ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.18, No.10, October 2007, pp.2403−2411
DOI: 10.1360/jos182403
© 2007 by *Journal of Software*. All rights reserved.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于云模型的协同过滤推荐算法*

张光卫[1,4+], 李德毅[2], 李 鹏[3], 康建初[1], 陈桂生[2]

[1](北京航空航天大学 软件开发环境国家重点实验室,北京 100083)

[2](中国电子工程系统研究所,北京 100840)

[3](哈尔滨工业大学 深圳研究生院 信息安全中心,广东 深圳 518055)

[4](山东建筑大学 计算机科学与技术学院,山东 济南 250101)

A Collaborative Filtering Recommendation Algorithm Based on Cloud Model

ZHANG Guang-Wei[1,4+], LI De-Yi[2], LI Peng[3], KANG Jian-Chu[1], CHEN Gui-Sheng[2]

[1](State Key Laboratory of Software Development Environment, BeiHang University, Beijing 100083, China)

[2](Institute of Electronic System Engineering, Beijing 100840, China)

[3](Information Security Center, Shenzhen Graduate School, Harbin Institute of Technology, Shenzhen 518055, China)

[4](School of Computer Science and Technology, Shandong Jianzhu University, Ji'nan 250101, China)

+ Corresponding author: Phn: +86-10-82316732, E-mail: ezhang@263.net, http://nlsde.buaa.edu.cn

Zhang GW, Li DY, Li P, Kang JC, Chen GS. A collaborative filtering recommendation algorithm based on cloud model. *Journal of Software*, 2007,18(10):2403−2411. http://www.jos.org.cn/1000-9825/18/2403.htm

Abstract: Recommendation system is one of the most important technologies applied in e-commerce. Similarity measuring method is fundamental to collaborative filtering algorithm, and traditional methods are inefficient especially when the user rating data are extremely sparse. Based on the outstanding characteristics of Cloud Model on the process of transforming a qualitative concept to a set of quantitative numerical values, a novel similarity measuring method, namely the likeness comparing method based on cloud model (LICM) is proposed in this paper. LICM compares the similarity of two users on knowledge level, which can overcome the drawback of attributes' strictly matching. This work analysis traditional methods throughly and puts forward a novel collaborative filtering algorithm, which is based on the LICM method. Experiments on typical data set show the excellent performance of the present collaborative filtering algorithm based on LICM, even with extremely sparsity of data.

Key words: cloud model; collaborative filtering; similarity; recommendation system; voting

摘 要: 协同过滤系统是电子商务系统中最重要的技术之一,用户相似性度量方法是影响推荐算法准确率高低的关键因素.针对传统相似性度量方法存在的不足,利用云模型在定性知识表示以及定性、定量知识转换时的桥梁作用,提出一种在知识层面比较用户相似度的方法,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.以该方法为核心,在全面分析传统方法的基础上,提出一种新的协同过滤推荐算法.实验结果表明,算法在用户评分数据极端稀疏的情况下,仍能取得较理想的推荐质量.

* Supported by the National Natural Science Foundation of China under Grant Nos.60496323, 60375016 (国家自然科学基金); the National Basic Research Program of China under Grant No.G2004CB719401 (国家重点基础研究发展计划(973))

Received 2006-05-18; Accepted 2007-02-05

© 中国科学院软件研究所 http://www.jos.org.cn

关键词: 云模型;协同过滤;相似性;推荐系统;投票
中图法分类号: TP311 文献标识码: A

推荐系统已逐渐成为电子商务IT技术的一个重要研究内容,日益受到研究者的关注.目前,几乎所有大型的电子商务系统,如 Amazon,eBay,阿里巴巴等,都不同程度地使用了各种形式的推荐系统.

为了在保证推荐系统实时性的条件下产生相对精确的推荐,研究者提出了多种不同的推荐算法,如协同过滤推荐系统、聚类技术[1,2]、关联规则技术[3]、Horting图技术[4]等.

Typestry[5]是最早提出来的基于协同过滤的推荐系统,但需要目标用户明确指出与自己行为比较类似的其他用户.GroupLens[6]是基于用户评分的自动化协同过滤推荐系统,用于推荐影片和新闻.Ringo推荐系统[7]和Video推荐系统[8]通过电子邮件的方式分别推荐音乐和影片.Breese等人[9]对各种协同过滤推荐算法及其改进进行了深入分析.传统的协同过滤推荐通过用户的最近邻居产生最终的推荐,基于项目的协同过滤推荐首先计算项目之间的相关性,然后通过用户对相关项目的评分预测用户对未评分项目的评分.

