融合评分差异和兴趣相似性的协同过滤推荐算法

魏慧娟 戴牡红

(湖南大学信息科学与工程学院 长沙 410082)

摘 要 为了解决在传统的协同过滤推荐算法中存在的相似性计算不准确的问题,并提高推荐系统的质量,提出一种用户相似度计算方法。在用户共同评分的基础上,该方法根据评分差值和时间特征来计算评分差值的信息熵;然后,利用用户评分差值的信息熵和评分项目属性计算出用户的相似度;最后,根据用户相似度计算出用户的最近邻居,以此预测目标项目的评分。实验结果表明,所提算法更加准确地实现了目标用户最近邻居的查找,有效地提高了推荐的准确性。

关键词 协同过滤,相似性计算,共同评分,项目属性

中图法分类号 TP311.5 **文献标识码** A

Collaboration Filtering Recommendation Algorithm Based on Ratings Difference and Interest Similarity

WEI Hui-juan DAI Mu-hong

(College of Information Science and Engineering, Hunan University, Changsha 410082, China)

Abstract In order to improve the quality of recommendation system and solve the existing similarity calculation inaccuracy problem of traditional collaborative filtering algorithm, this paper put forward a method to calculate user similarity. Based on the user common ratings, this method firstly calculates the information entropy of rating differentials according to rating differentials and time features. Then it evaluates the similarity of the user by utilizing the information entropy of rating differentials and the rated item attributes. Finally, the nearest neighbors would be calculated according to the user similarity, which helps predict the rating of the target item. The experimental results show that the proposed algorithm makes the target user find the nearest neighbors more accurately and improves the recommendation accuracy effectively.

Keywords Collaborative filtering, Similarity measure, Common ratings, Item attributes

1 引言

信息技术的发展在方便用户检索出自己所需信息的同时,也产生了海量数据,导致信息过载。为解决信息过载的问题,各门户网站和电子商务系统都提供了主动推荐的功能。目前,协同过滤算法是推荐系统中最受欢迎、应用最广泛的推荐算法。协同过滤的推荐包括基于邻居集和基于模型的推荐。基于邻居集的推荐又可分为基于用户的推荐和基于项目的推荐。基于用户的协同过滤推荐建立在用户过去的项目评分行为上,利用相似用户来预测对未产生行为项目的评分,然后再根据评分推荐项目给用户[1]。

但是,由于用户评分数据的稀疏性和新注册用户没有历史数据,从而导致了传统的协同过滤算法存在着用户相似性计算不准确、推荐的准确性偏低以及新用户的冷启动问题[2]。针对传统的协同过滤算法存在相似度计算不准确的问题,文献[3]提出了利用用户的共同评分数据来计算用户的相似性,并利用用户对项目的共同评分数据来计算项目相似性,再分别计算基于用户和基于项目的预测评分,最后通过相似性权重,结合前两种方法得到最终的预测结果。文献[4]根据现有用户的实际评分,利用神经网络方法输入新用户的推荐准确率 *MAE* 和所有用户的 *MAE* 值,动态调整与其他用户的各项评分差值和 Jaccard 相似度的权值大小,以实现用户的相似性度量,然后根据协同过滤的算法产生推荐结果,最后通过交叉验证的方法获得推荐系统的质量,解决了新用户的冷启动问题。文献[5]提出了一种信任模型来评测用户评分的可靠性。文献[6]提出了利用聚类方法来提高相似性度量的准确度。这些算法在一定程度上改善了协同过滤算法的推荐效果,但存在一定的缺陷,即都是只考虑评分数据带来的影响。

在计算用户相似性的同时,要考虑影响用户相似的各种因素。除了用户的评分能反映用户的兴趣偏好,影响用户的相似性度量,还能从用户所评分的项目中挖掘出用户潜在的兴趣。文献[7]提出以用户对项目属性的偏好度代替评分数据对新项目进行推荐,有效解决了新项目的冷启动问题,明显提高了推荐系统的准确度。文献[8]提出了基于项目属性的协同过滤算法,首先分析项目属性并计算项目属性的权重,然后利用属性重心协调模型和项目属性权重计算项目间的相似性,进而产生推荐预测。文献[9]指出,在用户不同描述习惯和多样性语言表达的差异下,很难辨认用户兴趣的相似性,为此利用项目特征构建用户偏好模型,然后把用户偏好的相似度引入到协同过滤的推荐算法中,以产生推荐列表。

