推荐系统中一种新的相似性计算方法*)

李 涛[1,2] 王建东[1] 叶飞跃[3]

(南京航空航天大学信息科学与技术学院 南京 210016)[1]

(南京信息工程大学电子与信息工程学院 南京 210044)[2]

(江苏技术师范学院计算机科学与工程学院 常州 213001)[3]

摘 要 随着互联网的发展,推荐系统逐步得到广泛应用,协同过滤是其中的关键技术之一,它根据相似用户的喜好产生对目标用户的推荐。随着用户和项目数量的增加,用于产生推荐的数据集将极端稀疏,协同过滤系统的性能下降。为此,提出了一种新的用户多层相似性度量,不仅降低数据稀疏性的影响,而且克服了相似不相同的问题。实验表明,该度量方式能够提高协同过滤系统的推荐质量。

关键词 推荐算法,协同过滤,相似性

A New Similarity Method Used in Recommender Systems

LI Tao[1,2] WANG Jian-Dong[1] YE Fei-Yue[1]

(College of Information Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 210016)[1]

(College of Electronic and Information Technology, Nanjing University of Information Science and Technology, Nanjing 210044)[2]

(College of Computer Science and Technology, Jiangsu Teachers University of Technology, Changzhou 213001)[3]

Abstract Recommender systems are becoming increasingly popular with the evolution of the Internet, and collaborative filtering (CF) is one of the most important technologies in recommender systems. The performance of CF systems degrades with increasing number of customers and items. So, a new multiple-level user similarity is presented, which not only overcomes the difficulty of data sparsity, but also solves the "similar but not same" problem. The experimental results show that the presented algorithm can improve the performance of CF systems in both the recommendation quality and efficiency.

Keywords Recommendation algorithm, Collaborative filtering, Similarity

1 引言

随着电子商务的蓬勃发展,推荐系统被越来越广泛地应用于电子商务网站中,扮演传统商业中销售人员的角色:向顾客提供商品信息和建议,模拟销售人员帮助顾客完成购买过程。

根据推荐的生成方式,可以将目前的推荐系统分为两类:基于内容的和协同推荐系统。基于内容的推荐系统是根据目标用户自己以往对项目(如电子商务中的商品、电影、音乐等)的喜好或评分,判断其对目标项目的喜好程度,从而产生推荐[1~3]。协同过滤推荐算法基本思想是通过计算目标用户与各个基本用户对项目评分之间的相似性,搜索目标用户的最近邻居,然后根据最近邻居的评分数据对目标用户产生推荐,即目标用户对未评分项目的评分可以通过最近邻居对该项目评分的加权平均值进行逼近,从而产生推荐[4,5]。其中,协同过滤推荐系统应用更为广泛,因而本文主要针对协同过滤推荐系统进行讨论。

在协同过滤系统中,为了对目标用户产生推荐,需要搜索目标用户最近邻居,在此过程中,定义用户之间的相似性成为关键问题之一。自第一个协同过滤系统 Tapestry[4] 诞生以来,多种用户相似性度量被用于协同过滤算法,主要包括 Pearson 相关相似性[5],向量相似性[6],余弦相似性[7],修正的余弦相似性[7],统计贝叶斯方法[8],最小平方差[9]等。其中,Pearson 相关相似性是目前协同过滤系统中最为常用的相似性度量之一。

文[6,7,10]对比了上述各度量方法,总体来说,使用 Pearson 相关相似性度量与统计贝叶斯网络的协同过滤系统性能要优于使用其他度量的系统。基于贝叶斯网络的协同过滤系统需要相对较少的内存容量,但是其训练时间过长。基于 Pearson 相关相似性的协同过滤系统无需训练过程,易于理解,而且具有较高的推荐质量,因而 Pearson 相关相似性得到广泛的应用。

随着电子商务系统规模的扩大,用于产生推荐的数据将极端稀疏,使用上述度量的产生推荐效果将逐步减弱。同时,在使用这些相似性度量的协同过滤系统中,部分用户具有相似的喜好,但是由于他们仅对相似的项目进行了评分,而没有评分相同的项目,系统并不将其视为近邻,这就产生了所谓的"相似不相同"问题。

