doi: 10.3969/j.issn.0490-6756.2018.03.012

基于用户非对称相似性的协同过滤推荐算法

黄贤英, 龙姝言, 谢　晋
(重庆理工大学计算机科学与工程学院, 重庆 400054)

摘　要: 针对传统协同过滤推荐算法的数据稀疏以及用户关系衡量不准确的问题,提出了基于用户非对称相似关系的推荐算法.利用用户的潜在特征的样本数量,结合奇异值矩阵分解,计算用户之间非对称的相似度,明确用户间关系.仿真结果表明,随着邻居数量的增加,该算法的平均绝对误差始终优于传统算法,误差值在邻居数量为 40～60 之间值为最小,约为 0.682,传统算法平均绝对误差值约为 0.758,可以看出该算法判断用户关系较为准确,预测评分比传统算法更接近实际评分.

关键词: 协同过滤; 推荐算法; 非对称相似; 矩阵分解
中图分类号: TP391　　**文献标识码**: A　　**文章编号**: 0490-6756(2018)03-0489-05

Collaborative filtering recommendation algorithm based on asymmetric similarity of users

HUANG Xian-Ying, *LONG Shu-Yan*, *XIE Jin*
(School of Computer Science and Engineering, Chongqing University of Technology, Chongqing 400054, China)

Abstract: Aiming at data sparseness and inaccurate user relationship in traditional collaborative filtering recommendation algorithm, an improvement recommendation algorithm based on asymmetric similarity relationship of users is proposed. By using the sample number of potential features and the decomposition of singular value matrix, the asymmetric similarity between users is calculated, and the relation between users is defined. The simulation results show that , the Mean Absolute Error of the algorithm is superior to the traditional algorithm with the increase of the number of neighbours, the minimal value is about 0.682 between the number of neighbors is 40～60. The value of Mean Absolute Error is about 0.758 of the traditional algorithm. It can be seen that the algorithm to determine the user relationship is more accurate, predictive score is relatively close to the actual score.

Keywords: Collaborative filtering; Recommendation algorithm; Asymmetric similarity; Matrix factorization

1　引　言

随着网络通讯和电子商务的兴起,互联网成为人们获取信息以及购物的重要工具,导致了数据的爆炸式增长,也就是信息过载.用户要从互联网上的海量信息中找到自己需要的信息变得比较困难,因此,推荐系统应运而生.推荐系统是根据喜好为用户推荐物品的自动化系统[1-3],大大节约了用户

收稿日期: 2017-10-20
基金项目: 教育部人文社科青年项目(16YJC860010); 重庆市社会科学规划博士项目(2015BS059); 国家自然科学基金(61603065); 国家统计局全国统计科学研究重点项目(2016LZ08); 教育部人文社会科学研究项目(15YJC790061)
作者简介: 黄贤英(1967－), 女, 汉族, 重庆万州人, 教授, 硕士, 研究方向为数据挖掘, 推荐系统,嵌入式系统,信息检索.
通讯作者: 龙姝言. E-mail: lsy0727@foxmail.com

花费的时间,并且在各大电子商务网站,音乐网站以及在线视频网站都有一定的应用.因此,推荐系统成为一个越来越受学者关注的研究领域.由于推荐系统的发展,各类技术也被应用到推荐系统中,其中应用最广,效率最高的就是协同过滤算法.

协同过滤推荐系统根据其推荐依据又分为两种方法:基于项目的协同过滤和基于用户的协同过滤[4,5].虽然推荐系统应用于各个领域,但随着数据量的不断增加,协同过滤推荐算法又存在着几个问题:(1)数据稀疏[6]:大多数用户对系统中的项目评分较少,因此这些用户可能找不到类似的用户;(2)冷启动[7,8]问题:这是指新项目或新用户出现的问题.一个没有任何评分项目的新用户且尚未找到类似的用户进行推荐.另一方面,当一个新的项目被引入到系统中,并且没有用户对该项目评分,它就不可能被推荐给用户;(3)推荐精度[9,10]不高:由于对用户关系以及其喜好判断的不准确,会导致给用户推荐的物品并不是用户所喜欢的.