魏慧娟(1992—),女,硕士生,主要研究方向为数据挖掘,E-mail:2435768798@qq.com;**戴牡红**(1964—),男,研究员,主要研究方向为数据科学,E-mail:dmh@hnu.edu.cn(通信作者)。

基于以上研究方法，本文提出了一种新的计算用户相似性的方法。首先，利用时间特征和评分差值来计算用户评分差异值的信息熵，以反映用户兴趣相似度的大小；然后，根据用户评分的项目属性计算用户的相似度；最后，综合考虑用户的评分差异、时间特征和评分项目的属性来评估用户的相似度，找出目标用户的邻居用户，并利用 Top-N 推荐算法生成推荐列表。

2　用户相似性度量方法

在寻找目标用户的邻居用户的过程中，传统的相似性度量方法主要有 3 种：余弦(cosine)相似性，person 相关系数，修正的余弦相似性。

2.1　余弦相似性

用户的偏好显式地用 $m*n$ 的评分矩阵表示，如图 1 所示。其中，m 代表用户的数量；n 代表项目的数量；r_{ij} 表示用户 i 对项目 j 的评分，评分值在 1 到 5 之间变动。如果用户 i 没有对项目 j 进行评分，则把评分默认值设为 0。

$$R=\begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1j} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2j} & \cdots & r_{2n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ r_{i1} & r_{i2} & \cdots & r_{ij} & \cdots & r_{in} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ r_{m1} & r_{m2} & \cdots & r_{mj} & \cdots & r_{mn} \end{bmatrix}$$

图 1　用户项目评分矩阵

把用户评分看作 n 维项目空间上的向量，用户间的相似性通过向量间的余弦夹角来判断。用户 U 的评分向量表示为$\vec{U}$，用户 V 的评分向量表示为 $\vec{V}$，用户 U 和用户 V 之间的相似度 $sim(U,V)$ 如式(1)所示。

$$sim(U,V)=\cos(\vec{U},\vec{V})=\frac{\vec{U}\vec{V}}{\|\vec{U}\|\|\vec{V}\|} \tag{1}$$

2.2　person 相关系数

在用户共同评分的基础上度量用户之间的相似度。设用户 U 评分的项目集合为 I_U，用户 V 评分的项目集合为 I_V，则用户 U 和用户 V 之间的相似度 $sim(U,V)$ 如式(2)所示。

$$sim(U,V)=\frac{\sum_{i\in I_U\cap I_V}(r_{Ui}-\overline{r_U})(r_{Vi}-\overline{r_V})}{\sqrt{\sum_{i\in I_U\cap I_V}(r_{Ui}-\overline{r_U})^2}\sqrt{\sum_{i\in I_U\cap I_V}(r_{Vi}-\overline{r_V})^2}} \tag{2}$$

其中，$I_U\cap I_V$ 表示用户 U 和用户 V 共同评分的项目，$\overline{r_U}$ 表示用户 U 所评分项目的平均值，$\overline{r_V}$ 表示用户 V 所评分项目的平均值。

2.3　修正的余弦(adjusted cosine)相似性

修正的余弦相似性计算方法通过减去用户对项目的平均评分来解决余弦相似性不同用户评分尺度的问题。I_U 表示用户 U 评分的项目集合，I_V 表示用户 V 评分的项目集合，用户 U 和用户 V 之间的相似度计算如式(3)所示。

$$sim(U,V)=\frac{\sum_{i\in I_U\cap I_V}(r_{Ui}-\overline{r_U})(r_{Vi}-\overline{r_V})}{\sqrt{\sum_{i\in I_U}(r_{Ui}-\overline{r_U})^2}\sqrt{\sum_{i\in I_V}(r_{Vi}-\overline{r_V})^2}} \tag{3}$$

