协同过滤推荐算法的研究与改进

范 虎[1],花伟伟[2]
(1. 安徽理工大学 计算机科学与工程学院,安徽 淮南 232001;
2. 安徽省淮南市田家庵区检察院,安徽 淮南 232001)

摘 要:传统的基于用户的协同过滤推荐算法在计算用户间相似性时依赖于用户-项目评分矩阵,但在实际的商业系统中,用户参与的评价往往非常少,这样计算出的相似性精确度通常很低。文中提出结合用户相似性和基于项目分类特征的相似性计算方法,计算用户间的相似性,形成目标用户的近邻集合,完成向目标用户的推荐。文中在 MovieLens 数据集上的实验结果表明,相对于 Pearson 相似性的协同过滤推荐算法,文中提出的改进算法在推荐质量方面有明显提高。

关键词:基于用户;协同过滤;推荐系统;项目分类;相似性计算
中图分类号:P315.69 **文献标识码**:A **文章编号**:1673-629X(2013)09-0066-04
doi:10.3969/j.issn.1673-629X.2013.09.017

Research and Improvement of Collaborative Filtering Recommendation Algorithm

FAN Hu[1], HUA Wei-wei[2]
(1. College of Computer Science and Engineering, Anhui University of Technology, Huainan 232001, China;
2. Tianjia'an District Prosecutor, Huainan 232001, China)

Abstract: The traditional user-based collaborative filtering algorithm calculates users' similarity according to user-item rating matrix, but in real business system, the user-ratings data is very sparse, so the calculation accuracy is very low. The calculation method mixing user similarity and project classification features based similarity is proposed for similarity calculation between users, get target user's close neighbor set, calculate recommended results. The experimental results on the MovieLens data set show that, compared with the Pearson similar collaborative filtering algorithm, the above improved algorithm raises the recommendation quality significantly.

Key words: user-based; collaborative filtering; recommendation system; project classification; similar calculations

0 引 言

随着互联网和电子商务的发展,协同过滤推荐[1]技术越来越多地在互联网和社交网络中使用。它能帮助用户更快地找到用户所需要的信息,在实际的商业应用中,推荐系统根据海量数据挖掘[2]用户的潜在需求并主动向用户做出推荐,让用户能方便快捷地找到自己所需要的东西。基于用户的协同过滤算法首先根据用户-项目评分矩阵计算用户的相似性[3],其次根据用户间的相似度选取目标用户的相似用户集,然后根据相似用户集来预测目标用户对项目的偏好程度,最后根据预测值来对目标用户实施推荐。协同过滤算法必须依靠用户对项目的评分,如果项目得到的用户评分很稀少[4-5],那么根据较少的评分计算出的相似用户的精度也很低,在实际的商业应用中,往往只有少数的用户会参与评价,而项目总数又是巨大的,数据的稀疏性降低了相似性计算的精度,进而导致推荐效果也很不理想。

针对传统协同过滤推荐算法的不足,文中在协同过滤推荐算法的相似度计算这一环节,引进基于项目分类的计算用户间相似性[6]的方法,提高用户相似度计算的精确度,最后通过实验对比来验证改进算法对推荐结果的改善。

收稿日期:2012-11-23 **修回日期**:2013-02-25 **网络出版时间**:2013-05-09
基金项目:国家自然科学基金资助项目(61170060);安徽省自然科学基金(11040606M135);安徽省高等学校自然科学基金重点项目(KJ2011A083)
作者简介:范 虎(1986-),男,硕士,研究方向为网络监控。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20130509.1057.015.html

1　基于用户的相似性计算方法

推荐系统首先从网站获取大量用户对项目的反馈信息,然后对数据进行预处理,产生一个如表 1 所示的用户 – 项目评分矩阵 $\boldsymbol{R}$,行为项目,列为用户,$R_{i,j}$ 为用户集合中第 i 个用户对项目集合第 j 个项目的评分。