在推荐系统这一领域,许多学者进行了研究,例如文献[11]利用正则化梯度下降法,即 Regularized SVD(RSVD)模型来进行预测;文献[12]提出了概率矩阵分解算法,利用概率图模型对评分矩阵进行建模,从而得出矩阵分解优化公式;文献[13]利用融合特征的奇异值矩阵分解为微博用户推荐好友等;文献[14]提出一种采用结合修正公式改进的 Jaccard 相似性系数计算用户之间的相似度,在计算过程中考虑用户之间共同评分项和所有评分项的关系,以及用户在共有的评价项目集合上的评分差别对用户的相似度的影响,从而获取更加精确的用户相似度矩阵[15].但在这类方法中,并没有考虑到用户之间由于评分数量和分值的不同,其潜在特征向量的样本数量也不相同,就会导致其相似关系是非对称的,并且统一的正则化参数会导致算法本身的不平衡,对于一部分的项目和用户参数值偏小,而对另一部分项目和用户的参数值偏大.

针对上述问题,本文提出了一个以用户为中心的协同过滤算法,将用户的偏好、意见、行为和反馈认为是用户特征,并与项目特征和上下文信息集成在一起,并且支持不同的应用程序中使用适当的自定义,提出处理过度专业化或意外问题的模型,进而向用户推荐可能没有发现的项目.该算法使用基于矩阵分解的推荐算法,使系统能够充分了解项目,发现项目之间的隐藏关系,使一些冷门项目可能被推荐.

2 传统的协同过滤推荐算法

2.1 问题定义及参数

本文分别用 u,v 表示两个不同的用户;i,j 表示不同的项目;r_{ui} 表示用户 u 对项目 i 的已有评分;I_u 为用户 u 已评分的电影集合;$\hat{r}_{vj}$ 表示用户 v 对项目 j 的预测评价分数,分值为 1~5 分,分数越高,表示该用户对该物品的喜欢程度越深.用户的所有评分记录可以看作是一个用户-项目评分矩阵 R,其中包括 m 个用户 $\{U_1,U_2,U_3,\cdots,U_m\}$ 以及 n 个项目 $\{I_1,I_2,I_3,\cdots,I_n\}$ 如式(1)所示.

$$\begin{array}{c} \\ U_1 \\ U_2 \\ U_3 \\ \cdots \\ U_m \end{array}\begin{array}{c} \begin{array}{ccccc} I_1 & I_2 & I_3 & \cdots & I_n \end{array} \\ \begin{bmatrix} r_{11} & r_{12} & r_{13} & \cdots & r_{1n} \\ r_{21} & r_{22} & r_{23} & \cdots & r_{2n} \\ r_{31} & r_{32} & r_{33} & \cdots & r_{3n} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ r_{m1} & r_{m2} & r_{m3} & \cdots & r_{mn} \end{bmatrix} \end{array} \tag{1}$$

2.2 基础偏移量

一般来说,用户评分的时候,存在个人基础偏好这个因素,也就是说,用户在对项目评分的时候,可能分数普遍比其他用户高或者比其他用户低,这是用户的评分偏好,也有可能存在有的物品得到的评分普遍比其他物品高,因此,文献[14]提出基础偏移量来表示用户或者物品的偏好.

$$b_{ui}=\bar{r}+b_u+b_i \tag{2}$$

其中,b_u 表示用户 u 的评分偏好;b_i 表示用户对项目 i 的评分偏好;$\bar{r}$ 表示用户评分矩阵中所有用户评分的平均值.例如,预测用户 U_1 对电影《金刚狼:殊死一战》的评分,平均分为 4 分,用户 U_1 平均打分比别的用户高 0.5 分,而电影《金刚狼:殊死一战》的平均打分低于其他电影 0.3 分,因此用户 U_1 对电影《金刚狼:殊死一战》的预测评分为:$b_{ui}=4+0.5-0.3=4.2$.

