Journal of Computer Applications
计算机应用,2016,36(8):2246-2251,2281
ISSN 1001-9081
CODEN JYIIDU
2016-08-10
http://www.joca.cn

文章编号:1001-9081(2016)08-2246-06
doi:10.11772/j.issn.1001-9081.2016.08.2246

基于改进型启发式相似度模型的协同过滤推荐方法

张　南,林晓勇*,史晟辉
(北京化工大学 信息科学与技术学院,北京 100029)
(*通信作者电子邮箱 linxy@mail.buct.edu.cn)

摘　要:为提高协同过滤推荐方法的准确性和有效性,提出一种基于改进型启发式相似度模型的协同过滤推荐方法 PSJ。该方法考虑了用户评分差值、用户全局评分偏好和用户共同评分物品数三个因素。PSJ 方法的 Proximity 因子使用指数函数反映用户评分差值对用户相似度的影响,这样也可避免零除问题;将 NHSM 方法中的 Significance 因子和 URP 因子合并成 PSJ 方法的 Significance 因子,这使得 PSJ 方法的计算复杂度低于 NHSM 方法;而且为了提高在数据稀疏情况下的推荐效果,PSJ 方法同时考虑了用户间的评分差值和用户全局评分两个因素。实验采用 Top-*k* 推荐中的查准率和查全率作为衡量标准。实验结果表明,当推荐物品数大于 20 时,与 NHSM、杰卡尔德算法、自适应余弦相似度(ACOS)算法、杰卡尔德均方差(JMSD)算法和皮尔逊相关系数算法(SPCC)相比,PSJ 方法的查准率与查全率均有提升。

关键词:协同过滤推荐方法;启发式相似度模型;用户相似度;推荐效果;数据稀疏
中图分类号:TP181　　**文献标志码**:A

Collaborative filtering recommendation method based on improved heuristic similarity model

ZHANG Nan, LIN Xiaoyong*, SHI Shenghui
(*College of Information Science and Technology, Beijing University of Chemical Technology, Beijing* 100029, *China*)

Abstract: In order to improve the accuracy and efficiency of collaborative filtering recommendation method, a collaborative filtering recommendation method based on improved heuristic similarity model, namely PSJ, was proposed, which considered the difference of user ratings, the user global rating preferences and the number of common rating items. The Proximity factor of PSJ method used the exponential function to reflect the influence of the difference of user ratings, which avoided the problem of zero divider. The Significance factor of NHSM (New Heuristic Similarity Model) method and the URP (User Rating Preference) factor were merged to build the Significance factor of PSJ method, which makes the computational complexity of the PSJ method be lower than that of NHSM. To improve the recommendation performance in data sparsity conditions, both the variance value of user ratings and user global rating preferences were considered in PSJ method. In experiments, precision and recall of Top-*k* recommendation were used to evaluate the results. The results show that compard with NHSM, Jaccard algorithm, Adjust COSine similarity (ACOS) algorithm, Jaccard Mean Squared Difference (JMSD) algorithm and Sigmoid function based Pearson Correlation Coefficient method (SPCC), the precision and recall of PSJ method are improved.

Key words: collaborative filtering recommendation method; heuristic similarity model; user similarity; recommendation performance; data sparsity

0　引言

互联网的信息量正变得越来越大,然而大量的线上信息也带来一些不便。例如,某个消费者希望在网上买一部智能手机,在做出决定之前他将非常困惑,因为他将浏览和比较大量的互联网上提供的智能手机信息。推荐系统就是为了解决这种问题而被发明的,它会自动为用户提供购买物品的建议。准确的推荐可以帮助用户快速找到他感兴趣的物品并且避免用户浏览大量的相关商品。推荐系统同时对经销商来说也是一个很好的助手,它可以向访问经销商网站的浏览者推荐网站内的商品,通过这样的方式经销商有可能将这些浏览者变成长期客户。

协同过滤方法[1]首先被发明用于邮件过滤,现在它已经非常广泛地用于推荐系统。它依据激活用户的相似度或者被激活用户评过分的物品的相似度提供推荐结果。协同过滤技术可以被分成两类:基于内存的方法和基于模型的方法[2]。基于内存的方法首先利用用户评分数据计算用户之间的相似度,然后将相似度值超过某个阈值的用户作为目标用户的邻居,最后根据这些邻居产生推荐结果。基于模型的方法的工作方式完全不同。它首先建立一个模型用于描述用户的行为,并用这个模型来预测用户对物品的评分。协同过滤方法的关键是计算用户之间的相似度,因此相似度计算的准确性将影响推荐的准确性。现在有许多相似度算法,例如:杰卡尔德相似度(Jaccard similarity, Jaccard)算法[3]、余弦相似度

收稿日期:2016-01-13;**修回日期**:2016-03-03。　**基金项目**:中央高校基本科研业务费资助项目(JD1413)。
作者简介:张南(1988—),男,湖南邵阳人,硕士研究生,主要研究方向:人工智能、数据挖掘;　林晓勇(1979—),男,福建浦城人,副教授,博士研究生,主要研究方向:基于 Web 2.0 的社会性网络服务、数据挖掘;　史晟辉(1974—),女,河北河间人,副教授,博士研究生,主要研究方向:大数据分析、编译技术、生物信息、自然语言处理。

(COSine similarity, COS)算法[4]、皮尔逊相关系数(Pearson Correlation Coefficient, PCC)算法[5]。除了协同过滤方法之外,还有许多方法被用于推荐系统,例如:语义推荐、社交推荐、基于内容的技术。

1　相关原理

假设 $U=\{u_1,u_2,\cdots,u_n\}$ 代表用户集合,$I=\{i_1,i_2,\cdots,i_n\}$ 代表物品集合,$\boldsymbol{R}=(r_{i,j})_{M\times N}$ 代表用户评分矩阵($i=1,2,\cdots,M$; $j=1,2,\cdots,N$)。