3　基于评分差异和兴趣的相似性度量方法

3.1　现有相似性度量方法的不足

第 2 节介绍了余弦相似性、person 相关系数和修正的余弦相似性，但这 3 种相似性的计算方法都是从用户的评分出发，没有考虑用户评分的项目内容对用户相似性的影响。在余弦相似性的计算方法中，用户 U 的默认评分设置为 0 或用户评分的平均值$\overline{r_U}$。在大规模的评分矩阵和数据稀疏的环境下，设置相同的默认评分值是不合理。因此，在评分矩阵稀疏的情况下，余弦相似性计算方法不能准确计算出用户的最近邻居。文献[10]对余弦相似性的计算结果进行了统计，目标用户至少有一个邻居用户的占 95%，用户间的相似性计算不准确。修正的余弦相似性计算方法也存在同样的问题。

在 person 相关系数计算方法中，首先计算用户 U 和用户 V 共同评分的项目集，然后通过对项目集的评分计算用户间的相似性，这种计算方法比前两种相似性的计算方法更准确。但是，在 person 相关系数的计算式(2)中，若分母为 0，则相似性的计算方法就没有意义。在数据稀疏的环境下，person 相关系数计算方法的准确率也不高。

为了缓解评分数据稀疏带来的影响，本文重新定义了相似性的计算方法。相似度由两部分构成：1)引入信息熵计算用户评分差异的熵值，根据熵值的大小判断用户的相似性；2)根据用户评分的项目属性计算用户间的相似性，用户所评分项目的属性越相似，则用户的相似性越高。最后，将两部分相似度线性组合，从而计算出用户最终的相似度。

3.2　基于评分差异的用户相似性计算方法

设用户 U 和用户 V 共同评分的项目集合为 $I(I_1,I_2,\cdots,I_n)$，共同评分的数据分别为 $R_U(r_{U1},r_{U2},\cdots)$ 和 $R_V(r_{v1},r_{v2},\cdots)$。假定用户 U 和用户 V 对项目 I_1 的评分分别为 5 和 1，虽然用户 U 和用户 V 共同评分了 I_1，但是评分的差异反映了用户兴趣的差异。

为了反映共同评分用户兴趣的差异，本文引入了信息熵的概念。在信息论中，信息熵用来度量信息分布混乱的程度，信息排列越不整齐，信息熵越大。给定样本集 X，信息熵的计算公式如式(4)所示。

$$H(X)=-\sum_{i=1}^{N}p(x_i)\log_2 p(x_i) \tag{4}$$

其中，N 为 X 中分类的个数，$p(x_i)$ 表示 X 中第 i 类元素出现的概率。将信息熵用于计算共同评分用户之间的相似度，评分差异的信息熵越大，两用户的评分差异信息就越不整齐，用户之间的相似度也就越低。

计算用户间相似度的步骤如下：

第 1 步　计算用户 U 和用户 V 共同评分的差异值 $D(U,V)$。

$$\begin{aligned} D(U,V)&=(r_{U1}-r_{V1},r_{U2}-r_{V2},\cdots,r_{Un}-r_{Vn}) \\ &=(d_1,d_2,\cdots,d_n) \end{aligned}$$

第 2 步　计算用户差异值的信息熵。根据用户 U 和用户 V 的共同项目评分差异值 $d_i(i=1,2,\cdots,n)$，计算出 $D(U,V)$ 中 d_i 的概率 $p(d_i)$，并将其代入式(3)：

$$H(D(U,V))=-\sum_{i=1}^{N}p(d_i)\log_2 p(d_i) \tag{5}$$

其中，N 为评分差异值的种类个数，如果用户评分差异值的信息熵为 0，则表示用户 U 和用户 V 对于共同项目的评分差异完全相同，即这两个用户的相似程度最好；反之，信息熵越高，说明这两个用户对共同项目的评分差异越大。