最近邻协同过滤推荐是当前最成功的推荐技术之一[9],它基于这样一个假设:如果用户对一些项目的评分比较相似,则他们对其他项目的评分也会比较相似.算法的基本思想是:目标用户对未评分项目的评分,可以通过其最近邻居对该项目的评分来逼近.

为了找出目标用户的最近邻居,需要度量用户之间的相似性.然而,随着电子商务系统规模的扩大,用户数目和项目数据急剧增加,用户评分数据出现极端稀疏性[1],使得利用传统相似性度量方法得到的最近邻居集合不够准确,导致算法的推荐质量降低.

针对用户评分数据极端稀疏的问题,文献[10,11]提出通过奇异值分解(singular value decomposition,简称SVD)减少项目空间维数的方法,但降维会导致信息损失,使得该方法在项目空间维数较高的情况下难以保证推荐效果[12].文献[13]提出一种基于项目评分预测的协同过滤推荐技术,通过估计用户评分的办法补充用户-项矩阵,减小数据稀疏性对计算结果的负面影响.由于该算法在计算项目相似性时仍然沿用传统的相似性计算方法,而且没有考虑项目的分类信息,从而影响了推荐质量.文献[14]在此基础上引入了项目分类信息,采用修正的条件概率计算项目之间的相似性,并用于对用户没有评价过的项目进行评分估计,填充用户-项矩阵,进而根据填充了的用户-项矩阵计算用户相似性,取得了不错的效果.然而,该算法依然把用户对所有项目的评分作为单个向量进行用户相似度的计算和指导最近邻居的选择,在一定程度上影响了算法的效能.

本文在分析传统用户相似度计算方法的基础上,利用云模型在定性知识表示以及定性、定量知识转换时的桥梁作用,提出一种基于云模型的相似度计算方法(likeness comparing method based on cloud model,简称LICM).该方法在知识层面完成相似度的比较,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.进而以 LICM 方法为基础,提出一种新的协同过滤算法.

本文第 1 节为传统相似性度量方法的介绍及分析.第 2 节在简单介绍云模型之后,基于云模型给出用户投票偏好的知识表示及其相似性度量方法(LICM).第 3 节给出基于 LICM 的协同过滤推荐算法的详细步骤.第 4 节是仿真对比实验和结果分析.最后给出全文总结及进一步研究的方向.

1 传统的相似性度量方法及其分析

度量用户间相似性的方法,目前主要有余弦(cosine)相似性、相关(correlation)相似性和修正的余弦(adjusted cosine)相似性 3 种:

(1) 余弦相似性: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度量.设用户 i 和用户 j 在 n 维项目空间上的评分分别表示为向量 $\vec{i}$, $\vec{j}$,则用户 i 和用户 j 之间的相似性为

$$sim(i,j)=\cos(\vec{i},\vec{j})=\frac{\vec{i}\cdot\vec{j}}{\|\vec{i}\|\|\vec{j}\|} \tag{1}$$

(2) 修正的余弦相似性:由于在余弦相似性度量方法中没有考虑不同用户的评分尺度问题,修正的余弦相

© 中国科学院软件研究所 http://www.jos.org.cn

似性度量方法通过减去用户对项目的平均评分来改善上述缺陷.设经用户i和用户j共同评分的项目集合用I_{ij}表示,I_i和I_j分别表示经用户i和用户j评分的项目集合,则用户i和用户j之间的相似性为

$$sim(i,j)=\frac{\sum_{c\in I_{ij}}(R_{i,c}-\bar{R}_i)(R_{j,c}-\bar{R}_j)}{\sqrt{\sum_{c\in I_i}(R_{i,c}-\bar{R}_i)^2}\sqrt{\sum_{c\in I_j}(R_{j,c}-\bar{R}_j)^2}} \tag{2}$$

$R_{i,c}$表示用户i对项目c的评分,$\bar{R}_i$和$\bar{R}_j$分别表示用户i和用户j对项目的平均评分.

(3) 相关相似性:设经用户i和用户j共同评分的项目集合用I_{ij}表示,则用户i和用户j之间的相似性$sim(i,j)$通过Pearson相关系数来度量:

$$sim(i,j)=\frac{\sum_{c\in I_{ij}}(R_{i,c}-\bar{R}_i)(R_{j,c}-\bar{R}_j)}{\sqrt{\sum_{c\in I_{ij}}(R_{i,c}-\bar{R}_i)^2}\sqrt{\sum_{c\in I_{ij}}(R_{j,c}-\bar{R}_j)^2}} \tag{3}$$

$R_{i,c}$表示用户i对项目c的评分,$\bar{R}_i$和$\bar{R}_j$分别表示用户i和用户j对项目的平均评分.