COS 方法[4] 是最常用的相似度算法,计算公式如下:

$$sim(u,v)^{\mathrm{COS}}=\frac{\boldsymbol{r}_u\cdot\boldsymbol{r}_v}{\|\boldsymbol{r}_u\|\cdot\|\boldsymbol{r}_v\|} \tag{1}$$

其中,$\boldsymbol{r}_u$ 和 $\boldsymbol{r}_v$ 分别表示用户 u 和用户 v 的评分向量。

用户评分偏好是一个相似度计算的因素。研究人员在研究中发现:一些人可能很喜欢一个物品,但是他的评分相对偏低;一些人可能不喜欢某个物品,但是他给出的评分可能很高。一些研究人员认为余弦相似度算法没有考虑用户的评分偏好,为此提出了自适应余弦相似度(Adjust COSine similarity, ACOS)算法[6],计算公式如下:

$$sim(u,v)^{\mathrm{ACOS}}=\frac{\sum_{p\in I}(r_{u,p}-\bar{r}_u)(r_{v,p}-\bar{r}_v)}{\sqrt{\sum_{p\in I}(r_{u,p}-\bar{r}_u)^2}\cdot\sqrt{\sum_{p\in I}(r_{v,p}-\bar{r}_v)^2}} \tag{2}$$

其中:$r_{u,p}$ 和 $r_{v,p}$ 分别表示用户 u 和用户 v 对物品 p 的评分,$\bar{r}_u$ 和 $\bar{r}_v$ 分别表示用户 u 和用户 v 的评分均值。

PCC 方法[5]是和 COS 方法一样的常用相似度计算方法,计算公式如下:

$$sim(u,v)^{\mathrm{PCC}}=\frac{\sum_{p\in I_{uv}}(r_{u,p}-\bar{r}_u)(r_{v,p}-\bar{r}_v)}{\sqrt{\sum_{p\in I_{uv}}(r_{u,p}-\bar{r}_u)^2}\cdot\sqrt{\sum_{p\in I_{uv}}(r_{v,p}-\bar{r}_v)^2}} \tag{3}$$

其中,I_{uv} 表示用户 u 和用户 v 共同评分物品的集合。

一些研究人员认为 PCC 算法没有考虑到用户评分的消极程度和积极程度,他们认为评分域的中间值可以用于区分评分的积极和消极程度,为此提出了约束皮尔逊相关系数算法(Constrained Pearson Correlation Coefficient measure, CPCC)[7],计算公式如下:

$$sim(u,v)^{\mathrm{CPCC}}=\frac{\sum_{p\in I_{uv}}(r_{u,p}-r_{\mathrm{med}})(r_{v,p}-r_{\mathrm{med}})}{\sqrt{\sum_{p\in I_{uv}}(r_{u,p}-r_{\mathrm{med}})^2}\cdot\sqrt{\sum_{p\in I_{uv}}(r_{v,p}-r_{\mathrm{med}})^2}} \tag{4}$$

其中,r_{med} 表示评分域中间值。

一些研究者认为共同评分的物品数量也是影响相似度计算的因素,因此将曲线方程和 PCC 方法相结合提出了基于曲线方程的皮尔逊相关系数算法(Sigmoid function based Pearson Correlation Coefficient method, SPCC)[8],计算公式如下:

$$sim(u,v)^{\mathrm{SPCC}}=\frac{\sum_{p\in I_{uv}}(r_{u,p}-\bar{r}_u)(r_{v,p}-\bar{r}_v)}{\sqrt{\sum_{p\in I_{uv}}(r_{u,p}-\bar{r}_u)^2}\cdot\sqrt{\sum_{p\in I_{uv}}(r_{v,p}-\bar{r}_v)^2}}\cdot\frac{1}{1+\exp\left(-\frac{|I_{uv}|}{2}\right)} \tag{5}$$

Jaccard 方法[2] 和均方差相似度算法(Mean Squared Difference, MSD)[9]也是很常用的计算相似度的方法,其中:Jaccard 方法只考虑到了用户共同评分物品的数量,没有考虑用户评分的数值;而 MSD 方法只考虑到了用户评分的数值,但没有考虑到用户共同评分的物品数量。有研究人员认为可以将两种方法的优缺点相互弥补,为此提出了杰卡尔德均方差(Jaccard Mean Squared Difference, JMSD)相似度算法[10]。这三种相似度计算方法的计算公式分别如下:

$$sim(u,v)^{\mathrm{Jaccard}}=\frac{|I_u\cap I_v|}{|I_u\cup I_v|} \tag{6}$$

其中,I_u 和 I_v 分别表示用户 u 和用户 v 评分物品的集合。

$$sim(u,v)^{\mathrm{MSD}}=1-\frac{1}{|I_{uv}|}\sum_{p\in I_{uv}}(r_{u,p}-r_{v,p})^2 \tag{7}$$

$$sim(u,v)^{\mathrm{JMSD}}=\frac{|I_u\cap I_v|}{|I_u\cup I_v|}\cdot\left(1-\frac{1}{|I_{uv}|}\sum_{p\in I_{uv}}(r_{u,p}-r_{v,p})^2\right) \tag{8}$$

Liu 等[11] 提出一种叫 NHSM(New Heuristic Similarity Model)的相似度计算方法,该方法由五个因子组成:Proximity、Significance、Singularity、Jaccard factor 和 URP(User Rating Preference)。该方法的计算公式如下:

$$sim(u,v)^{\mathrm{NHSM}}=sim(u,v)^{\mathrm{PSS}}\cdot sim(u,v)^{\mathrm{Jaccard'}}\cdot sim(u,v)^{\mathrm{URP}} \tag{9}$$

