面向用户兴趣密度分布的协同过滤推荐算法①

毕孝儒

(四川外国语大学 重庆南方翻译学院 管理学院, 重庆 401120)

摘 要: 针对评分数据稀疏和单一评分相似性计算不准确导致推荐质量不高的问题, 提出一种面向用户兴趣密度分布的协同过滤推荐算法. 在计算项目类别相似度的同时, 引入类别的信息熵以确定项目之间距离, 在此基础上采用 Parzen 窗估计方法获取用户在整个项目空间上的兴趣密度分布, 最后结合用户属性差异性和兴趣密度之间相对熵以确定目标用户的最近邻居用户集. 实验结果表明, 该算法在避免数据填充所引入误差的同时, 有效提升数据稀疏情况下的推荐质量.

关键词:协同过滤; Parzen 窗估计; 相对熵; 用户属性

Collaboration Filtering Recommendation Algorithm Faced Distribution of User Interest Density

BI Xiao-Ru

(School of Management, Chongqing Nanfang Translators College of University SISU, Chongqing 401120, China)

Abstract: Aiming to such the problems that sparse data and poor calculation of score similarity result in low quality of recommendation, a collaborative filtering recommendation algorithm based on distribution of user interest density is proposed in the paper. After calculating the similarity of items, classification and entropy are calculated to get finally similarity between two items. Parzen window estimation is applied to get user interest density distribution in total item space. Finally user's attribute similarity and relative entropy are used to determine nearest neighbour user set. Experimental result shows that the algorithm effectively raises recommendation quality of spare data while avoiding error of filling data.

Key words: collaborative filtering; Parzen window estimation; relative entropy; user's attributes

近年来, 随着互联网的发展, 网络数据也呈现指数增长, 致使用户无法有效地在海量的信息中获取对自己有用的信息. 因而各种形式的推荐系统应运而生. 其中, 协同过滤推荐是当前应用很广泛, 且很成功的一种推荐技术, 其基本思想是基于用户-项目评分数据集, 通过收集相似用户的兴趣信息进而对目标用户推荐, 但也存在数据稀疏性、冷启动和扩展性等问题[1]. 针对这些问题, 研究者提出了多种解决方案, 主要包括矩阵填充[1]和矩阵降维[2,3]两种技术. 矩阵填充技术包括默认的值[4]填充, 以及项目本身预测缺失[5]的评分. 该技术在填充缺失数据的同时也引入了新误差; 矩阵降维技术则采用 SVD 等矩阵分解技术将高维数据投影到低维空间以发现项目或用户之间相似性, 但其存在计算量大, 降维导致信息丢失等问题. 文献[6]将核方法用于推荐系统, 较好地解决了以上两种技术存在的缺陷. 但该方法存在以下不足: 第一、未考虑项目的类别权重对项目间距离计算的影响; 第二、在计算用户相似性时未引入用户属性间的差异性. 针对这一现状, 提出一种面向用户兴趣密度分布的协同过滤推荐算法(Collaborative Filtering Recommendation Algorithm Faced Distribution of User Interest Density, DUID-CF). 在计算项目类别相似度的同时, 引入类别的信息熵概念以计算项目之间距离, 并采用 Parzen 窗估计方法获取用户在整个项目空间上的兴趣密度分布, 最后结合用户属性差异性和兴趣密度之间相对熵以确定目标用户的最近邻居用户集. 实验结果表明, 该算

① 收稿时间:2014-03-25;收到修改稿时间:2014-04-23

法在避免数据填充所引入误差的同时, 有效提升数据稀疏情况下的推荐质量.

1 相关工作

基于用户的协同过滤算法主要分为三个步骤: 首先采用用户-项目评分矩阵表示用户评分数据; 其次运用某种相似性度量方法计算用户间相似性; 最后产生目标用户的的邻居集(K 个), 并将目标用户所感兴趣的项目通过一定推荐策略返回给用户.

1.1 用户评分数据表示

推荐系统中存储的用户评分数据中一般包括用户 id、项目 id 和用户对项目的评分信息. 设有 m 个用户和 n 个项目, $U=\{U_1,U_2,...,U_m\}$ 表示用户集, $I=\{I_1,I_2,...,I_n\}$ 表示项目集, 则用户评分数据可采用一个 $m\times n$ 阶的用户-项目评分矩阵 $R=\{r_{i,j}\}$ 表示.