总体来讲,3 种方式均为基于向量的相似度计算方式,进行对象属性之间的严格匹配.

余弦相似性度量方法把用户评分看作一个向量,用向量的余弦夹角度量用户间的相似性,然而没有包含用户评分的统计特征;修正的余弦相似性方法在余弦相似性基础上,减去了用户对项目的平均评分,然而该方法更多体现的是用户之间的相关性而非相似性.相关性和相似性是两个不同的概念,相似性反映的是聚合特点,而相关性反映的是组合特点[15];相似相关性方法,依据双方共同评分的项目进行用户相似性评价,如果用户间的所有评分项目均为共同评分项目,那么相似相关性和修正的余弦相似性是等同的.用户对共同评分项目的评分确实能够很好地体现用户的相似程度,但由于用户评分数据的极端稀疏性,用户间共同评分的项目极为稀少,使得相似相关性评价方法实际不可行,文献[13,14]等提出通过填充用户-项矩阵进行改进的方法.

2 基于云模型的相似性度量方法

2.1 云模型简介

云模型是李德毅院士提出的一种定性定量转换模型[16−19],能够实现定性概念与其数值表示之间的不确定性转换,已经在智能控制、模糊评测等多个分类得到应用.正态云模型是最重要的一种云模型,由于其具有良好的数学性质,可以表示自然科学、社会科学中大量的不确定现象[17].

定义 1. 云和云滴:设U是一个用数值表示的定量论域,C是U上的定性概念,若定量值$x\in U$是定性概念C的一次随机实现,x对C的确定度$\mu(x)\in[0,1]$是有稳定倾向的随机数,$\mu:U\to[0,1]$ $\forall x\in U$ $x\to\mu(x)$,则x在论域U上的分布称为云,记为云$C(X)$.每一个x称为一个云滴[16].如果概念对应的论域是n维空间,那么可以拓广至n维云.

定义中提及的随机实现是概率意义下的实现,每一次实现的随机样本又具有一个随机的确定度;定义中提及的确定度是模糊集意义下的隶属度,同时又具有概率意义下的分布,这些体现了模糊性和随机性的关联性.

云模型所表达的概念的整体特性可以用云的数字特征来反映,云用期望 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 是云滴在论域空间分布的期望,是最能够代表定性概念的点,或者说是这个概念量化的最典型样本;熵 En 代表定性概念的可度量粒度,熵越大,通常概念越宏观,也是定性概念不确定性的度量,由概念的随机性和模糊性共同决定.一方面,En 是定性概念随机性的度量,反映了能够代表这个定性概念的云滴的离散程度;另一方面,又是定性概念亦此亦彼性的度量,反映了在论域空间可被概念接受的云滴的取值范围;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.

通过正向云算法,可以把定性概念的整体特征变换为定量数值表示,实现概念空间到数值空间的转换;通过逆向云算法,可以实现从定量值到定性概念的转换,将一组定量数据转换为以数字特征$\{Ex,En,He\}$来表示的定性概念.

算法 1 和算法 2 分别给出了正向云算法和逆向云算法.

© 中国科学院软件研究所 http://www.jos.org.cn

算法 1. 正向云算法[16].

输入:表示定性概念 C 的 3 个数字特征值 Ex,En,He($En \geq He \geq 0$);云滴数 N;

输出:N 个云滴的定量值以及每个云滴属于概念 C 的确定度.

步骤:

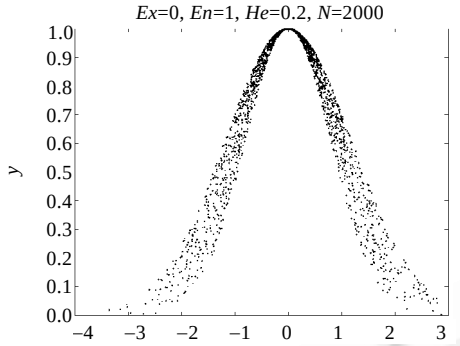


Fig.1 Distributing of one-dimension cloud

图 1 一维正态云模型的分布

(1) 生成区间[$En-He$,$En+He$]上的一个均匀随机数 En';

(2) 生成以Ex为期望值,$(En')^2$为方差的一个正态随机数x;

(3) 令 x 为定性概念 C 的一次具体量化值,称为云滴;

(4) 计算 $y=\mathrm{e}^{-(x-Ex)^2/(2(En')^2)}$,令 y 为 x 属于定性概念 C 的确定度,{x,y}完整地反映了这一次定性定量转换的全部内容;

(5) 重复步骤(1)~步骤(4),直到产生 N 个云滴.

