CN 43-1258/TP
ISSN 1007-130X

计算机工程与科学
Computer Engineering & Science

第 36 卷第 6 期 2014 年 6 月
Vol. 36, No. 6, Jun. 2014

文章编号:1007-130X(2014)06-1172-05

基于用户特征属性和云模型的协同过滤推荐算法*

刘发升,洪　营
(江西理工大学信息工程学院,江西 赣州 341000)

摘　要:随着数据的极端稀疏性,仅仅依赖于传统的协同过滤相似性的度量方法已无法取得精确的推荐结果。针对这一问题,提出基于用户特征属性和云模型的协同过滤算法。首先,算法利用云模型计算用户评分云相似性,结合用户打分偏好对原矩阵进行填充,在此基础上得到用户的评分云相似性;其次,再结合用户特征属性相似性通过加权因子计算用户的最终相似性,得到一种新的相似性度量方法;最后,得到算法的评分预测。实验结果表明,该方法能够提高推荐质量。

关键词:协同过滤;云模型;用户特征属性相似性;打分偏好;云相似性

中图分类号:TP311　　**文献标志码**:A

doi:10.3969/j.issn.1007-130X.2014.06.028

A collaborative filtering recommendation algorithm based on user characteristic attribute and cloud model

LIU Fa-sheng, HONG Ying
(School of Information Engineering, Jiangxi University of Science and Technology, Ganzhou 341000, China)

Abstract: With the extreme sparsity of the data, traditional collaborative filtering similarity metrics are unable to obtain accurate recommendation results. In order to solve this problem, a collaborative filtering algorithm based on user feature and cloud model is proposed. Firstly, it takes advantage of the cloud model to calculate the similarity of user rating cloud, combines with user scoring preference to fill the original matrix, and then get user cloud similarity. Secondly, combining with user feature similarity and user cloud similarity, a new similarity measure method is proposed to calculate the final similarity by using weighting factor. Finally, the final rating prediction is obtained. The experimental results show that this approach can improve the recommended quality.

Key words: collaborative filtering; cloud model; user characteristic attribute similarity; scoring preference; cloud similarity

1　引言

随着电子商务的迅猛发展,商品的种类和数量呈几何式的增长,导致用户很难搜索到自己想要的产品。协同过滤技术作为现今最成功的个性化推荐技术之一[1],已广泛用于各大商务网站。比如,外国的 Amazon 的购书推荐、MovieLens 的电影推荐、Google 的 Orkut 社区推荐,国内的当当、淘宝等。

协同过滤算法基于用户-评分矩阵,度量用户或项目之间的相似性,寻找与目标用户相似的邻居用户,然后根据邻居用户对项目的评分预测目标用户的评分,从而产生推荐。但是,近些年来随着用户和商品数量的剧增,导致用户-评分矩阵极端得稀疏,使得相似性的度量不够精准,推荐算法质量降低。针对稀疏性问题,有研究人员提出改进:文献[2]提出基于云模型的协同过滤算法,计算用户

* **收稿日期**:2013-01-15;**修回日期**:2013-04-27
基金项目:江西省教育厅科技项目(GJJ08283,GJJ11463);江西省高等学校智能计算与网络测控技术重点实验室 2012 年资助项目
通信地址:341000 江西省赣州市红旗大道 86 号江西理工大学信息工程学院
Address: School of Information Engineering, Jiangxi University of Science and Technology, 86 Hongqi Avenue, Ganzhou 341000, Jiangxi, P. R. China

的评分相似性,避免了传统严格匹配对象属性不足的情况。文献[3]提出一种基于项目评分预测的协同过滤推荐算法,根据项目之间的相似性预测用户对项目的评分填充矩阵,一定程度上缓解了数据的稀疏性。文献[4]采用贝叶斯算法分析不同用户特征对项目的喜好程度,提高推荐的质量。文献[5]提出了一种基于云模型填充的协同过滤推荐算法,该算法利用相似用户计算目标用户评分缺失项,填充数据,再采用传统的协同过滤推荐算法得到理想的推荐结果。该算法在一定程度上解决了数据的稀疏性问题。

