DOI:10.16451/j.cnki.issn1003-6059.2013.10.006

协同过滤中一种有效的最近邻选择方法*

冷亚军[1,2] 梁昌勇[1,2] 丁 勇[1] 陆 青[3]

[1](合肥工业大学 管理学院 合肥 230009)
[2](过程优化与智能决策教育部重点实验室 合肥 230009)
[3](上海电力学院 经济与管理学院 上海 201300)

摘 要 协同过滤中的评分数据稀疏性使得最近邻搜寻不够准确,导致推荐质量较差. 基于此,文中提出一种有效的针对稀疏评分的最近邻选择方法——两阶段最近邻选择算法(TPNS). TPNS 分为两个步骤,首先计算用户间的近邻倾向性,选择近邻倾向性较高的用户组成初始近邻集合;然后根据初始近邻集合计算目标用户与其他用户间的等价关系相似性,使用等价关系相似性对目标用户的初始近邻集合进行修正,得到最近邻集合. 在 MovieLens 数据集上对比常用的推荐算法,实验结果表明文中方法在协同过滤推荐的应用中具有更高的准确性.

关键词 推荐系统,协同过滤,最近邻选择,近邻倾向性,近邻修正
中图法分类号 TP 311

Method of Neighborhood Formation in Collaborative Filtering

LENG Ya-Jun[1,2], LIANG Chang-Yong[1,2], DING Yong[1], LU Qing[3]
[1](*School of Management*, *Hefei University of Technology*, *Hefei* 230009)
[2](*Key Laboratory of Process Optimization and Intelligent Decision-Making*, *Ministry of Education*, *Hefei* 230009)
[3](*College of Economics and Management*, *Shanghai University of Electric Power*, *Shanghai* 201300)

ABSTRACT

In collaborative filtering, sparsity in ratings makes inaccurate neighborhood formation, thereby resulting in poor recommendations. To address this issue, a method of neighborhood formation, two-phase neighbor selection method (TPNS), is proposed. The definition of neighbor tendency is given. Based on the neighbor tendency, the preliminary neighborhood is formed. Then, the equivalence relation similarity is applied to modify the preliminary neighborhood, which makes the neighborhood formation more accurate. Experimental results on MovieLens dataset show that compared with the existing algorithms, TPNS performs better in the application of personalized recommendation.

Key Words Recommender System, Collaborative Filtering, Neighborhood Formation, Neighbor Tendency, Neighborhood Modification

* 国家自然科学基金(No. 71271072,71201145)、高等学校博士学科点专项科研基金(No. 20110111110006)、教育部人文社会科学研究基金(No. 09YJC630055, 11YJC630283)资助项目
收稿日期:2012-05-14;修回日期:2012-12-24
作者简介 冷亚军(通讯作者),男,1985 年生,博士研究生,主要研究方向为电子商务、数据挖掘. E-mail: huayi2001@163.com. 梁昌勇,男,1965 年生,教授,博士生导师,主要研究方向为智能决策支持系统、企业信息化. 丁勇,男,1969 年生,博士,副教授,主要研究方向为决策分析、电子商务. 陆青,男,1982 年生,博士,讲师,主要研究方向为进化计算、数据挖掘.

1　引　言

推荐系统根据用户偏好,向用户提供个性化信息、商品和服务推荐,帮助用户解决信息超载(Information Overload)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(Recommender System)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].

根据推荐过程中使用方法的不同,推荐系统[2]可分为,基于内容(Content-Based)的推荐系统、协同过滤(Collaborative Filtering)推荐系统和混合(Hybrid Approach)推荐系统. 协同过滤推荐系统根据其他用户的偏好向目标用户进行推荐,首先找出一组与目标用户偏好一致的邻居用户,然后对邻居用户进行分析,把邻居用户喜欢的项目推荐给目标用户. 由于协同过滤不需要考虑被推荐项目的内容,且易于实现,因此它成为一种较流行的推荐技术[3]. 许多大型网站,如 Amazon.com、Yahoo.com、Netflix.com 等都使用协同过滤推荐技术.