1.2 相似性度量

具有代表性的相似性度量方法有Pearson相关系数和修正余弦相似性. 设 $I_{U_i}=\{h:h\in I,r_{i,h}\neq\varnothing\}$ 为用户 U_i 评过分的项目集合, $\overline{r_{i,*}}$ 为用户 U_i 产生的评分均值. 则Pearson相关系数计算用户 U_i 与 U_j 相似性方法如式(1)所示:

$$corr(U_i,U_j)=\frac{\sum_{k\in I_{U_i}\cap I_{U_j}}(r_{i,k}-\overline{r_{i,*}})(r_{j,k}-\overline{r_{j,*}})}{\sqrt{\sum_{k\in I_{U_i}\cap I_{U_j}}(r_{i,k}-\overline{r_{i,*}})^2}\sqrt{\sum_{k\in I_{U_i}\cap I_{U_j}}(r_{j,k}-\overline{r_{j,*}})^2}} \tag{1}$$

修正余弦相似性计算用户相似性方法如式(2)所示:

$$corr(U_i,U_j)=\frac{\sum_{k\in I_i\cap I_j}(r_{i,k}-\overline{r_{i,*}})(r_{j,k}-\overline{r_{j,*}})}{\sqrt{\sum_{k\in I_i}(r_{i,k}-\overline{r_{i,*}})^2}\sqrt{\sum_{k\in I_j}(r_{j,k}-\overline{r_{j,*}})^2}} \tag{2}$$

1.3 评分预测

根据用户间的相似度可以获取目标用户的最近邻居集合, 并将其相似性作为权重预测目标用户对未评分项目的评分, 故目标用户 u_t 对项目 i 的评分 $P_{u_t,i}$ 预测如式(3)所示:

$$P_{u_t,i}=\overline{r_t}+\frac{\sum_{V\in K_{u_t}}corr(U_t,V)(r_{v,i}-\overline{r_t})}{\sum_{V\in K_{u_t}}(|corr(U_t,V)|)} \tag{3}$$

式(3)中, $\overline{r_t}$ 为用户 u_t 的评分均值, K_{u_t} 为用户 u_t 的 K 最近邻居集.

2 面向用户兴趣密度分布的推荐算法

针对基于核方法协同过滤推荐算法的不足, DUID-CF 算法将项目类别权重引入到项目间距离度量中的同时, 结合用户属性间的差异性确定用户之间的相似性, 实现评分推荐. DUID-CF 算法主要分为三个步骤.

2.1 项目间距离的度量

考虑到传统项目相似度度量方法存在的不足, 文献[6]文中从项目类别角度给出了项目之间相似性的度量方法. 设项目类别矩阵 $A_{I,C}$, 其中 I 表示商品项, C 表示类别. 设项目类别 $C=\{c_1,c_2,...,c_n\}$ 表示所有项目组成的集合, 则项目类别矩阵 $A_{I,C}$ 为

$$A_{I,C}=\begin{bmatrix} I_1 & c_{1,1} & c_{1,2} & \cdots & c_{1,m} \\ I_2 & c_{2,1} & c_{2,2} & \cdots & c_{2,m} \\ & & & & \\ I_l & c_{l,1} & c_{l,2} & \cdots & c_{k,m} \end{bmatrix} \tag{4}$$

式(4)中, I_1~I_k 表示项目 id, $c_{k,m}$ 值为 0 或 1, 当其值为 1 时, 表示第 k 个项目属于第 m 类别; 当其值为 0 时, 表示第 k 个项目不属于第 m 类别. 则定义项目间的分类相似度[7]如式(5)所示:

$$sim_C(i,j)=\frac{|\sum_{r=1}^{m}c_{i,r}\ \&\&\ c_{j,r}|^2}{m\times|\sum_{r=1}^{m}c_{i,r}||c_{j,r}|} \tag{5}$$

式(5)中, m 为总的类别数目, $c_{i,r}\ \&\&\ c_{j,r}$ 表示逻辑与运算, 即项目 i,j 的同属于 r 类时值为 1; $c_{i,r}\|c_{j,r}$ 表示逻辑或运算, 即项目 i,j 的任一个属于 r 类时值为 1. 该方法虽较好地反映了项目间差异, 但其并未考虑每一类别权重大小的影响. 以下以表 1 的电影推荐系统为例予以解释.

