云南民族大学学报(自然科学版),2024,33(6):736 - 745　　CN 53 - 1192/N　ISSN 1672 - 8513
doi:10.3969/j.issn.1672 - 8513.2024.06.010　　http://ynmz.cbpt.cnki.net

Pearson 相关系数下非对称相似度计算及其应用

郑英丽,朴丽莎,王丽珍
(云南大学滇池学院 理工学院,云南 昆明 650228)

摘要: 稀疏性是推荐算法存在的问题之一,解决稀疏性问题的常用方法是矩阵分解,矩阵分解结合用户相似度可以提高推荐的准确率,但是传统的相似度计算方法并未考虑用户对项目评分数量的差异,因此构建的相似度矩阵是对称的.针对这一问题,结合 Pearson 相关系数,给出一种新的计算方法——用户非对称相似度.在考虑用户对相同项目评分的同时,计算用户间评分相同的项目数与用户所有评分项目数的比值,以此拉近用户之间相似的程度,且得到用户之间的非对称关系.其次,利用用户非对称相似度方法计算用户间相似度矩阵,将相似度矩阵与用户评分矩阵融入到概率矩阵分解框架中,实现用户的社会化推荐.在公开数据集上测试,结果显示改进的非对称相似度公式相比传统的相似度计算公式,在稀疏的数据集上进行社会化推荐能得到更准确的推荐结果.

关键词: 社会化推荐;非对称相似度;概率矩阵分解
中图分类号:TP393　　**文献标志码:**A　　**文章编号:**1672 - 8513(2024)06 - 0736 - 10

随着社会科技的快速发展,我们已经进入 5G 时代,社交媒体是人们生活必不可少的一部分,每天在其上产生与传播的信息不计其数[1],为了向不同用户推荐合适的信息,个性化推荐应运而生[2].一种专门针对用户的个性化推荐服务系统[3]进入了人们的视野,许多学者对其进行研究并改进.社会化推荐[4]的推荐方法有很多,协同过滤[5](collaborative filtering,简称 CF)是应用较多的算法之一,CF 又分为基于内存(memory - based collaborative filtering algorithm,简称 memory - based CF)和基于模型(model - based collaborative filtering algorithm,简称 Model - based CF)的推荐算法[6].相似度的计算对推荐算法有重要作用,本文聚焦于用户相似度的度量的研究.

度量相似度的传统方法,包括 Jaccard 相似度、余弦相似度、Pearson 相关系数等[4,7].文献[7]针对在稀疏数据集上使用传统相似性度量方法计算用户相似性时仅利用共同评级项目的评分这一问题,提出了基于邻域的协同过滤相似性度量方法(bhattacharyya coefficient for collaborative filtering,简称 BCF),BCF 方法在定位稀疏评级数据集中目标用户的邻居时全面地利用所有评级信息.因此,在稀疏数据上,BCF 方法可以可靠的为目标用户提供推荐.Liu 等[8]提出了一种基于 PIP 测量的相似性度量方法.考虑到不同的用户具有不同的评判标准,改进的相似性度量考虑了 2 个用户之间的共同评级的比例,使用评级的均值和方差来描述用户的评级偏好.实验结果显示该方法可以获得比大多数其他方法更好的性能.武聪等[9]基于用户标签与评分建立了用户标签相似度,基于 Jaccard 相似度计算用户相似度,再将 2 个相似度结合进行矩阵分解,提出了 UTagJMF 算法.李一野等[10]针对传统余弦相似度计算方法对数值不够敏感这一问题,使用调整余弦相似度,在将数据转换为特征矩阵时采用了神经网络的方法,数据经嵌入层映射后进行转换,减小了训练过程中负样本造成的不利影响,提高了稀疏性下数据分类的准确率.陈功平等[11]在传统的 Pearson 相关系数下引进了热门项目这一度量指标,但因为热门项目在相似度计算中影响较小,他们采用了评价数量作为热门项目的度量指标,同时针对共同评价项目过少这一问题设置了惩罚阈值,对 Pearson 相关系数进行了改进,实验结果显示

收稿日期:2023 - 09 - 09.
基金项目:云南大学滇池学院校级项目(2022XZC12);云南大学滇池学院校级重点项目(2022XZD03).
作者简介:郑英丽(1993 -),女,硕士,讲师.主要从事数据挖掘、应用数学研究.

改进的相关系数比原来的 Pearson 相关系数在预测评分的准确率更高. 虽然目前结合用户相关关系进行推荐的算法很多,计算相似度的改进方法也很多,但是大多都停留在对称相似度的计算上,并不能体现用户间的差异.

1　相关理论基础

1.1　矩阵分解

在推荐系统中,用户集 I 对项目集 J 的打分一般用一个二维矩阵 R 表示,R_{ij}的含义为用户 i 给项目 j 的评分值,没有给出评分的记为 0 或 null. 评分预测的目的就是要将这些 0 值补全,因此可以使用矩阵分解的方法. 矩阵分解[12](matrix factorization,简称 MF),顾名思义就是把一个用户评分矩阵 R 分解为一个用户特征矩阵 U 与一个项目特征矩阵 V 的乘积,即

$$R = U \times V. \tag{1}$$

分解后对未评分项的预测值为:

$$\hat{r}_{ij} = u_i^T v_j = \sum_{k=1}^{K} u_{ik} v_{kj}. \tag{2}$$

损失函数采用式(3):

$$e_{ij}^2 = (r_{ij} - \hat{r}_{ij})^2 = \left(r_{ij} - \sum_{k=1}^{K} u_{ik} v_{kj}\right)^2. \tag{3}$$