尽管协同过滤在个性化推荐方面取得较大成功,但却面临着严峻的稀疏性问题(Sparsity Problem)[4-5]. 实际网站中用户和项目数量庞大,而用户通常只对一小部分项目进行评分,可用于计算用户间相似性的数据非常有限. 经常可看到的现象是两个用户间没有任何共同评分项,导致相似性无法计算. 即使有的用户间相似性可计算,可靠性也难以保证. 评分数据的稀疏性使得最近邻搜寻不够准确,从而导致推荐质量较差.

基于此,本文提出一种有效的针对稀疏评分的最近邻选择方法——两阶段最近邻选择算法(Two-Phase Neighbor Selection, TPNS). TPNS 分为两个步骤,首先计算用户间的近邻倾向性,选择近邻倾向性较高的用户组成初始近邻集合; 然后根据初始近邻集合计算目标用户与其他用户间的等价关系相似性,使用等价关系相似性对目标用户的初始近邻集合进行修正,得到最近邻集合. 本文算法具有以下优点: 1) 首阶段的近邻倾向性综合考虑目标用户评分数量、用户共同评分数量、用户评分差异和项目信息熵 4 种信息,充分利用有限评分数据所包含的各种有用信息. 与传统的仅考虑用户评分差异的评分相似性相比,近邻倾向性能更准确地反映用户间的真实关系; 2) 第二阶段的等价关系相似性考虑目标用户与其他用户的共同邻居数,实现对目标用户邻居集合的二次计算,使搜寻到的最近邻更可靠.

2　相关工作介绍

协同过滤是推荐系统中广泛使用的最成功的推荐技术[6-8]. 传统的协同过滤算法分为 3 个阶段[9]: 用户偏好表示、邻居用户形成、推荐生成. 用户的偏好信息可用一个用户-项目评分矩阵 $\boldsymbol{R}$ 来表示,$\boldsymbol{R}$ 是一个 $m \times n$ 阶矩阵,m 表示用户的数目,n 表示项目的数目,矩阵中的每一元素 $R_{i,j}$ 表示第 i 个用户对第 j 个项目的评分值. 传统的协同过滤算法在 $\boldsymbol{R}$ 上计算目标用户与其他用户间的评分相似性,选择相似性最高的前 k 个用户组成目标用户的最近邻集合. 常用的相似性度量方法有 Pearson 相关系数、cosine 相似性和均方差相似性(Mean Squared Differences, MSD)[7,10].

最近邻集合形成后,可采用两种方法生成目标用户的 top-N 推荐集: 1) 综合所有最近邻评分以预测目标用户对未评分项目的评分,将预测评分最高的前 N 个项目推荐给目标用户[2,11]. 2) 采用最频繁项推荐(Most-Frequent Item Recommendation, MFIR)[9,12]完成对目标用户的 top-N 推荐,即将用户的评分划分为积极评分 R^p 和消极评分 R^n(如在 5 分制推荐系统中,$R^p = \{3, 4, 5\}$,$R^n = \{1, 2\}$),扫描所有最近邻用户的评分数据,将最近邻用户积极评分数最多且目标用户尚未评分的前 N 个项目推荐给目标用户. Symeonidis 等[12]的研究表明,方法 2) 的推荐质量高于方法 1),本文后续研究采用方法 2) 完成对目标用户的 top-N 推荐.

最近邻选择是协同过滤中最重要的步骤[9,13]. 传统的协同过滤算法在两个用户都有评分的项目集合上计算用户间相似性,筛选最近邻. 众所周知,用户只有在较多的项目上评分较相似,对用户间相似性的确定程度才会较高[14]. 实际上,用户在项目空间上的评分很少,经两个用户共同评分的项目更少,通常只有一两个,即使用户在这样小的项目集合上评分非常相似,也不能肯定他们之间具有很高的相似性. 传统的协同过滤算法在整个用户空间上搜寻最近邻,很多与目标用户共同评分较少,原本与目标用户差别较大的用户,由于计算出的评分相似性较高而被选为最近邻,在一定程度上降低最近邻搜寻的准确性,导致推荐质量较差.