以上文献中的算法都在不同程度上给出了解决数据稀疏性问题的方法。但是,以上算法均忽略了用户特征之间的相似性和用户的打分偏好,使得算法性能降低。本文围绕上述问题展开研究,在已有的协同过滤算法的基础上,提出一种基于用户特征属性和云模型的协同过滤算法。它首先将云模型的云相似度计算引入到用户评分相似度计算中,结合用户打分偏好,对原始数据进行填充,得到用户云相似性;然后结合用户特征属性的相似性对最终用户相似性计算方法做出改进,使得计算结果更加接近用户的实际评分水平。实验表明,该算法能够提高系统的推荐质量。

2 传统的基于用户的协同过滤算法

传统的基于用户的协同过滤算法主要是根据相似用户之间对相同项目的评分也较为相似,产生邻居集,根据邻居集预测目标用户对项目的评分并推荐给用户。可用下式表示:

$$i \Rightarrow u: \exists v, (v \leftrightarrow u, v \rightarrow i) \tag{1}$$

其中 $i \Rightarrow u$ 表示项目 i 推荐给 u,$v \leftrightarrow u$ 表示用户 v 和用户 u 相似,$v \rightarrow i$ 表示用户 v 对项目 i 给予过评分。

2.1 用户相似性的度量

相似性的计算是算法中很重要的步骤,传统的相似性度量方法[6]主要有:余弦相似性、修正的余弦相似性和 Pearson 相关系数。

(1)余弦相似性。设用户 u 和用户 v 分别对 n 个项目有过访问记录,向量 $\boldsymbol{u}$ 表示 u 访问过的 n 个项目,向量 $\boldsymbol{v}$ 表示 v 访问过的 n 个项目,那么相似性可表示为:

$$sim(u,v) = \cos(\boldsymbol{u},\boldsymbol{v}) = \frac{\boldsymbol{u} \cdot \boldsymbol{v}}{\|\boldsymbol{u}\| \times \|\boldsymbol{v}\|} \tag{2}$$

(2)修正的余弦相似性。余弦相似性度量方法中没有考虑不同用户对项目的评价评分问题。修正的余弦相似性度量方法通过减去用户对项目的评价评分来改善上述缺陷,设经过用户 u 和用户 v 评分的项目集合为 $I_{u,v}$,则用户 u 和用户 v 之间的相似性为:

$$sim(u,v) = \frac{\sum_{i \in I_{u,v}} (r_{u,v} - \overline{u})(r_{u,v} - \overline{v})}{\sqrt{\sum_{i \in I_u} (r_{u,i} - \overline{r_u})^2} \sqrt{\sum_{i \in I_v} (r_{v,i} - \overline{r_v})^2}} \tag{3}$$

其中,$r_{u,i}$ 为用户 u 对项目 i 的评分,$\overline{r_u}$ 和 $\overline{r_v}$ 分别表示用户 μ 和用户 v 对项目的平均评分。

(3)Pearson 相关系数。Pearson 相关系数建立在用户的共同评分的基础上,Pearson 相关系数是最常用的一种方法,设 $I_{u,v}$ 为用户 u 和用户 v 的共同评分项目集合,$sim(u,v)$ 为两者的相似性,表示如下:

$$sim(u,v) = \frac{\sum_{i \in I_{u,v}} (r_{u,v} - \overline{r_u})(r_{v,i} - \overline{r_v})}{\sqrt{\sum_{i \in I_{u,v}} (r_{u,v} - \overline{r_u})^2} \sqrt{\sum_{i \in I_{u,v}} (r_{v,i} - \overline{r_v})^2}} \tag{4}$$

2.2 推荐的生成

通过 2.1 节提到的相似性度量方法得到目标用户的最近邻居集,预测目标用户对项目的评分,从而产生推荐。设目标用户 u 的最近邻居集合为 N_u,$R_{u,i}$ 为用户 u 对项目 i 的预测评分,用如下公式表示:

$$R_{u,i} = \overline{r_u} + \frac{\sum_{v \in N_u} sim(u,v) \times (r_{u,v} - \overline{r_v})}{\sum_{v \in N_u} (|sim(u,v)|)} \tag{5}$$