用梯度下降对损失函数最小化:

$$\frac{\partial e_{ij}^2}{\partial u_{ik}} = -2(r_{ij} - \hat{r}_{ij}) v_{kj} = -2e_{ij} v_{kj}, \frac{\partial e_{ij}^2}{\partial v_{kj}} = -2(r_{ij} - \hat{r}_{ij}) u_{ik} = -2e_{ij} u_{ik}. \tag{4}$$

更新之后的 u'_{ik}, v'_{kj}如下:

$$\begin{aligned} u'_{ik} &= u_{ik} - \alpha \frac{\partial e_{ij}^2}{\partial u_{ik}} = u_{ik} + 2\alpha e_{ij} v_{kj}, \\ v'_{kj} &= v_{kj} - \alpha \frac{\partial e_{ij}^2}{\partial v_{kj}} = v_{kj} + 2\alpha e_{ij} u_{ik}. \end{aligned} \tag{5}$$

矩阵分解有 2 个特点:1)可以得到用户偏好和项目特征;2)降低了评分矩阵的维度.

1.2　概率矩阵分解

在 MF 方法的基础上,概率矩阵分解[13](probabilistic matrix actorization,简称 PMF)增加正则项来防止过拟合,并假设评分矩阵 R 中的元素 R_{ij}是由用户偏好向量 U_i 和项目特征向量 V_j 的内积所决定的,且服从均值为 $U_i^T V_j$,方差为 σ^2 的正态分布,即

$$R_{ij} \sim N(U_i^T V_j, \sigma^2). \tag{6}$$

则可观测到评分矩阵 R 的条件概率为:

$$p(R \mid U, V, \sigma_R^2) = \prod_{i=1}^{M} \prod_{j=1}^{N} [N(U_i^T V_j, \sigma_R^2)]^{I_{ij}^R}. \tag{7}$$

假设 U 和 V 均服从均值为 0 的球形高斯先验,则:

$$\begin{aligned} p(U \mid \sigma_U^2) &= \prod_{i=1}^{M} N(U_i \mid 0, \sigma_U^2 I), \\ p(V \mid \sigma_V^2) &= \prod_{j=1}^{N} N(V_j \mid 0, \sigma_V^2 I). \end{aligned} \tag{8}$$

由贝叶斯推断,可以得到 U 和 V 的联合后验概率分布为:

$$\begin{aligned} p(U, V \mid R, \sigma_R^2, \sigma_U^2, \sigma_V^2) &\propto p(R \mid U, V, \sigma_R^2) p(U \mid \sigma_u^2) p(V \mid \sigma_V^2) \\ &= \prod_{i=1}^{M} \prod_{j=1}^{N} N(U_i^T V_j, \sigma_R^2)^{I_{ij}^R} \times \prod_{i=1}^{M} N(U_i \mid 0, \sigma_U^2 I) \times \prod_{j=1}^{N} N(V_j \mid \sigma_V^2 I). \end{aligned} \tag{9}$$

对等式两边取对数可以得到

$$\ln p(U,V\mid R,\sigma_R^2,\sigma_U^2,\sigma_V^2)=-\frac{1}{2\sigma_R^2}\sum_{i=1}^{M}\sum_{j=1}^{N}I_{ij}^R(R_{ij}-U_i^TV_j)^2-\frac{1}{2\sigma_U^2}\sum_{i=1}^{M}U_i^TU_i-\frac{1}{2\sigma_V^2}\sum_{j=1}^{N}V_j^TV_j$$
$$-\frac{1}{2}\left[\left(\sum_{i=1}^{M}\sum_{j=1}^{N}I_{ij}^R\right)\ln\sigma_R^2+\frac{1}{2}(ml\ln\sigma_U^2+nl\ln\sigma_V^2)\right]+C. \tag{10}$$

Salakhutdinov 等[12]在以上的基础上使用逻辑函数 $g(x)=1/(1+e^{-x})$ 将预测值 $(U_i^TV_j)$ 限定在区间[0,1]内. 他们得到的条件概率分布为:

$$p(R\mid U,V,\sigma_R^2)=\prod_{i=1}^{M}\prod_{j=1}^{N}[N(R_{ij}\mid g(U_i^TV_j),\sigma_R^2)]^{I_{ij}^R}. \tag{11}$$

同时他们使用函数 $f(x)=(x-1)/(R_{\max}-1)$ 将评分限制在[0,1]内.

1.3 相似度计算方法

余弦相似度:将用户 i 对所有项目的评分看作一个向量,则所有评分够成了 n 维项目空间上的向量,通过计算两向量间夹角余弦值来度量两用户间的相似程度[13],夹角越小,余弦值就越接近1,相似度就越大. 余弦相似度取值范围为[0,1],计算公式如下.

$$\mathrm{Cos}(x,y)=\frac{\sum_{i\in I}r_{x,i}\cdot r_{y,i}}{\sqrt{\sum_{i\in I}(r_{x,i})^2}\sqrt{\sum_{i\in I}(r_{y,i})^2}}. \tag{12}$$

其中,I 为全部评分数据集,$r_{x,i}$,$r_{y,i}$分别表示用户 x 和 y 对项目 i 的评分值. 由于评分矩阵中零值较多,用公式(12)对相似度进行计算,导致误差变大. 例如,两个用户 x 和 y 分别对物品 i 和 j 进行评分(5分制),x 的评分为(5,4),y 的评分为(2,1),使用余弦相似度得到 $\mathrm{Cos}(x,y)=0.98$,从计算结果上看用户 x 和 y 是极为相似的,但是从用户评分可以很容易的看出用户 x 相对于用户 y 更加喜欢这两个物品. 因此余弦相似度对数据的不敏感性容易导致不准确的预测.