2.3 矩阵分解

矩阵分解(Matrix Factorization,MF)方法就是将用户-项目评分矩阵分解为若干个低阶矩阵的乘积.本文采用的是增量奇异值矩阵分解,通过找到一个特征向量维度 F,将用户-项目评分矩阵 R 分解为两个低阶矩阵 P,Q 的乘积,P 为 $F\times m$ 阶的用户特征矩阵,Q 为 $F\times n$ 的项目特征矩阵.

$$R=P^T\cdot Q \tag{3}$$

然后利用用户特征矩阵 P 和项目特征矩阵 Q 预测用户 u 对项目 i 的偏好值,如式(4).

$$\hat{r}_{ui} = \bar{r} + b_u + b_i + \sum_{f=1}^{F} p_{fu}^T \times q_{fi} \tag{4}$$

其中，p_{fu} 为用户特征矩阵 P 中的第 f 行，第 u 列的值；q_{fi} 为项目特征矩阵 Q 中的第 f 行，第 i 列的值. 由于实验目标是找到最优特征矩阵 P 和 Q，使得目标用户对目标项目的评分预测值与真实值之间的误差尽量小，因此，定义目标函数 G 为式(5).

$$G = \sum (r_{ui} - \hat{r}_{ui})^2 = \sum (r_{ui} - \bar{r} - b_u - b_i - (\sum_{f=1}^{F} p_{fu}^T \times q_i))^2 \tag{5}$$

为了避免相对数量比较小的项目和数据过拟合的现象，需要加入惩罚因子来约束递归调整过程，如式(6).

$$G = \sum (r_{ui} - \bar{r} - b_u - b_i - (\sum_{f=1}^{F} p_{fu}^T \times q_{fi}))^2 + \beta(b_u^2 + b_i^2 + \| p_u \|^2 + \| q_i \|^2) \tag{6}$$

其中，p_u 为用户特征矩阵 P 中的第 u 行；q_i 为项目特征矩阵 Q 中的第 i 行.

3　基于用户非对称相似度的推荐算法

不同用户之间评分的项目以及评分的项目数量的不同导致两个用户之间的相对相似度也是不对等的. 例如用户 A 购买了项目 a、b、c、d、e，用户 B 购买了 a、b、e、f，用户 A 与用户 B 的相似度与用户 B 与用户 A 的相似度明显不一样，其非对称相似度示例见表 1.

表 1　非对称相似度示例

Tab. 1　The example of asymmetric similarity

用户	购买项目	相同项目比例
A	a,b,c,d,e,	2/5
B	a,c,f	2/3

因此，引入式(7)计算非对称相似度.

$$Sim(u,v) = \frac{|I_u \cap I_v|}{I_u} \tag{7}$$

而该非对称相似性度量计算为两个用户评分的项目之间的比率，因此，将该公式加入一个 Sorenson 指数，得到式(8).

$$Sim(u,v) = \frac{|I_u \cap I_v|}{I_u} \cdot 2 \cdot \frac{|I_u \cap I_v|}{|I_u| + |I_v|} \tag{8}$$

由于该非对称相似性只考虑了用户总体的相似性，因此将该相似性度量与余弦相似度的实际公式结合，得到式(9).

$$ACOS(u,v) = \frac{\vec{r}_u \cdot \vec{r}_v}{\| \vec{r}_u \| \cdot \| \vec{r}_v \|} \cdot \frac{|I_u \cap I_v|}{I_u} \times 2 \cdot \frac{|I_u \cap I_v|}{|I_u| + |I_v|} \tag{9}$$

将非对称相似度与余弦相似度相结合称为非对称余弦相似度($ACOS$). 这种相似度计算模式不仅区分了 $Sim(u,v)$ 和 $Sim(v,u)$，还能够为用户提供更好的推荐.

结合上节提到的矩阵分解，将 $ACOS(u,v)$ 作为权重 ω_{uv} 表示用户 u 与用户 v 之间的评分关系，进一步提高推荐的准确性，因此，对目标函数更新如式(10).

$$G = \sum (r_{ui} - \bar{r} - b_u - b_i - \sum_{v \in R_k(u)} (r_{vi} - b_{vi}) \times \omega_{uv} - (\sum_{f=1}^{F} p_{fu}^T \times q_{fi}))^2 + \beta(b_u^2 + b_i^2 + \sum_{v \in R_k(u)} \omega_{uv} + \| p_u \|^2 + \| q_i \|^2) \tag{10}$$

其中，$R_k(u)$ 表示用户 u 的邻居集合的前 k 个，由上述的非对称相似度计算得到，$(r_{vi} - b_{vi})$ 表示 ω_{uv} 的系数，当相关系数 ω_{uv} 值和 $(r_{vi} - b_{vi})$ 的值较高时，意味着预测值低于实际值，则将 $(r_{vi} - b_{vi}) \times \omega_{uv}$ 作为基础偏移量的偏移量，调整参数.

