

# 一种基于云模型数据填充的算法

余志虎, 戚玉峰

(南京邮电大学 计算机学院, 江苏 南京 210003)

**摘要:**协同过滤推荐技术是现如今电子商务系统中最重要的技术之一。针对目前协同过滤推荐技术中存在的因数据极度稀疏而导致相似性度量不准,推荐质量严重受到影响的问题,利用云模型在定性知识表示及定性、定量知识转换时的作用,提出一种基于云模型的数据填充算法,它利用相似用户计算目标用户评分缺失项。利用经典实验数据进行验证比较,结果表明,即使在用户评分数据极端稀疏的情况下,利用此算法对数据进行填充之后,再采用传统的协同过滤推荐算法能取得较理想的推荐质量,从一定程度上解决了推荐系统中普遍存在的稀疏性问题。

**关键词:**云模型;数据稀疏;数据填充

**中图分类号:**TP301.6

**文献标识码:**A

**文章编号:**1673-629X(2010)12-0034-04

## A Data Filling Algorithm Based on Cloud Model

YU Zhi-hu, QI Yu-feng

(School of Computer, Nanjing University of Posts & Telecommunications, Nanjing 210003, China)

**Abstract:** The collaborative filtering is one of the most important technologies in current E-commerce system, in the view of data in collaborative filtering technology are extremely sparse resulting in bad similarity measure and recommend poor quality, using a cloud model's action in qualitative knowledge representation and the role of conversion among qualitative and quantitative knowledge, proposed a data filling algorithm based on cloud model, and using the classical experimental data to validate and compare, it calculate user score missing items by using similar user. The result shows, even if the user's rating data is extremely sparse, it can get better recommendation quality by filling data with the algorithm and adopting traditional collaborative filtering algorithm, to some extent it can solve common sparse problems in recommended system.

**Key words:** cloud model; data sparse; data filling

### 0 引言

现今世界信息量随着网络的发展而飞速膨胀,如从电子商务网站获得的产品数据,其数据量增长的如此快以致我们很难处理它,例如:Yahoo、Amazon和CDNow等等这些网站每天都提供了大量选择给它的潜在客户;如何从中找到最适合用户的需求和爱好的产品则成为难点,为了克服这个问题,推荐系统出现并变成了研究者的一个研究热点。

推荐系统中有一个最著名的方法就是协同过滤<sup>[1]</sup>,它分为基于记忆和模型两种协同过滤算法,本质是通过对已有数据通过与其相似性来推荐目标数据评分值,其中,基于记忆协同过滤算法包括基于产品协同过滤<sup>[2]</sup>和基于用户协同过滤,其思想是通过相似用户

或产品的特征来推荐给目标用户或产品的打分数据值,其中最常用的方法是K近邻(K-Nearest Neighbor, KNN)方法。从这种基于记忆的方法可以看出,算法比较简单性,但其推荐效率高,也因此在推荐算法中是一种较成功的推荐方法。基于模型的方法则是采用一种预测模型<sup>[3]</sup>,这种模型是通过早先从产品评价库中获得的一个模型,再把这种模型用于对应目标用户的推荐中去,目前比较流行的方法包括聚类模型<sup>[4]</sup>、SVD<sup>[5]</sup>等。上面几种常用的推荐方法中,都有一个推荐系统中常见的问题,那就是对冷启动问题没有很好的得到解决,从而会使推荐精确度受很大影响,表现为,当一个用户评分数据稀疏时,这很容易找到,如当当网中图书数据库,一个庞大的产品库,一些冷门书会很难得到用户的评分以及一些新手购书很少有评价价值,从而很难从中找到相似的用户,这就导致最后推荐质量严重下降<sup>[6]</sup>,如图1为一电子商务推荐系统中用户-资源评价矩阵来表示用户对资源的评价情况,矩阵的每一行代表一个用户,每一列代表一个资源,中间

收稿日期:2010-04-21;修回日期:2010-07-03

基金项目:国家863计划(2007AA701302, 2009AA701202)