调整余弦相似度:考虑到余弦相似度对数值的不敏感,为了修正这种不合理性,将用户的评分均值考虑进来,便得到了调整余弦相似度,取值范围为[-1,1],公式如下.

$$\mathrm{Aco}(x,y)=\frac{\sum_{i\in I}(r_{x,i}-\bar{r}_x)(r_{y,i}-\bar{r}_y)}{\sqrt{\sum_{i\in I}(r_{x,i}-\bar{r}_i)^2}\sqrt{\sum_{i\in I}(r_{y,i}-\bar{r}_i)^2}}. \tag{13}$$

其中,$\bar{r}_x$ 和 $\bar{r}_y$ 分别表示用户 x 和 y 评分的均值. 在上面的例子中,若 $\bar{r}_x=3$,$\bar{r}_y=2$ 则根据改进的余弦相似度公式,$\mathrm{Aco}(x,y)=-0.45$,相似度为负值,更加符合实际.

Jaccard 相似度:Jaccard 相似度主要应用在布尔值或符号度量的个体间相似度计算上,取值范围为[0,1],如式(14)所示.

$$\mathrm{Jac}(x,y)=\frac{N(x)\cap N(y)}{N(x)\cup N(y)}. \tag{14}$$

其中,$N(x)\cap N(y)$表示用户 x 和用户 y 共同有评分的项目数量,$N(x)\cup N(y)$表示用户 x 与用户 y 评分项目数量的并集. 该方法的缺点就是元素的取值只能是0或1,不能利用用户更丰富的评分信息,也没有将用户的具体评分值考虑进去.

Pearson 相关系数取值范围为[-1,1],如式(15)所示.

$$\mathrm{Pear}(x,y)=\frac{\sum_{i\in I_{xy}}(r_{x,i}-\bar{r}_x)(r_{y,i}-\bar{r}_y)}{\sqrt{\sum_{i\in I_x}(r_{x,i}-\bar{r}_x)^2}\sqrt{\sum_{i\in I_y}(r_{y,i}-\bar{r}_y)^2}}. \tag{15}$$

I_{xy}表示用户 x 与用户 y 共同评分的项目集,I_x 为用户 x 的评分项目集,I_y 为用户 y 的评分项目集. Pearson 相关系数计算时考虑的是用户 x 与用户 y 的共同评分集,去除了未评分项目的影响,但是该方法在计算时使用的是用户对项目的绝对评分值,忽略了对共同评分项目数量上的比例依赖关系[14].

2　相似度计算方法改进

无论是以上介绍的哪个公式,计算得到的用户间相似度都是对称的,即任意 2 个用户 x 和 y 之间的相似度 $\mathrm{sim}(x,y)$ 和 $\mathrm{sim}(y,x)$ 是一样的,但是在实际场景中,由于不同用户所评分的项目和评分的项目数不同[15],相似度 $\mathrm{sim}(x,y)$ 和 $\mathrm{sim}(y,x)$ 通常是不相等的. 比如用户 x 对电影 1,2,3,4 进行了评分,用户 y 对电影 2,3,6 进行了评分,则由表计算可得,$\mathrm{sim}(x,y)$ 和 $\mathrm{sim}(y,x)$ 是不一样的.

表 1　非对称相似度计算举例

用户	评分电影编号	相同项目比例
x	1,2,3,4	1/2
y	2,3,6	2/3

余弦相似度公式在度量用户间相似程度时没有考虑到用户的评分等级差异,往往会把 2 个不相似的用户视为有相同的兴趣爱好. 而调整余弦相似度和 Pearson 相关系数则将用户的评分均值结合进来,规避了评分等级区别大这一问题,且得到的值有正有负,相比余弦相似度更加符合实际情况,但是得到的相似度矩阵仍然是对称的. Jaccard 相似度则是对用户都对同一项目进行观看、浏览、点击等行为来进行度量,即 2 个用户都观看过同一部电影,且计算 Jaccard 相似度时取值只能是 0 或 1,并没有考虑用户的具体评分值,因此同样具有同余弦相似度一样的缺点.

基于以上考虑,针对 Jaccard 相似度和 Pearson 相关系数的优点与存在的不足,对相似度公式进行改进.

定义 1　通过对现有方法的分析,定义一种新的相似度计算方法,称为 PSR 相关系数,其计算公式如下.

$$\mathrm{PSR}(x,y)=\begin{cases} s(x,y)\cdot\mathrm{Pear}(x,y)=\dfrac{I(x)\cap I(y)}{N(x)}\cdot\dfrac{\sum\limits_{i\in I_{xy}}(r_{x,i}-\bar r_x)(r_{y,i}-\bar r_y)}{\sqrt{\sum\limits_{i\in I_x}(r_{x,i}-\bar r_x)^2}\sqrt{\sum\limits_{i\in I_y}(r_{y,i}-\bar r_y)^2}}, I(x)\cap I(y)\neq\varnothing; \\ \mathrm{Pear}(x,y)=\dfrac{\sum\limits_{i\in I_{xy}}(r_{x,i}-\bar r_x)(r_{y,i}-\bar r_y)}{\sqrt{\sum\limits_{i\in I_x}(r_{x,i}-\bar r_x)^2}\sqrt{\sum\limits_{i\in I_y}(r_{y,i}-\bar r_y)^2}}, I(x)\cap I(y)=\varnothing. \end{cases} \tag{16}$$