如图 1 所示,就是用上述算法生成的一维正态云,横坐标是云滴 x,纵坐标是云滴 x 的确定度 y.云的 3 个参数为 Ex=0,En=1,He=0.2,共生成了 2 000 个云滴.

算法 2. 逆向云算法[16].

输入:N个云滴{x_1,x_2,…,x_N};

输出:这 N 个云滴表示的定性概念的期望值 Ex、熵 En 和超熵 He.

步骤:

(1) 根据x_i计算这组数据的样本均值 $\bar{X}=\frac{1}{N}\sum_{i=1}^{N}x_i$,一阶样本绝对中心矩 $\frac{1}{N}\sum_{i=1}^{N}|x_i-\bar{X}|$,样本方差 $S^2=\frac{1}{N-1}\sum_{i=1}^{N}(x_i-\bar{X})^2$;

(2) Ex 的估计值为 $E\hat{x}=\bar{X}$;

(3) He 的估计值为 $\hat{He}=\sqrt{\frac{\pi}{2}}\times\frac{1}{N}\sum_{i=1}^{N}|x_i-E\hat{x}|$;

(4) En 的估计值为 $\hat{En}=\sqrt{S^2-\frac{1}{3}H\hat{e}^2}$.

2.2 基于云模型的用户相似度计算

电子商务系统一般通过投票(voting)的办法得到用户对项目的满意情况,假设评分标准为 5 个级别{很不满意、不满意、一般、满意、很满意},相应的分值分别为{1,2,3,4,5}.以表 1 中给出 A,B,C,D 这 4 个用户对 10 个项目的评分情况.

Table 1 User scoring matrix

表 1 用户评分矩阵

User \ Item	1	2	3	4	5	6	7	8	9	10
A	2	1	1	1	2	1	1	2	1	2
B	5	4	5	4	5	4	5	4	5	4
C	4	5	3	4	5	5	4	4	5	3
D	2	1	2	2	1	1	2	2	1	2

统计用户对各个项目打分分值出现频度,记作用户的评分频度向量,记为 $\vec{U}$ =($u1$,$u2$,$u3$,$u4$,$u5$),其中,$u1$~$u5$ 分别为用户给出的相应于 5 个等级的评价次数.用户评分频度向量不关心具体项目的评分,而是关心用户对项

© 中国科学院软件研究所 http://www.jos.org.cn

目集的评分特征,表中给出的 4 个用户的评分频度向量分别为 $\vec{U}_A$=(6,4,0,0,0), $\vec{U}_B$=(0,0,0,5,5), $\vec{U}_C$=(0,0,2,4,4), $\vec{U}_D$=(4,6,0,0,0).

利用逆向云算法,根据用户对各个项目的评分频度向量,可以计算出用云的 3 个参数表示的用户评分偏好,我们把由云的 3 个参数组成的用户评分偏好称为用户评分特征向量,记为 $\vec{V}=(Ex,En,He)$,其中,期望 *Ex* 反映了用户对所有项目的平均满意度,为偏好水平;熵 *En* 反映了用户投票的集中程度,为投票偏好的离散度;*He* 为熵的稳定度.

分别计算表中给出的 4 个用户的评分特征向量: $\vec{V}_A$=(1.5,0.62666,0.339), $\vec{V}_B$=(4.6,0.60159,0.30862), $\vec{V}_C$=(4.4,0.75199,0.27676), $\vec{V}_D$=(1.6,0.60159,0.30862),显然,用户 *A*,*D* 倾向投低分,*B*,*C* 倾向投高分,且 *C* 投票的离散度大于 *B*.用户之间相似程度取决于他们在对同一组项目是否具有相似的评分倾向.

首先我们给出云的相似度定义:

定义 2. 给定由两个云 *i*,*j* 的数字特征组成的向量 $\vec{V}_i$ 和 $\vec{V}_j$,它们之间的余弦夹角称为云 *i* 和 *j* 之间的相似度:

$$sim(i,j)=\cos(\vec{V}_i,\vec{V}_j)=\frac{\vec{V}_i\cdot\vec{V}_j}{\|\vec{V}_i\|\|\vec{V}_j\|},$$

其中,

$$\vec{V}_i=(Ex_i,En_i,He_i),\ \vec{V}_j=(Ex_j,En_j,He_j).$$

两个云之间的相似度有以下特点:

Sim(*i*,*i*)=1,即一个云与其自身的相似度为 1.

