基于评分支持度的最近邻协同过滤推荐算法*

陶维安，范会联
（长江师范学院 数学与计算机学院，重庆 408100）

摘　要：针对传统协同过滤推荐算法存在推荐质量不高的局限性，提出一种基于评分支持度的最近邻协同过滤推荐算法。该算法用调整后的共同评分次数动态调节相似度的值，以更真实地反映彼此间的相似性。然后计算目标用户和目标项目的最近邻集合及各自评分和支持度，根据评分支持度自适应调节基于目标用户和目标项目的评分对最终推荐结果影响的权重。与其他算法的对比实验结果表明，该算法能有效避免传统相似度度量方法存在的问题，从而提高了推荐质量。

关键词：协同过滤；最近邻居；评分支持度；相似度
中图分类号：TP18　　**文献标志码**：A　　**文章编号**：1001-3695(2012)05-1723-03
doi:10.3969/j.issn.1001-3695.2012.05.032

Collaborative filtering recommendation algorithm based on nearest-neighborhood and rating support

TAO Wei-an, FAN Hui-lian
(*School of Mathematics & Computer Science*, *Yangtze Normal University*, *Chongqing* 408100, *China*)

Abstract: To solve the shortcomings of the traditional collaborative filtering recommendation algorithms, this paper proposed an improved collaborative filtering recommendation algorithm for the nearest neighbors based on rating support. First on the basis of correlation similarity, this algorithm adopted an improved similarity measure method which could dynamically adjust the value of similarity according to the modified common rating. Then, computed predicting rating and rating support of the active user and item based on the nearest neighbor sets. Finally, according to the rating support data, adjusted different self adaptive influence weights of the neighbor sets of the active user and the active item, and obtained the final recommendation results. The experimental results show that compared with the other recommendation algorithms, the algorithm can effectively avoid the defects of traditional similarity measure and improve the recommendation quality.

Key words: collaborative filtering; nearest neighborhood; rating support; similarity

个性化推荐系统能基于用户个人偏好为用户提供定制信息，其中协同过滤是当前最成功的推荐技术之一[1]，已广泛应用于电子商务的各个领域。在基于协同过滤的推荐系统中，一种被广泛采用的经典模型是 K-近邻模型，其工作原理是利用评分相似度来构造目标用户或项目的 K-近邻集合[2,3]，再基于 K-近邻集合进行推荐。由于评分数据的稀疏性导致传统相似度的计算结果往往不能准确反映与目标对象的真实联系，加上该模型需要事先确定目标对象的最近邻居数量 K，而未知用户或项目的相似性近邻数量往往不是固定的，过大和过小的 K 值对算法推荐结果会产生很大影响。同时，K-近邻模型往往只单独考虑用户或项目因素而忽视两者之间的关联性，从而影响推荐系统的推荐质量。

1　问题定义及相关工作

1.1　问题定义

协同过滤推荐问题的定义如下[4]：给定用户集 U 和项目集 I，用户对于项目的兴趣可以表示为一个 $|U| \times |I|$ 的用户—项目评分矩阵 $R(m,n)$，矩阵中每一元素 $R_{u,i}$ 是用户 u 对于项目 i 的评分，体现用户 u 对项目 i 的兴趣和偏好程度。在该矩阵中，m 个行向量分别代表用户集 $U=\{u_1,u_2,\cdots,u_m\}$ 中的每个用户对项目的评分集合，n 个列向量分别表示项目集 $I=\{i_1,i_2,\cdots,i_n\}$ 中各个项目的被评分集合。一般情况下，R 中已知评分的数量远远小于未知评分的数量。协同过滤推荐算法要解决的问题是给定评分集 $T \subset R(|T| << |R|)$ 作为训练集，根据 T 中已知的评分，构造一个推荐系统，该系统要能以最小的累积误差对 R 中未知评分项目进行评分预测。