针对上述问题,研究人员陆续提出一些解决方案. 最简单的做法是将评分矩阵中的未评分项设定为一个缺省值,这个值可以是评分中值、众数、用户评分均值、项目评分均值等[15]. 由于未评分项获得的评分值不可能完全相同,设定缺省值方法的可信

度并不高. 邓爱林等[14] 采用基于项目的协同过滤方法对用户评分项并集中的评分空值进行填补,在填补后的评分项并集上计算用户间的相似性. 基于项目的协同过滤方法本身也存在着稀疏性问题,这影响了填补效果,从而导致后续的计算不够准确. 李聪等[16] 对邓爱林等的方法进行改进,采用领域最近邻方法预测评分项并集中的未评分值. 领域最近邻方法实际上是在一个项目类别中计算用户间的相似性,实际网站中用户的评分非常稀疏,一个项目类别中的评分就更稀疏,这将导致很多用户间的相似性无法计算. Goldberg 等[17] 从 Jester 数据集中抽取一部分项目作为给定集(Gauge Set),用户使用系统时必须对给定集中的全部项目进行评分. Goldberg 等以增加用户-系统交互的方式来降低评分数据的稀疏性,然而在实际网站中,很难要求用户对所有指定的项目都进行评分. Kim 等[11] 提出一种偏差映射模型,采用协同过滤方法对用户的已有评分进行预测,将预测偏差存储到一个矩阵中. 对于用户的评分空值,计算出预测结果,并使用矩阵中的预测偏差对预测结果进行修正. Kim 等的方法只适用于评分数据极端稀疏的情况,当用户评分数量普遍大于 5 时,该方法的准确性反而不如传统的协同过滤算法. 文献[8] 使用奇异值分解(Singular Value Decomposition, SVD) 将 $m \times n$ 阶用户评分矩阵 $\boldsymbol{R}$ 分解为 3 个低阶矩阵: $\boldsymbol{R} \approx \boldsymbol{U}_h \times \boldsymbol{S}_h \times \boldsymbol{V}_h'$, $h < \min\{m, n\}$,然后基于低阶矩阵 $\boldsymbol{U}_h\boldsymbol{S}_h^{1/2}$ 进行协同过滤推荐,从而在降低稀疏性的同时也提高可扩展性. 进行 SVD 操作之前,需要采用项目评分均值对评分矩阵 $\boldsymbol{R}$ 进行填充,这与设定缺省值方法存在着同样的问题. 且降维会导致信息损失,降维效果与数据集密切相关,在项目空间维数很高的情况下,降维效果难以保证[19].

3 两阶段最近邻选择算法

两阶段最近邻选择算法(TPNS) 分为两个步骤,首先计算用户间的近邻倾向性,形成初始近邻集合; 然后通过等价关系相似性对初始近邻集合进行修正,使最近邻的搜寻结果更合理.

3.1 近邻倾向性的计算

通常用 U 表示推荐系统中的用户集合,I 表示项目集合. 假设 x 为一目标用户,

$$I_x = \{ i \in I \mid r_{x,i} \neq \bullet \}$$

表示 x 已评分的项目集合. 令 x 对项目的总体关注度为 1,则 x 对 I_x 中每一项目的关注度为 $1/\#I_x$,$\#I_x$ 表示集合 I_x 中含有的元素个数. 任取一项目 $i \in I$,

$$U_i^p = \{ u \in U \mid r_{u,i} \in R^p \}$$

表示对 i 进行积极评分的用户集合,

$$U_i^n = \{ u \in U \mid r_{u,i} \in R^n \}$$

表示对 i 进行消极评分的用户集合. 则对于用户 $y \neq x$,任取一项目 $i \in I_{xy}$,

$$I_{xy} = \{ i \in I \mid r_{x,i} \neq \bullet \ \wedge \ r_{y,i} \neq \bullet \}$$

为两个用户共同评分的项目集合,在项目 i 上目标用户 x 选择用户 y 作为最近邻的倾向性 $t_{xy}(i)$:

$$t_{xy}(i) = \begin{cases} \dfrac{1}{\#I_x}\#U_i^p, & r_{x,i} \in R^p, r_{y,i} \in R^p \\ -\dfrac{1}{\#I_x}\#U_i^n, & r_{x,i} \in R^p, r_{y,i} \in R^n \\ -\dfrac{1}{\#I_x}\#U_i^p, & r_{x,i} \in R^n, r_{y,i} \in R^p \\ \dfrac{1}{\#I_x}\#U_i^n, & r_{x,i} \in R^n, r_{y,i} \in R^n \end{cases} \tag{1}$$

计算 I_{xy} 中每一项目上 x 选择 y 作为最近邻的倾向性,则 x 选择 y 作为最近邻的总体倾向性:

$$t(x,y) = \sum_{i \in I_{xy}} t_{xy}(i). \tag{2}$$

根据目标用户 x 对所有用户近邻倾向性的大小,选择倾向性最高的前 k' 个用户组成 x 的初始近邻集合 $U_x^c = \{u_1, u_2, \cdots, u_k\}$.