为了最小化目标函数 G，需要对参量赋初值，首先初始化 p_u 和 q_i，随机对 p_u 和 q_i 中的 f 维向量进行随机赋值. 得到初始特征矩阵之后，对用户偏好 b_u 赋初值，为用户 u 评分的所有项目的评分均值，项目偏好 b_i 为所有用户对项目 i 评分的均值，通过应用随机梯度下降法，对评分数据做迭代更新.

$$b_u \leftarrow b_u + \alpha(e_{ui} + \beta b_u) \tag{11}$$

$$b_i \leftarrow b_i + \alpha(e_{ui} + \beta b_i) \tag{12}$$

$$\omega_{uv} \leftarrow \omega_{uv} + \alpha(e_{ui}(r_{vi} - b_{vi}) - \beta\omega_{uv}) \tag{13}$$

$$p_{fu} \leftarrow p_{fu} + \alpha(e_{ui} \times q_{fi} - \beta p_{fu}) \tag{14}$$

$$q_{fi} \leftarrow q_{fi} + \alpha(e_{ui} \times p_{fu} - \beta q_{fi}) \tag{15}$$

其中，e_{ui} 为 $r_{ui} - \hat{r}_{ui}$ 的误差；α 为梯度下降的步长；β 为惩罚因子. 在经过迭代更新之后，目标函数 G 小于某一阈值，得到 P，Q 两个矩阵，就是所需的分解矩阵.

在训练集中应用上述的迭代过程，得到用户和项目的偏移量 b_u 和 b_i，以及它们的向量表示 p_u 和 q_i. 这样通过奇异值矩阵分解，降低了时间复杂度，减少了空间的消耗，结合用户的非对称相似性产生推荐，有效的提高了推荐精度.

4 仿真结果分析

4.1 数据处理

仿真采用的是 GroupLens 项目研究组提供的 MovieLens 数据集(http://grouplens.org/datasets/movielens/)对算法进行评估,其中包含 943 个用户对 1682 部电影的 100000 条评分记录(可以计算出该数据集的稀疏度为$(943\times1682-1.0\times10^5)/(943\times1682)=93.695\%$),每个用户至少有 20 条评分记录,评分范围为 1(非常差) ~ 5(非常好).数据集中还包括用户和电影的基本信息,如用户职业、年龄,电影类型、上映时间等.实验之前需要对数据集进行划分.本文实验将数据集的 80% 作为训练集,剩余的 20%作为测试集.

4.2 度量方法

采用度量标准为平均绝对误差(Mean Absolute Error, MAE)以及标准误差(Root Mean Square Error,RMSE).平均绝对误差的指标是用来衡量统计的准确性和比较.它是用来衡量预测的用户评分与实际用户评分之间的误差,*MAE* 值越小,推荐准确度越高,计算方法如下式.

$$MAE=\frac{\sum_{i=1}^{n}|r_{ui}-\hat{r}_{ui}|}{n} \tag{16}$$

标准误差及均方根误差,*RMSE* 值越小,表示推荐效果越好,计算方法如下式

$$RMSE=\sqrt{\frac{\sum_{i=1}^{n}(r_{ui}-\hat{r}_{ui})^2}{n}} \tag{17}$$

4.3 仿真结果及分析

设计了两组实验,对 Movielens 数据集中的 7 个小数据集分别进行实验比较;将 Movielens-100k 的 7 个小数据集总合成一个数据集,将改进的算法 AMCF 与文献[16]中的算法以及 cosine 算法进行比较.

实验 1 将 Movielens 的 7 个小数据集 u_1,u_2,u_3,u_4,u_5,u_a,u_b 分别进行实验.其中,$u_1\sim u_5$ 数据集分别将用户数据集分为 80%的训练集和 20%测试集,且数据不相交,用于交叉实验;u_a 和 u_b 的测试集(ua.test 和 ub.test 不相交)为用户数据集中评分了 10 部电影的用户,去掉这些用户的其他数据为训练集.实验结果如图 1 和图 2 所示.