对称性 *Sim*(*i*,*j*)=*Sim*(*j*,*i*),即云 *i* 对 *j* 的相似度等于 *j* 对 *i* 的相似度.

基于云的相似度定义,对用户相似度定义如下:

定义 3. 给定项目集合和用户对其中一个或多个项目的评分集合,两个用户关于项目集合的评分特征向量的余弦夹角,称为他们之间的用户相似度.这是一种基于项目集合的用户相似度定义.

对于上面得到的 *A*,*B*,*C*,*D* 这 4 个用户的评分特征向量,按照云的相似性定义,计算出它们两两之间的相似度,结果为如表 2 所示的对角线元素均为 1 对称矩阵.

Table 2 User similarity matrix

表 2 用户相似度矩阵

	A	*B*	*C*	*D*
A	1	0.956 06	0.964 75	0.999 01
B	0.956 06	1	0.999 22	0.967 94
C	0.964 75	0.999 22	1	0.975 45
D	0.999 01	0.967 94	0.975 45	1

由于用户评分特征向量中包含了用户对项目的平均满意度、离散度和不确定度信息,通过向量间余弦夹角的计算得到的相似度中包含了用户评分的统计特征,因而得到的相似性是合理的.为了表示方便,我们记这种基于云的相似性比较方法为 LICM 方法.

为了对 LICM 方法与第 1 节列出的 3 种传统方法进行对比,分别用 LICM 方法、余弦相似性方法、修正余弦相似性方法计算以上给出的 *A*,*B*,*C*,*D* 这 4 个用户的相似性(对于本例,相关相似性方法等同于修正余弦相似性方法),结果见表 3.

Table 3 User similarity computed by three methods

表 3 3 种方法计算的用户相似性

	LICM method				Cosine similarity method				Adjusted cosine similarity method			
	A	*B*	*C*	*D*	*A*	*B*	*C*	*D*	*A*	*B*	*C*	*D*
A	1	0.956 06	0.964 75	**0.999 01**	1	**0.938 11**	0.916 6	0.926 7	1	0	–0.218 22	**0.25**
B	0.956 06	1	**0.999 22**	0.967 94	0.938 11	1	**0.978 47**	0.950 33	0	1	0	0
C	0.964 75	**0.999 22**	1	0.975 45	0.916 6	**0.978 47**	1	0.896 53	–0.218 22	**0**	1	–0.872 87
D	**0.999 01**	0.967 94	0.975 45	1	0.926 7	**0.950 33**	0.896 53	1	**0.25**	0	–0.872 87	1

© 中国科学院软件研究所 http://www.jos.org.cn

根据以上计算结果,分别为 *A*,*B*,*C*,*D* 选出最相似的用户,结果见表 4.可见,通过 LICM 方法得到的计算结果最为合理,与我们的主观判断非常一致,优于其他两种方法的计算结果.

Table 4 Result of similarity computation

表 4 相似度计算结果

	LICM method	Cosine similarity method	Adjusted cosine similarity method
A	***D***	*B*	*D*
B	***C***	*C*	Unknown
C	***B***	*B*	*B*
D	***A***	*B*	*A*

LICM 方法是本文提出的协同过滤推荐算法的基础,具有以下特点:

(1) 在用户投票整体层面上粗粒度考虑用户的相似性,考虑了对象类的整体信息,避免了基于向量的相似度计算方式严格匹配对象属性的不足;

(2) 充分利用了用户投票数据的统计信息;

(3) 避免了传统相似度比较方法中侧重利用相关性而非相似性的弱点;

(4) 使得那些虽然缺少共同评分项目,但有整体共同偏好的用户变得可比较;

(5) 更加适合用户评分数据稀疏的现实情况.

3 基于云模型的协同过滤推荐算法

基于知识层面的用户相似性比较方法,本文提出一种新的协同过滤算法:首先用 LICM 方法计算相似性矩阵,然后根据待推荐用户和待评估项目找出用户的最近邻居,再通过加权平均策略预测项目的评分,详细过程见算法 3.

算法 3. 基于云模型的协同过滤推荐算法.

输入:用户评分表;

输出:预测目标用户 UID 关于项目 IID 的推荐评分.

步骤:

(1) 计算用户-项目矩阵

根据用户评分表,计算用户项目矩阵 *R*,行代表用户,用户数为 *m*,列表示项目,项目数为 *n*.

$$R=\begin{pmatrix} r_{11} & r_{12} & \cdots & r_{1n} \\ r_{21} & r_{22} & \cdots & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ r_{m1} & r_{m2} & \cdots & r_{mn} \end{pmatrix}$$

Fig.2 User-Item scoring matrix

图 2 用户-项目评分矩阵

对用户没有评价过的项目的评分设为 0,即 $r_{ij}=\begin{cases}\text{实际评分}, & \text{如果用户}i\text{对项目}j\text{投票} \\ 0, & \text{如果用户}i\text{对项目}j\text{没有投票}\end{cases}$.