表1　电影类别示例

	c_1	c_2	c_3	c_4	c_5
I_1	0	0	1	1	1
I_2	1	0	0	1	1
I_3	1	0	0	1	1
I_4	0	0	1	1	1
I_5	0	0	0	0	1
I_6	0	0	1	1	1

设电影类别为 $C=\{c_1,c_2,c_3,c_4,c_5\}$, 其中 c_1 代表动作片, c_2 代表喜剧片, c_3 代表恐怖片, c_4 代表记录片, c_5 代表冒险片, 一部电影可能同时属于几个类别. 由该表 1 可知, 6 部电影中同属于冒险片的电影均不属于喜剧片, 因而类别 c_2, c_5 对项目没有区分能力; 另外 6 部

电影中有一半属于恐怖片, 一半不属于恐怖片, 因而类别 c_3 对项目有很好的区分能力.

以上述分析为基础, 文中采用项目类别的信息熵来描述每个类别对项目的重要程度. 设类别 c_i 在整个项目空间取值为集合 $V(c_i)=\{v_{i1},v_{i2},...,v_{il}\}$, v_{il} 表示类别 c_i 的第 l 个项目取值, 则定义项目取值 v_{il} 出现的概率如式(6)所示[7]:

$$P(v_{il})=\frac{count(v_{il})}{total\_count} \tag{6}$$

式(6)中, $count(v_{il})$ 表示系统中类别 c_i 的值为 v_{il} 的项目总数量, $total\_count$ 表示所有项目的总数量, 则项目类别 c_i 信息熵如式(7)所示[7]:

$$H(c_i)=-\sum_{k=1}^{l}P(v_{ik})\ln(P(v_{ik})) \tag{7}$$

采用式(7)计算表2中项目类别 c_1~c_5 的信息熵分别为0.6365、0、0.6931、0.4506、0. 因而表1中类别 c_3 对项目相似度度量影响最大.

综合考虑式(5)、式(7), DUID-CF 算法给出了新的项目间相似性计算方法, 如式(8)所示:

$$sim_C^*(i,j)=\alpha\frac{|\sum_{r=1}^{m}(c_{i,r}\ \&\&c_{j,r})|^2}{m\times|\sum_{r=1}^{m}c_{i,r}||c_{j,r}|}+(1-\alpha)\frac{\sum_{x\in I_{U_i}\cap I_{U_j}}H(c_x)}{\sum_{r=1}^{m}H(c_r)} \tag{8}$$

式中, I_{U_i}、I_{U_i} 分别为用户 U_i、U_j 为评过分的项目集, $\alpha\in[0,1]$ 为比例因子. 该相似度定义一方面考虑到了2个项目所占类别中的重合比例, 另一方面页考虑了重合类别的信息熵(区分能力), 因而其对项目之间相似度描述更为全面. 式(9)给出了 2 个项目在项目空间上的距离度量.

$$d_{i,j}^*=1-sim_C^*(i,j) \tag{9}$$

由式(9)可知, 2 个项目相似性越大, 则其在项目空间的距离越小.

2.2 用户兴趣度估计

在传统的用户相似算法中, 仅仅考虑由共同评分的那些项目, 而实际上, 用户对于尚未评分的哪些项目也有自己的喜好. 因而若能估计用户在整个项目空间上的兴趣密度分布, 再计算两用户兴趣密度分布的相似性更为符合实际情况. 考虑到用户兴趣的多模性(即有多个局部极大值), 文献[6]采用统计学中的密度估计方法进行用户兴趣分布估计. 设 $x_1,x_2,...,x_n$ 是独立分布样本 $X_1,X_2,...,X_n$ 一组采样值, 则任意一个采样值 x 的密度函数 $f(x)$ 的核密度估计定义为