其中,$s(x,y)=\dfrac{I(x)\cap I(y)}{N(x)}$,$s(y,x)=\dfrac{I(x)\cap I(y)}{N(y)}$这里 $I(x)\cap I(y)$ 表示用户 x 与用户 y 对项目评分相同的个数,$N(x)$ 代表用户 x 评分项目的总数量. $s(x,y)$ 计算得到的是用户 x 与用户 y 评分相同的项目数量占用户 x 所有评分项目数量的比例,反之,$s(y,x)$ 计算得到的是用户 x 与用户 y 评分相同的项目数量占用户 y 所有评分项目数量的比例. 当用户有评分相同的项目时,在 Pearson 相似度前乘上 $s(x,y)$,用户之间的非对称关系由用户的所有评分项目数解释,实际相似程度由两用户相同评分项目数来修正;当没有评分相同的项目时,用 Pearson 系数填充.

引理 1　PSR 相关系数的取值范围为 $[-1,1]$.

证明　(1)当 $I(x)\cap I(y)\neq\varnothing$ 时,$\mathrm{PSR}(x,y)=s(x,y)\cdot\mathrm{Pear}(x,y)$.

①$I(x)\cap I(y)=N(x)=N(y)$,即 x 与 y 评分与评分项数完全相同时,$s(x,y)=1$,此时 $r_{x,i}=r_{y,i}$,$\bar r_x=\bar r_y$,因此 Pear = 1,PSR = 1;

②$I(x)\cap I(y)\subset N(x)$ $(I(x)\cap I(y)\subset N(y))$,即 x 与 y 有评分相同的项,$0<s(x,y)<1$,当 x 与 y 出现评分极端的情况下,即 $r_{x,i}\equiv5$,$r_{y,i}\equiv0$ $(r_{x,i}\equiv0,r_{y,i}\equiv5)$ 时,$Pear=-1$,因此 $Pear(x,y)\in[-1,0)\cup(0,1]$,因此 $\mathrm{PSR}(x,y)\in(-1,0)\cup(0,1]$.

(2)当 $I(x)\cap I(y)=\varnothing$ 时,$\mathrm{PSR}(x,y)=\mathrm{Pear}(x,y)$:当 $I_{xy}=\varnothing$ 时,即 x 与 y 没有共同评分项目,此时 Pear = 0,在(1)中已证明 $\mathrm{Pear}(x,y)\in[-1,0)\cup(0,1]$,因此 $Pear(x,y)\in[-1,1]$.

综上,$\mathrm{PSR}(x,y)\in[-1,1]$. 证毕

例如有 5 个用户 $(u_i,i=1,\cdots,5)$ 对 4 个项目 $(v_j,j=1,\cdots,4)$ 的评分如表 2 所示,用户未对项目进行评分的用 * 表示.

对表 2 采用公式(12)~(16)计算其相似度如表 3 ~ 表 7 所示:

表2 用户-项目评分表

项目用户	v_1	v_2	v_3	v_4
u_1	1	*	2	3
u_2	*	*	2	1
u_3	5	1	*	3
u_4	1	*	*	1
u_5	*	3	5	*

表3 用户 u_1 与其它用户的相似度

用户方法	u_2	u_3	u_4	u_5
Cos	0.837	0.632	0.756	0.458
Acos	0.556	0.376	0.397	-0.277
Jaccard	0.667	0.5	0.667	0.250
Pearson	0.473	-0.048	0.426	0.286
PSR	0.158	-0.016	0.142	0.286

表4 用户 u_2 与其它用户的相似度

用户方法	u_1	u_3	u_4	u_5
Cos	0.837	0.227	0.316	0.767
Acos	0.556	-0.422	0.486	-0.055
Jaccard	0.667	0.250	0.333	0.333
Pearson	0.473	0.047	0.139	0.930
PSR	0.237	0.047	0.069	0.930

表5 用户 u_3 与其它用户的相似度

用户方法	u_1	u_2	u_4	u_5
Cos	0.632	0.227	0.956	0.087
Acos	0.376	-0.422	0.364	-0.641
Jaccard	0.500	0.250	0.667	0.250
Pearson	-0.048	0.047	0.795	-0.127
PSR	-0.161	0.047	0.795	-0.127

表6 用户 u_4 与其它用户的相似度

用户方法	u_1	u_2	u_3	u_5
Cos	0.756	0.316	0.956	0
Acos	0.397	0.486	0.364	-0.766
Jaccard	0.667	0.333	0.667	0
Pearson	0.426	0.139	0.795	0
PSR	0.213	0.069	0.795	0

表7 用户 u_5 与其它用户的相似度

用户方法	u_1	u_2	u_3	u_4
Cos	0.458	0.767	0.087	0
Acos	-0.277	-0.055	-0.641	-0.766
Jaccard	0.250	0.333	0.250	0
Pearson	0.286	0.930	-0.127	0
PSR	0.286	0.930	-0.127	0

由表2可知,用户 u_1 与 u_2 对4个项目的评分分别为(1,*,2,3)和(*,*,2,1),有2个共同评分项目,但只有一项评分相同的项目,对项目 v_4 的评分上出现了差异,因此2个用户评分相似度不会很高,而由表3可得传统的4种方法得到的相似度都高于0.47,即传统方法会得到无论2个用户评分有多大差异,都可能得到较高的相似度. 本文方法得到 $\mathrm{sim}(u_1,u_2)=0.158$,从评分项目的数量上来看 u_1 与 u_2 的相似度是非对称的. 用户 u_3 与 u_4 的评分分别(5,1,*,3)和(1,*,*,1),与 u_1 都有两项共同评分项,其中一项评分相同的项目,但是 u_1 与 u_3 对 v_1 评分分别为1、5,而 u_1 与 u_4 对 v_4 评分分别为3、1,前者差异更大. 因此 u_1 与 u_3、u_4 相似度不会很高,且前者小于后者,但余弦、Jaccard得到的相似度都大于0.5,调整余弦都为正数. 本文方法得到的相似度分别为-0.016、0.142,u_1 与 u_3 评分差异大,相似度为负数更合理. 综合来看,本文的方法得到的相似度更加准确.