(2) 计算用户评分特征向量

根据用户项目矩阵 *R*,统计各个用户的评分频度向量 $\vec{U}_i=(u1,u2,u3,u4,u5)(1\le i\le m)$;

根据评分频度向量,通过逆向云算法计算每个用户的评分特征向量 $\vec{V}_i=(Ex_i,En_i,Ee_i)(1\le i\le m)$.

(3) 计算用户相似度矩阵

用户相似度矩阵可表示为 $Sim=\begin{pmatrix} sim(1,1) & sim(1,2) & \cdots & sim(1,m) \\ sim(2,1) & sim(2,1) & \cdots & sim(2,m) \\ \vdots & \vdots & \ddots & \vdots \\ sim(m,1) & sim(m,2) & \cdots & sim(m,m) \end{pmatrix}$,其中,$Sim(i,j)$表示用户 *i* 与用户 *j* 的相

© 中国科学院软件研究所 http://www.jos.org.cn

似度:

$$sim(i,j)=\cos(\vec{V}_i,\vec{V}_j)=\frac{\vec{V}_i\times\vec{V}_j}{\|\vec{V}_i\|\|\vec{V}_j\|} \tag{4}$$

其中,$\vec{V}_i=(Ex_i,En_i,He_i)$,$\vec{V}_j=(Ex_j,En_j,He_j)$ 分别为用户 i,j 的评分特征向量.

(4) 产生推荐

首先寻找最近邻居,根据待推荐用户UID、用户相似矩阵Sim和用户项目矩阵R,在整个用户空间中查找用户UID的最近邻居用户且该用户对项目IID有评价记录,得到最近邻居集$F_{\mathrm{UID}}=\{F1,F2,\ldots,Fk\}$,并且$F1$ 与UID的相似性$sim(\mathrm{UID},F1)$最高,$F2$ 与UID的相似性$sim(\mathrm{UID},F2)$次之,依此类推.

根据最近邻居集合F_{UID}产生推荐,预测UID对待推荐项目IID的打分,本文采用加权平均策略:用户UID对项目IID的预测评分$P_{\mathrm{UID,IID}}$可以通过F_{UID}中各个用户对项目IID评分的加权平均得到[9],计算方法如下:

$$P_{\mathrm{UID,IID}}=\bar{R}_{\mathrm{UID}}+\frac{\sum_{u\in F_{\mathrm{UID}}}sim(\mathrm{UID},u)\times(r_{u,\mathrm{IID}}-\bar{R}_u)}{\sum_{u\in F_{\mathrm{UID}}}|sim(\mathrm{UID},u)|} \tag{5}$$

其中,$r_{u,\mathrm{IID}}$为用户u对项目IID的评分,$sim(\mathrm{UID},u)$为用户UID与u的相似度,$\bar{R}_{\mathrm{UID}}$,$\bar{R}_u$ 分别表示用户UID和u对已评分项目评分的算术平均值.

4 实验及其分析

本文使用MovieLens站点(http://movielens.umn.edu)提供的测试数据集,该站点是一个基于Web的研究型推荐系统,用于接收用户对电影的评分并提供相应的电影推荐列表.从该站点下载 1997 年 9 月 19 日~1998 年 5 月 22 日的数据集,包括 943 个用户对 1 682 个项目(影片)的 10 万条投票记录,用户评分数据集的稀疏等级[13]为 1−100000/(9431682)=0.9370.把记录按照 80%和 20%的比例划分为训练集和测试集.

推荐质量的评价标准主要有两类:统计精度度量方法和决策支持精度度量方法[20,21].统计精度度量方法中的平均绝对偏差MAE(mean absolute error),是一种常用度量方法.

平均绝对偏差(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.假设预测的用户评分集合为$\{p_1,p_2,\ldots,p_N\}$,对应的实际评分集合为$\{q_1,q_2,\ldots,q_N\}$,则MAE可由下式计算[20,21]:

$$MAE=\frac{\sum_{i=1}^{N}|p_i-q_i|}{N} \tag{6}$$

把本文的算法与传统的基于余弦法、基于修正余弦的协同过滤推荐算法以及文献[22]提出的 BP-CF(back propagation-collaborative filtering)算法进行比较,使用 MAE 为度量标准,得到如图 3 所示的 MAE 随最近邻居数变化而变动的折线图.可见,本文提出的算法有较好的性能表现.