$$\overline{f}(x)=\frac{1}{nh}\sum_{i=1}^{n}k(\frac{\|x-x_i\|}{h}) \tag{10}$$

式(10)中, $k(\frac{\|x-x_i\|}{h})$ 为核函数, h 为核函数的窗宽. 常用的核函数有三角核函数、均匀核函数、高斯核函数等, 文中选用应用广泛的高斯函数作为核函数, 如式(11)所示:

$$k(z)=\frac{1}{\sqrt{2\pi}}\exp(-\frac{z^2}{2h^2}) \tag{11}$$

则用高斯核估计用户 u 兴趣分布 P_u 公式为

$$\overline{f_{P_u}}(k)=\frac{1}{|I_u|\times\sqrt{2\pi}h}\sum_{i\in I_u}R_{u,i}\times\exp(-\frac{(d_{i,j}^*)^2}{2h^2}) \tag{12}$$

2.3 用户相似性计算

考虑到KL散度(相对熵)是2个概率分布之间距离的非对称性度量, 文献[6]采用相对熵计算用户相似性. 设 P_{U_i}、P_{U_j} 分别为核密度估计方法得到的用户 U_i、U_j 兴趣密度分布, 则 U_i、U_j 散度定义为

$$D_{KL}(\mathrm{P}_{U_i}\|\mathrm{P}_{U_j})=\sum_{w=1}^{n}\mathrm{P}_{U_i}(w)\log\frac{\mathrm{P}_{U_i}(w)}{\mathrm{P}_{U_j}(w)} \tag{13}$$

上式定义中规定: $0\log(0/0)=0$, $0\log(0/\mathrm{P}_{U_j}(w))=0$, $0\log(\mathrm{P}_{U_i}(w)/0)=0$. 同时由于 KL 散度不具有对称性, 因而一般采用式(14)计算用户间的相似性:

$$sim_{KL}(U_i,U_j)=\frac{1}{2}(\mathrm{D}_{KL}(\mathrm{P}_{U_i}\|\mathrm{P}_{U_j})+\mathrm{D}_{KL}(\mathrm{P}_{U_j}\|\mathrm{P}_{U_i})) \tag{14}$$

式(14)虽然能够描述用户间的兴趣分布相似性, 但并未考虑用户每一属性取值对其兴趣分布的影响. 比如用户的性别、年龄、学历背景、职业等属性差异均会反映出其不同的兴趣偏好, 即属性取值相近的用户一般具有相似兴趣取向, 而对于属性取值相差甚远用户, 其兴趣偏好差异较大. 设矩阵 $A=\{a_{i,j}\}_{m\times l}$ 为用户属性, $a_{i,j}$ 为用户 U_i 的第 j 个属性的取值. 设 $a_{i,j}\Leftrightarrow a_{k,j}$ 为双向蕴涵运算, 即用户 U_i 与 U_j 的第 j 个属性相同时值为1, 否则为0. 则用户 U_i 与 U_j 的属性相似度表述为

$$sim_A(U_i,U_j)=\frac{\sum_{t=1}^{l}(a_{i,j}\Leftrightarrow a_{k,j})}{l} \tag{15}$$

上式表明两用户属性相似度取值在[0,1]之间, 即取值越接近1, 表明两用户属性相似度越高, 否则相似度越低. 以下通过表2以解释上式的具体运用. 为了表述方便, 将表中的年龄字段 a_4 取值离散化, 即用户年龄小于20岁, 其年龄属性取值为1, 用户年龄大于等于20且

小于30岁, 其属性取值为2等等. 因而可以得到用户 U_1 至 U_5 的离散化后年龄属性值为4,2,3,4,4. 设 U_5 为目标用户, 则依据式(15)可得用户 U_5 与 U_1, U_2, U_3, U_4 的相似度分别为0.5,0,0,0.75, 即 U_5 与 U_4 具有较高属性相似度, 而与 U_2, U_3 属性相似度最低.