1.2　相关工作研究现状

常用的最近邻协同过滤推荐算法可分为基于用户和基于项目两种。基于用户的推荐算法通过分析不同用户对项目的评分，为目标用户寻找若干个最相似的邻居用户（最近邻居），根据最近邻居对该项目的评分来对目标用户尚未评分的项目作出评分预测。这种算法主要关注用户与用户之间兴趣的关联性[5]。2001 年，Sarwar 等人[6] 提出了一种基于项目的协同过滤推荐算法，该算法基于这样一种前提：当大部分用户对一些项目的评分比较相似时，当前用户对这些项目的评分也是相似的。算法通过寻找尚未评分项目的最近邻项目集合，然后根

收稿日期：2011-10-16；**修回日期**：2011-11-17　　**基金项目**：重庆市教委科学技术资助项目(KJ111304)；重庆市涪陵区科委资助项目(FLKJ,2011ABA2043)

作者简介：陶维安(1956-)，男，副教授，主要研究方向为智能信息处理；范会联(1971-)，男，重庆石柱人，副教授，硕士，主要研究方向为智能信息处理、软件工程等(fhlmx@163.com).

据这些最近邻居项目评分值来对目标项目进行评分预测。

近年来,对协同过滤推荐算法的研究集中在如何同时结合基础于用户和基于项目的推荐算法对推荐进行优化。Xue 等人[7]提出基于聚类的协同过滤推荐算法,利用聚类技术进行空缺值的填补和用户邻居以及项目邻居的选取。Wang 等人[8]通过建立概率模型来预测空缺的用户评分数据,并提出融合基于用户和基于项目算法的预测值计算方法。汪静等人[9]提出基于共同评分和相似度权重的协同过滤推荐算法,根据用户相似度和项目相似度值的相对大小来动态加权各自的评分预测值。黄裕洋等人[10]提出的综合用户和项目因素的推荐算法,根据最近邻用户集的用户数和最近邻项目集的项目数的多少来加权各自评分预测值。以上这些算法考虑了用户和项目之间的联系,但是这些算法要么以固定权重来结合基于用户和基于项目的评分预测值(如文献[7,8]),但在对数据集没有充分了解的情况下难以给出恰当的固定权重。文献[9,10]以基于用户和基于项目算法的各自预测结果去动态加权预测值,但其加权因子中没有充分考虑用户和项目各自最近邻的质量。

本文基于评分支持度的不确定近邻协同过滤推荐算法,在研究相似度的度量合理性基础上,提出对基于用户和基于项目算法各自评分预测值的支持度概念,据此解决融合基于用户和基于项目算法时对尚未评分项目评分预测值的动态影响程度。

2 基于不确定近邻推荐策略

用户或项目间相似度的度量是否准确,直接关系到推荐算法的推荐质量。因此,如何衡量目标对象与其他对象的相似度成为提高系统推荐质量的关键,本文在 Pearson 相关系数基础上改进相似度的计算。

2.1 传统的相似度度量方法

用户间相似度用于度量用户兴趣偏好的相似程度,通过不同用户对共同项目的评分数据来反映。衡量用户相似度的 Pearson 相关系数计算方法如下[11]:

$$\mathrm{USim}(u,v)=\frac{\sum_{i\in I(u)\cap I(v)}(R_{u,i}-\overline{R_u})(R_{v,i}-\overline{R_v})}{\sqrt{\sum_{i\in I(u)\cap I(v)}(R_{u,i}-\overline{R_u})^2}\sqrt{\sum_{i\in I(u)\cap I(v)}(R_{v,i}-\overline{R_v})^2}} \tag{1}$$

其中:$\mathrm{USim}(u,v)$表示用户 u 和 v 的相似度;$I(u)$ 和 $I(v)$ 分别表示用户 u 和 v 评分的项目集合;i 表示用户 u 和 v 共同评分的项目,是 $I(u)$ 和 $I(v)$ 的交集;$R_{u,i}$ 和 $R_{v,i}$ 分别表示用户 u 和 v 对项目 i 的评分;$\overline{R_u}$ 和 $\overline{R_v}$ 分别表示用户 u 和 v 各自评分的均值。