其中,$\overline{r_u}$ 和 $\overline{r_v}$ 分别表示用户 u 和 v 对已评分项目的算术平均值。

2.3 现有算法不足

现有的协同过滤推荐算法普遍存在两个问题:

(1) 用户项目评分矩阵的稀疏性造成相似性度量的不准确性。用户项目数量较多时,用户项目评分数据十分稀疏,如果两个可能有相似兴趣的用户很少甚至没有对相同项目做出过评分,那么相似的用户就难以辨别。在进行算法推荐时,当新用户加入时,系统很难找到相似的对象,出现冷启动问题。

(2) 忽略了用户特征属性的相似性。现有算法对于用户相似度的计算仅限于采用用户对项目的评分作为数据集,忽略了用户特征属性。在现实生活中,具有相同特征属性的用户往往具有相同的兴趣爱好。比如,女性顾客对爱情类影片的喜好程

度一般都比男性顾客高,男性顾客购买的电子产品比女性顾客购买得多。现有算法忽略用户特征属性,无法正确反映用户的真正兴趣爱好,从而导致推荐算法的结果偏离了用户的需求,推荐结果不精确。

3 相关工作

3.1 用户特征属性相似性

在现实生活中,具有不同特征的用户对同一个物品的喜好程度往往也不一样,而特征相似的用户喜好一般都相似。加入用户特征属性相似能更加精确地计算用户之间的相似,有利于推荐的结果。

利用向量 $\boldsymbol{U}_u=(Attr_1,Attr_2,\cdots,Attr_n)$ 表示用户的特征属性,其中,$\boldsymbol{U}_u$ 表示第 u 个用户,$Attr_n$ 表示该用户的第 n 个特征属性的取值。利用用户属性特征向量建立用户特征属性矩阵,如表 1 所示。

Table 1 Table of user characteristic attributes

表 1 用户特征属性表

	$Attr_1$	$Attr_2$	$Attr_3$	…	$Attr_n$
U_1	A_{11}	A_{12}	A_{13}	…	$Attr_{1n}$
U_2	A_{21}	A_{22}	A_{22}	…	$Attr_{2n}$
⋮	⋮	⋮	⋮		⋮
U_u	A_{u1}	A_{u2}	A_{u3}	…	$Attr_{un}$

用户间的特征属性相似度量方法有多种,常见的有欧氏距离、向量余弦、Pearson 相关等。本文采用下式计算用户特征属性相似性,假设属性与属性之间相互独立,则计算公式如下:

$$sim_{\text{attr}}(u,v)=\frac{|U_u\cap U_v|}{N} \tag{6}$$

其中,$U_u\cap U_v$ 表示用户 u 和用户 v 所有特征属性值中相同特征属性的个数。N 表示所有特征属性的个数。

3.2 打分偏好

在用户特征中存在着打分偏好这一因素,即用户的打分记录普遍比其他用户高或低[7]。用户心理行为会在用户决定的时候具有潜在的影响,在给电影打分的时候,部分用户给出的分数比较宽松,给出的分数普遍高于其他用户;部分用户给出的分数比较严谨,给出的分数普遍低于其他用户。用户的打分偏好可以由云模型建立的用户特征向量中的参数云期望(Ex)中体现出来。以 3 分为参数,当云期望 Ex 低于参数时,用户打分偏低,视用户为严谨评分;当云期望 Ex 高于参数时,用户打分偏高,视用户为宽松评分。计算公式如下:

$$R_u=\begin{cases}\lfloor R'_u\rfloor, Ex_u\leqslant 3\\ \lceil R'_u\rceil, Ex_u>3\end{cases} \tag{7}$$