在计算评分差异的信息熵时，发现有以下 3 种影响用户相似度大小的因素。

1)评分差值。用户 U 和用户 V 的评分差值 d_i 影响着用

户相似度的大小，评分差值$|d_i|$越大，用户 U 和用户 V 的差异度越高。

2)时间特征。用户的兴趣随着时间变化，如果要预测用户目前的兴趣，就应该关注用户最近的行为。为了找出用户的近期最相似用户，引入用户评分的时间间隔函数 $f(t)$，用户 U 和用户 V 对项目 i 评分的时间间隔 t 越大，则它们的兴趣相似度越低。时间间隔函数 $f(t)$如式(6)所示。

$$f(t)=\frac{1}{1+e^{-t}} \tag{6}$$

其中，$-1\leqslant t\leqslant 1$，$0<f(t)<1$。$f(t)$是单调递增函数，函数值随着时间间隔 t 的增加而增加，并始终保持在(0,1)范围内。首先将时间间隔值标准化，将其映射到区间(−1,1)，再代入到时间间隔函数进行计算。

3)评分项目比重。两个用户共同评分项目的数量影响着用户的兴趣相似度，如果两个用户共同评分项目的数量很少，但此时评分差异度的信息熵很小，即两个用户的相似度很高，但实际上这两个用户的相似度很低，因此在计算评分差异的信息熵时，增加一个共同评分项目比重的权重。用户 U 和用户 V 共同评分的项目越多，$\frac{N(U)\bigcup N(V)}{N(U)\bigcap N(V)}$越小，用户 U 和用户 V 的信息熵越小，则这两个用户的相似度就越大。

利用评分差值、时间特征、评分项目比重 3 种影响因素对评分差异的信息熵式(5)进行加权，加权后的信息熵计算方法如式(7)所示。

$$S(U,V)=-\frac{N(U)\bigcup N(V)}{N(U)\bigcap N(V)}\sum_{i=1}^{N}p(d_i)\log_2 p(d_i)\,|d_i|\,f(t) \tag{7}$$

第 3 步　最小-最大规范化，把 $S(U,V)$映射到区间[0,1]。用最小-最大规范化对用户间相似度 $S(U,V)$进行线性变换，归一化的相似度 $sim_1(U,V)$越大，则用户 U 和用户 V 之间的相似度就越大。其计算公式如式(8)所示。

$$sim_1(U,V)=\frac{MAX(S(U,a)-S(U,V)}{MAX(S(U,a)-MIN(S(U,a))} \tag{8}$$

其中，$S(U,a)$表示用户 U 与其他用户的相似度，$MAX(S(U,a))$代表与用户 U 相似度最大的值；$MIN(S(U,a))$代表与用户 U 相似度最小的值。

3.3　用户兴趣相似度

仅仅依靠用户的评分数据并不能完全评估用户的兴趣，用户对项目的评分只能从行为上反映用户感兴趣的项目，不能反映用户感兴趣的项目内容。每个项目都包含自身的属性，这些属性特征不仅能帮助用户更好地理解项目，还能发掘用户潜在的兴趣。用户产生行为的项目特征反映了用户的兴趣，为此，在评估用户评分相似度的同时，考虑用户评分项目的属性，以用户对项目所具有的相关属性的偏好代表用户的兴趣，以对项目属性的偏好度来反映两个用户的兴趣相似度。

假设用户所评分的 n 个项目具有 p 个属性，项目属性矩阵如图 2 所示。其中，I_{ij} 表示第 i 个项目的第 j 个属性，I_{ij} 的值为 0 和 1，如果项目 i 具有第 j 个属性，则 I_{ij} 为 1，否则为 0。

$$I=\begin{bmatrix} I_{11} & I_{12} & \cdots & I_{1j} & \cdots & I_{1p} \\ I_{21} & I_{22} & \cdots & I_{2j} & \cdots & I_{2p} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ I_{i1} & I_{i2} & \cdots & I_{ij} & \cdots & I_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ I_{n1} & I_{n2} & \cdots & I_{nj} & \cdots & I_{np} \end{bmatrix}$$