项目相似度反映的是用户对项目同时感兴趣的程度,如果不同用户对两个项目的共同评分趋向一致,说明他们对这两个项目的偏好趋向一致。因此,可通过比较对同一个项目存在评分值的用户集合来计算项目相似度,衡量项目相似度的 Pearson 相关系数计算方法如下[11]:

$$\mathrm{ISim}(i,j)=\frac{\sum_{u\in U(i)\cap U(j)}(R_{u,i}-\overline{R_i})(R_{u,j}-\overline{R_j})}{\sqrt{\sum_{u\in U(i)\cap U(j)}(R_{u,i}-\overline{R_i})^2}\sqrt{\sum_{u\in U(i)\cap U(j)}(R_{u,j}-\overline{R_j})^2}} \tag{2}$$

其中:$\mathrm{ISim}(i,j)$表示项目 i 和 j 的相似度;$U(i)$ 和 $U(j)$ 分别表示项目 i 和 j 中存在评分值的用户集合;u 是对项目 i 和 j 共同评分的用户,是 $U(i)$ 和 $U(j)$ 的交集;$R_{u,i}$ 和 $R_{u,j}$ 分别表示用户 u 对项目 i 和 j 的评分值;$\overline{R_i}$ 和 $\overline{R_j}$ 分别表示项目 i 和 j 的平均被评分值。

2.2 相似度度量的改进

以上相似度的计算依赖于用户共同评分的项目,在实际的推荐系统中,由于用户评分数据极端稀疏,共同评分的项目数量极少。因此,相似度的计算结果存在较大偶然因素[12]。为了消除这种偶然性带来的影响,Herlocker 等人[13,14]提出增加一个关联权重因子来计算相似度。按此思想,根据用户间共同评分项目个数越多则他们的兴趣偏好可能越相似的原则,用不同用户共同评分项目个数来调整相似度。同理,项目相似度也存在类似情况,即两个项目被不同用户共同评分的次数越多,则说明他们都能满足较多用户的兴趣偏好,因此,项目的相似程度就可能越高。

基于以上分析,定义用户 u 和 v 共同评分的项目集用 $I(u)\cap I(v)$ 表示,对项目 i 和 j 共同评分的用户集用 $U(i)\cap U(j)$ 表示,$|^*|$ 运算表示集合包含的元素个数,用 $\max_{w\in U}|I(u)\cap I(w)|$ 表示用户 u 与用户集中其他用户共同评分项目个数的最大值,用 $\max_{k\in I}|U(i)\cap U(k)|$ 表示项目 i 与项目集中其他项目同时被用户评分的最大用户数。本文用自然指数函数调整共同评分次数对原相似度的影响,将其影响范围控制在约 $(0.36788,1]$ 内。改进的用户相似度计算方法如式(3)所示,改进的项目相似度计算如式(4)所示。

$$\mathrm{USim}'(u,v)=\exp\left(\frac{|I(u)\cap I(v)|}{\max_{w\in U}|I(u)\cap I(w)|}-1\right)\times\mathrm{USim}(u,v) \tag{3}$$

$$\mathrm{ISim}'(i,j)=\exp\left(\frac{|U(i)\cap U(j)|}{\max_{k\in I}|U(i)\cap U(k)|}-1\right)\times\mathrm{ISim}(i,j) \tag{4}$$

式(3)的调整系数能反映在用户 u 所关注的项目中,与用户 v 重合得越多,则 u 与 v 的相似度可能越大。式(4)的调整系数表示对项目 i 评分的用户中,同时对项目 j 也进行评分的越多,则项目 i 与 j 的相似度可能就越大。