5 结论和进一步工作

本文在分析传统协同过滤推荐算法用户相似度计算方法的基础上,利用云模型在定性知识表示以及定性、定量知识转换时的桥梁作用,提出一种基于云模型的用户相似度比较方法(LICM).该方法在知识层面完成用户相似度的比较,改善了传统基于向量的相似度比较方法必须严格匹配对象属性的不足,而且算法在一定程度上克服了用户评分数据极端稀疏的负面影响.如何基于定性知识进行用户项稀疏矩阵填充等技术,都是有意义的下一步研究方向.

© 中国科学院软件研究所 http://www.jos.org.cn

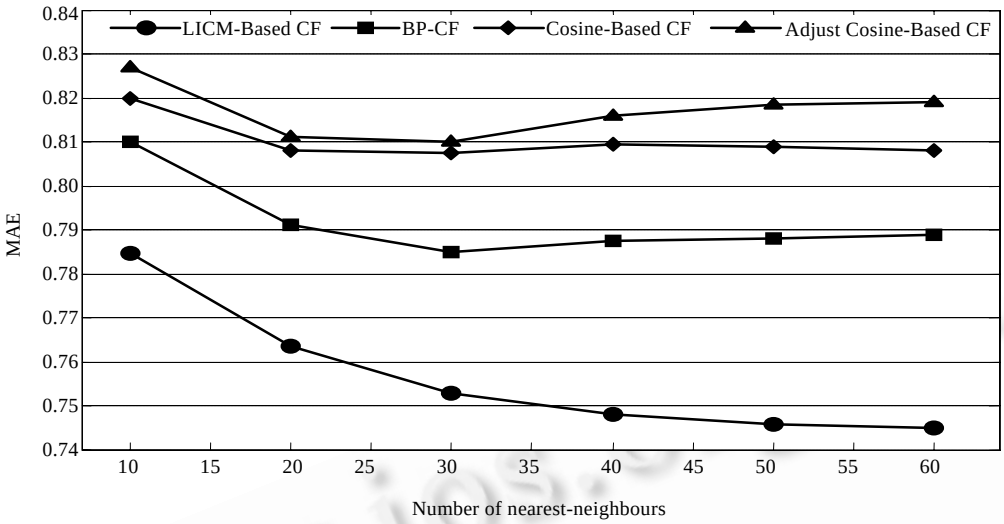


Fig.3 MAE-Nearest neighbours number graph

图 3 MAE 随最近邻居数变化图

References:

[1] Zan H, Hsinchun C, Daniel Z. Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering. ACM Trans. on Information Systems, 2004,22(1):116−142.

[2] Thiesson B, Meek C, Chickering DM, Heckerman D. Learning mixture of DAG models1 microsoft research. Technical Report, MSR2TR297230, 1997.

[3] Sarwar B, Karypis G, Konstan J, Riedl J. Analysis of recommendation algorithms for E-commerce. In: Proc. of the 2nd ACM Conf. on Electronic Commerce. New York: ACM Press, 2001. 158−167. http://www.research.ibm.com/iac/ec00/

[4] Aggarwal CC,Wolf J, Wu KL, Yu PS. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. In: Proc. of the 5th ACM SIGKDD Int'l Conf. on Knowledge Discovery and Data Mining. New York: ACM Press, 1999. 201−212.

[5] Goldberg D, Nichols D, Oki BM, Terry D. Using collaborative filtering to weave an information tapestry. Communications of the ACM, 1992,35(12):61−70.

[6] Resnick P, Iacovou N, Suchak M, Bergstrom P, Riedl J. Grouplens: An open architecture for collaborative filtering of netnews. In: Proc. of the ACM CSCW'94 Conf. on Computer-Supported Cooperative Work. Chapel Hill: ACM, 1994. 175−186.

[7] Shardanand U, Maes P. Social information filtering: Algorithms for automating "Word of Mouth". In: Proc. of the ACM CHI'95 Conf. on Human Factors in Computing Systems. New York: ACM Press/Addison-Wesley Publishing Co., 1995. 210−217.

[8] Hill W, Stead L, Rosenstein M, Furnas G. Recommending and evaluating choices in a virtual community of use. In: Proc. of the CHI'95. New York: ACM Press/Addison-Wesley Publishing Co., 1995. 194−201.

[9] Breese J, Hecherman D, Kadie C. Empirical analysis of predictive algorithms for collaborative filtering. In: Proc. of the 14th Conf. on Uncertainty in Artificial Intelligence (UAI'98). San Francisco: Morgan Kaufmann Publishers, 1998. 43−52.