作者简介:余志虎(1986-),男,湖南岳阳人,硕士研究生,研究方向为计算机在通信中应用。

即表示用户评分情况,由于用户对资源的评价并非用户浏览网站的必要行为,所以用户对资源的评价存在一定的稀疏性,即在图矩阵中存在大量的空元,且空元呈不规则分布,所以用户-资源评价矩阵是一个稀疏矩阵。

	i1	i2	i3	i4	i5	...	im
u1		3	5				
u2	1		4				7
u3							
u4	4		7	8			
u5		3		5	3		
...	4				4		6
un	3		8				9

图 1 用户-资源评价矩阵

通过上面的分析,考虑到推荐系统中对于数据十分稀疏而导致得不到准确推荐质量的常见问题,提出了一种通过云模型技术<sup>[7]</sup>填充用户评分矩阵,通过云模型的定性概念来反映用户评分的总体情况,发现用户满意度(即评分高低程度)、评分离散度以及这种离散度的稳定性,然后在知识层面完成相似度的比较,通过这种评分总体情况来发现用户之间的相似性,然后根据这种用户相似性去填充待填充用户评分数据,这种方法克服了传统基于向量的相似度比较方法严格匹配对象属性的不足,用这种定性概念来较好地解决数据稀疏性的问题。

## 1 数据稀疏性问题的解决方案

通过参考文献,得出现有的协同推荐系统中针对数据稀疏性较常见的解决方法主要有零值填充、均值填充、SlopeOne<sup>[8]</sup>填充方法以及基于奇异值分解的降维技术等等。

零值填充以及均值填充算法是最普遍的填充算法,在一般对预测的实时性以及准确性要求不太高的地方都采用这两种方法。优点就是简单,从而减少协同过滤系统数据预处理时间。

其中,Daniel Lemire 教授提出了一种 SlopeOne 算法,它的基本思想比较简单,是利用各用户对不同不充分项的差异来决定不同项间的平均评分差异,然后利用这个评分差异来求得目标用户需要填充的缺失目标评分。算法简单,易实现和维护,但针对趋势不规律的评分推荐的效果很不理想。

基于奇异值分解则是一种隐性语义检索,其优点是它允许存在一个简化的近似矩阵,通过在原始的矩阵基础上采用一定办法来降低噪音,从而能够更有效地表明在用户与项目间的潜在关联性。但付出的代价是推荐的质量却有一定程度的下降;基于聚类的协同

过滤推荐算法则是通过用户或者项目的相似性进行聚类。把先前全部用户按照一定的规则分成多个簇,品味和爱好相似的用户则会被分配到相同的聚类中去。在一定程度上缓解数据稀疏性的问题。

文中提出一种基于云模型的数据填充方法,利用云模型的定性定量转换特点,通过用户评分项来求得定性的用户评分特征向量,然后利用这些特征向量计算用户的相似度,针对目标用户评分缺失项,利用其  $K$  个邻近用户对目标项的评分相似度以及相关因子加权计算目标用户缺失项评分。最后在推荐过程中结合基于项目推荐算法来作为一种推荐系统模型。

## 2 基于云模型的数据填充算法

### 2.1 云模型

云模型是李德毅院士提出的作为不确定性知识的定性定量转换的数学模型<sup>[9,10]</sup>,能够实现定性概念与其数值表示之间的不确定性转换,目前云模型已成功应用于智能控制、数据挖掘、大系统评估等领域。云的数字特征用期望  $Ex$ (expected value)、熵  $En$ (entropy)、超熵  $He$ (hyper entropy)这 3 个数字特征来整体表征一个概念。期望  $Ex$  在数域空间最能够代表定性概念的点,即这个概念量化的最典型样本点;熵  $En$  是定性概念不确定性的度量,熵越大,通常概念越宏观,也是定性概念不确定性的度量;超熵  $He$  是熵的不确定性度量,即熵的熵,由熵的随机性和模糊性共同决定。用 3 个数字特征表示的定性概念的整体特征记作  $C(Ex, En, He)$ ,称为云的特征向量。计算如下:

$$Ex: \bar{Ex} = \bar{X}$$

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

$$En: \bar{En} = \sqrt{S^2 - \frac{1}{3} He^2}$$

其中,  $\bar{X}$  为样本均值,  $S^2$  为样本方差。

把云的特征向量应用于数据填充算法中,首先根据每个用户评分项来计算其用户的评分特征向量,这里采用逆云计算,把用户单个评分项看作是云滴,从而每个用户就可以看作是一个云的特征向量,即用户评分的特征向量,利用评分特征向量采用下面的云相似度量方法来计算用户间的相似性,最后再通过选取最近的  $K$  个邻近用户来通过加权来计算目标用户的缺少评分项。

### 2.2 相似性度量度量

云的相似度量利用云的整体特征向量来计算,其公式如下:

$$\text{sim}(m, n) = \frac{C_m \times C_n}{\|C_m\| \|C_n\|}$$

其中  $C_m = (Ex_m, En_m, He_m)$ ,  $C_n = (Ex_n, En_n, He_n)$  为云的特征向量, 把每个用户评分看作是云滴, 则每个用户的评分就是一个云的特征向量, 这里即为用户的评分特征向量, 从而可以利用逆向云计算得到。这三个数字特征表示定性概念, 利用云模型的定性概念能够完成用户相似性比较, 克服传统基于向量的相似性比较方法严格匹配对象属性的不足。

其中, 用户相似性计算中还需要引入相关因子的概念, 因为在计算用户相似性时, 只是考虑了用户整体评分特性, 没有考虑具体是用户的哪些项的评分, 故最后可能会导致具有不相同评分项的用户也具有相似性, 从而使得结果有效性不高, 这也是云模型需要考虑的, 从而引入相关因子的概念。

定义 1: 相关因子  $\Phi(x_{i,j})$ , 计算如下:

$$\Phi(x) = 1 - \frac{1}{2^{\lambda x}}$$

其中,  $x_{i,j}$  为两用户  $i, j$  的共同评分的项目个数,  $\lambda$  为权衡系数, 可以根据需评分项来由用户指定,  $\lambda$  越大, 表示需要共同评分的项越多, 当  $x$  越大时, 即共同评分越多时, 相关因子越大, 从而使得最后求填充项数据时, 其准确性越高。

利用逆向云算法算出每个用户获取评分的三个特征值  $Ex, En, He$ , 如果用户间相关因子大于指定值, 这里的指定值视数据集的大小由用户来指定, 然后再利用相似性公式计算出用户相似度。

### 2.3 基于云模型数据填充算法

其基本思想: 利用用户评分来找相似用户, 然后根据相似用户对指定项目评分来填充目标用户相应项目的缺失评分, 因为考虑相似用户是通过评分特征向量来计算的相似性, 故相似用户对同项目的评分值也具有相似值。具体过程是, 先找到需填充数据评分项, 计算目标用户与其他各用户间的相关因子  $\Phi(x_{i,j})$ , 采用云模型的用户评分特征向量来计算用户相似值; 然后根据相似值来计算此用户相似列表, 找出此用户最近  $K$  邻居, 再利用  $K$  个邻居的评分情况采用加权平均策略把相关因子考虑其中, 由此计算客户的评分缺失项。

算法: 基于云模型填充算法

输入: 用户评分矩阵

输出: 填充较完整的评分矩阵

第一步: 通过给出用户评分矩阵, 利用逆向云算法计算每个用户的评分特征值 ( $Ex, En, He$ );

第二步: 计算用户相似度矩阵;

根据用户评分项, 利用余弦距离函数计算用户相似度, 同时计算所有用户间的相关因子  $\Phi(x_{i,j})$ , 最后

存入相似度矩阵与相关因子矩阵中, 其中用户  $U_i$  和其自身的相似度为 1, 用户  $U_i$  对用户  $U_j$  的相似度与项目  $U_j$  对项目  $U_i$  的相似度相等;

第三步: 产生填充数据:

对于用户  $U_i$  没有评分的指定项目  $I_1$ , 通过用户  $U_i$  的最近  $K$  邻居, 利用其用户评分相似性原则采用加权平均策略填充用户评分项  $S_{i1}$ , 方法如下:

$$S_{i1} = S_i + \frac{\sum_{u \in \text{Ng}(i)} (r_{u,I} - S_u) \text{sim}(i, u) \Phi(x_{i,u})}{\sum_{u \in \text{Ng}(i)} |\text{sim}(i, u)| |\Phi(x_{i,u})|}$$

其中,  $r_{u,I}$  为  $i$  的邻近用户  $u$  对项目  $I$  的评分,  $\text{Ng}(i)$  为用户  $i$  的  $K$  个邻近用户,  $S_i$  为用户  $i$  的评分平均值,  $S_u$  为用户  $u$  的用户评分均值,  $\text{sim}(i, u)$  为用户  $i$  与其邻近用户  $u$  的相似度,  $\Phi(x_{i,u})$  为用户  $i$  和邻近用户  $u$  的相关因子。

## 3 实验及结果分析

文中提出一种基于云模型方法来对稀疏数据评分进行填充, 对原始评分数据表中除去目标用户对目标项目评分之外, 其它所有未评分项目则依次用此方法填补。最后再采用常用的 Item-based 方法对目标预测项进行评分。

### 3.1 实验数据

为了分析所采用算法的推荐效果, 设计合理的实验过程和评价标准, 在协同过滤推荐过程中, 把基于云模型数据填充方法结合 Item-based 过滤推荐算法, 来与其他过滤推荐算法作比较。在邻居个数不同的情况下分别对 User-based, Item-based 协同过滤算法进行实验, 这两种协同过滤算法不采用任何数据填充方法的前提下与采用基于云模型的方法进行比较, 同时与加入 SlopeOne 填充方法进行比较。

采用 Grouplens 研究小组收集的 MovieLens 数据集, 数据集 MovieLens 来自 (<http://MovieLens.umn.edu/>), 它是用于接收用户对电影的评分并提供相应的电影推荐列表。下载的数据是其中的一部分, 具体为 100000 个评分数据记录, 是 943 个用户对 1682 部电影的评价结果, 可以看出评分数据的稀疏值为  $1 - 100000 / (943 * 1682) = 0.936953$ , 网站还规定一个用户至少要对 20 部不同电影进行了评分。这里将相关因子中的  $\lambda$  为权衡系数选择 10, 然后将数据导入 Access 数据库中作为实验数据。

### 3.2 评价指标

衡量推荐系统质量的方法主要有统计精度标准和决策支持精度标准, 一般采用前者, 即分析系统对用户的推荐值和用户的真实评价之间的误差距离, 这里

采用常用的平均绝对偏差 MAE (Mean Absolute Error)<sup>[11,12]</sup>,在评价度量过程中将数据记录分为训练集和测试集两部分,推荐算法工作在训练集中,目标用户为  $U_i$ ,来测试训练集中的数据预测测试集中的项目,然后 MAE 则是对测试集中目标用户已经评分过的电影项目进行度量,规定项目数为  $N_i \leq N$ ,用户  $U_i$  平均绝对偏差  $MAE_i$  的公式为:

$$MAE_i = \frac{\sum_{i=1}^{N_i} |p_i - q_i|}{N_i}$$

其中,  $p_i$  表示系统对项目  $i$  的推荐值,  $q_i$  则为对应的实际用户评分, MAE 为计算得到的平均绝对误差。则 MAE 值越小,则表示系统预测的质量就越高。

### 3.3 实验结果

从图2可以看出没有采用任何填充方法的传统 Item-based 和 User-based 算法产生协同推荐计算,相对来说,其 MAE 值较高,从而预测的质量较差,而采用的加入基于云模型数据填充方法 MAE 要比 Slope One 方法预测质量要好,从一定程序说明此方法的优越性。