2.3 最近邻居的选取

对于每个未知评分数据 $R_{u,i}$,用户 u 的最近邻居集合 $\mathrm{NearU}(u)$ 的选取规则为

$$\mathrm{NearU}(u)=\{v|\mathrm{USim}(u,v)>\alpha,v\neq u\} \tag{5}$$

项目 i 的最近邻居集合 $\mathrm{NearI}(i)$ 的选取规则为

$$\mathrm{NearI}(i)=\{j|\mathrm{ISim}(i,j)>\beta,j\neq i\} \tag{6}$$

其中:α 和 β 分别是用于确定是否属于最近邻居用户集和最近邻居项目集的阈值。

2.4 未评分项目的预测评分

获得目标用户的最近邻居集后,根据其邻居用户对尚未评分项目的评分来预测目标用户对该项目的评分值。基于用户的协同过滤推荐算法中,用户 u 对其未评分项目 i 的评分预测值计算如下:

$$P_{\mathrm{user}}(R_{u,i})=\overline{R_u}+\frac{\sum_{v\in\mathrm{NearU}(u)}\mathrm{USim}'(u,v)\times(R_{v,i}-\overline{R_v})}{\sum_{v\in\mathrm{NearU}(u)}\mathrm{USim}'(u,v)} \tag{7}$$

获得目标项目的最近邻居集后,根据其邻居项目被评分情况来预测当前用户对目标项目的评分预测值。基于项目的协同过滤推荐算法中,用户 u 对其未评分项目 i 的评分预测值计算如下:

$$P_{\mathrm{item}}(R_{u,i})=\overline{R'}_i+\frac{\sum_{j\in\mathrm{NearI}(i)}\mathrm{ISim}'(i,j)\times(R_{u,j}-\overline{R_j})}{\sum_{j\in\mathrm{NearI}(i)}\mathrm{ISim}'(i,j)} \tag{8}$$

以用户最近邻居集和项目最近邻居集为基础,本文提出评分支持度概念,再结合两种推荐算法,用评分支持度对基于用户的评分和基于项目的评分预测结果进行自适应动态调整,以调节基于用户和基于项目的评分对最终推荐结果影响的权重,从而获得更合适的推荐结果。

基于用户的评分支持度定义为

$$U\lambda_{u,i}=\frac{\sum_{v\in \mathrm{NearU}(u)}|I(u)\cap I(v)|}{\sum_{v\in \mathrm{NearU}(u)}|I(u)\cap I(v)|+\sum_{j\in \mathrm{NearI}(i)}|U(i)\cap U(j)|} \tag{9}$$

基于项目的评分支持度定义为

$$I\lambda_{u,i}=\frac{\sum_{j\in \mathrm{NearI}(i)}|U(i)\cap U(j)|}{\sum_{v\in \mathrm{NearU}(u)}|I(u)\cap I(v)|+\sum_{j\in \mathrm{NearI}(i)}|U(i)\cap U(j)|} \tag{10}$$

其中:$\sum_{v\in \mathrm{NearU}(u)}|I(u)\cap I(v)|$ 表示当前用户 u 与属于其最近邻居集 NearU(u)的用户 v 的共同评分项目个数的和,$\sum_{j\in \mathrm{NearI}(i)}|U(i)\cap U(j)|$ 表示同时对未评分项目 i 与属于其最近邻居集 Near$I$(i)的项目 j 进行共同评分的用户个数的和。显然,$U\lambda_{u,i}+I\lambda_{u,i}=1$。

最后,经过自适应动态加权后,本文算法用户 u 对项目 i 的评分预测值计算公式为

$$P(R_{u,i})=U\lambda_{u,i}\times P_{\mathrm{user}}(R_{u,i})+I\lambda_{u,i}\times P_{\mathrm{item}}(R_{u,i}) \tag{11}$$