表2　用户不同属性取值

	a_1	a_2	a_3	a_4
U_1	男	本科	教师	42
U_2	女	大专	程序员	23
U_3	女	本科	律师	35
U_4	男	硕士	科员	40
U_5	男	硕士	医生	45

根据上述分析, DUID-CF算法将用户属性相似度引入用户相似度度量中, 其改进后的用户相似性度量方法如式(16)所示.

$$sim(U_i,U_j)=\beta sim_{KL}(U_i,U_j)+(1-\beta)sim_A(U_i,U_j) \quad (16)$$

式中, $\beta\in[0,1]$ 为比例因子, 该相似度计算方法挖掘了用户属性之间相似度, 并结合相对熵, 通过适当调节 β 因子, 使用户相似度更能客观反映用户之间的兴趣偏好.

2.4 算法描述

输入: 用户-项目评分矩阵;

输出: 用户 U_i 对项目 I_j 的预测评分.

步骤1. 运用式(9)计算项目 I_j 与其他项目距离;

步骤2. 依据式(12)估计用户兴趣在项目空间上的分布;

步骤3. 重复步骤1、2, 估计所有用户兴趣分布;

步骤4. 依据式(16)计算2个用户间的相似性;

步骤5. 利用式(3)做出预测;

在算法步骤 1、步骤 4 中, 由于分别计算了项目类别信息熵和用户间的属性相似度, 因而与文献[6]算法比较, 其计算量较大. 因而如何降低 DUID-CF 算法耗费时间是今后研究重点.

3　实验与分析

3.1 用户实验数据集

实验采用 GroupLens 研究小组提供的 MovieLens 数据集(http://movieslens.umn.edu), 它包括 943 个用户对1 682个项目(影片)的10万条投票记录, 其稀疏等级是 0.9370. 其中, 用户属性有年龄、性别和职业三个; 电影类别分为科幻、冒险、动作、喜剧、恐怖等 18 个类别. 每一部电影分别属于 18 个类别中的 1 个或几个. 实验把数据集按 80%和 20%的比例划分为训练集和测试集.

3.2 评价指标

实验将正确率(*Accuracy_rate*)和平均绝对误差(*Mean Absolute Error*, *MAE*)作为算法性能评价标准. 正确率表示所有测试用例中预测分数与测试分数相等的用例所占比率. MAE通过计算预测的用户评分与实际的用户评分之间的偏差度量预测的准确性, MAE越小, 推荐质量越高. 设预测的用户评分集为 $\{p_1,p_2,...,p_n\}$, 对应实际评分集为 $\{q_1,q_2,...,q_n\}$, 则MAE计算公式如式(17)所示.

$$MAE=\frac{\sum_{i=1}^{n}|p_i-q_i|}{n} \quad (17)$$

3.3 实验环境

实验硬件环境为Intel inside CORE-i5系列CPU、2.2GHz主频、2GB内存; 实验软件环境为Windows7操作系统、Microsoft Visual Studio 2008集成环境、SQL Server 2008数据库.

3.4 不同核函数下窗宽估计

实验分析了不同核函数下窗宽对用户兴趣分布估计影响, 令 $h\in\{0.1,0.2,0.3,...,1\}$, 对每一个 h 取值测试其 *Accuracy_rate* 与 *MAE*. 其中, 核函数分别选择高斯函数、三角函数和均匀函数. 实验结果如图(1)、(2)所示.

由图1、图2实验结果可知, 在窗宽 h=0.4时, DUID-CF算法的推荐准确率高且MAE较低. 因此后续实验中核函数的窗宽设置为0.4.