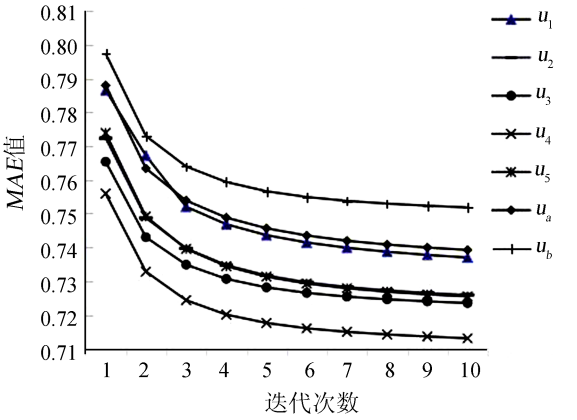

图 1 不同数据集迭代次数对 *MAE* 值的影响

Fig. 1 The impact of the number of iterations on *MAE* of different data sets

从图 1 中可以看出,算法在数据集 u_4 上效果最好,误差值最小,迭代完成后的 *MAE* 值约为 0.713,在数据集 u_b 上效果最差,误差值偏高,迭代完成后的 *MAE* 值约为 0.751.从图 1 中曲线走向可以看出,随着迭代次数的增加,误差值减小,并且减小的速度变慢,由此可以得出,误差缩小到一定值就不会再减小.

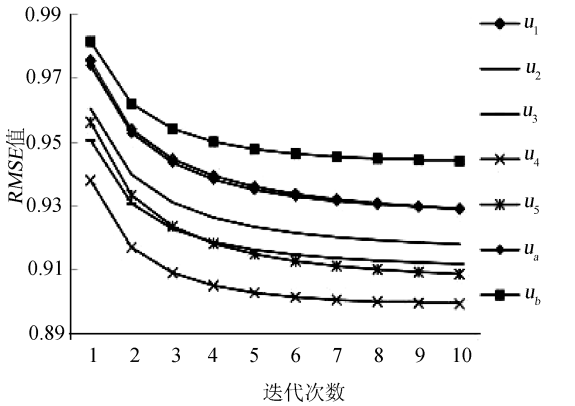

图 2 不同数据集迭代次数对 *RMSE* 值的影响

Fig. 2 The impact of the number of iterations on *RMSE* of different data sets

由图 2 可以看出,随着迭代次数的增加,*RMSE* 值会逐渐缩小,误差在缩小到一定值之后就不再减小,当邻居个数属于(45,50)区间时,误差最小,推荐精度最高.

实验 2 分别采用本文方法,文献[17]方法以及余弦相似度算法在 Movielens-100 k 数据集上进行实验,实验结果如图 3 和图 4 所示.

在图 3 中,通过对比实验分析,本文的算法明显优于文献[17]改进的 cosine 算法,有效地提高了推荐精度.