[10] Sarwar BM, Karypis G, Konstan JA, Riedl J. Application of dimensionality reduction in recommender system-A case study. In: Proc. of the ACM WebKDD 2000 Workshop. 2000. http://robotics.stanford.edu/~ronnyk/WEBKDD2000/

[11] Zhao L, Hu NJ, Zhang SZ. Algorithm design for personalization recommendation systems. Journal of Computer Research and Development, 2002,39(8):986−991 (in Chinese with English abstract).

[12] Aggarwal CC. On the effects of dimensionality reduction on high dimensional similarity search. In: Proc. of the ACM PODS Conf. Santa Barbara: ACM, 2001.

[13] Deng AL, Zhu YY, Shi BL. A collaborative filtering recommendation algorithm based on item rating prediction. Journal of Software, 2003,14(9):1621−1628 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/14/1621.htm

[14] Zhou JF, Tang X, Guo JF. An optimized collaborative filtering recommendation algorithm. Journal of Computer Research and Development, 2004,41(10):1842−1847 (in Chinese with English abstract).

© 中国科学院软件研究所 http://www.jos.org.cn

[15] Liu Q. Research on some key technologies of Chinese-English machine translation [Ph.D Thesis]. Beijing: Peking University, 2004 (in Chinese with English abstract).

[16] Li DY. Artificial Intelligence with Uncertainty. Beijing: National Defense Industry Press, 2005. 171−177 (in Chinese).

[17] Li DY, Liu CY. Study on the universality of the normal cloud model. Engineering Science, 2004,6(8):28−34 (in Chinese with English abstract).

[18] Li DY, Liu CY, Du Y, Han X. Artificial intelligence with uncertainty. Journal of Software, 2004,15(11):1583−1594 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/15/1583.htm

[19] Li DY. Uncertainty in knowledge representation. Engineering Science, 2000,2(10):73−79 (in Chinese with English abstract).

[20] Zhang BQ. A collaborative filtering recommendation algorithm based on domain knowledge. Computer Engineering, 2005,31(21):7−9 (in Chinese with English abstract).

[21] Sarwar B, Karypis G, Konstan J, Riedl J. Item-Based collaborative filtering recommendation algorithms. In: Proc. of the 10th Int'l World Wide Web Conf. Hong Kong: ACM Press, 2001. 285−295.

[22] Zhang F, Chang HY. Employing BP neural networks to alleviate the sparsity issue in collaborative filtering recommendation algorithms. Journal of Computer Research and Development, 2006,43(4):667−672 (in Chinese with English abstract).

附中文参考文献:

[11] 赵亮,胡乃静,张守志.个性化推荐算法设计.计算机研究与发展,2002,39(8):986−991.

[13] 邓爱林,朱扬勇,施伯乐.基于项目评分预测的协同过滤算法.软件学报,2003,14(9):1621−1628. http://www.jos.org.cn/1000-9825/14/1621.htm

[14] 周军锋,汤显,郭景峰.一种优化的协同过滤算法.计算机研究与发展,2004,41(10):1842−1847.

[15] 刘群.汉英机器翻译若干关键技术研究[博士学位论文].北京:北京大学,2004.

[16] 李德毅.不确定性人工智能.北京:国防工业出版社,2005.171−177.

[17] 李德毅,刘常昱. 论正态云模型的普适性.中国工程科学,2004,6(8):28−34.

[18] 李德毅,刘常昱,杜鹢,韩旭.不确定性人工智能.软件学报,2004,15(11):1583−1594. http://www.jos.org.cn/1000-9825/15/1583.htm

[19] 李德毅.知识表示中的不确定性.中国工程科学,2000,2(10):73−79.

[20] 张丙奇.基于领域知识的个性化推荐算法研究.计算机工程,2005,31(21):7−9.

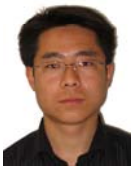
[22] 张锋,常会友.使用 BP 神经网络缓解协同过滤推荐算法的稀疏性问题.计算机研究与发展,2006,43(4):667−672.

张光卫(1970－),男,山东济南人,博士生,主要研究领域为人工智能,数据挖掘.

康建初(1953－),女,副教授,主要研究领域为人工智能,新一代网络,中间件技术.

李德毅(1944－),男,研究员,博士生导师,中国工程院院士,主要研究领域为人工智能,指挥自动化.

陈桂生(1965－),男,博士后,讲师,主要研究领域为人工智能,复杂网络,指挥自动化.



李鹏(1983－),男,硕士生,主要研究领域为人工智能,信息安全,数据挖掘.