3　实验结果及分析

3.1　数据集

实验使用的测试数据集来自 GroupLens 研究产品组(http://www.grouplens.org/)提供的一个著名电影评分数据 MovieLens,它有 10 万条记录,存储了 943 个用户对 1 682 部电影的评分,每个用户至少对 20 部电影进行了评分,评分的范围是 1 ~5,不同的评分表达了自己的兴趣。本实验从 10 万条记录随机抽取 80% 的记录组成训练集,剩余记录组成测试集。

3.2　推荐质量的度量标准

本文评价推荐质量的度量标准采用平均绝对偏差(mean absolute error, MAE)[15]或准确率(precision)[16]。MAE 通过计算预测的用户评分与实际用户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。设预测的用户评分集合为$\{p_1,p_2,\cdots,p_N\}$,相应的实际用户评分集合为$\{q_1,q_2,\cdots,q_N\}$,则 $\mathrm{MAE}=\frac{\sum_{i=1}^{N}p_i-q_i}{N}$。

准确率表示算法正确推荐数目占整个推荐集的比例。正确推荐定义为:如果算法推荐集中某个项目出现在目标用户测试集的访问记录里,则表示该推荐是正确的。具体计算为:$\mathrm{precision}=\frac{\mathrm{hits}}{N}$,其中,hits 表示算法产生的正确推荐数目,$N$ 表示算法生成的推荐总数。

3.3　实验及结果分析

本文通过三个实验来检验算法的有效性,

实验 1　用 MAE 评测本文算法的推荐质量。将本文 2.4 节提出的对未评分项目预测评分(RSCF)算法与文献[5]基于用户的最近邻协同过滤推荐(UCFR)算法、文献[6]基于项目的最近邻协同过滤推荐(ICFR)算法和文献[9] 基于共同评分和相似度权重的协同过滤推荐(CRSW)算法进行了对比实验,实验结果如图 1 所示。

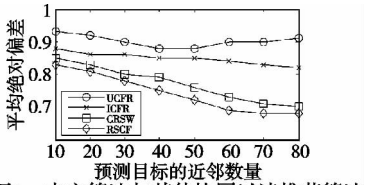

图1　本文算法与其他协同过滤推荐算法的对比测试结果

由图 1 可以看出,相比其他三种推荐算法,随着邻居数目的数量的不断变化,本文算法均可以获得更低的 MAE 值,因此算法有更好的推荐质量。

实验 2　用 precision 评测本文相似度度量改进的有效性。采用传统 K-近邻协同过滤推荐算法,从 10 个最近邻开始逐步递增,分别用本文提出的改进相似度(mysim)、Pearson 相关系数(psim)方法进行推荐准确率比较,实验结果如图 2 所示。由图 2 可以看出,本文提出的相似度量方法在不同的邻居个数下均能获得比 Pearson 相关系数方法更好的推荐准确率。这是因为本文相似度的计算中融入了共同评分次数的影响,使得选取的用户或项目与当前用户或项目具有更好的相似性。

实验 3　测试本文算法邻居阈值 α 和 β 对 MAE 的影响。这两个阈值的大小直接决定用户及项目的相似性邻居集合的元素个数。实验中让 α 和 β 同时从 0.2 开始变化到 0.7,间隔设为 0.05。图 3 是 α 和 β 对 MAE 的影响情况。