式(9)中 PSS(Proximity-Significance-Singularity)因子由 Proximity、Significance 和 Singularity 三个因子组成,$sim(u,v)^{\mathrm{PSS}}$ 计算方式如下:

$$sim(u,v)^{\mathrm{PSS}}=\sum_{p\in I}PSS(r_{u,p},r_{v,p}) \tag{10}$$

其中 $PSS(r_{u,p},r_{v,p})$ 是由三个部分构成,它的组成方式如下:

$$PSS(r_{u,p},r_{v,p})=Proximity(r_{u,p},r_{v,p})\cdot Significance(r_{u,p},r_{v,p})\cdot Singularity(r_{u,p},r_{v,p}) \tag{11}$$

$Proximity(r_{u,p},r_{v,p})$ 用于描述用户之间的评分差值对相似度的影响,它的计算公式如下:

$$Proximity(r_{u,p},r_{v,p})=1-\frac{1}{1+\exp(-|r_{u,p}-r_{v,p}|)} \tag{12}$$

$Significance(r_{u,p},r_{v,p})$ 描述的是用户评分与评分域的中值之间的关系,研究人员认为评分域的中值可以用来区分用户对该物品喜欢或不喜欢。它的计算方式如下:

$$Significance(r_{u,p},r_{v,p})=\frac{1}{1+\exp(-|r_{u,p}-r_{\mathrm{med}}|\cdot|r_{v,p}-r_{\mathrm{med}}|)} \tag{13}$$

$Singularity(r_{u,p},r_{v,p})$ 描述的是两个用户对一个共同评分物品的评分均值与该物品全局评分均值的差值对相似度的影响。它的计算公式如下:

$$Singularity(r_{u,p},r_{v,p})=1-\frac{1}{1+\exp\left(-\left|\frac{r_{u,p}+r_{v,p}}{2}-\mu_p\right|\right)} \tag{14}$$

其中,μ_p 表示物品 p 的评分均值。

$sim(u,v)^{\mathrm{Jaccard'}}$ 是一种改进型的 Jaccard 算法,它考虑用户共同评分物品数对相似度的影响。

$$sim(u,v)^{\mathrm{Jaccard'}}=\frac{|I_u\cap I_v|}{|I_u|\times|I_v|} \tag{15}$$

$sim(u,v)^{\mathrm{URP}}$ 考虑用户之间的评分均值的差值和用户评分方差的差值对相似度的影响。它的计算公式如下:

$$sim(u,v)^{URP} = 1 - \frac{1}{1+\exp(-|\mu_u-\mu_v|\cdot|\sigma_u-\sigma_v|)} \tag{16}$$

其中,σ_u 和 σ_v 分别表示用户 u 和用户 v 的评分方差。

以上九种算法可以分为三类:第一类只考虑了用户之间对同一个物品的评分而没有考虑共同评分物品的数量,包括 COS、ACOS、PCC、CPCC 和 MSD 方法。这类算法的缺点是会出现用户之间相似值偏高或偏低的现象。第二类只考虑了用户之间共同评分物品数而没有考虑用户之间对同一个物品的评分,Jaccard 方法属于这一类算法;这类算法在数据不是很稀疏时会表现出大量用户之间相似度值相等的情况,这样就难以发现用户组之间的差异。第三类是混合型算法,这类算法至少考虑了两点与用户间相似度计算相关的因素,如 SPCC、JMSD 和 NHSM 方法。这些与相似度计算相关的因素都在前两类算法的设计思想中有体现,例如 ACOS 方法考虑了用户评分均值对相似度的影响,MSD 方法考虑了用户评分差值对相似度的影响,Jaccard 方法考虑了共同评分物品数目对相似度计算的影响。这类算法与前两类算法相比在计算用户之间的相似度值时更加准确。另外根据文献[11]中的实验结果显示,NHSM 方法在数据稀疏的情况下能取得更好的推荐效果。

2 本文算法

NHSM 方法的五个因素中,有些因素还需要额外的计算,这使得 NHSM 方法十分复杂,而各个计算环节都会带来误差,多个计算误差相互叠加会增大偏离实际值的可能性,因此该方法的推荐效果在某些情况下不是很理想。本文将提出简化的 NHSM 方法。

2.1 设计思想

首先介绍一下实际应用场景中的状况。在现实场景中,系统中用户只给系统中少部分物品评分,这样用户-物品评分矩阵是非常稀疏的,所以应该将数据稀疏性问题引入到考虑范围中。下面是本文方法的考虑因素:

1)考虑用户评分的不同非常重要。在文献[12]的实验部分可以发现,大部分相似度计算方法考虑到了用户评分的不同。综合第 1 章介绍的九种方法可以发现,有三种方式可以引入用户评分的不同,一种是类似余弦相似度方法的那种乘积形式的计算公式,一种是 MSD 方法中的差值平方的方式,还有一种是 NHSM 方法中的指数函数的形式。在实际情况中,用户间评分差值和相似度是成反比的关系,即相似度越高评分差值越小。因此可以直接采用指数函数倒数的形式,这样处理没有 NHSM 方法那么复杂,同时还可以避免乘积形式中零除现象的出现。

2)用户共同评分物品所占的比例不能被忽略。一些研究者认为应该考虑共同评分物品所占的比例对相似度计算的影响,在前面的介绍中可以发现,SPCC[8]、JMSD[10]和 NHSM 方法[11]都引入了用户共同评分物品所占的比例,它们的实验结果显示这三种方法都提升了推荐效果。

3)应该考虑到用户评分的局部上下文和用户评分全局偏好。在 NHSM 方法[11]的设计思想中提到了这一条设计思想,而且它的实验结果显示,当数据集稀疏度越高时,NHSM 方法相比那些只考虑了用户评分的局部上下文的方法推荐效果要好。