3.3 云模型

云模型是李德毅院士提出的一种定性定量之间转换的数学模型[8,9],目前云模型已经在数据挖掘预测、人工智能控制、系统综合评估等领域得到广泛的应用。云模型用云的数字特征期望 Ex、熵 En、超熵 He 这三个数字特征来表征,记作 C(Ex,En,He),称为云的特征向量。根据用户-项目矩阵,将每个用户看作一朵“云”,每个评分看作一个“云滴”,采用逆向云算法可以计算出用户的评分特征向量:$\boldsymbol{U}=(Ex,En,He)$。其中期望 Ex 表示用户评分的期望值,熵 En 表示用户的评分等级集中程度,超熵 He 为熵的稳定度。计算公式[2]如下:

$$Ex=\overline{X}=\frac{1}{N}\sum_{i=1}^{N}x_i \tag{8}$$

$$He=\sqrt{\frac{\pi}{2}}\times\frac{1}{N}\sum_{i=1}^{N}|x_i-Ex| \tag{9}$$

$$En=\sqrt{S^2-\frac{1}{3}He} \tag{10}$$

其中,S^2 为样本方差,表示如下:

$$S^2=\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\overline{X})^2 \tag{11}$$

由用户的评分特征向量可以计算用户的评分相似性,用向量的夹角余弦公式计算两云之间的相似性,云相似性的计算公式如下:

$$sim_{\text{C}}(u,v)=\cos(\boldsymbol{u},\boldsymbol{v})=\frac{\boldsymbol{u}\cdot\boldsymbol{v}}{\|\boldsymbol{u}\|\times\|\boldsymbol{v}\|} \tag{12}$$

4 基于用户特征属性和云模型的协同过滤算法

基于用户特征属性和云模型的协同过滤算法的主要思想:首先利用用户矩阵和云模型计算,以每个评分为一朵“云”,每个用户为一个“云滴”计算用户 u 和用户 v 的云相似性。并利用式(5)和式(7)计算预测的评分,将其填充到原矩阵当中。并在此基础上计算用户的云相似性 $sim_C(u,v)$。结合用户特征属性相似性采用式(13)得到用户最终相似性 $sim_{\text{all}}(u,v)$[10],最后产生 top-N 的推荐集。最终相似性计算公式为:

$$sim_{\text{all}}(u,v)=\lambda sim_{\text{attr}}(u,v)+(1-\lambda)sim_{\text{C}}(u,v) \tag{13}$$

其中,λ 为加权因子,可根据数据集的不同调整 λ 取值。

算法具体步骤如下:

输入:用户评分矩阵 $\boldsymbol{R}_{u,i}$,用户特征属性矩阵 $\boldsymbol{U}_{u,attr}$,最近邻居集 k,加权因子 λ。

输出:预测出的目标用户 R_u 的 top-N 的推荐集$\{I_1,I_2,I_3,\cdots,I_k\}$。

步骤 1 根据用户评分矩阵,利用公式(8)~(12)计算用户之间的云相似性;

步骤 2 采用式(5)的评分预测方法得出每个用户初步的预测结果,利用式(7)判定修正预测结果,将修正后的结果以并集的方式填入原始用户评分矩阵中,得到新的评分矩阵;

步骤 3 采用公式(4)处理新的评分矩阵,得到用户 u 和 v 的云相似性 $sim_C(u,v)$,对用户特征属性矩阵采用式(6)得到用户特征属性相似性 $sim_{attr}(u,v)$;

步骤 4 利用公式(13)计算得出最终评分相似性 $sim_{all}(u,v)$;

步骤 5 选取与目标用户最大的前 k 个用户作为邻居集,利用式(5)产生 top-N 的推荐集。

5 实验结果及分析

5.1 数据集和评估标准

实验采用公开的 MovieLens 数据集(http://movielens.umn.edu)进行实验。该数据集包括 943 个用户对 1 682 个影片的 10 万条评分记录,评分范围为 1~5,且每个用户至少对 20 部以上的电影进行了评分。本实验把评分记录按照 80% 和 20%的比例划分为训练集和测试集。数据集中还包括每个评分用户的基本信息,取其年龄、性别、职业为主要特征属性进行实验。