3 在社会化推荐中的应用

3.1 模型建立

假设用户集用 $U=\{u_k\}_{k=1}^{m}$ 表示,用户间关联关系用集合 E 表示,则用户点与关系边构成图 $G=(U,E)$. 由图 G 可得到用户的相关关系矩阵 $S=\{s_{ik}\}(i,k=1,2,\cdots,m)$,对 $\forall u_i、u_k\in U$,如果 u_i 和 u_k 相关,则连接 u_i 与 u_k 的边的权重为 $s_{ik}\in(0,1]$,如果 u_i 和 u_k 不相关,则 $s_{ik}=0$.

用户相似度概率模型如图1所示.

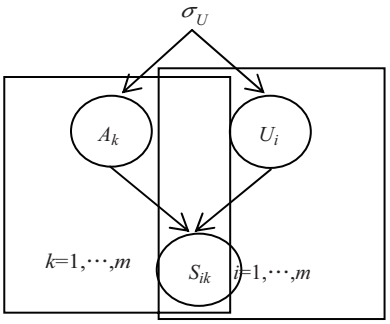

图1 用户相似度概率模型图

为了得到矩阵 S 中缺失的值,对矩阵 S 进行分解以降低维度,得到降维之后的潜在特征矩阵 U 和 A,而求解 U_i 和 A_k 的关键元素是相似权重 s_{ik},定义 S 的后验概率分布如下.

$$p(S\mid U,A,\sigma_S^2)=\prod_{i=1}^{M}\prod_{k=1}^{M}[N(s_{ik}\mid g(U_i^{\mathrm{T}}A_k),\sigma_S^2)]^{I_{ik}^S}. \tag{17}$$

其中,I_{ik}^{S}为指示函数,若 $s_{ik}\in(0,1]$,则值为 1,否则为 0.

此时,结合矩阵 S 和矩阵 R 进行联合分解,其概率模型如图 2 所示. U,V,A 的联合后验概率分布表示为:

$$p(U,V,A\mid R,S,\sigma_R^2,\sigma_S^2,\sigma_U^2,\sigma_V^2,\sigma_A^2)\propto p(R\mid U,V,\sigma_R^2)p(S\mid U,A,\sigma_S^2)\cdot p(U\mid\sigma_U^2)p(V\mid\sigma_V^2)p(A\mid\sigma_A^2)=$$
$$\prod_{i=1}^{M}\prod_{j=1}^{N}\left[N(R_{ij}\mid g(U_i^{\mathrm{T}}V_j),\sigma_R^2)\right]^{I_{ij}^{R}}\times\prod_{i=1}^{M}\prod_{k=1}^{M}\left[N(S_{ik}\mid g(U_i^{\mathrm{T}}A_k),\sigma_S^2)\right]^{I_{ik}^{S}}\times\prod_{i=1}^{M}N(U_i\mid 0,\sigma_U^2I)$$
$$\times\prod_{j=1}^{N}N(V_j\mid 0,\sigma_V^2I)\times\prod_{k=1}^{M}N(A_k\mid 0,\sigma_A^2I). \tag{18}$$

对(18)式两边取对数后可以得到:

$$\ln p(U,V,A\mid R,S,\sigma_R^2,\sigma_S^2,\sigma_U^2,\sigma_V^2,\sigma_A^2)=-\frac{1}{2\sigma_R^2}\sum_{i=1}^{M}\sum_{j=1}^{N}I_{ij}^{R}(R_{ij}-g(U_i^{\mathrm{T}}V_j))^2$$
$$-\frac{1}{2\sigma_S^2}\sum_{i=1}^{M}\sum_{k=1}^{M}I_{ik}^{S}(S_{ik}-g(U_i^{\mathrm{T}}A_k))^2-\frac{1}{2\sigma_U^2}\sum_{i=1}^{M}U_i^{\mathrm{T}}U_i-\frac{1}{2\sigma_V^2}\sum_{j=1}^{N}V_j^{\mathrm{T}}V_j-\frac{1}{2\sigma_A^2}\sum_{k=1}^{M}A_k^{\mathrm{T}}A_k$$
$$-\frac{1}{2}\left(\sum_{i=1}^{M}\sum_{j=1}^{N}I_{ij}^{R}\right)\ln\sigma_R^2-\frac{1}{2}\left(\sum_{i=1}^{M}\sum_{k=1}^{M}I_{ik}^{S}\right)\ln\sigma_S^2-\frac{1}{2}(md\ln\sigma_U^2+nd\ln\sigma_V^2+md\ln\sigma_A^2)+C. \tag{19}$$

其中 C 是不依赖参数的常量.