4)应该使用用户全局评分均值区分用户对商品是喜欢还是讨厌。在 ACOS 方法[6]的设计思想中提到了用户评分偏好对相似度计算的影响。通过观察不同网站的评分页面可以发现,大多数页面中没有明显地提醒用户评分域的中间值是一个区分喜欢和讨厌的中间值,因此用户会根据自己的偏好进行评分。根据上述情况可以将 NHSM 方法中的 Significance 因子和 URP 因子合并。

5)NHSM 方法[11]中 Singularity 因子可能不如作者认为的那么有效。可以假设这样的情景,当评分域为 1~5,只有 5 个评分域且只有 5 个不同的评分时,假设用户 1 给物品 1 评分 1,用户 2 给物品 1 评分 5,用户 3 给物品 1 评分 3,用户 4 给物品 1 评分 3,可以发现用户 1 与用户 2 的 Singularity 值和用户 3 与用户 4 的 Singularity 值相同。这显然给相似度计算带来了误导。因为大多数评分数据集的评分粒度不够大,使用该因子不一定能使用户组之间区分度增大,反而增加了计算复杂度和计算误差。

综合所述,本文将 NHSM 方法中的五个因子简化为三个因子,并用一个更简化的方程来重新定义 Proximity 因子,同时将 Significance 因子和 URP 因子结合在一起重新定义了新的 Significance 因子,而且在将 Singularity 因子移除的同时保留原来的 Jaccard′因子。

2.2 数学表达式

将本文方法称为 PSJ(Proximity-Significance-Jaccard),其数学表达式定义如下:

$$sim(u,v)^{PSJ} = sim(u,v)^{PS}\cdot sim(u,v)^{Jaccard'} \tag{17}$$

式(17)中的 PS(Proximity-Significance)因子是由 Proximity 因子和 Significance 因子组成的,$sim(u,v)^{PS}$ 的计算方式如下:

$$sim(u,v)^{PS} = \sum_{p\in I} PS(r_{u,p},r_{v,p}) \tag{18}$$

其中,$PS(r_{u,p},r_{v,p})$ 因子是由 $Proximity(r_{u,p},r_{v,p})$ 和 $Significance(r_{u,p},r_{v,p})$ 两个因子组成的,计算方式如下:

$$PS(r_{u,p},r_{v,p}) = Proximity(r_{u,p},r_{v,p})\cdot Significance(r_{u,p},r_{v,p}) \tag{19}$$

根据 2.1 节中的设计思想中提到的方案,PSJ 方法中 Proximity 因子和 Significance 因子的计算方式定义如下:

$$Proximity(r_{u,p},r_{v,p}) = \frac{1}{\exp(|r_{u,p}-r_{v,p}|)} \tag{20}$$

$$Significance(r_{u,p},r_{v,p}) = \frac{1}{1+\exp(-|r_{u,p}-\mu_u|\cdot|r_{v,p}-\mu_v|)} \tag{21}$$

由于在 PSJ 方法的 Significance 因子已经考虑了用户全局评分喜好,所以 PSJ 方法中的 Significance 因子是将 NHSM 方法中的 Significance 因子和 URP 因子合并。

为了考虑共同评分物品的所占比例,PSJ 方法仍然使用 NHSM 算法中的 Jaccard′方法,通过实际的实验对比可知,它融合到 PSJ 方法中产生的推荐效果比原始的 Jaccard 方法融入到 PSJ 方法产生的推荐效果更好,它的计算方式定义如下:

$$sim(u,v)^{Jaccard'} = \frac{|I_u\cap I_v|}{|I_u|\times|I_v|} \tag{22}$$

通过实验发现 NHSM 方法还存在一些问题,即计算用户与用户自己的相似度时不是 1,而 PSJ 方法中也存在这样的问题。为了解决该问题,最终使用分段函数的方式定义 PSJ 方法:

$$sim(u,v)^{PSJ} = \begin{cases} sim(u,v)^{PS}\cdot sim(u,v)^{Jaccard'}, & u\neq v \\ 1, & u=v \end{cases} \tag{23}$$

2.3　讨论

为了说明 PSJ 方法能提高用户相似度计算的准确性,将引入一个用户-物品评分矩阵,如表 1 所示。这个矩阵中 $U_1 \sim U_6$ 代表 6 个用户,$I_1 \sim I_7$ 代表 7 个物品,评分域是 1 ~ 5,“—”代表用户没有评分的物品。

表 1　一个用户-物品评分矩阵

用户	评分						
	I_1	I_2	I_3	I_4	I_5	I_6	I_7
U_1	5	—	1	5	—	—	2
U_2	4	1	—	5	—	4	1
U_3	5	—	1	—	5	5	1
U_4	—	—	5	—	—	3	—
U_5	2	—	—	—	—	—	5
U_6	—	2	—	—	—	5	—

图 1 表示各个相似度算法的相似度计算结果。由于这些矩阵是对称的,这里只列举出了矩阵的上半部分,行从左到右表示用户 1 ~ 6,列从上到下表示用户 1 ~ 6。

1)可以在图 1(a)中发现用户 1 与用户 2 的相似度高于用户 1 与用户 3 之间的相似度。用户 2 的评分向量为(4,1,—,5,—,4,1),并且用户 3 的评分向量为(5,—,1,—,5,5,1)。通过观察可以发现用户 3 有两个评分分值和用户 1 相同,而用户 2 只有一个评分和用户 1 相同,这两个用户的其他的评分大致与用户 1 相同。因此用户 1 与用户 3 的相似度应该高于用户 1 与用户 2。这个错误也发生在 ACOS 方法中,PSJ 方法纠正了这个相似度计算错误。

2)仔细观察用户 1 与用户 3 的评分向量,在图 1(c)中可以发现,用户 1 与用户 3 的相似度达到了 0.997,用户 1 与用户 3 之间只有两个相同评分物品,这个值跟实际情况相比偏高,这个问题也发生在 CPCC、SPCC 和 MSD 方法中。在 PSJ 方法中的计算结果是 0.646,该结果足够用于描述用户 1 与用户 3 的相似度。