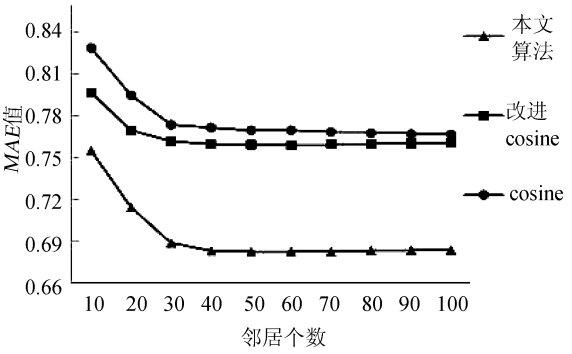

图 3　与改进算法 1 *MAE* 值对比图

Fig. 3　The comparison results on *MAE* with the improved algorithm 1

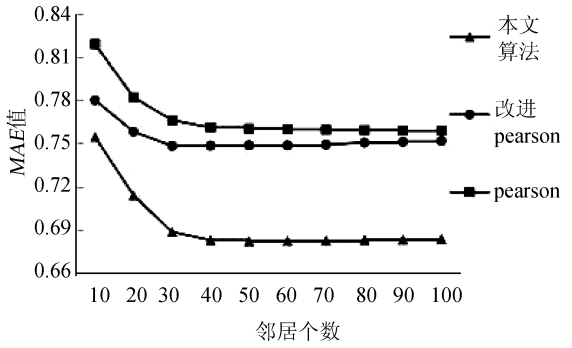

图 4　与改进算法 2 *MAE* 值对比图

Fig. 4　The comparison results on *MAE* with the improved algorithm 2

图 4 中实验结果表明，改进的 pearson 算法 *MAE* 值约为 0.749 左右，而本文算法 *MAE* 值趋于稳定之后一直保持在 0.682 左右，且各算法的 *MAE* 值随着邻居个数的增加而减小，但当邻居个数大于 50 时，误差不减反增，因此最佳邻居个数在 40～50 之间.

5　结　论

协同过滤推荐算法是推荐系统中热门的研究课题. 本文针对用户偏好关系不准确和数据稀疏提出了基于非对称用户相似度与矩阵分解的推荐算法，在传统的协同过滤算法的基础上，根据用户的潜在特征向量计算用户间的相似度，并利用矩阵分解进行降维，降低计算所需的时间空间，经试验表明. 该算法有效地利用了用户的评分偏好以及用户间关系，能够有效对矩阵进行降维和判断用户相似度，提高了预测的准确性. 但在本文算法中并未对实验数据做过多处理，下一步会考虑对用户数据进行聚类.

参考文献：

[1] Sun Z, Han L, Huang W, *et al*. Recommender systems based on social networks[J]. Syst Softw, 2015, 99: 109.

[2] Borràs J, Moreno A, Valls A. Intelligent tourism recommender systems: A survey[J]. Expert Syst Appl, 2014, 41: 7370.

[3] Dehghani Z, Reza S, Salwah S, *et al*. A systematic review of scholar context-aware recommender systems[J]. Expert Syst Appl, 2015, 42: 1743.

[4] 刘汉清，朱敏，苏亚博，等. 一种考虑用户兴趣转移特征的协同预测模型[J]. 四川大学学报：自然科学版，2016，53：548.

[5] 王永，万潇逸，陶娅芝，等. 基于 K-medoids 项目聚类的协同过滤推荐算法[J]. 重庆邮电大学学报：自然科学版，2017，29：521.

[6] Paterek A. Improving regularized singular value decomposition for collaborative filtering[C]//Proc KDD Cup Workshop at SIGKDD'07, 13th ACM Int Conf on Knowledge Discovery and Data Mining, 2007.

[7] 冷亚军，陆青，梁昌勇. 协同过滤推荐技术综述[J]. 模式识别与人工智能，2014，27：720.

[8] 孟祥武，纪威宇，张玉洁. 大数据环境下的推荐系统[J]. 北京邮电大学学报，2015，38：1.

[9] 彭石，周志彬，王国军. 基于评分矩阵预填充的协同过滤算法[J]. 计算机工程，2013，39：175.

[10] 冷亚军，梁昌勇，丁勇，等. 协同过滤中一种有效的最近邻选择方法[J]. 模式识别与人工智能，2013，26：968.

[11] Mnih A, Salakhutdinov R. Probabilistic matrix factorization[C]//Advances in neural information processing systems. 2007.

[12] 印鉴，王智圣，李琪，等. 基于大规模隐式反馈的个性化推荐[J]. 软件学报，2014，25：1953.

[13] Chen T, Zheng Z, Lu Q, *et al*. Informative ensemble of multi-resolution dynamic factorization models[C]// In KDD-Cup Workshop, [s. l.]: [s. n.], 2011.

[14] 张鹏，葛小青. 融合标签相似度的 k 近邻 Slope One 算法[J]. 重庆邮电大学学报：自然科学版，2016，28：518.

[15] 任看看，钱雪忠. 协同过滤算法中的用户相似性度量方法研究[J]. 计算机工程，2015，41：18.

[16] Pirasteh P, Hwang D, Jung J J. Exploiting matrix factorization to asymmetric user similarities in recommendation systems[J]. Knowledge-Based Syst, 2015, 83: 51.

[17] 王威，郑骏. 基于用户相似度的协同过滤算法改进[J]. 华东师范大学学报：自然科学版，2016，2016：60.