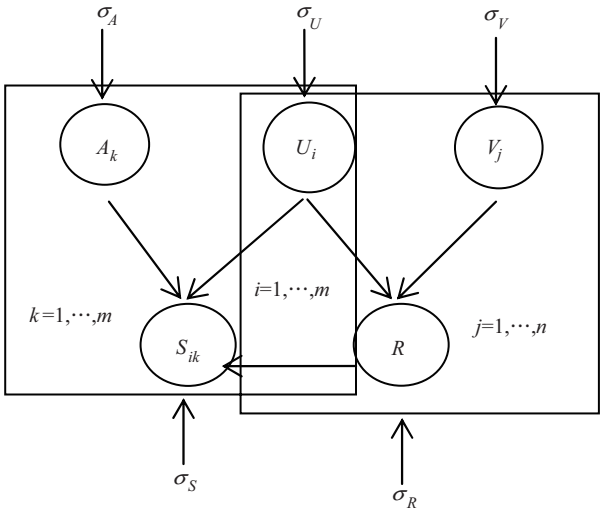

图 2　结合相似度的概率矩阵分解模型图

要求式(18)的概率,可转化为求式(20)的最小化.

$$E(R,S,U,V,A)=\frac{1}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}I_{ij}^{R}(R_{ij}-g(U_i^{\mathrm{T}}V_j))^2+\frac{\lambda_S}{2}\sum_{i=1}^{M}\sum_{k=1}^{N}I_{ik}^{S}(S_{ik}-g(U_i^{\mathrm{T}}A_k))^2+$$
$$\frac{\lambda_U}{2}\|U\|_F^2+\frac{\lambda_V}{2}\|V\|_F^2+\frac{\lambda_A}{2}\|A\|_F^2. \tag{20}$$

其中,$\lambda_S=\dfrac{\sigma_R^2}{\sigma_S^2},\lambda_U=\dfrac{\sigma_R^2}{\sigma_U^2},\lambda_V=\dfrac{\sigma_R^2}{\sigma_V^2},\lambda_A=\dfrac{\sigma_R^2}{\sigma_A^2}$,$\|\cdot\|_F^2$ 表示 F 范数. 用式(20)分别对 U,V,A 求偏导得式(21),以计算式(20)的最小值.

$$\frac{\partial E}{\partial U_i}=\sum_{j=1}^{n}I_{ij}^{R}g'(U_i^{\mathrm{T}}V_j)(g(U_i^{\mathrm{T}}V_j)-R_{ij})V_j+\lambda_S\sum_{k=1}^{m}I_{ik}^{S}g'(U_i^{\mathrm{T}}A_k)(g(U_i^{\mathrm{T}}A_k)-S_{ik})A_k+\lambda_uU_i,$$
$$\frac{\partial E}{\partial V_j}=\sum_{i=1}^{m}I_{ij}^{R}g'(U_i^{\mathrm{T}}V_j)(g(U_i^{\mathrm{T}}V_j)-R_{ij})U_i+\lambda_VV_j, \tag{21}$$
$$\frac{\partial E}{\partial A_k}=\lambda_S\sum_{i=1}^{m}I_{ik}^{S}g'(U_i^{\mathrm{T}}A_k)(g(U_i^{\mathrm{T}}A_k)-S_{ik})U_i+\lambda_AA_k.$$

其中:$g'(x)=g(x)(1-g(x))$是逻辑函数 $g(x)$的导数.

3.2 算法步骤

算法:基于 PSR 用户相似度的社会化推荐算法

输入:用户评分数据集

输出:用户相似度矩阵 S 和预测评分矩阵 R'

Step 1 根据导入的数据集生成用户评分矩阵 R

Step 2 根据评分矩阵 R 利用公式(16)计算用户相似度矩阵 S

Step 3 使用函数 $f(x)=x/R_{\max}$ 对评分矩阵归一化处理

Step 4 随机初始化特征矩阵 U,V,A,迭代次数 L

Step 5 根据公式(21)对梯度 $U_i^{L+1},V_j^{L+1},A_k^{L+1}$ 进行更新

Step 6 求损失函数 $E(R,S,U,V,A)$,$until E(L)-E(L-1)<\varepsilon$ 返回 U,V,A

Step 7 根据返回的特征矩阵得到填充后的评分矩阵 R' 和相似度矩阵 S'

Step 8 使用预测函数对评分进行预测

3.3 算法复杂度分析

分解模型是本文模型最耗时的部分,即目标函数 E 和各个目标变量梯度($\frac{\partial E}{\partial U_i},\frac{\partial E}{\partial V_j},\frac{\partial E}{\partial A_k}$)的计算. 因为评分矩阵 R 和相似度矩阵 S 中存在很多缺失值,用 ρ_R,ρ_S 分别表示 R、S 中非零数目,则 E 的计算复杂度为 $O(\rho_R d+\rho_S d)$,其中,d 为潜在特征维数. 目标变量梯度 $\frac{\partial E}{\partial U_i},\frac{\partial E}{\partial V_j},\frac{\partial E}{\partial A_k}$ 的计算复杂度分别为 $O(\rho_R d+\rho_S d)$、$O(\rho_R d)$、$O(\rho_S d)$. 所以经过一次迭代后的算法复杂度为 $O(\rho_R d+\rho_S d)$,因此,该模型算法复杂度与矩阵 R、S 的有效数据呈线性关系[15].

4 实验分析

为了验证上文提出方法的优越性,本节通过实验对比上文提出的相似度方法与一些传统相似度方法在评分预测上的效果. 先根据用户评分得到用户间相似度,再将相似度与用户评分融入概率矩阵分解模型中,在用户评分相同的情况下,以不同相似度方法得到的预测准确率来衡量方法的好坏.