3)在图 1(f)中可以发现,用户 2 和用户 5 的相似度与用户 2 和用户 6 的相似度相等。实际上,在表 1 中用户 2 与用户 5 对共同评分物品的评分差值还是比较大的,用户 2 与用户 6 的相似度应该高于用户 2 与用户 5 之间的相似度。在 PSJ 方法中用户 2 与用户 5 中的相似度是 0.037,并且用户 2 与用户 6 的相似度为 0.117,与实际情况相符。

4)在图 1(c)中有许多相似度值为 NaN,其产生是因为零除问题。这样的问题也发生在 CPCC、SPCC、MSD 和 JMSD 方法中。将 JMSD 和 MSD 方法比较可以发现,JMSD 方法中 NaN 值的数量相对少一些。PSJ 方法的计算结果中没有 NaN 值,这体现了 PSJ 方法使用指数函数的倒数的优势。

(a) COS

1.0	0.825	0.430	0.117	0.501	0.0
	1.0	0.608	0.268	0.314	0.532
		1.0	0.391	0.317	0.529
			1.0	0.0	0.478
				1.0	0.0
					1.0

(b) ACOS

1.0	0.729	0.136	0.298	0.580	0.210
	1.0	0.283	0.234	0.283	0.480
		1.0	0.340	0.284	0.474
			1.0	0.572	0.776
				1.0	0.559
					1.0

(c) PCC

1.0	0.949	0.997	-1.0	-1.0	NaN
	1.0	1.0	NaN	-1.0	1.0
		1.0	-1.0	-1.0	NaN
			1.0	NaN	NaN
				1.0	NaN
					1.0

(d) CPCC

1.0	0.887	0.962	-1.0	-0.800	NaN
	1.0	0.943	NaN	-1.0	0.800
		1.0	-0.707	-0.949	1.0
			1.0	NaN	NaN
				1.0	NaN
					1.0

(e) SPCC

0.818	0.694	0.729	-0.622	-0.622	NaN
	0.818	0.622	NaN	-0.622	0.622
		0.731	-0.622	-0.622	NaN
			0.622	NaN	NaN
				0.622	NaN
					0.622

(f) Jaccard

1.0	0.5	0.5	0.2	0.5	0.0
	1.0	0.429	0.167	0.4	0.4
		1.0	0.4	0.4	0.167
			1.0	0.0	0.333
				1.0	0.0
					1.0

(g) MSD

1.0	0.5	1.0	-15.0	-8.0	NaN
	1.0	0.0	NaN	-3.0	0.0
		1.0	-15.0	-8.0	NaN
			1.0	NaN	NaN
				1.0	NaN
					1.0

(h) JMSD

1.0	0.167	0.334	-3.0	-4.0	NaN
	1.0	0.142	0.0	-3.6	0.0
		1.0	-3.6	-4.6	0.167
			1.0	NaN	-1.0
				1.0	NaN
					1.0

(i) NHSM

0.080	0.036	0.039	0.0007	0.0025	0.0
	0.102	0.007	0.0	0.004	0.019
		0.059	0.001	0.004	0.0
			0.016	0.0	0.0
				0.022	0.0
					0.069

(j) PSJ

1.0	0.285	0.646	0.006	0.015	0.0
	1.0	0.051	0.0	0.037	0.117
		1.0	0.008	0.023	0.0
			1.0	0.0	0.0
				1.0	0.0
					1.0

图 1　十种相似度算法对于表 1 中的矩阵的计算结果

5)在图 1(f)中许多的相似度值为 0.4 和 0.5。当仔细观察表 1 中各个用户的评分向量后可以发现,这些用户之间是彼此不同的,这些用户之间的相似度值也应该是不同的,这样才能更好地区分他们。在 PSJ 方法的计算结果中这个问题得到了解决。

6)在图 1(g)中许多用户的相似度值偏低,有些相似度值过于接近 0。同时 NHSM 方法中用户和用户自己的相似度还是不同,即不等于 1,这个问题也发生在 SPCC 方法中。在 PSJ 方法中这个问题得到了解决。

3 实验与分析

3.1 数据集

在实验中使用了最新的 MovieLens 数据集,该数据集被叫作 ml-latest-small。它包括 706 个用户,8 570 部电影和 100 023 个评分,每个用户至少给 20 部电影评过分。它与其他 MovieLens 数据集的不同之处是:它的评分粒度更细一些,其评分域是 0.5 ~ 5.0,有 10 个不同的评分值。它的用户-物品评分矩阵的密度是 1.7%。

本文选取了该数据集中的所有用户和前 5000 部电影,每个用户至少给 15 部电影评过分,用于实验的用户-物品评分矩阵密度仍然是 1.7%。在实验过程中,数据集被分成两部分,80% 的数据用于训练,剩下 20% 的数据用于测试。按照这样的数据处理方式在实验过程中一共进行了五次交叉验证,每次交叉验证的训练集和测试集都不相同。

3.2 相似度计算对推荐的影响

实验设计:首先使用数据集里的数据建立一个用户-物品矩阵;然后,用相似度计算方法计算用户之间的相似度,并建立用户相似度矩阵;接着,建立推荐列表。建立推荐列表的操作又分成两步:1)通过用户相似度矩阵找出与用户最相似度值排在前 k 的用户,用链表将他们从高到低记录下来;2)根据相似度值最高的用户给出的评分排列这前 k 位用户关注过的而目标用户从未关注过的物品,最终选取排在前 n 的物品推荐给目标用户。

对比方法分别是 ACOS 方法、SPCC 方法、Jaccard 方法、JMSD 方法和 NHSM 方法。因为推荐的物品数量和最近邻居的数量将影响到推荐效果,所以本文将实验情景分成两种状况:第一种状况是当邻近用户数是定值而推荐物品数是变量,第二种情况是推荐物品数是定值而邻近用户数是变量。最终的每条实验数据都由五次交叉实验的结果求平均值得出。