近邻倾向性以目标用户的已评分项目为基础进行计算. 它为目标用户的每一评分项目分配相同的权重值. 任取一目标用户的已评分项目,当另一用户 $y(y \in U)$ 与目标用户在该项目上的偏好相同时(同为积极评分或同为消极评分),y 在该项目上被选为最近邻的倾向性为正. 当 y 与目标用户在该项目上的偏好不同时,y 在该项目上被选为最近邻的倾向性为负. 当 y 没有对该项目进行评分时,则无法识别 y 与目标用户在该项目上的关系,故忽略该项目. 确定用户 y 在某一项目上被选为最近邻的倾向性大小时,还考虑项目信息熵的作用: 与 y 偏好相同的信息熵越大,则该项目在用户近邻倾向性的确定过程中所起的作用就越小; 反之则越大. 近邻倾向性综合考虑目标用户评分数量、用户共同评分数量、用户评分差异和项目信息熵 4 种信息,充分利用有限评分数据所包含的各种有用信息.

3.2 初始近邻的修正

高维稀疏的评分数据使得 3.1 节搜寻到的初始近邻集合中或多或少存在一些偏好差异较大的用户. 文献[20] 采用 Rough 集理论的分类观点对高维

稀疏数据进行聚类,定义等价关系相似性和广义等价关系,避免相似对象被划分到不同聚类中,达到抑制噪声的作用. 本文受文献[20]的启发,采用等价关系相似性和广义等价关系对初始近邻集合进行修正,使得到的近邻集合更合理.

定义 1 近似空间[21]　定义一个二元序对 $K = (S, D)$,其中,$S = \{e_1, e_2, \cdots, e_f\}$ 是有限非空论域,D 是 S 上的一个二元等价关系,$D \subseteq S \times S$ 也称为 S 上的一个不可分辨关系. $S/D = \{[e]_D \mid e \in S\}$ 表示 D 在 S 上的划分,由 S 中每个对象 e 所在的 D–等价类 $[e]_D$ 组成.

定义 2 初始等价关系[22]　对象 e_i 的初始等价关系定义为 $D_i = \{[e_i], S - [e_i]\}$,其中

$$[e_i] = \{e_j \mid \xi(e_i, e_j) \geqslant \alpha\},$$

$\xi(e_i, e_j)$ 为对象 e_i 和 e_j 的稀疏相似性,$j = 1, 2, \cdots, f$,α 为所取的对象稀疏相似性阈值.

定义 3 等价关系相似性[20]　设 D_i、D_j 为任意的两个初始等价关系,定义等价关系相似性:

$$\mu(D_i, D_j) = w_{ij} \frac{|[e_i] \cap [e_j]|}{|[e_i] \cup [e_j]|}, \tag{3}$$

其中

$$w_{ij} = \begin{cases} 1, & \text{if } [e_i] \cap [e_j] \neq \varnothing \\ 0, & \text{otherwise} \end{cases}$$

定义 4 广义等价关系[20]　设 D_i、D_j 为任意的两个初始等价关系,定义广义等价关系:

$$D_i' = \{Q_i, S - Q_i\},$$

其中

$$Q_i = \bigcup_{1 \leqslant j \leqslant f} \{e_j \mid \mu(D_i, D_j) \geqslant \beta\},$$

β 为所取的等价关系相似性阈值.

根据以上定义对初始近邻集合进行修正. 将用户集合 $U = \{u_1, u_2, \cdots, u_m\}$ 看作一个有限非空论域. $\forall u_i \in U$,计算 u_i 对所有用户近邻倾向性的大小,则 u_i 的初始等价关系:

$$R_i = \{[u_i], U - [u_i]\},$$

其中

$$[u_i] = \{u_j \mid t(u_i, u_j) \geqslant \alpha\}.$$

本文取近邻倾向性最高的前 k' 个用户组成 $[u_i]$,即 $[u_i] = U^c_{u_i}$. 对于目标用户 u_a 和其他用户 u_j 的初始等价关系 D_a 和 D_j,根据式(3)计算等价关系相似性 $\mu(D_a, D_j)$,则 u_a 的广义等价关系:

$$D_a' = \{Q_a, U - Q_a\},$$

其中

$$Q_a = \bigcup_{1 \leqslant j \leqslant m} \{u_j \mid \mu(D_a, D_j) \geqslant \beta\},$$

本文取等价关系相似性最高的前 k 个用户组成 Q_a. 将 Q_a 作为目标用户 u_a 的最近邻集合,完成对 u_a 初始近邻集合的修正.