图 2　项目属性矩阵

用$\vec{I_i}$表示第 i 个项目的属性向量，也就是属性矩阵中第 i 行的行向量，用户 U 评分的项目集合表示为 N_U，用 $U(W_1,W_2,\cdots,W_j,\cdots,W_p)$表示用户 U 的偏好向量，则用户 U 的偏好向量$\vec{U}$如式(9)所示。

$$U(W_1,W_2,\cdots,W_j,\cdots,W_p)=\frac{\sum_{i\in N_U}\vec{I_i}}{N_U} \tag{9}$$

用户 U 的偏好向量为$\vec{U}$，用户 V 的偏好向量为$\vec{V}$，则以用户对项目属性的偏好度来反映用户 U 和用户 V 之间的兴趣相似度 $sim_2(U,V)$。$sim_2(U,V)$越大，用户间的兴趣越相似。

$$sim_2(U,V)=\cos(\vec{U},\vec{V})=\frac{\vec{U}\ \vec{V}}{\|\vec{U}\|\|\vec{V}\|} \tag{10}$$

3.4　用户相似度的计算

用户的兴趣相似度受到评分差异度和评分项目属性的相似度两方面因素的影响，通过对 $sim_1(U,V)$和 $sim_2(U,V)$进行线性组合，得到用户 U 和用户 V 的兴趣相似度 $sim(U,V)$，如式(11)所示。

$$sim(U,V)=\alpha sim_1(U,V)+(1-\alpha)sim_2(U,V) \tag{11}$$

其中，权重 α 的值在实验中设定，当没有与用户 U 共同评分的项目时，$\alpha=0$。通过计算用户的偏好相似度 sim_2 找到用户 U 的邻居集用户。$sim(U,V)$越大，用户 U 和用户 V 越相似。

4　融合评分差异和兴趣相似度的协同过滤算法

融合评分差异和兴趣相似度的协同过滤算法主要包括 3 个步骤：1)利用用户-项目评分矩阵 R 代表用户对项目的评分；2)通过计算用户间的相似性找出目标用户的邻居用户；3)目标用户通过邻居用户对未产生行为的项目做出评分预测，并把评分高的项目推荐给用户。

算法 1　基于评分差异和用户兴趣相似度的协同过滤算法

输入：目标用户 U，项目属性矩阵 I(n×p)，项目评分矩阵 R(m×n)，邻居用户个数 K

输出：用户 U 的前 N 个推荐项目

方法：

```
1. train(R_{m×n}, I_{n×p});//随机将 80%的项目评分矩阵和项目属性矩阵作为训练集
2. test(R_{m×n}, I_{n×p});//将剩下的 20%作为测试集
3. for 对于每个项目 i{
4.    for R 中的每个训练用户 V{
5.       Commonrate(U,V);//找出每个用户 V 与目标用户 U 的共同评分的项目
6.       Sim_1(U,V);//使用式(7)、式(8)计算出用户 U 和用户 V 评分差异的熵值
7.       Interest(R_{m×n}, I_{n×p});//使用式(9)计算出用户的偏好向量Ū和V̄
8.       Sim_2(U,V);//使用式(10)计算出用户 U 和用户 V 的偏好相似度
9.       Sim(U,V);//使用式(11)计算出用户 U 和用户 V 的兴趣相似度}
10. end for;
11. Mean(U);//计算用户 U 对所有项目评分的平均值r̄_U
12. for (a=1;a<=K;a++)  {
13.    Mean();//计算每个邻居用户的平均值r̄_a
    }
14. end for;
```

15. $r_{Ui}=\overline{r_U}+\frac{\sum_{a\in U_K}(r_{ai}-\overline{r_a})\times sim(a,U)\times f(t_0-t_{ai})}{\sum_{a\in U_K}sim(a,U)}$；//计算用户 U 对项目 i 的预测评分，$t_0$ 表示当前时间，t_{ai} 表示邻居用户 a 对项目 i 的评分时间}