表 1　用户–项目评分矩阵

	item_1	…	item_i	…	item_n
user_1	$R_{1,1}$	…	$R_{1,i}$	…	$R_{1,n}$
…	…	…	…	…	…
user_i	$R_{i,1}$	…	$R_{i,i}$	…	$R_{i,n}$
…	…	…	…	…	…
user_m	$R_{m,1}$	…	$R_{m,i}$	…	$R_{m,n}$

1.1　用户的相似性计算

如表 1 所示,在用户–项目评分矩阵中,两个行向量可以计算用户间的相似性,两个列向量可以反映项目间的相似性。夹角越小,用户/项目间的相似度越大。下面是几种常用的相似性计算方法:

(1)余弦相似性(Cosine Similarity)[7]。

余弦相似性和计算向量间的相似性是等同的。首先计算两个用户评分向量的夹角的大小,计算的结果越小表示用户相似度越大,计算方法如公式(1)所示:

$$\text{sim}(u,v)=\cos(\vec{u},\vec{v})=\frac{\sum_{i\in I}R_{u,i}R_{v,i}}{\sqrt{\sum_{i\in I}(R_{u,i})^2}\sqrt{\sum_{i\in I}(R_{v,i})^2}} \tag{1}$$

$\vec{u},\vec{v}$ 为待比较的两个用户 u 和 v 的评分向量,I 为所有项目的集合,u 对项目 i 的评分大小记为 $R_{u,i}$,v 对项目 i 的评分大小记为 $R_{v,i}$。

(2)皮尔逊(Pearson)相关性[7]。

$$\text{sim}_p(u,v)=\frac{\sum_{i\in I_{uv}}(R_{u,i}-\vec{R}_u)(R_{v,i}-\vec{R}_v)}{\sqrt{\sum_{i\in I_{uv}}(R_{u,i}-\vec{R}_u)^2}\sqrt{\sum_{i\in I}(R_{v,i}-\vec{R}_v)^2}} \tag{2}$$

I_{uv} 为用户 u 和 v 均有过评分记录的项目集合,u 和 v 各自在所评项目上的评分均值为 $\vec{R}_u$ 和 $\vec{R}_v$,由于不同用户的评分尺度不一样,通过引入评分均值这个参数,有利于减少用户评分尺度不一致造成的误差。

1.2　最近邻居的产生

利用上述相似性计算方法算得的用户相似性结果可以用一个 $m\times m$ 的二维矩阵 $\boldsymbol{S}(m,m)$ 表示,$\boldsymbol{S}$ 中的行向量和列向量均表示用户,元素值表示对应行的用户和列的用户的相似度,通过矩阵 $\boldsymbol{S}$ 产生目标用户的近似邻居集通常有两种方法:

(1)通过一个预先设定的阈值来选取目标用户的近似邻居。

$$M(u)=\{u_n \mid \text{Sim}(u_n,u)>\partial, u_n\neq u\} \tag{3}$$

$M(u)$ 表示用户 u 的相似用户集,当用户 u_n 和用户 u 的相似性超过预先设定的阈值 ∂ 时,就将 u_n 作为 u 的近似邻居,这样产生的相似邻居数目是不确定的,但相似度较为可靠。

(2)k 邻居法[8]:选取目标用户的近似邻居,将目标用户与除目标本身外的所有用户比较,然后选取与目标用户最相似的前 k 个用户作为目标用户的近似邻居,这样产生的相似邻居数目是确定的,但相似度不能很好地保证。

2　基于项目分类的相似性计算

当数据比较稀疏时,可以通过用户对某一类项目的喜好程度来衡量项目间的相似性[9-11]。以电影为例,假如两人都喜欢看喜剧片,但两个人看过的具体喜剧片并不同,结果他们的共同评价项目并不多。如果用 Pearson 的相似性计算方法,这两个用户的相似度应该会很低,但如果用分类的相似性来度量,他们俩就可能是较为相似的用户。