等价关系相似性实现用户间相互关系的二次计算. 其基本思想是两用户的共同邻居数越多,他们偏好相似的可能性就越大. 等价关系相似性以第一阶段产生的近邻集合为基础进行计算,它实际上是对最近邻集合的一种修正,使搜寻到的最近邻更可靠.

3.3　本文算法描述

将 TPNS 应用到协同过滤推荐中,提出基于 TPNS 的协同过滤推荐算法,算法的具体描述如下.

算法　基于 TPNS 的协同过滤推荐算法

输入　用户–项目评分矩阵 R、初始近邻数 k'、最近邻数 k、被推荐项目数 N

输出　目标用户的 top–N 推荐集 I_r

step 1　将所有用户的评分划分为积极评分 R^p 和消极评分 R^n.

step 2　$\forall u_i \in U$,令 u_i 对项目的总体关注度为 1,采用式(1)、式(2)计算 u_i 选择其他用户作为最近邻的倾向性,取倾向性最高的前 k' 个用户组成 u_i 的初始近邻集合 $U^c_{u_i} = \{u_1, u_2, \cdots, u_{k'}\}$.

step 3　循环执行 step 2,得到所有用户的初始近邻集合.

step 4　采用式(3)计算目标用户 u_a 与其他用户的等价关系相似性,选择等价关系相似性最高的前 k 个用户作为 u_a 的最近邻集合

$$U^n_{u_a} = \{u_1, u_2, \cdots, u_k\}.$$

step 5　扫描所有最近邻用户的评分数据,将最近邻积极评分数最多且 u_a 尚未评分的前 N 个项目作为 u_a 的 top–N 推荐集 $I_r = \{i_1, i_2, \cdots, i_N\}$,并输出.

4　实验及结果分析

4.1　实验数据集

本文使用 MovieLens 站点[23] 提供的数据集对算法进行评估,该站点由美国 Minnesota 大学的 GroupLens 研究小组创建并维护,是一个基于 Web 的研究型推荐系统,通过用户对电影的评分进行电影推荐. 目前该站点公布 3 个评分数据集,本文采用 MovieLens100K 数据集. 该数据集包含 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),数据的稀疏等级为

$$1 - \frac{100000}{943 \times 1682} = 0.9370.$$

实验中将数据集按照 80% 和 20% 的比例划分为训

练集和测试集,采用 5 折交叉法进行验证.

4.2 评价标准

实验采用精确率(Precision)[12,23]作为度量算法优劣的标准. *precision* 值越大,算法的推荐准确性越高. 令用户 u 测试集中的积极评分项集合为 A,算法为其生成的 top-N 推荐集合为 B,则算法相对于 u 的 *precision*:

$$precision_u = \frac{|A \cap B|}{|B|}.$$

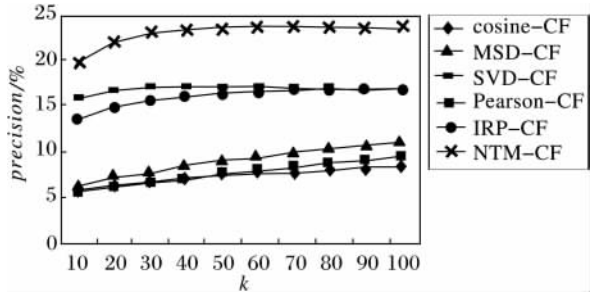
算法的总体 *precision*:

$$precision_w = \frac{\sum_{u \in U} precision_u}{|U|}.$$

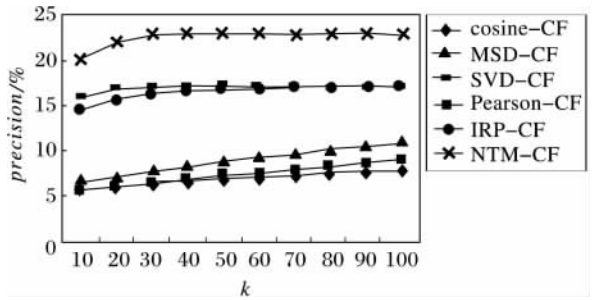
4.3 实验结果

4.3.1 近邻倾向性的有效性

首先考查本文提出的近邻倾向性方法(Neighbor Tendency Method, NTM)的有效性,并确定初始近邻数 k' 的大小. 对基于 cosine 相似性的协同过滤算法(cosine-CF)[24]、基于 Pearson 相关系数的协同过滤算法(Pearson-CF)[25]、基于 MSD 的协同过滤算法(MSD-CF)[10]、文献[14]提出的基于项目评分预测的协同过滤算法(IRP-CF)、文献[18]提出的基于 SVD 的协同过滤算法(SVD-CF)和采用 NTM 的协同过滤算法(NTM-CF)对比. 最近邻数在 10 ~ 100 之间变动,实验结果如图 1 所示.



(a) $N = 10$



(b) $N = 20$

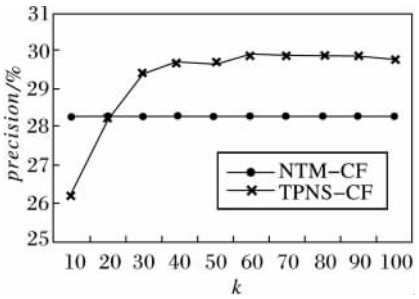
图 1 近邻倾向性方法的效果

Fig. 1 Effect of NTM

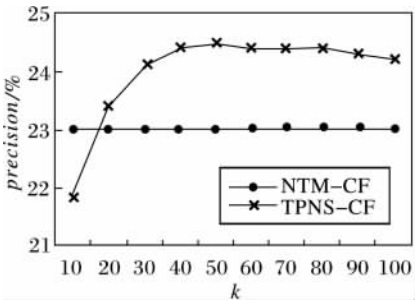
被推荐项目数 N 为 10 和 20 两种情况下实验的结果基本相同,N 为 10 时各算法的 *precision* 值稍大. 在任意最近邻数目下,采用 NTM 的协同过滤算法都取得比其它 5 种算法更大的 *precision*. 这表明本文提出的近邻倾向性方法可有效缓解数据稀疏性,提高协同过滤算法的推荐质量. 当最近邻数从 10 增加到 40 时,NTM-CF 的推荐准确性不断升高,此后再增加最近邻数,NTM-CF 的准确性变动很小,曲线趋于平缓. 这表明对于 NTM-CF 来说,40 个邻居用户便可提供质量够高的推荐,因此取初始近邻数 $k' = 40$ 进行后续的实验.

4.3.2 近邻修正的作用

实验考查初始近邻集合修正对算法推荐质量的影响. 实验结果如图 2 所示,可看到,被推荐项目数 N 为 10 和 20 两种情况下实验的结果基本相同. 对于 $N = 10$ 的情况,当最近邻数 $k \geqslant 30$ 时,TPNS-CF 的 *precision* 优于 NTM-CF. 对于 $N = 20$ 的情况,当最近邻数 $k \geqslant 20$ 时,TPNS-CF 的 *precision* 优于 NTM-CF. 这表明选取适当数量等价关系相似性较高的用户对初始近邻集合进行修正,确实可进一步提高算法的推荐质量. 两种情况下,TPNS-CF 的推荐质量均在 $k = 40$ 时达到最优,因此修正后的最近邻集合大小应为 40.



(a) $N = 10$



(b) $N = 20$

图 2 近邻修正的影响

Fig. 2 Impact of neighborhood modification

4.3.3　算法性能对比

最后对原始的协同过滤算法(Original-cosine、Original-Pearson、Original-MSD、Original-IRP、Original-SVD、Original-NTM)和采用近邻修正的算法(Modified-cosine、Modified-Pearson、Modified-MSD、Modified-IRP、Modified-SVD、Modified-NTM)对比. 对于采用近邻修正的6种算法,为便于对比,初始近邻数 k' 都设置为40. 最近邻数 k 在10 ~ 100之间变动,所有算法的 *precision* 均为10个 k 值下 *precision* 的平均值.