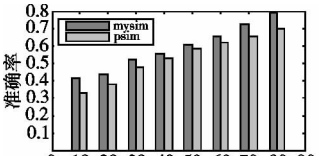

图2　基于准确率的不同相似度度量方法比较

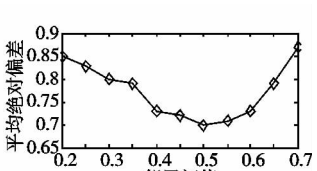

图3　邻居阈值 α 和 β 对MAE的影响

从实验 3 的结果可以看出,随着阈值的变化,MAE 值不断改变:当阈值设为 0.4 ~0.6 区间中的值时,MAE 值较小且波动不大,以后随着阈值的增大 MAE 呈上升的趋势。由于本文算法在相似度的计算上乘了一个取值在(0.36788,1]区间的自适应调整系数,因此,邻居阈值 α 和 β 取值不能设得过高。阈值较小时,导致相似近邻数量增多,但其中有些项目或用户实际的相似度并不大,从而使得推荐偏差较高。随着阈值的不断增大,对相似度的要求越来越高,邻居集合的相似度也随之增大,因此能取得较高的推荐质量。当阈值为 0.5 左右时,平均绝对偏差最小,随着近邻阈值的进一步增大,相似近邻数量越来越少,导致平均绝对偏差就再次增大。

4　结束语

本文用共同评分次数自适应调整相似度的计算方法,在不需要人为干预的情况下,有效解决了评分数据极端稀疏情况下传统相似度度量方法存在的不足。在产生推荐时,分别计算目标用户和目标项目的最近邻集合并分别评分;根据用户评分和项目评分的支持度动态调节用户和项目因素对最终推荐结果的影响权重。实验结果表明,该算法能较好地提高推荐质量。

参考文献:

[1] LINDEN G, SMITH B, YORK J. Amazon.com recommendations: item to item collaborative filtering[J]. IEEE Internet Computing, 2003,7(1):76-80.

(下转第 1728 页)

(上接第 1725 页)

[2] 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.

[3] BELL R M, KOREN Y. Improved neighborhood-based collaborative filtering[C]//Proc of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2007:7-14.

[4] 陈逸,于洪. 一种基于相同评分矩阵的协同过滤补值算法[J]. 计算机应用研究,2009,26(12):4513-4515,4519.

[5] LEE H C,LEE S J,CHUNG Y J. A study on the improved collaborative filtering algorithm for recommender system[C]//Proc of the 5th International Conference on Software Engineering Research, Management and Applications. 2007:297-304.

[6] SARWAR B M, KARYP I S G, KONSTAN J,*et al*. Item-based collaborative filtering recommendation algorithms[C]//Proc of the 10th International World Wide Web Conference. 2001:285-295.

[7] XUE G R, LIN C X, YANG Q,*et al*. Scalable collaborative filtering using cluster-based smoothing[C]//Proc of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York:ACM Press,2005:114-121.

[8] WANG Jun, De VRIES A P, REINDERS M J T. Unifying user-based and item-based collaborative filtering approaches by similarity fusion [C]//Proc of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM Press,2006:501-508.

[9] 汪静,印鉴,郑利荣,等. 基于共同评分和相似性权重的协同过滤推荐算法[J]. 计算机科学,2010,37(2):99-103.

[10] 黄裕洋,金远平. 一种综合用户和项目因素的协同过滤推荐算法[J]. 东南大学学报:自然科学版,2010,40(5):917-921.

[11] 王茜,杨莉云,杨德礼. 面向用户偏好的属性值评分分布协同过滤算法[J]. 系统工程学报,2010,25(4):562-568.

[12] 黄创光,印鉴,汪静,等. 不确定近邻的协同过滤推荐算法[J]. 计算机学报,2010,33(8):1369-1376.

[13] HERLOCKER J, KONSTAN J A, RIEDL J. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms [J]. Information Retrieval, 2002,5(4):287-310.

[14] McLAUGHLIN M R, HERLOCKER J L. A collaborative filtering algorithm and evaluation metric that accurately model the user experience[C]// Proc of the 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York:ACM Press,2004:329-336.

[15] HERLOCKER J, KONSTAN J, TERVEEN L,*et al*. Evaluating collaborative filtering recommender systems[J]. ACM Trans on Information Systems,2004,22(1):5-53.

[16] 于洪,李转运. 基于遗忘曲线的协同过滤推荐算法[J]. 南京大学学报:自然科学版,2010,46(5):520-527.