16. end for

17. rank(r_{Ui})；//对r_{Ui}的值进行降序排序

18. out(i)；//输出预测评分高的前 N 个项目

5　实验结果和分析

为了验证本文提出的基于评分差异和用户兴趣相似度的协同过滤算法的有效性，本节将其与已有的用户相似性计算算法（余弦相似度、person 相关系数、修正的余弦相似性）进行实验比较。实验环境：MATLAB 实验平台，2.94 GHz CPU，4.00 GB 内存，Window 7 操作系统。

5.1　数据集

采用美国 Minnesota 大学 GroupLens 研究小组创建的 MovieLens 100k 的公开数据集[12]作为实验数据。该数据集包括 3 张重要的数据表，分别是 u. data，u. item 和 u. user，实验中主要用到 u. data 和 u. item 数据表，u. data 记录了 943 个用户对 1682 部电影的 100000 个评分，其中每个用户评分的电影至少有 20 部，评分值为 15。u. data 中包括用户 ID、项目 ID、评分数据和评分时间，本实验中主要用到了 u. item 数据表的 19 个项目属性。实验中将数据集的 80% 作为训练集，剩余的 20% 作为测试集，首先在训练集上建立用户兴趣模型，然后在测试集上对用户评分进行预测。

5.2　评测指标

本文采用平均绝对误差 MAE（Normalized Mean Absolute Error）作为预测打分准确率的评价指标，通过计算预测评分值和真实评分值的差异来度量预测的准确率。平均绝对误差的计算公式为：

$$MAE=\frac{\sum_{u,i\in Test}|r_{Ui}-\hat{r}_{Ui}|}{\sum_{U\in Test}|Test|)} \tag{12}$$

其中，r_{Ui} 表示用户 U 对项目 i 的预测评分值，$\hat{r}_{Ui}$ 表示用户 U 对项目 i 的真实评分值，$Test$ 为测试集。MAE 的值越小，表示算法的推荐精度越高。

Top-N 推荐的预测准确率一般通过准确率（precision）和召回率（recall）度量。$R(u)$ 表示为用户 u 推荐的 N 个项目，$T(u)$ 表示用户在测试集上产生评分行为的项目。

准确率衡量的是在 Top-N 的推荐列表中被正确作出推荐的项目所占的比例。推荐结果的准确率计算公式为：

$$precision=\frac{\sum_{u\in U}R(u)\bigcap T(u)}{\sum_{u\in U}R(u)} \tag{13}$$

召回率衡量的是在全部的测试样本中被正确做出推荐的项目所占的比例。推荐结果的召回率的计算公式为：

$$Recall=\frac{\sum_{u\in U}R(u)\bigcap T(u)}{\sum_{u\in U}T(u)} \tag{14}$$

5.3　实验结果

首先对算法中的参数 α 进行评估。由式（11）可知，α（$0\leqslant\alpha\leqslant1$）表示在用户相似性计算中评分差异和项目属性所占的权重，$\alpha=0$ 表示通过计算项目属性来计算用户间的相似性，$\alpha=1$ 表示通过计算用户间的共同评分差异来计算用户间的相似性。以 α 为变量，确定 MAE 的值，如图 3 所示。$\alpha=0.6$ 时，MAE 的值最小，推荐结果最好。实验结果表明，综合考虑评分差异和项目属性来计算用户的相似性要比只考虑评分差异和项目属性的推荐结果好。

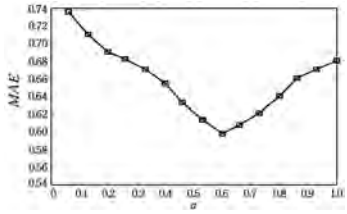

图 3　α 对 MAE 大小的影响

接下来，确定 $\alpha=0.6$，比较 cosine 相似性、person 相关、修正的余弦相似性和本文的方法的 MAE 值，以及推荐结果的准确率和召回率。图 4 给出了 4 种相似性计算方法在邻居个数变化的情况下 MAE 值的比较结果，可以看出，随着邻居个数 K 的增加，MAE 值逐渐减小，推荐精度变大，在 $K=80$ 之后，MAE 值趋于平缓。cosine 相似性和修正的余弦相似性两种方法的推荐精度相差不大，本文方法的 MAE 值最小，比 person 相关系数方法的 MAE 值小，推荐精度更高。