实验结果如表1所示(粗体表示最优结果). Original-NTM 的 *precision* 大于 Original-cosine、Original-Pearson、Original-MSD、Original-IRP 和 Original-SVD,这再次证明本文提出的近邻倾向性方法在协同过滤推荐的应用中具有更高的准确性. 采用近邻修正的6种算法,其 *precision* 分别大于各自没有采用时,说明本文提出的近邻修正策略可广泛提高多种协同过滤算法的准确性. Modified-NTM(即TPNS-CF)的 *precision* 大于其它算法,这表明采用本文两阶段最近邻选择方法TPNS的协同过滤算法,在所有算法中具有最高的推荐质量.

表1　不同算法精确率对比

Table 1　Precision comparison of different methods

		cosine	Pearson	MSD	IRP	SVD	NTM
top-10	Original	0.089	0.094	0.110	0.193	0.203	0.278
	Modified	0.267	0.265	0.256	0.244	0.204	**0.293**
top-20	Original	0.070	0.074	0.089	0.165	0.169	0.225
	Modified	0.220	0.218	0.205	0.203	0.170	**0.240**

5　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越来越重要的作用,许多大型网站都应用推荐系统. 协同过滤是推荐系统中广泛使用的较成功的推荐技术,但却面临着严峻的稀疏性问题. 评分数据的稀疏性使得最近邻搜寻不够准确,导致推荐质量较差.

本文提出一种最近邻选择方法——两阶段最近邻选择算法. 首先计算用户间的近邻倾向性,形成初始近邻集合. 然后基于初始近邻集合,计算用户间的等价关系相似性,完成对初始近邻集合的修正. 首阶段的近邻倾向性方法(NTM)综合考虑目标用户评分数量、用户共同评分数量、用户评分差异和项目信息熵4种信息,充分利用有限评分数据所包含的各种有用信息,因此可有效缓解评分数据的稀疏性,保证初始近邻集合的质量. 第二阶段的近邻修正策略考虑用户间的共同邻居数量,实现对近邻集合的二次计算,从而进一步提高最近邻搜寻结果的可靠性. 本文提出的近邻修正策略不仅适用于NTM,还可提高其它多种协同过滤算法的推荐质量,具有较广泛的应用价值.

参　考　文　献

[1] Wu Hu, Wang Yongji, Wang Zhe, *et al*. Two-Phase Collaborative Filtering Algorithm Based on Co-Clustering. Journal of Software, 2010, 21(5): 1042 - 1054 (in Chinese)
(吴 湖,王永吉,王 哲,等. 两阶段联合聚类协同过滤算法. 软件学报, 2010, 21(5): 1042 - 1054)

[2] Adomavicius G, Tuzhilin A. Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions. IEEE Trans on Knowledge and Data Engineering, 2005, 17(6): 734 - 749

[3] Leung C W K, Chan S, Chung F L. A Collaborative Filtering Framework Based on Fuzzy Association Rules and Multiple-Level Similarity. Knowledge and Information Systems, 2006, 10(3): 357 - 381

[4] Sarwar B, Karypis G, Konstan J, *et al*. Item-Based Collaborative Filtering Recommendation Algorithms // Proc of the 10th International Conference on World Wide Web. Hong Kong, China, 2001: 285 - 295

[5] Kim H N, Ji A T, Ha I, *et al*. Collaborative Filtering Based on Collaborative Tagging for Enhancing the Quality of Recommendation. Electronic Commerce Research and Applications, 2010, 9(1): 73 - 83

[6] Herlocker J, Konstan J, Terveen L, *et al*. Evaluating Collaborative Filtering Recommender Systems. ACM Trans on Information Systems, 2004, 22(1): 5 - 53

[7] Ahn H J. A New Similarity Measure for Collaborative Filtering to Alleviate the New User Cold-Starting Problem. Information Sciences, 2008, 178(1): 37 - 51

[8] Lee S K, Cho Y H, Kim S H. Collaborative Filtering with Ordinal Scale-Based Implicit Ratings for Mobile Music Recommendations. Information Sciences, 2010, 180(11): 2142 - 2155

[9] Sarwar B, Karypis G, Konstan J, *et al*. Analysis of Recommendation Algorithms for E-Commerce // Proc of the 2nd ACM Conference on Electronic Commerce. New York, USA, 2000: 158 - 167