针对这些问题,本文提出了一种用于协同过滤的多层相似性,将概念分层引入相似性的计算,在一定程度上缓解了数据稀疏性带来的推荐质量问题,克服相似不相同问题,提高系统的推荐生成质量。实验也表明,本文提出的相似性度量要优于目前使用的其他相似性计算方法。

2 相关概念

定义 1 推荐系统中的数据源 $S=(I,U,R)$,其中 $I=\{I$-

*)本研究得到南京信息工程大学科研基金资助项目(Y507)资助。**李 涛** 博士生,主要研究方向:数据挖掘、人工智能、电子商务。**王建东** 教授,博士生导师,主要研究方向:数据挖掘、人工智能。**叶飞跃** 教授,主要研究方向:数据挖掘、人工智能。

$tem_1, Item_2, \cdots, Item_n\}$是推荐系统中的项目集合，$\| I \| = n$；$U=\{User_1, User_2, \cdots, User_m\}$是推荐系统中基本用户的集合，$\| U \| = m$；$m \times n$ 阶矩阵 R 是基本用户对各项目的评分矩阵，其中 R 中第 i 行第 j 列的元素 r_{ij} 表示 U 中第 i 用户对 I 中第 j 个项目的评分。

在现实环境中，随着系统规模的扩大，m 和 n 的值都会迅速增大，由于用户不可能对所有的项目都进行评分，$R(m,n)$ 中空值数目将急剧增加，从而导致评分数据极端稀疏。数据集的稀疏性不仅会影响最近邻的查找效果，降低推荐质量，而且会导致推荐系统近邻选择时相似不相同的问题。

概念分层就是定义一个映射序列，将低层概念映射到更一般的高层概念[11]。它是数据挖掘中的一种数据预处理方法。在电子商务系统中，项目通常属于一定的类别，在此基础上，可以对项目进行概念分层[12,13]。

定义 2 给定数据源 $S=(I,U,R)$，项目集 I 中的元素按照类别分为 l 层 $\{C^0, C^1, C^2, \cdots, C^l\}$，其中 $C^0 = I$。$C^i = \{c_1^i, c_2^i, \cdots, c_{s_i}^i\}$ 为第 i 层类别的集合，其中 c_j^i 表示第 i 层类别集合 C^i 中的第 j 个类别，其表征的类别包含了集合 C^{i-1} 中的若干元素所表征的类别，同时隶属于集合 C^{i+1} 中的一个或多个元素所表征的类别。$n_i = |C^i|$ 为第 i 层类别的个数，一般而言，$n=n_0>n_1>\cdots>n_l$ 总能成立。

例如，雪碧、可乐等隶属于高一层的软饮料类，啤酒、白酒等隶属于酒精饮料类，而这两类又都隶属于更高一层的饮品类。

设 G_k 是类别关系矩阵，表示第 $k-1$ 层的类别与第 k 层的类别之间的关系，则

$$g_k(i,j)=\begin{cases}1 & \text{类别 } c_i^{k-1} \text{ 隶属于类别 } c_j^k \\ 0 & \text{类别 } C_i^{k-1} \text{ 不属于类别 } c_j^k\end{cases},$$

其中 $k \geqslant 1$。

定义 3 R^k 是分层评分矩阵，表示用户对第 k 层类别的评分。其中元素 $r^k(i,j)$ 表示在用户 i 对第 k 层的类别 j 的评分，其中 $0 \leqslant k \leqslant l, 1 \leqslant i \leqslant m, 1 \leqslant j \leqslant n_k$。当 $k=0$ 时，R^0 即数据源 D 中的原始评分矩阵 R。

$$\text{本文中，} r^k(i,j)=\frac{\text{向量 } R^{k-1}(i) \times G_k \text{ 中非 0 元素的和}}{\text{向量 } R^{k-1}(i) \times G_k \text{ 中非 0 元素的个数}},$$

其中 $k \geqslant 1$，$R^{k-1}(i)$ 表示用户 i 对第 $k-1$ 层类别的评分向量。可见，用户对第 k 层类别的评分矩阵 R^k 可以由 R^{k-1} 计算，已知用户对项目的原始评分矩阵 R^0，以及各层的类别关系矩阵，可以计算出各层的类别评分矩阵。