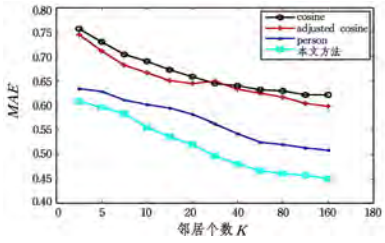

图 4　不同推荐算法的 MAE 比较

图 5 给出了 4 种相似性计算方法在邻居个数变化的情况下准确率的比较。随着邻居个数 K 的增加，4 种方法的推荐准确率增加，在 $K=80$ 时，准确率最大，修正的余弦相似性比 person 相关的相似性计算方法的准确率低，而本文方法的准确率明显高于其他 3 种算法，推荐效果最好。

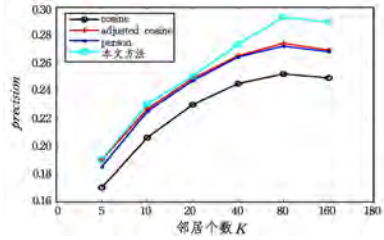

图 5　不同推荐算法的准确率比较

图 6 给出了 4 种相似性计算方法在邻居个数变化的情况下召回率的比较。实验结果表明，随着邻居个数 K 的增加，4 种方法的推荐召回率增加，在 $K=80$ 时，召回率最大，本文方法的召回率高于其他 3 种算法。

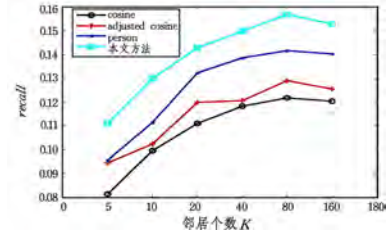

图 6　不同推荐算法的召回率比较

（下转第 422 页）

[13] ZHANG Z K,ZHOU T,ZHANG Y C. Personalized recommendation via integrated diffusion on user-item-tag tripartite graphs [J]. Physica A:Statistical Mechanics and its Applications,2010,389(1):179-186.
[14] 刘建勋,石敏,周栋,等. 基于主题模型的 Mashup 标签推荐方法[J]. 计算机学报,2017,40(2):520-534.
[15] 李锡荣,许洁萍,薛盛博,等. 基于软近邻投票的图像标签相关性计算[J]. 计算机学报,2014,37(6):1365-1371.
[16] 张斌,张引,高克宁,等. 融合关系与内容分析的社会标签推荐[J]. 软件学报,2012,23(3):476-488.
[17] YANG S,LU Z,GILES C L. Automatic tag recommendation algorithms for social recommender systems [J]. ACM Transactions on the Web,2011,5(1):1-31.
[18] 孔欣欣,苏本昌,王宏志,等. 基于标签权重评分的推荐模型及算法研究[J]. 计算机学报,2017,40(6):1440-1452.
[19] JOMSRI P,SANGUANSINTUKUL S,CHOOCHAIWATTANA W. A framework for tag-based research paper recommender system:An IR approach[C]// Proceedings of the 2010 IEEE 24th Int'l Conf. on Advanced Information Networking and Applications Workshops. 2010:103-108.
[20] 郭彩云,王会进. 改进的基于标签的协同过滤算法[J]. 计算机工程与应用,2016,52(8):56-61,147.
[21] 李慧,马小平,胡云,等. 融合主题与语言模型的个性化标签推荐方法研究[J]. 计算机科学,2015,42(8):70-74.
[22] AR Y,BOSTANCI E. A genetic algorithm solution tothe collaborative filtering problem [J]. Expert Systemswith Applications,2016,61:122-128.
[23] 李瑞敏,林鸿飞,闫俊. 基于用户-标签-项目语义挖掘的个性化音乐推荐[J]. 计算机研究与发展,2014,51(10):2270-2276.
[24] BREESE J S,HECKERMAN D,KADIE C. Empirical analysis of predictive algorithms for collaborative filtering[C]// Proc. of the 14th Conference on Uncertainty in Artificial Intelligence. Madison,Wisconsin,USA,1998:43-52.
[25] HERLOCKER J L,KONSTAN J A,BORCHERS A,et al. An algorithmic framework for performing collaborative filtering[C]// 22th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Berkeley,CA,USA,1999:230-237.
[26] JIN R,CHAI J Y,SI L. An automatic weighting scheme for collaborative filtering[C]// 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. Sheffield,UK,2004:337-344.
[27] RESNICK P,IACOVOU N,SUCHAK M,et al. An open architecture for collaborative filtering of netnews[C]// 1994 ACM Conference on Computer Supported Cooperative Work. Chapel Hill,NC,USA,1994:175-186.
[28] SARWAR B,KARYPIS G,KONSTAN J,et al. Item-based collaborative filtering recommendation algorithms[C]// 10th International Conference on World Wide Web. Hong Kong,China,2001:285-295.
[29] DESHPANDE M,KARYPIS G. Item-based top-n recommendation algorithms[J]. ACM Transactions on Information System,2004,22(1):143-177.