将整个项目集按不同的属性分成主要的几大类[12],项目类别可以用一个集合来表示 $\{f_1,f_2,\cdots,f_n\}$,对于某个项目来说,它可能同时分属于几个不同的类型,一部电影就可以既是爱情片又是动作片,在实际研究中,将每个类型属性分成不同的等级(例如 1 到 5 级),这样每个项目包含不同类型的属性等级是不同的,如一部电影包含动作片属性的等级可能为 4,而包含爱情片的等级大小为 2,这样对于 n 个项目来说,就可以建立项目–类别属性矩阵 $\boldsymbol{A}$,$\boldsymbol{A}$ 中的元素 $A(i,j)$ 表示一个项目 i 具有的属性 j 的等级大小(见表 2)。

表 2　项目–类别属性矩阵

	f_1	…	f_i	…	f_k
item_1	$A_{1,1}$	…	$A_{1,i}$	…	$A_{1,k}$
…	…	…	…	…	…
item_i	$A_{i,1}$	…	$A_{i,i}$	…	$A_{i,k}$
…	…	…	…	…	…
item_n	$A_{n,1}$	…	$A_{n,i}$	…	$A_{n,k}$

用户 u 对某类型项目 i 的偏好程度 $I_{u,i}$ 可以用如下公式(4)计算:

$$I_{u,i}=\text{Score}_{u,i}/\text{Score}_u \tag{4}$$

$\text{Score}_{u,i}$ 表示用户 u 所有喜好项目包含属性 i 的大小之和,Score_u 表示用户所有喜好项目包含的所有属性等级之和。假设项目共有 k 个类别,可以通过公式(4)计算出用户–类别偏好矩阵 $\boldsymbol{D}$,$\boldsymbol{D}$ 是一个二维矩阵,$D(i,j)=2$ 表示用户 i 对属性 j 的偏好程度为 2(见表 3)。

表 3 用户–类别属性矩阵

	f_1	…	f_i	…	f_k
user_1	$D_{1,1}$	…	$D_{1,i}$	…	$D_{1,k}$
…	…	…	…	…	…
user_i	$D_{i,1}$	…	$D_{i,i}$	…	$D_{i,k}$
…	…	…	…	…	…
user_m	$D_{m,1}$	…	$D_{m,i}$	…	$D_{m,k}$

用户相似性计算：

分类的相似性计算方法跟传统的基于用户–项目评分矩阵[13]的方法相似，也是基于向量(Vector)的。通过表 3 可以取得用户类别偏好向量 $\boldsymbol{D}_u=(D_{u,1},D_{u,2},\cdots,D_{u,k})$，这样两个用户 u 和 v 之间的相似性就可以按照公式(5)计算：

$$\text{Sim}_D(u,v)=\frac{\sum_{i=1}^{k}D_{u,i}D_{v,i}}{\sqrt{\sum_{i=1}^{k}(D_{u,i})^2}\sqrt{\sum_{i=1}^{k}(D_{v,i})^2}} \tag{5}$$

公式(5)和余弦相似性类似，可以计算出一个 $m\times m$ 的二维矩阵来表示用户之间的相似性，行向量和列向量均表示用户。

3 产生推荐结果

3.1 混合的相似度计算

基于 Pearson 的用户相似度计算[14]和基于项目分类的相似度计算是从不同的方面体现了用户间的相似程度。

第一种方法在数据比较完整时计算更精确，第二种方法在数据严重稀疏时也能较好地反映用户间的相似程度，根据公式(2)和公式(5)，通过设置一个权值 a 来加权两种相似度计算，如公式(6)：

$$\text{Sim}(u,v)=a\frac{\sum_{i\in I_{uv}}(R_{u,i}-\vec{R}_u)(R_{v,i}-\vec{R}_v)}{\sqrt{\sum_{i\in I_{uv}}(R_{u,i}-\vec{R}_u)^2}\sqrt{\sum_{i\in I}(R_{v,i}-\vec{R}_v)^2}}+(1-a)\frac{\sum_{i=1}^{k}D_{u,i}D_{v,i}}{\sqrt{\sum_{i=1}^{k}(D_{u,i})^2}\sqrt{\sum_{i=1}^{k}(D_{v,i})^2}} \tag{6}$$