通常用平均绝对偏差 MAE(Mean Absolute Error)[11]来验证推荐算法的精确度,该方法主要是通过计算预测用户评分与实际的用户评分之间的偏差来度量预测的准确性。假设用户的预测评分集合为$\{pre_1,pre_2,\cdots,pre_N\}$,对应的实际评分集合为$\{R_1,R_2,\cdots,R_N\}$,则可由如下表达式计算 MAE:

$$MAE=\frac{\sum_{i=1}^{N}|pre_i-R_i|}{N}$$

5.2 实验结果

为了验证本文中提出的算法,设计了两组实验:首先验证影响因子对实验结果的影响,其次在固定影响因子的前提下,将本文算法所得实验结果与传统的基于用户的协同过滤算法 TCF(Traditional Collaborative Filtering)、基于云模型的协同过滤算法 CMCF(Cloud Model Collaborative Filtering)进行比较,以验证基于用户特征属性和云模型协同过滤推荐算法 UFCMCF(User Feature and Cloud Model Collaborative Filtering)的有效性。实验具体过程如下:

(1) 加权因子的验证:在取其最近邻居集为 50 的情况下,实验给出了不同 λ 的 MAE 值。实验的结果如图 1 所示。

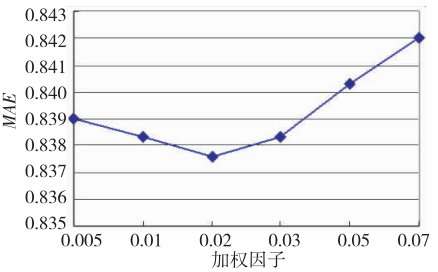

Figure 1 Effects of λ on MAE

图 1 λ 对 MAE 的影响

由图 1 可知,在取用户特征属性值为年龄、性别、职业的情况下,随着加权因子 λ 变化,MAE 值会随之发生变化,所以正确地设置参数加权因子 λ 值会直接影响算法的推荐精度。当算法有新用户加入时,冷启动问题出现,可将加权因子 λ 设置为 1,即可解决算法的冷启动问题;当算法无冷启动问题时,由实验结果得知,λ=0.02 的情况下 MAE 值为最小。所以接下来的实验验证取 λ 的值为0.02。

(2) 基于用户特征属性和云模型的协同过滤推荐算法验证:将 UFCMCF 算法与传统的 UCF 算法、CMCF 算法做比较。在 λ 取值确定的情况下,将邻居集的个数从 10 递增到 50,间隔为 10。实验结果如图 2 所示。

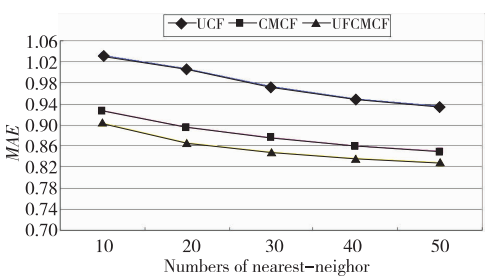

Figure 2 Comparison of three algorithms

图 2 三种算法的性能比较

由图 2 可知,随着邻居数目的变化,本文所提出的 UFCMCF 算法具有最小的 MAE 值。这是由

于算法首先采用云模型计算用户云相似性,并结合用户打分偏好,对原矩阵进行数据填充,很大程度上解决了数据稀疏性的问题,提高了相似性计算的精度;其次在采用云模型计算用户云相似性的基础上,结合用户特征属性相似性,站在用户的层面对算法进行改进,使得算法更具有个性化,推荐结果更加精确(由图中 CMCF 算法和 UFCMCF 算法比较可知)。此外用户特征属性相似度的计算,根据数据集离线进行计算,对于线上推荐结果在时间上的影响可以忽略不计,保证了算法的实时性。因此,与传统的 UCF 算法、CMCF 算法相比较,本文所提出的基于用户特征属性和云模型的协同过滤推荐算法 UFCMCF 预测结果更加准确,从而提高推荐系统的质量。

6 结束语

传统的协同过滤推荐算法在对用户做评分预测时,不考虑用户自身的特征属性,使得算法性能降低。本文提出的基于用户特征属性和云模型的协同过滤算法在利用云模型进行数据填充的时候,将利用云相似所得的预测结果进行了用户偏好的处理,在计算用户相似度的时候加入了用户特征属性,最后实验表明提出的算法能够提高系统的推荐质量。如何更有效地发现用户的偏好改变,比如用户兴趣会怎样随着时间而反复变化,改变后用户兴趣对算法结果的影响等,都是下一步将要进行的工作。

参考文献:

[1] Dasu T, Johnson T. Exploratory data mining and data cleaning[M]. Hoboken:John Wiley & Sons Press,2003.