4.1 实验数据集

在实验中,采用 MovieLens Latest Datasets,详细信息如表 8 所示.

表 8 实验数据集描述

数据集	类别	用户数	项目数	评分数	评分等级	稀疏等级
MoviesLen's Latest Datasets	电影	610	9 742	100 836	[0.5-5]	0.983 0

4.2 分析比较

文中采用 MAE(mean absolute error 平均绝对误差)和 RMSE(root mean square error 均方根误差)这 2 个指标进行评估,它们的计算公式如下.

$$\mathrm{MAE}=\frac{1}{N}\sum_{i,j}|r_{i,j}-\hat{r}_{i,j}|. \tag{22}$$

其中,$r_{i,j}$ 为用户真实评分,$\hat{r}_{i,j}$ 为预测评分,N 为样本容量.

$$\mathrm{RMSE}=\sqrt{\frac{1}{N}\sum_{i,j}(r_{i,j}-\hat{r}_{i,j})^2}. \tag{23}$$

评价指标 MAE 和 RMSE 值越大,预测效果越差,所作出的推荐质量越差.

本文采用如下预测函数对评分进行预测[17].

$$\hat{R}_{u_a,i}=\bar{r}_{u_a}+\frac{\sum_{u_k\in U_n}(R_{u_k,i}-\bar{r}_{u_k})\times\mathrm{sim}(u_a,u_k)}{\sum_{u_k\in U_n}|\mathrm{sim}(u_a,u_k)|}. \tag{24}$$

其中，$\hat{R}_{u_a,i}$为预测的用户 u_a 对项目 i 的评分，$\bar{r}_{u_a}$和 $\bar{r}_{u_k}$分别表示用户 u_a、u_k 的评分均值，sim(u_a,u_k)代表用户 u_a 和 u_k 的相似度.

为了验证上文提出改进的相似度方法(PSR)的优越性，将该方法分别与 Jaccard 相似度(Jac)、余弦相似度(Cos)、调整余弦相似度(Aco)、Pearson 相关系数(Pear)的推荐方法进行评分预测比较.

实验参数选择：

在进行实验前，先对实验中涉及到的参数进行设置，在指标 MAE 和 RMSE 下对各种方法中的参数 λ_s 计算最优值，如表 9 所示.

表 9　实验参数选择

方法	参数				
	Jac	Cos	Aco	Pear	PSR
λ_U	0.01	0.01	0.01	0.01	0.01
λ_V	0.01	0.01	0.01	0.01	0.01
λ_S	10	20	5	5	15

4.3　实验结果

基于以上参数值，表 10 和表 11 分别展现了特征维度 d 分别为 5 维和 10 维时本文提出的改进相似度方法较其它方法在指标 MAE、RMSE 上的提升(表中数据为多次试验的均值).

表 10　$d=5$ 时实验结果对比

方法	指标					
	Jac	Cos	Aco	Pear	PRS	PSR 降低/%
MAE	0.655	0.676	0.678	0.674	0.639	1.60
RMSE	0.800	0.815	0.816	0.814	0.790	1.00

表 11　$d=10$ 时实验结果对比

方法	指标					
	Jac	Cos	Aco	Pear	PRS	PSR 降低/%
MAE	0.735	0.744	0.746	0.744	0.725	1.00
RMSE	0.854	0.860	0.861	0.860	0.848	0.6

根据表 10 可以得到，在 Jaccard、余弦、调整余弦、Pearson 这 4 种方法中效果最好的是 Jac 相似度，指标 MAE 为 0.655，指标 RMSE 为 0.800，而本文提出的 PRS 方法 MAE 为 0.639，RMSE 为 0.790，比 Jac 相似度在指标 MAE 上降低了 1.60%，在指标 RMSE 上降低了 1.00%.

图 3、图 4 展现了特征维度 d 为 5 维时各种方法随着实验迭代次数的增加，指标 MAE 和 RMSE 的变化.

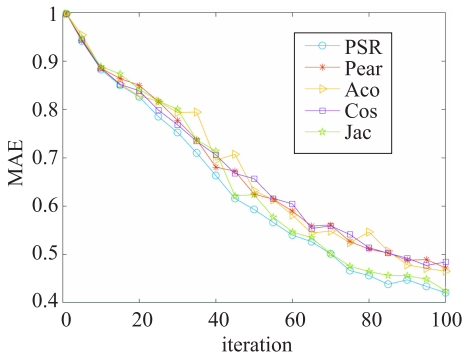

图 3　$d=5$ 时 5 种相似度在指标 MAE 下的比较

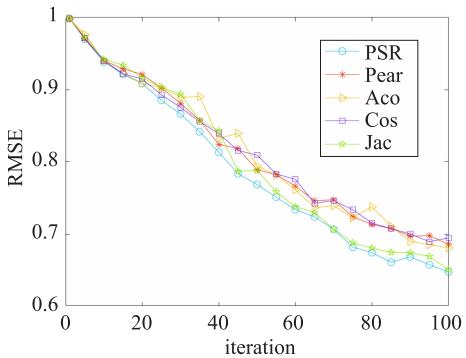

图 4　$d=5$ 时 5 种相似度在指标 RMSE 下的比较

由图 3 和图 4 可以看出,当特征维度为 5 维时,各种相似度方法随着实验迭代次数的不断增加,指标 MAE 和 RMSE 都在减少,但是本文方法 PSR 整体都比其它方法低.