公式(6)中 a 的取值范围为 0 到 1，当 $a=1$ 时，公式(6)就等价于 Pearson 相似性计算方法，当 $a=0$ 时，公式(6)就等价于基于项目分类的相似性计算方法。使用公式(6)可以计算出用户相似性矩阵 M，如表 4 所示，行向量和列向量均表示用户，矩阵元素表示对应行的用户和列的用户的相似度。

3.2 产生推荐

对于一个目标用户 u，根据表 4 选取与 u 最相似

表 4 用户相似性矩阵

	user_1	…	user_i	…	user_m
user_1	$M_{1,1}$	…	$M_{1,i}$	…	$M_{1,m}$
…	…	…	…	…	…
user_i	$M_{i,1}$	…	$M_{i,i}$	…	$M_{i,m}$
…	…	…	…	…	…
user_m	$M_{m,1}$	…	$M_{m,i}$	…	$M_{m,m}$

的 k 个用户组成用户 u 的近邻集合，然后根据 u 的相似用户对目标项目 i 的评分来预测 u 对 i 的评分，计算方法如下：

$$W(u,i)=\bar{u}+\frac{\sum_{v\in\text{KNB}}\text{sim}(u,v)(R_{v,i}-\bar{v})}{\sum_{v\in\text{KNB}}\text{sim}(u,v)} \tag{7}$$

公式(7)中，$\bar{u}$ 和 $\bar{v}$ 分别表示用户 u 和 v 在各自所有打分项目上的评分均值，$R_{v,i}$ 表示 v 对项目 i 的评分值。

通过公式(7)可以计算出用户 u 对所有项目的评分预测值，然后选取预测值最高的项目对用户 u 实施推荐。

4 实验结果及分析

4.1 数据集

实验采用 MovieLens 站点提供的数据集，MovieLens 是一个非盈利性的、以研究为目的的实验性站点。用户对该站点电影打分后，数据会被记录下来反映用户的偏好，随着用户对电影打分的增多，系统对用户的推荐也越来越准确。

该站点提供三个不同规模的数据集，从中选取一个包含 100 000 条评分的数据集，该数据集是通过 943 个用户 1 682 部电影进行评分产生的，每个用户对 20 部以上的电影做过评价，将1 682部电影分成 18 个不同的类型：Action，Adventure，Animation，Children’s，Comedy，Documentary，Crime，Drama，Fantasy，Film–Noir，Horror，Musical，Mystery，Romance，Sci–Fi，Thriller，War，Western，取数据集中的 80% 作为训练集，20% 作为预测集，数据的稀疏程度为：

$$1-100\,000/(943*1\,682)=0.936\,95$$

4.2 评价标准

实验采用平均绝对偏差(Mean Absolute Error，MAE)[5]来衡量算法的优劣，MAE 表示用户对项目预测评分与实际评分之间的偏差，计算得到的 MAE 越小，表明算法的准确度越高。

假设测试集中共用 r 条数据，对这 r 条数据的实际评分值为 $\{q_1,q_2,\cdots,q_r\}$，通过计算得到的预测值为 $\{p_1,p_2,\cdots,p_r\}$，那么 MAE 的计算如公式(8)所示：

$$\mathrm{MAE}=\frac{\sum_{i=1}^{r}|q_i-p_i|}{r} \tag{8}$$

4.3　实验结果及分析

首先确定公式(6)中权值 a 的合理取值,因为 a 的取值区间为[0,1],设置 a 的取值从 0 到 1,每次增加 0.1,计算不同的 a 对应的 MAE 大小,从而确定 a 的合理取值,实验结果如图 1 所示。从图 1 中可以看出,当 $a=0.6$ 时,MAE 接近最小值,因此在后面的试验中,选择 $a=0.6$。