基于前述分析，我们选用 Pearson 相关相似性作为本文多层相似性度量方式的基础。

定义 4 设在第 k 层项目类别中，经用户 i 和用户 j 共同评分的类别集合用 S_k 表示，则在第 k 层类别中，用户 i 和用户 j 之间的相似性 $\mathrm{sim}_k(i,j)$ 通过 Pearson 相关系数度量：

$$\mathrm{sim}_k(i,j)=\frac{\sum_{u \in S_k}(r_{iu}-\overline{r_i})(r_{ju}-\overline{r_j})}{\sqrt{\sum_{u \in S_k}(r_{iu}-\overline{r_i})^2}\sqrt{\sum_{u \in S_k}(r_{ju}-\overline{r_j})^2}} \quad (1)$$

其中，$k \geqslant 0$，当 $k=0$ 时，计算的是用户 i 和用户 j 在原始项目评分上的相似性。r_{iu} 和 r_{ju} 分别表示用户 i 和 j 对第 k 层类别 u 的评分，$\overline{r_i}$ 和 $\overline{r_j}$ 分别表示用户 i 和 j 对第 k 层类别的平均评分。

定义 5 用户 i 与用户 j 之间的多层综合相似性为

$$\mathrm{Sim}(i,j)=\sum_l \alpha_k \times \mathrm{sim}_k(i,j) \quad (2)$$

其中 α_k 为用户 i 与用户 j 在第 k 层类别相似性的权重 $\mathrm{sim}_k(i,j)$ 所占的权重，且 $\sum_l \alpha_k = 1$。$\alpha=[\alpha_0, \alpha_1, \cdots, \alpha_l]$ 可以根据实际情况或经验选取[11]。

定义 6 已知数据源 $S=(I,U,R)$，给定目标用户 a，及其对 I 中项目评分向量 $A(1,n)$，对于 $\forall i \in U$，将 $Sim(a,i)$ 最大的 l 个基本用户 i 组成集合 NS_a，则称该集合中的元素为目标用户 a 的最近邻居。

定义 7 已知数据源 $S=(I,U,R)$，给定目标用户 a，及其最近邻居集合 NS_a，则用户 a 对项目 j 的预测评分 P_{aj} 为

$$P_{aj}=\overline{r_a}+\frac{\sum_{i \in NS_a}\mathrm{Sim}(a,i)\times(r_{ij}-\overline{r_i})}{\sum_{i \in NS_a}|\mathrm{Sim}(a,i)|} \quad (3)$$

其中，$\overline{r_a}$ 和 $\overline{r_i}$ 分别表示用户 a 和用户 i 对项目 j 的平均评分。

3 实验结果及分析

本节介绍实验所采用的数据集和实验结果评价标准及结果分析。我们以传统的协同过滤算法为基础，通过实验将本文提出的多层综合相似性度量与其他相似性进行了对比。

3.1 实验环境及实验数据集

性能测评的硬件平台是配置 Intel Pentium Ⅳ 2.4GHz CPU 和 256MB RAM 的 PC，操作系统运行 Windows XP Professional，所有程序均采用 Matlab 实现。

本文的实验数据采用 MovieLens 站点提供的数据集。MovieLens 是一个基于 Web 的研究型推荐系统，用于接收用户对电影的评分并提供相应的电影推荐列表。目前，该 Web 站点的用户已经超过 43,000 人，可供用户评分的电影超过 3,500 部。我们从用户评分数据库中选择 100,000 条评分数据作为实验数据集，其中包括 943 个用户和 1,682 部电影，其中每个用户至少对 20 部电影进行了评分。该数据集可以分为两层类别，所有电影（项目）分为 19 个电影流派（类别），其中一部电影可以属于一个或多个较高层次的电影流派。实验过程中，我们将原始电影作为 C^0，各电影流派作为 C^1，计算用户之间的两层相似性。

实验过程中，我们将整个实验数据集进一步划分为训练集和测试集，并使用 5-折交叉验证（5-fold cross-validation）方法。随机地把数据集分成 5 个互不相交的子集，每个子集的大小大致相等。每次选择一个项目子集作为测试集，其他 4 个作为训练集，然后运行 5 次测试。

3.2 度量标准

评价推荐系统质量的度量标准主要包括统计精度度量方法和决策支持精度度量方法[8]。统计精度度量方法中的平均绝对误差（mean absolute error，MAE）易于计算和理解，可以直观地对推荐质量进行度量，是最常用的一种推荐质量度量方法，本文即是采用 MAE 作为度量标准。其中 MAE 越小，推荐质量越高。