(上接第 401 页)

结束语 本文针对传统协同过滤算法中用户相似性计算不准确和推荐准确率偏低的问题,将时间特征、评分差异、项目属性综合考虑到相似性的计算方法中,然后再将改进的相似性计算方法融入到基于用户的协同过滤算法中,提出了一种基于评分差异和用户兴趣相似度的协同过滤算法。在计算用户相似性时,考虑到用户的兴趣随时间变化,把时间因素融合到基于改进信息熵的相似性计算方法中,评分项目属性也影响着用户的兴趣相似度,因此综合两部分的相似性计算方法得出一种新的用户相似性计算方法。实验结果表明,该相似性计算方法更能准确地找出目标用户的邻居集用户,有效地提高了推荐系统的准确率。

用户兴趣可能随时间和情景而发生改变,因此,在下一步的工作中,将建立情景模型以反映用户兴趣变化的因素,并将其加入到用户相似性的计算方法中,以更真实地描述用户兴趣和查找用户的最近邻居。

参 考 文 献

[1] CAI Y,LEUNG H,LI Q,et al. Typicality-based collaborative filtering recommendation[J]. IEEE Transactions on Knowledge and Data Engineering,2014,26(3):766-779.
[2] KALELI C. An entropy-based neighbor selection approach for collaborative filtering[J]. Knowledge-Based Systems,2014,56(C):273-280.
[3] 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):99-104.
[4] BOBADILLA J S,ORTEGA F,HERNANDO A,et al. A collaborative filtering approach to mitigate the new user cold start problem[J]. Knowledge-Based Systems,2012,26:225-238.
[5] JIA D,ZHANG F,LIU S. A robust collaborative filtering recommendation algorithm based on multidimensional trust model [J]. Journal of Software,2013,8(1):11-18.
[6] JU C,XU C. A new collaborative recommendation approach based on users clustering using artificial bee colony algorithm [J]. The Scientific World Journal,2013,2013(3):869658.
[7] 陈志敏,李志强. 基于用户特征和项目属性的协同过滤推荐算法[J]. 计算机应用,2011,31(7):1748-1750.
[8] HUANG M,SUN L,DU W. Collaborative filtering recommendation algorithm based on item attributes [C] // 2014 15th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). IEEE,2014:101-106.
[9] ZHANG J,PENG Q,SUN S,et al. Collaborative filtering recommendation algorithm based on user preference derived from item domain features[J]. Physica A:Statistical Mechanics and its Applications,2014,396(2):66-76.
[10] PIAO C H,ZHAO J,ZHENG L J. Research on entropy-based collaborative filtering algorithm and personalized recommendation in e-commerce[J]. Service Oriented Computing and Applications,2009,3(2):147-157.