在上述实验中每个算法都要进行 35 次用户-用户相似度矩阵计算,在本章的 3.2.4 节中将通过用户-用户相似度矩阵计算的平均时间说明 PSJ 方法的优越性。

3.2.1 衡量尺度

在商业化推荐系统中,总是推荐给用户一个他或她可能喜欢的 k 件物品的列表,这种方式被称为 Top-k 推荐。本文对比实验也使用 Top-k 推荐,因此使用查全率和查准率[14-16]来衡量实验结果。

查准率定义如下:

$$Precision = \frac{1}{nk}\sum_{u} N(k,u) \tag{24}$$

其中:n 表示用户总数,$N(k, u)$ 表示推荐给用户 u 的 k 个物品中用户 u 实际接受的个数。

查全率定义如下,其中 M 表示测试集中数据总数:

$$Recall = \frac{1}{M}\sum_{u} N(k,u) \tag{25}$$

3.2.2 推荐物品数

本节实验将邻近用户数设为定值 20,推荐物品数从 10 到 70 变化。

图 2 展现的是当 Top-k 中 k 变化时不同方法查准率的变化。与 NHSM、ACOS 和 SPCC 方法相比,PSJ 方法的查准率提升比较明显;当 $k = 10$ 时,Jaccard 和 JMSD 方法的查准率与 PSJ 方法很接近;当 $k > 20$ 时,NHSM 方法获得了比 SPCC、JMSD 和 ACOS 方法更好的查准率,而 PSJ 方法能获得最好的查准率。

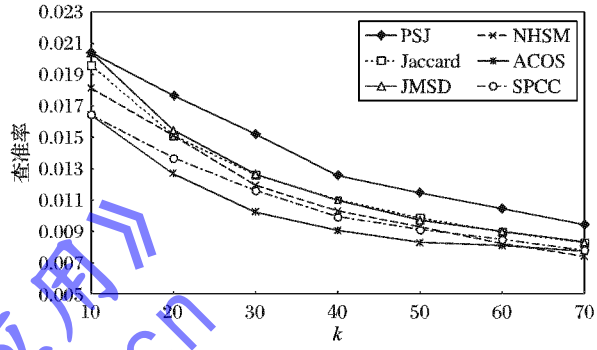

图 2 k 取不同值时数据集 ml-latest-small 上的查准率

图 3 展现的是当 Top-k 中 k 变化时不同方法查全率的变化。与 ACOS 和 SPCC 方法相比,PSJ 方法的查全率提升明显;当 $k = 10$ 时,Jaccard、JMSD 和 NHSM 方法的查全率与 PSJ 方法很接近;当 $k > 20$ 时,NHSM 方法获得了比 SPCC、JMSD 和 ACOS 方法更好的查全率,而 PSJ 方法获得了最好的查全率。

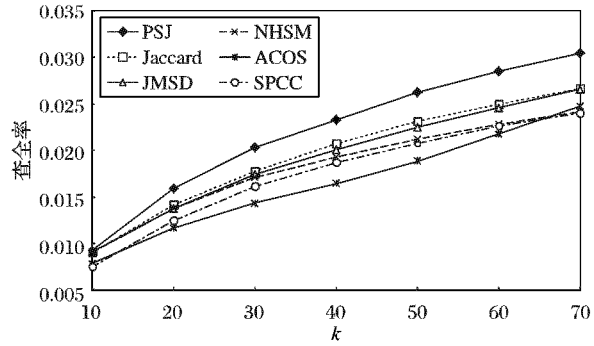

图 3 k 取不同值时数据集 ml-latest-small 上的查全率

图 2 和图 3 的实验结果表明,PSJ 方法考虑的各种因素对提升推荐效果是有效的。

3.2.3 最近邻居数量

本节实验将推荐物品数量设为定值 50,最近邻居数从 10 到 70 变化。

图 4 展现的是当最近邻居数变化时不同方法查准率的变化。在整个实验过程中,PSJ 方法能获得最高的查准率,与其他五种协同过滤方法相比,PSJ 方法的查准率提升较为明显。当最近邻居数大于 20 时,除了 ACOS 方法外,其他方法的查准率都比较稳定;当最近邻居数在 10 ~ 20 时,ACOS 方法的查准率下降很快,这是用户相似度计算偏差太大造成的。