3.3 实验结果

根据文[6,7,10]，在协同过滤系统中，Pearson 相关相似性的推荐质量要优于其他相似性度量，因而在实验过程中，以传统的过滤算法为基础，将多层相似性度量与 Pearson 相关相似性进行了对比。

在实验过程中，我们首先将共同评分项目阈值设定为 20，即用户之间共同评分项目的数量大于 20 时，我们计算其相似性，否则直接将其相似性置为 0。目标用户的最近邻居数量 l 从 10 变化到 30，间隔为 5。在多层相似性计算过程中，分别使用 $\alpha=[0.2,0.8]$，$\alpha=[0.3,0.7]$，$\alpha=[0.4,0.6]$，$\alpha=[0.5,0.5]$。实验结果如图 1 所示。

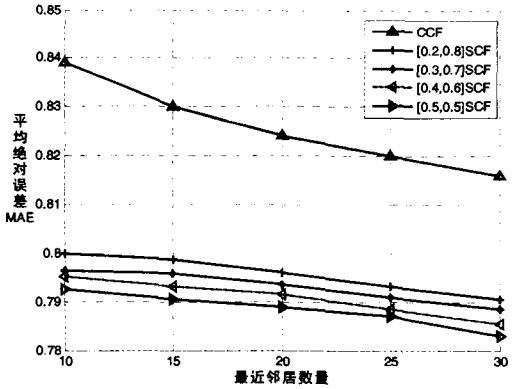

图 1 共同评分项目阈值固定为 20 时推荐精度比较

在进一步实验中，我们将目标用户的最近邻居数量 l 设定为 20。计算用户之间相似性时，共同评分项目数量阈值从 10 变化到 30，间隔为 5，再次进行对比实验，结果如图 2 所示。

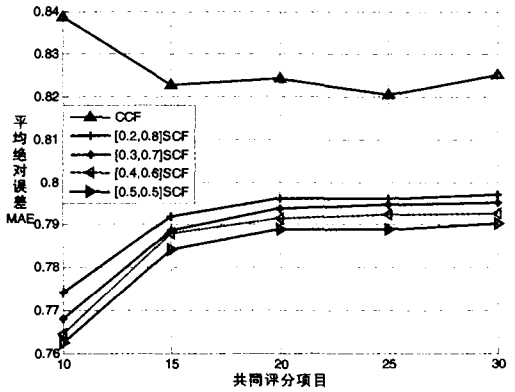

图 2 最近邻居阈值固定为 20 时推荐精度比较

由实验结果可以看出，在相同条件下，一般而言，基于多层相似性的协同过滤算法要的 MAE 均小于其他对比算法。同时，计算多层相似性时，我们特意变化了权重的 α 选取，实验结果也表明，推荐精度受其影响不大。也就是说，多层相似度对 α 的选取没有特别苛刻的要求。

传统的相似性计算方法仅仅考虑用户对各个项目的实际评分，并没有将项目之间的关系考虑进去，因而即使用户具有相似的喜好，由于没有对相同的项目进行评分，使用传统的相似性计算方法获得的结果并不将它们视为最近邻，这将影响推荐系统中实际最近邻的查找结果，进而影响推荐质量。多层综合相似性不仅考虑了用户对基本项目的评分情况，而且考虑了用户对项目类别的评分，即其对项目类别的喜欢程度，因而能够避免传统协同过滤系统中“相似不相同”的问题，同时能够获得更好的推荐质量。

结束语 针对协同过滤推荐系统的稀疏性，本文提出了一种新的用户相似性度量方法，在推荐系统项目集合中引入概念分层，并综合计算用户相似性。通过运用新的相似性度量方法，不仅缓解了系统数据稀疏性问题，而且避免了“相似不相同”的问题，同时实验表明新的相似性度量方法能够提高系统推荐质量。