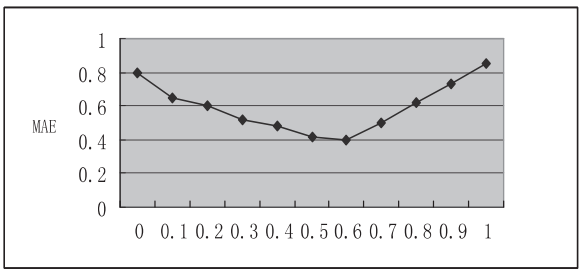

图 1　不同权值对应的 MAE 值的统计图

接下来验证文中提到的基于用户和项目分类的协同过滤算法的有效性,将其与传统的基于用户的协同过滤推荐算法进行比较,采用相同的数据集,计算出两种方法在近邻集合数量不同的情况下 MAE 的区别,这里近邻集合 KNB 大小分别为 10,20,30,40,50,60,实验结果如图 2 所示。

从图 2 中可以看出,采取文中提出的混合的协同过滤算法,在不同近邻集合的地方均取得了比基于 Pearson 相关性的协同过滤算法更低的误差,提高了推荐精度。

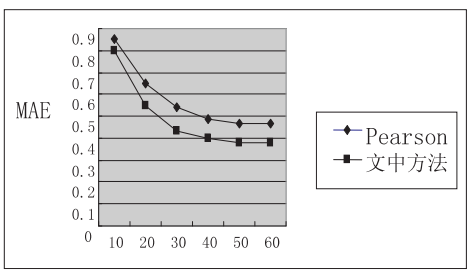

图 2　Pearson 度量法和文中的方法对应 MAE 值的对比图

5　结束语

基于用户的协同过滤推荐算法依赖于用户对项目的评分数据来计算推荐结果,而实际的推荐系统往往面临数据严重稀缺的问题,从而导致推荐结果不准确。文中在传统的 Pearson 相似性算法的基础上,提出了基于项目分类的相似性度量方法,将项目按照不同的属性分类,然后计算用户对不同属性的偏好程度并根据用户对不同属性的偏好来计算用户的相似性,最后结合两种相似性计算方法,用一个权值控制两种方法的重要性。

通过实验表明,在数据处于相同稀疏程度下,该方法比单纯的基于 Pearson 相似性的协同过滤技术取得了更高的推荐精度,在一定程度上缓解了数据稀疏性的问题。文中下一步的工作是研究如何对项目进行合理的分类以提高算法的推荐精度。

参考文献:

[1] 马宏伟,张光卫,李　鹏. 协同过滤推荐算法综述[J]. 小型微型计算机系统,2009,30(7):1282-1288.

[2] 陶启萍. 基于 Web 数据挖掘的电子商务推荐系统研究[D]. 阜新:辽宁工程技术大学,2005.

[3] 张海鹏,李烈彪,李　仙,等. 基于项目分类预测的协同过滤推荐算法[J]. 情报学报,2008,27(2):218-223.

[4] 张海燕,丁　峰,姜丽红. 基于模糊聚类的协同过滤推荐方法[J]. 计算机仿真,2005,22(8):144-147.

[5] 高凤荣,杜小勇,王　珊. 一种基于稀疏矩阵划分的个性化推荐算法[J]. 微电子学与计算机,2004,21(2):58-62.

[6] 孙　多. 基于兴趣度的聚类协同过滤推荐系统的设计[J]. 安徽大学学报(自然科学版),2007,31(5):19-22.

[7] 夏治勇. 个性化推荐技术中的协同过滤算法研究[D]. 青岛:中国海洋大学,2011.