[2] Zhang Guang-wei, Li De-yi, Li Peng, et al. A collaborative filtering recommendation algorithm based on cloud model[J]. Journal of Software,2007,18(10):2403-2411. (in Chinese)

[3] Deng Ai-lin, Zhu Yang-yong, Shi Bai-le. A collaborative filtering recommendation algorithm based on item rating prediction[J]. Journal of Software, 2003, 14(9):1621-1628. (in Chinese)

[4] Meng Xian-fu, Chen Li. Collaborative filtering recommendation algorithm based on Bayesian theory[J]. Journal of Computer Applications, 2009,29(10):2733-2735. (in Chinese)

[5] Yu Zhi-hu, Qi Yu-feng. A data fill algorithm based on cloud model[J]. Computer Technology and Development, 2010,20(12):35-37. (in Chinese)

[6] Su Xiao-yuan, Khoshgofataar T M. A survey of collaborative filtering techniques[J]. Advances in Artificial Intelligence, 2009:1-19:Article ID 421425.

[7] Yang Yang, Xiang Yang, Xiong Lei. Collaborative filtering and recommendation algorithm based on matrix factorization and user nearest neighbor model[J]. Journal of Computer Applications, 2012,32(2):395-398. (in Chinese)

[8] Li De-yi. Artificial intelligence with uncertainty[M]. Beijing:National Defence Industry Press,2005. (in Chinese)

[9] Li De-yi, Liu Chang-yu. Study on the universality of the normal cloud model[J]. Engineering Science, 2004,6(8):28-34. (in Chinese)

[10] Xu Xiang, Wang Xu-fa. Optimization method of similarity degree in collaborative filter algorithm[J]. Computer Engineering, 2010,36(6):52-57. (in Chinese)

[11] Sarwar B, Karypis G, Konstan J, et al. Analysis of recommendation algorithms for E-commerce[C]// Proc of the 2nd ACM Conference on Electronic Commerce,2000:158-295.

附中文参考文献:

[2] 张光卫,李德毅,李鹏,等. 基于云模型的协同过滤推荐算法[J]. 软件学报,2007, 18(10):2403-2411.

[3] 邓爱邻,朱扬勇,施伯乐. 基于项目评分的协同过滤推荐算法[J]. 软件学报,2003, 14(9):1621-1628.

[4] 孟宪福,陈莉. 基于贝叶斯理论的协同过滤推荐算法[J]. 计算机应用,2009,29(10):2733-2735.

[5] 余志虎,戚玉峰. 一种基于云模型数据填充的算法[J]. 计算机技术与发展,2010,20 (12):35-37.

[7] 杨阳,向阳,熊磊. 基于矩阵分解与用户近邻模型的协同过滤推荐算法[J]. 计算机应用,2012,32(2):395-398.

[8] 李德毅. 不确定性人工智能[M]. 北京:国防工业出版社,2005.

[9] 李德毅,刘长昱. 论正态云模型的普适性[J]. 中国工程科学,2004,6(8):28-34.

[10] 徐翔,王煦法. 协同过滤算法中相似度优化方法[J]. 计算机工程,2010,36(6):52-57.

作者简介:

刘发升(1963 -),男,江西大余人,博士,教授,研究方向为数据挖掘和数据库技术。**E-mail**:fashengliu@hotmail.com

LIU Fa-sheng, born in 1963, PhD, professor, his research interests include data mining, and database technique.

洪营(1990 -),男,江西鄱阳人,硕士生,研究方向为 Web 数据挖掘。**E-mail**:1052075015@qq.com

HONG Ying, born in 1990, MS candidate, his research interest includes web data mining.