参 考 文 献

1 Balabanovic M, Shoham Y. Fab: Content-Based, Collaborative Recommendation. Communications of ACM, 1997, 40(3): 66～72
2 Mooney R J, Roy L. Content-Based Book Recommending Using Learning for Text Categorization. In: Proceeding of ACM SIGIR'99 Workshop Recommender Systems: Algorithms and Evaluation, 1999
3 Pazzani M, Billsus D. Learning and Revising User Profiles: The Identification of Interesting Web Sites. Machine Learning, 1997, 27: 313～331
4 Goldberg D, Nichols D, O ki BM, et al. Using collaborative filtering to weave an information Tapestry. Communications of the ACM, 1992, 35(12): 61～70
5 Resnick P, Iacovou N, Suchak M, et al. GroupLens: An open architecture for collaborative filtering of Netnews. In: Proceeding of ACM 1994 Conference on Computer Supported Cooperative Work, 1994. 175～186
6 Breese J S, Heckerman D, Kadie C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. In: Fourteen Conference on Uncertainty in Artificial Intelligence, 1998
7 Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithm. In: Proceedings of the 10th International World Wide Web Conference, 2001. 285～295
8 Ha V, Haddawy P. Toward Case-Based Preference Elicitation: Similarity Measures on Preference Structures. In: Proceedings of 14th Conference on Uncertainty in Artificial Intelligence, 1998. 193～201
9 Shardanand U, Maes P. Social Information Filtering: Algorithms for Automating "Word of Mouth". In: Proceedings of ACM Conference on Human Factors in computing Systems, 1995. 210～217
10 Vozalis E V, Margaritis K G. Recommender systems: An experimental comparison of two filtering algorithms. http://macedonia.uom.gr/～mans/papiria/voz-epy9.pdf. (2006-07-06)
11 Han J, Kamber M. Data mining concepts and techniques [M]. Beijing: High Education Press, 2001. 87～93, 232～236
12 Leung C W, Chan S C, Chung F. A collaborative filtering framework based on fuzzy association rules and multiple-level similarity. Knowledge Information Systems, 2006, 9(4): 492～511
13 Kim C, Kim J. A recommendation algorithm using multi-level association rules. In: Proceedings of the IEEE/WIC International Conference on Web Intelligence, 2003

（上接第 167 页）

实验结果表明，算法 CROP_Index 在效率上优于 CROP。

结论 本文通过对 CROP 算法进行改进，提出了挖掘频繁闭项集的高性能算法 CROP_Index。该算法用“索引数组”来组织数据，通过为每个项目增加包含索引，找到频繁共同出现的项集。基于二进制位图，给出了一个包含索引的计算方法，并利用索引启发信息合并，得到复合型频繁模式树的初始结点；同时，给出一些新的性质，使得改进的算法只生成闭合结点，从而节省了大量不必要的操作，缩小了搜索空间。实验结果表明，CROP_Index 在效率上优于 CROP。

参 考 文 献

1 Pasquier N, Bastide Y, Taouil R, Lakhal L. Discovering Frequent Closed Itemsets for Association Rules [C]. In: Beeri C, Buneman P, eds. Proceedings of 7th International Conference Database Theory. Jerusalem: Springer-Verlag, 1999. 398～416
2 Zaki MJ, Hsiao CJ. CHARM: An Efficient Algorithm for Closed Itemset Mining [C]. In: Grossman R, et al, eds. Proceedings of the 2nd SIAM International. Conference on Data Mining. Arlington: SIAM, 2002. 12～28
3 Lucchese C, Orlando S, Perego R. Fast and Memory Efficient Mining of Frequent Closed Itemsets [J]. IEEE Transaction on Knowledge and Data Engineering, 2006, 18(1): 21～36
4 Pei J, Han J, Mao R. CLOSET: An Efficient Algorithm for Mining Frequent Closed Itemsets [C]. In: Gunopulos D, Rastogi R, eds. Proceedings of 2000 ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery. Dallas: ACM Press, 2000. 21～30
5 Wang J Y, Han J, Pei J. CLOSET+: Searching for the Best Strategies for Mining Frequent Closed Itemsets [C]. In: Getoor L, Senator T E, Domingos P, Faloutsos C, eds. Proceedings of the ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM Press, 2003. 236～245
6 刘君强，孙晓莹，庄越挺，潘云鹤. 挖掘闭合模式的高性能算法 [J]. 软件学报，2004，15(1)：94～102
7 缪裕青. 频繁闭合项目集的并行挖掘算法 [J]. 计算机科学，2004，31(5)：166～168
8 Frequent Itemset Mining Implementations Repository. http://fimi.cs.helsinki.fi/