[8] 罗　辛,欧阳元新,熊　璋,等. 通过相似度支持度优化基于 K 近邻的协同过滤算法[J]. 计算机学报,2010,33(8):1437-1445.

[9] Shapira B, Taieb-Maimon M. Study of the Usefulness of Known and New Implicit Indicators and Their Optimal Combination for Accurate Inference of Users Interests[C]//Proceeding of ACM Symposium on Applied Computing. [s. l.]: [s. n.],2006:18-19.

[10] Resnick P, Iakovou N, Sushak M, et al. GroupLens: An open architecture for collaborative filtering of netnews[C]//Proc. of 1994 Computer Supported Cooperative Work Conf.. Chapel Hill:[s. n.],1994:175-186.

[11] Hill W, Stead L, Rosenstein M, et al. Recommending and evaluating choices in a virtual community of use[C]//Proc. of Conf. on Human Factors in Computing Systems. Denver:[s. n.],1995:194-201.

[12] 邓爱林,左子叶,朱扬勇. 基于项目聚类的协同过滤推荐算法[J]. 小型微型计算机系统,2004,25(9):1665-1670.

[13] Goldberg D, Nichols D, Oki B M, et al. Using Collaborative Filtering to Weave an Information Tapestry[J]. Communications of the ACM,1992,35(12):61-70.

[14] Sarwar B M, Karypis G, Konstan J A, et al. Application of dimensionality reduction in recommender system-A case study[R]. Minneapolis, USA: University of Minnesota,2000.

协同过滤推荐算法的研究与改进

作者： 范虎， 花伟伟， FAN Hu， HUA Wei-wei

作者单位： 范虎,FAN Hu(安徽理工大学 计算机科学与工程学院,安徽 淮南,232001)， 花伟伟,HUA Wei-wei(安徽省淮南市田家庵区检察院,安徽 淮南,232001)

刊名： 计算机技术与发展

ISTIC

英文刊名： Computer Technology and Development

年，卷(期)： 2013(9)

参考文献(14条)

1. 马宏伟;张光卫;李鹏 协同过滤推荐算法综述[期刊论文]-小型微型计算机系统 2009(07)

2. 陶启萍 基于 Web 数据挖掘的电子商务推荐系统研究 2005

3. 张海鹏;李烈彪;李仙 基于项目分类预测的协同过滤推荐算法[期刊论文]-情报学报 2008(02)

4. 张海燕;丁峰;姜丽红 基于模糊聚类的协同过滤推荐方法[期刊论文]-计算机仿真 2005(08)

5. 高凤荣;杜小勇;王珊 一种基于稀疏矩阵划分的个性化推荐算法[期刊论文]-微电子学与计算机 2004(02)

6. 孙多 基于兴趣度的聚类协同过滤推荐系统的设计[期刊论文]-安徽大学学报（自然科学版） 2007(05)

7. 夏治勇 个性化推荐技术中的协同过滤算法研究 2011

8. 罗辛;欧阳元新;熊璋 通过相似度支持度优化基于K近邻的协同过滤算法[期刊论文]-计算机学报 2010(08)

9. Shapira B;Taieb-Maimon M Study of the Usefulness of Known and New Implicit Indicators and Their Optimal Combi-nation for Accurate Inference of Users Interests 2006

10. Resnick P;Iakovou N;Sushak M GroupLens:An open architecture for collaborative filtering of netnews 1994

11. Hill W;Stead L;Rosenstein M Recommending and eval-uating choices in a virtual community of use 1995

12. 邓爱林;左子叶;朱扬勇 基于项目聚类的协同过滤推荐算法[期刊论文]-小型微型计算机系统 2004(09)

13. Goldberg D;Nichols D;Oki B M Using Collaborative Filtering to Weave an Information Tapestry 1992(12)

14. Sarwar B M;Karypis G;Konstan J A Application of di-mensionality reduction in recommender system-A case study 2000