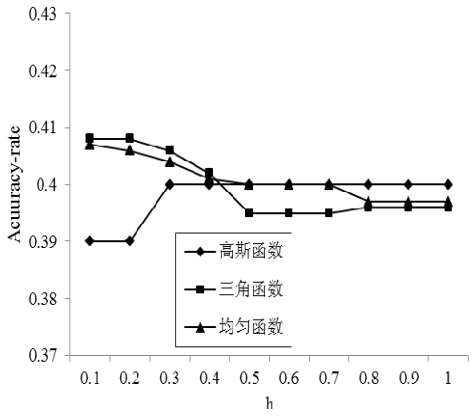

图 1　不同核函数下 DUID-CF 算法的推荐准确率

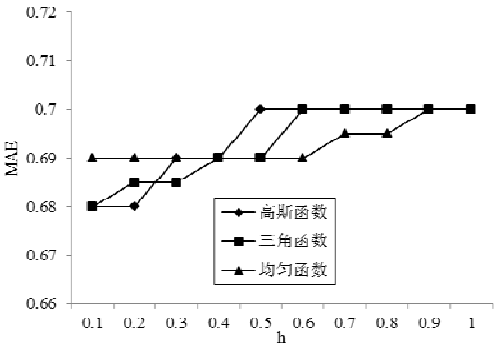

图 2 不同核函数下 DUID-CF 算法的平均绝对误差

3.5 推荐实验结果与分析

实验利用遗传算法分别对式(8)、式(16)中的两个参数 α，β 进行了寻优实验, 其适应度函数为式(17), 得到最优参数值 α=0.56，β=0.61 .

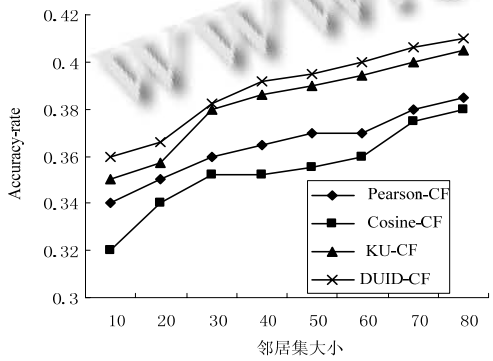

图 3 不同邻居集下四种算法准确率

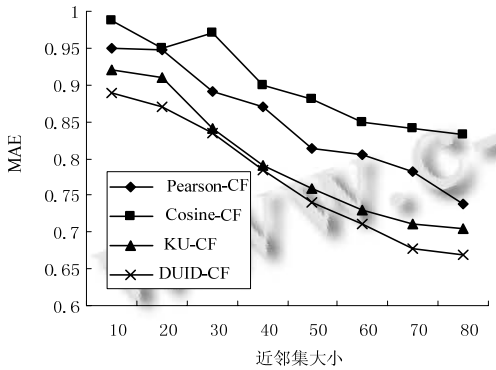

图 4 不同邻居集下四种算法平均绝对误差

在同等数据集下, 实验将基于修正余弦(Cosine)相似度的协同过滤算法, 基于Pearson相似度的协同过滤算法, 基于核方法的协同过滤(KU-CF)算法, 以及 DUID-CF 算法进行了实验比较, 由图(3)、(4)可以看出, 在不同邻居数目下, DUID-CF 算法较其他三种算法有较高的推荐准确率和更低的 MAE, 表明 DUID-CF 算法有更高的推荐质量.

4 结语

针对传统协同过滤算法存在的数据稀疏性等问题, 提出一种面向项目兴趣度分布的协同过滤算法. 在 MovieLens 数据集合上的仿真结果表明, 该算法能够避免数据填充引入的误差, 显著提高预测准确性. 如何降低算法计算量是下一步研究的主要内容.

参考文献

1 Huang CG, Yin J, Wang J, et al. Uncertain neighbors' collaborative filtering recommendation algorithms. Chinese Journal of Computers, 2010, 33(8): 1369–1377.

2 Kuruoz M, Benozur A, Csalogány K. Methods for largescale SVD with missing values. 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, California. 2007. 31–38.

3 方耀宁,郭云飞,丁雪涛等.一种基于局部结构的改进奇异值分解推荐算法.电子与信息学报,2013,35(6):1284–1289.

4 Deshpande M, Karypis G. Item-Based top-N recommendation algorithms. ACM Trans. on Information Systems, 2004, 22(1): 147–177.

5 Degemmis M, Lops P, Semeraro G. A content-collaborative recommender that exploits wordnet-based user profiles for neighborhood formation. Journal of User Modeling and User-Adapted Interaction, 2007, 17(3): 217–255.

6 王鹏,王晶晶,俞能海.基于核方法的 User-Based 协同过滤推荐算法.计算机研究与发展,2013,50(7):1444–1451.

7 彭石,周志彬,王国军.基于评分矩阵预填充的协同过滤算法.计算机工程,2013,39(1):175－178.