图 5 展现的是当最近邻居数变化时不同方法查全率的变化。与查准率实验结果一样,PSJ 方法能获得最高的查全率,

与其他五种协同过滤方法相比,PSJ 方法的查全率提升较为明显。

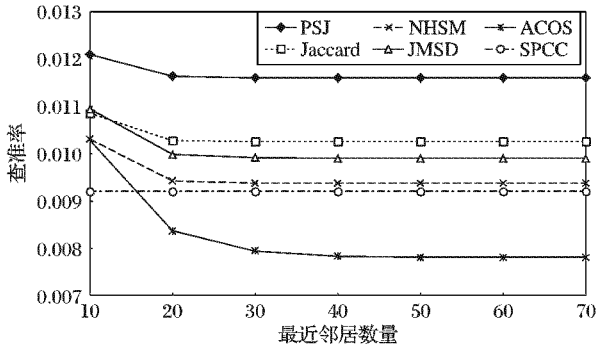

图 4　最近邻居数变化时数据集 ml-latest-small 上的查准率

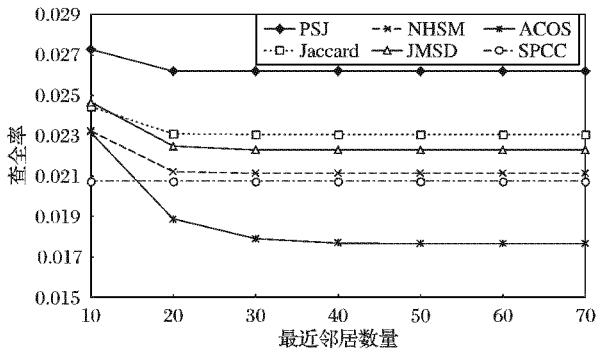

图 5　最近邻居数变化时数据集 ml-latest-small 上的查全率

图 4 和图 5 的实验结果表明,PSJ 方法在实验情景下与其他方法相比,其推荐效果也有明显提升。

3.2.4　不同算法的计算时间

本节通过用户-用户相似度矩阵的计算时间来对比 PSJ 算法与其他五种相似度算法。表 2 展示的是各个算法计算了 35 次用户-用户相似度矩阵的平均计算时间。从表 2 的数据中可以发现,PSJ 方法与 NHSM 方法相比计算时间缩短了 9.85%,但与其他四种相似度算法的计算时间仍有差距,继续降低 PSJ 算法的复杂度仍然很有必要。

表 2 不同方法计算相似度矩阵的平均计算时间　ms

方法	平均计算时间	方法	平均计算时间
PSJ	7930.21	NHSM	8796.56
Jaccard	6440.86	ACOS	5831.79
JMSD	6707.71	SPCC	6173.18

综合以上两组实验的实验结果,从两个不同情景说明了 PSJ 方法的有效性,其查准率与查全率与对比的五种协同过滤方法相比有不同幅度的提升;同时在实验中还对比了 PSJ 方法与其他五种方法计算用户-用户相似度矩阵的平均计算时间,这些实验结果说明了 PSJ 仍然需要降低复杂度。

4　结语

本文介绍了协同过滤推荐方法中使用的相似度计算方法,并在综合了这些相似度计算方法的优点之后,提出了 PSJ 方法的设计思路,并给出了 PSJ 方法的计算公式。PSJ 方法是基于 NHSM 方法的改进方法,并且在设计时考虑到了数据稀疏状况对产生推荐结果的影响。它简化了 NHSM 方法的 Proximity 因子的计算,将 NHSM 方法的 Significance 因子和 URP 因子合并组成了自己的 Significance 因子。通过这两步简化,使得 PSJ 方法的计算复杂度相较于 NHSM 方法明显降低。在 PSJ 方法的讨论中,通过一个用户-物品评分矩阵的例子验证了 PSJ 方法在相似度计算方面的准确性。在 MovieLens 的 ml-latest-small 数据集上对比实验结果验证了 PSJ 方法的有效性,结果表明,与对比的协同过滤方法相比,PSJ 方法可以有效提升推荐效果,同时在一定程度上克服了数据稀疏情况对推荐效果的影响。然而,在更为稀疏的数据集上推荐效果如何,以及如何改进是下一步需要深入研究的内容。

参考文献:

[1] YANG X, GUO Y, LIU Y, et al. A survey of collaborative filtering based social recommender systems [J]. Computer Communications, 2014, 41: 1 - 10.

[2] CACHEDA F, CARNEIRO V, FERNÁNDEZ D, et al. Comparison of collaborative filtering algorithms: limitations of current techniques and proposals for scalable, high-performance recommender systems [J]. ACM Transactions on the Web, 2011, 5(1): Article No. 2.

[3] KOUTRIKA G, BERCOVITZ B, GARCIA-MOLINA H. FlexRecs: expressing and combining flexible recommendations [C]// SIGMOD '09: Proceedings of the 2009 ACM SIGMOD International Conference on Management of Data. New York: ACM, 2009: 745 - 758.

[4] ADOMAVICIUS G, TUZHILIN A. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions [J]. IEEE Transactions on Knowledge and Data Engineering, 2005, 17(6): 734 - 749.

[5] RESNICK P, IACOVOU N, SUCHAK M, et al. GroupLens: an open architecture for collaborative filtering of netnews [C]// CSCW '94: Proceedings of the ACM Conference on Computer Supported Cooperative Work. New York: ACM, 1994: 175 - 186.

[6] AHN H J. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem [J]. Information Sciences, 2008, 178(1): 37 - 51.

[7] PATRA B K, LAUNONEN R, OLLIKAINEN V, et al. A new similarity measure using Bhattacharyya coefficient for collaborative filtering in sparse data [J]. Knowledge-Based Systems, 2015, 82: 163 - 177.

[8] JAMALI M, ESTER M. TrustWalker: a random walk model for combining trust-based and item-based recommendation [C]// KDD '09: Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2009: 397 - 406.

[9] BOBADILLA J, HERNANDO A, ORTEGA F, et al. Collaborative filtering based on significances [J]. Information Sciences, 2012, 185(1): 1 - 17.

[10] BOBADILLA J, SERRADILLA F, BERNAL J. A new collaborative filtering metric that improves the behavior of recommender systems [J]. Knowledge-Based Systems, 2010, 23(6): 520 - 528.

[11] LIU H, HU Z, MIAN A, et al. A new user similarity model to improve the accuracy of collaborative filtering [J]. Knowledge-Based Systems, 2014, 56: 156 - 166.

[12] BOBADILLA J, ORTEGA F, HERNANDO A, et al. A collaborative filtering approach to mitigate the new user cold start problem [J]. Knowledge-Based Systems, 2011, 26: 225 - 238.

[13] ANAND D, BHARADWAJ K K. Utilizing various sparsity measures for enhancing accuracy of collaborative recommender systems based on local and global similarities [J]. Expert Systems with Applications, 2011, 38(5): 5101 - 5109.

(下转第 2281 页)

pean Conference on Computer Vision. Berlin: Springer, 2014: 424 – 438.

[25] LIN Z, DING G, HU M, et al. Image tag completion via image-specific and tag-specific linear sparse reconstructions [C]// Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2013: 1618 – 1625.