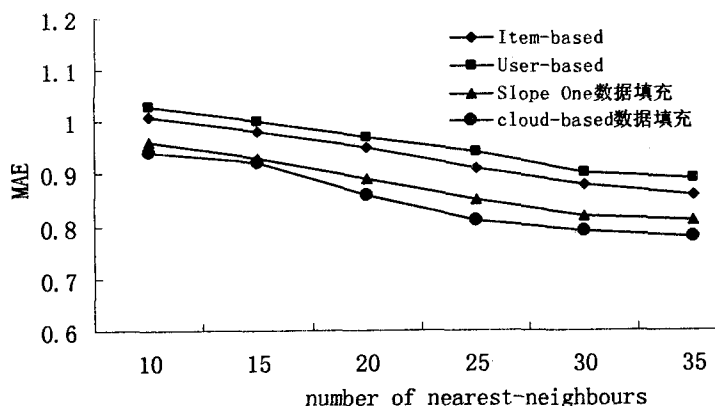


图2 各推荐算法 MAE 值比较

## 4 结束语

采用一种基于云模型的数据填充算法,通过云模型定性定量转换来求得相似用户,进而利用相似用户

集的评分数据来填充用户缺失评分项,从实验结果可以看出从一定程度上解决了协同过滤推荐系统中数据稀疏性的问题。下步工作可以在填充方法中考虑用户个人信息,不仅仅局限于用户评分来填充数据,从而表现一种更个性化的数据填充方法。

### 参考文献:

- [1] 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.
- [2] 邓爱林, 朱扬勇, 施伯乐. 基于项目评分预测的协同过滤推荐算法[J]. 软件学报, 2003, 14(9): 1621-1628.
- [3] 青 海. 电子商务推荐系统核心技术研究[D]. 北京: 北京工业大学, 2009.
- [4] 查文琴, 梁昌勇, 曹 镭. 基于用户聚类的协同过滤推荐方法[J]. 计算机技术与发展, 2009, 19(6): 69-75.
- [5] 曾小波, 魏祖宽, 金在弘. 协同过滤系统的矩阵稀疏性问题的研究[J]. 计算机应用, 2010, 34(4): 1079-1082.
- [6] 郭艳红, 邓贵仕. 协同过滤系统项目冷启动的混合推荐算法[J]. 计算机工程, 2008, 34(23): 11-13.
- [7] Zhang Guang-wei, Li De-Yi, Li Peng, et al. A Collaborative Filtering Recommendation Algorithm Based on Cloud Model[J]. Journal of Software, 2007, 118(10): 2403-2411.
- [8] 王立军. 基于协同过滤推荐系统的数据稀疏性问题的研究[D]. 长春: 东北师范大学, 2009.
- [9] 李德毅, 刘常昱. 论正态云模型的普适性[J]. 中国工程科学, 2004, 6(8): 28-34.
- [10] 李德毅, 刘常昱, 杜 鹄, 等. 不确定性人工智能[J]. 软件学报, 2004, 15(11): 1583-1594.
- [11] Sun Tieli, Wang Lijun, Guo Qinghe. A Collaborative Filtering Recommendation Algorithm Based on Item Similarity of User Preference[C]. USA: IEEE Computer Society, 2009: 60-63.
- [12] Lemire D, Maclachlan A. Slope One Predictors for Online Rating-Based Collaborative Filtering[C]//SIAM Data Mining. California: Newport Beach, 2005: 21-23.

(上接第7页)

- [8] Search Algorithm for Block Motion Estimation[J]. IEEE Trans. on Circuit and Systems for Video Technology, 1994, 4(4): 438-442.
- [9] 朱秀昌, 刘 峰, 胡 栋. 数字图像处理与图像通信[M]. 北京: 北京邮电大学出版社, 2002.
- [10] 梁 燕, 刘文研. 基于起点预测的自适应快速搜索算法

- [J]. 计算机工程, 2005, 31(19): 1-3.
- [11] 文 俊, 王 朋, 刘重庆. 一种预测三步搜索算法[J]. 上海交通大学学报, 2003, 37(6): 857-859.
- [12] Koga T, Iinuma K, Hirano A, et al. Motion compensated inter-frame coding for video conferencing[C]//Proc. NTC81. New Orleans, LA: [s. n.], 1981.