[10] Bobadilla J, Hernando A, Ortega F, *et al*. Collaborative Filtering Based on Significances. Information Sciences, 2012, 185(1): 1 - 17

[11] Kim N H, EI-Saddik A, Jo G S. Collaborative Error-Reflected Models for Cold-Start Recommender Systems. Decision Support Systems, 2011, 51(3): 519 - 531

[12] Symeonidis P, Nanopoulos A, Papadopoulos A N, *et al*. Collaborative Recommender Systems: Combining Effectiveness and Effi-

ciency. Expert Systems with Applications, 2008, 34(4): 2995 – 3013

[13] Luo H, Niu C Y, Shen R M, *et al*. A Collaborative Filtering Framework Based on Both Local User Similarity and Global User Similarity. Machine Learning, 2008, 72(3): 231 – 245

[14] Deng Ailin, Zhu Yangyong, Shi Bole. A Collaborative Filtering Recommendation Algorithm Based on Item Rating Prediction. 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. 软件学报, 2003, 14(9): 1621 – 1628)

[15] Sun Xiaohua. Research of Sparsity and Cold Start Problem in Collaborative Filtering. Ph. D Dissertation. Hangzhou, China: Zhejiang University, 2005 (in Chinese)
(孙小华. 协同过滤系统的稀疏性与冷启动问题研究. 博士学位论文. 杭州: 浙江大学, 2005)

[16] Li Cong, Liang Changyong, Ma Li. A Collaborative Filtering Recommendation Algorithm Based on Domain Nearest Neighbor. Journal of Computer Research and Development, 2008, 45(9): 1532 – 1538 (in Chinese)
(李 聪,梁昌勇,马 丽. 基于领域最近邻的协同过滤推荐算法. 计算机研究与发展, 2008, 45(9): 1532 – 1538)

[17] Goldberg K, Roeder T, Gupta D, *et al*. Eigentaste: A Constant Time Collaborative Filtering Algorithm. Information Retrieval, 2001, 4(2): 133 – 151

[18] Sarwar B M, Karypis G, Konstan J A, *et al*. Application of Dimensionality Reduction in Recommender System – A Case Study // Proc of the ACM WebKDD 2000 Web Mining for E-Commerce Workshop. Boston, USA, 2000: 82 – 90

[19] Aggarwal C C. On the Effects of Dimensionality Reduction on High Dimensional Similarity Search // Proc of the 20th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems. Santa Barbara, USA, 2001: 256 – 266

[20] Zhao Yaqin, Zhou Xianzhong, He Xin, *et al*. An Effective High Attribute Dimensional Sparse Clustering. Pattern Recognition and Artificial Intelligence, 2006, 19(3): 289 – 294 (in Chinese)
(赵亚琴,周献中,何 新,等. 一种有效的高属性维稀疏数据聚类算法. 模式识别与人工智能, 2006, 19(3): 289 – 294)

[21] Miao Duoqian, Wang Jue. An Information Representation of the Concepts and Operations in Rough Set Theory. Journal of Software, 1999, 10(2): 113 – 116 (in Chinese)
(苗夺谦,王 珏. 粗糙集理论中概念与运算的信息表示. 软件学报, 1999, 10(2): 113 – 116)

[22] An Qiusheng, Shen Junyi, Wang Guoyin. A Clustering Method Based on Information Granularity and Rough Sets. Pattern Recognition and Artificial Intelligence, 2003, 16(4): 412 – 417 (in Chinese)
(安秋生,沈钧毅,王国胤. 基于信息粒度与 Rough 集的聚类方法研究. 模式识别与人工智能, 2003, 16(4): 412 – 417)

[23] Liu D R, Lai C H, Lee W J. A Hybrid of Sequential Rules and Collaborative Filtering for Product Recommendation. Information Sciences, 2009, 179(20): 3505 – 3519

[24] Breese J, Heckerman D, Kadie C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering // Proc of the 14th Conference on Uncertainty in Artificial Intelligence. Madison, USA, 1998: 43 – 52

[25] Resnick P, Iacovou N, Suchak M, *et al*. GroupLens: An Open Architecture for Collaborative Filtering of Netnews // Proc of the ACM Conference on Computer Supported Cooperative Work. Chapel Hill, USA, 1994: 175 – 186