[26] LIN Z, DING G, HU M, et al. Image tag completion via dual-view linear sparse reconstructions [J]. Computer Vision and Image Understanding, 2014, 124: 42 – 60.

[27] WANG Q, RUAN L, ZHANG Z, et al. Learning compact hashing codes for efficient tag completion and prediction [C]// Proceedings of the 22nd ACM International Conference on Conference on Information & Knowledge Management. New York: ACM, 2013: 1789 – 1794.

[28] WU L, JIN R, JAIN A K. Tag completion for image retrieval [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(3): 716 – 727.

[29] ZNAIDIA A, LE BORGNE H, HUDELOT C. Tag completion based on belief theory and neighbor voting [C]// Proceedings of the 3rd ACM Conference on International Conference on Multimedia Retrieval. New York: ACM, 2013: 49 – 56.

[30] SHAFER G. A Mathematical Theory of Evidence [M]. Princeton: Princeton University Press, 1976: 35 – 46.

[31] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks [C]// NIPS 2012: Advances in Neural Information Processing Systems. Cambridge: MIT Press, 2012: 1106 – 1114.

[32] CIRESAN D, MEIER U, SCHMIDHUBER J. Multi-column deep neural networks for image classification [C]// Proceedings of the 25th IEEE Conference on Computer Vision and Pattern Recognition. Piscataway, NJ: IEEE, 2012: 3642 – 3649.

[33] SRIVASTAVA N, SALAKHUTDINOV R. Learning representations for multimodal data with deep belief nets [C]// Proceedings of the 29th International Conference on Machine Learning Workshop. New York: ACM, 2012: 1 – 8.

[34] FENG F X, LI R F, WANG X J. Deep correspondence restricted Boltzmann machine for cross-modal retrieval [J]. Neurocomputing, 2015, 154: 50 – 60.

[35] 杨阳,张文生.基于深度学习的图像自动标注算法[J].数据采集与处理,2015,30(1):88 – 98.(YANG Y, ZHANG W S. Image auto-annotation based on deep learning [J]. Journal of Data Acquisition and Processing, 2015, 30(1):88 – 98.)

[36] DUYGULU P, BARNARD K, DE FREITAS J F G, et al. Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary [C]// Proceedings of the 2002 European Conference on Computer Vision. Berlin: Springer, 2002: 97 – 112.

[37] RUSSELL B C, TORRALBA A, MURPHY K P, et al. LabelMe: a database and Web-based tool for image annotation [J]. International Journal of Computer Vision, 2008, 77(1/2/3): 157 – 173.

[38] SHOTTON J, WINN J, ROTHER C, et al. TextonBoost: joint appearance, shape and context modeling for multi-class object recognition and segmentation [C]// ECCV 2006: Proceedings of the 9th European Conference on Computer Vision. Berlin: Springer, 2006: 1 – 15.

[39] HUISKES M J, LEW M S. The MIR flickr retrieval evaluation [C]// Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval. New York: ACM, 2008: 39 – 43.

[40] CHUA T S, TANG J, HONG R, et al. NUS-WIDE: a real-world Web image database from National University of Singapore [C]// Proceedings of the 2009 ACM International Conference on Image and Video Retrieval. New York: ACM, 2009: Article No. 48.

[41] GRUBINGER M, CLOUGH P, MÜLLER H, et al. The IAPR TC-12 benchmark: a new evaluation resource for visual information systems [C]// Proceedings of the 2006 International Workshop OntoImage Language Resources for Content-Based Image Retrieval. Genoa, Italy: [s. n.], 2006: 13 – 23.

[42] MAKADIA A, PAVLOVIC V, KUMAR S. Baselines for image annotation [J]. International Journal of Computer Vision, 2010, 90(1): 88 – 105.

[43] JÄRVELIN K, KEKÄLÄINEN J. Cumulated gain-based evaluation of IR techniques [J]. ACM Transactions on Information Systems, 2002, 20(4): 422 – 44.

Background

This work is partially supported by the National Natural Science Foundation of China (61572263), the Postdoctoral Science Foundation of China (2015M581794), the Project of Natural Science Research of Jiangsu University (15KJB520027), the Postdoctoral Science Foundation of Jiangsu Province (1501023C), and the Scientific Research Foundation of Nanjing University of Posts and Telecommunications (NY214127, NY215096).

LIU Mengdi, born in 1993, M. S. candidate. Her research interests include large scale machine learning.

CHEN Yanli, born in 1969, Ph. D., professor. Her research interests include intelligent information processing, network information security.

CHEN Lei, born in 1975, Ph. D., associate professor. His research interests include large scale machine learning, pattern recognition, data mining.

(上接第 2251 页)

[14] CREMONESI P, KOREN Y, TURRIN R. Performance of recommender algorithms on top-n recommendation tasks [C]// RecSys '10: Proceedings of the fourth ACM conference on Recommender Systems. New York: ACM, 2010: 39 – 46.

[15] FOUSS F, PIROTTE A, RENDERS J-M, et al. Random-walk computation of similarities between nodes of a graph with application to collaborative recommendation [J]. IEEE Transactions on Knowledge and Data Engineering, 2007, 19(3): 355 – 369.

[16] YU S J. The dynamic competitive recommendation algorithm in social network services [J]. Information Sciences: an International Journal, 2012, 187: 1 – 14.

Background

This work is partially supported by the Fundamental Research Funds for the Central Universities (JD1413).

ZHANG Nan, born in 1988, M. S. candidate. His research interests include artifical intelligence, data mining.

LIN Xiaoyong, born in 1979, Ph. D. candidate, associate professor. His research interests include Web 2.0 based social networking services, data mining.

SHI Shenghui, born in 1974, Ph. D. candidate, associate professor. Her research interests include big data analytics, compiling technique, biological information, natural language processing.