通过表 11 可以得到 PSR 方法比 Jaccard、余弦、调整余弦、Pearson 这 4 种方法效果都好,其中效果最好的是 Jac 相关系数,指标 MAE 为 0.735,RMSE 为 0.854,PRS 的 MAE 为 0.725,RMSE 为 0.848. 同样的,PSR 方法比 Jac 相关系数在指标 MAE 上降低了 1.00%,在指标 RMSE 上降低了 0.6%.

图 5、图 6 展现了特征维度 $d=10$ 时各种方法随着实验迭代次数的不断增加,指标 MAE 和 RMSE 的变化.

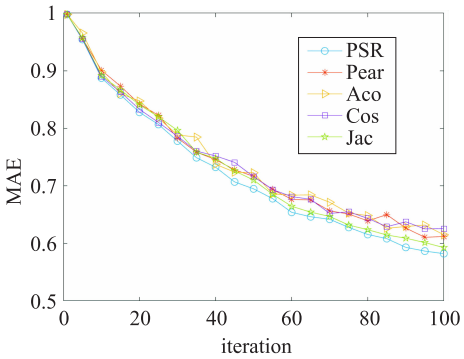

图 5 $d=10$ 时 5 种相似度在指标 MAE 下的比较

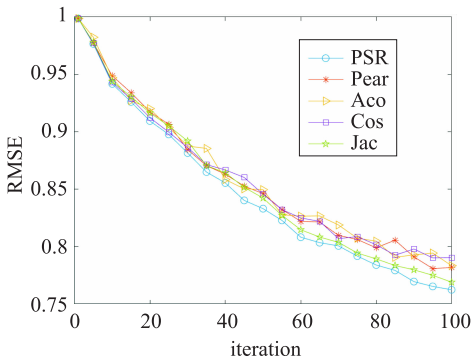

图 6 $d=10$ 时 5 种相似度在指标 RMSE 下的比较

由图 5 和图 6 可以看出,当特征维度为 5 维时,各种相似度方法随着实验迭代次数的不断增加,指标 MAE 和 RMSE 都在减少,但是本文方法 PSR 整体都比其它方法低.

5 结语

本文基于传统相似度度量方法存在的优缺点,在 Pearson 相关系数前增加了一个系数,给出了一种非对称相似度计算方法 PSR,基于该方法计算得到用户相似度矩阵,然后将相似度矩阵与评分矩阵通过 PMF 模型融合在一个统一的框架中,提出了基于 PSR 用户相似度的社会化推荐算法. 在公开数据集 MovieLens 上进行试验,与 Jaccard 相似度、Pearson 相关系数、余弦相似度、调整余弦相似度进行对比,验证本章提出的 PSR 方法提高了预测准确率,同时也缓解了数据稀疏性问题.

参考文献:

[1] ZHENG G P,YU H Y,XU W F. Collaborative filtering recommendation algorithm with item label features[J]. International Core Journal of Engineering,2020,6(1):160 – 170.

[2] 丁浩,艾文华,胡广伟,等. 融合用户兴趣波动时序的个性化推荐模型[J]. 数据分析与知识发现,2021,5(11):45 – 58.

[3] 陈碧毅,黄玲,王昌栋,等. 融合显式反馈与隐式反馈的协同过滤推荐算法[J]. 软件学报,2020,31(3):794 – 805.

[4] XU H L,WU X,LI X D,et al. Comparison study of internet recommendation system[J]. Journal of Software,2009,20(2):350 – 362.

[5] 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.

[6] 于阳,于洪涛,黄瑞阳. 基于熵优化近邻选择的协同过滤推荐算法[J]. 计算机应用研究,2017,34(9):2618 – 2623.

[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] 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.

[9] 武聪,马文明,王冰,朱建豪. 融合用户标签相似度的矩阵分解算法[J]. 南京大学学报(自然科学),2022,58(1):143 – 152.

[10] 李一野,邓浩江. 基于改进余弦相似度的协同过滤推荐算法[J]. 计算机与现代化,2020(1):69 – 74.

[11] 陈功平,王红. 改进 Pearson 相关系数的个性化推荐算法[J]. 山东农业大学学报(自然科学版),2016,47(6):940 – 944.

[12] SALAKHUTDINOV R, MNIH A. Probabilistic matrix factorization[C]//Advances in Neural Information Processing Systems. Massachusetts: MIT Press, 2007: 1257 – 1264.
[13] 于阳. 基于邻域的协同过滤推荐算法研究[D]. 郑州:解放军信息工程大学,2017.
[14] 褚宏林,刘其成,牟春晓. 针对修正余弦相似度改进的协同过滤推荐算法[J]. 烟台大学学报(自然科学与工程版),2021,34(3):330 – 336.
[15] 黄贤英,龙姝言,谢晋. 基于用户非对称相似性的协同过滤推荐算法[J]. 四川大学学报(自然科学版),2018,55(3):489 – 493.
[16] 郑英丽,王新,马倩等. 一种结合用户相似度的社会化推荐算法[J]. 云南民族大学学报(自然科学版),2019,28(1):93 – 99.
[17] 贾俊杰,张玉超. 基于用户模糊聚类的综合信任推荐算法[J]. 计算机工程,2021,47(6):60 – 67.

Asymmetric similarity calculation and application under pearson correlation coefficient

ZHENG Ying-li, PIAO Li-sha, WANG Li-zhen
(Institute of Technology, Dianchi College of Yunnan University, Kunming 650228, China)

Abstract: Sparsity is one of the problems of recommendation algorithms. The existing method to solve the problem of sparsity is the matrix factorization. The matrix factorization combined with user similarity can improve the accuracy of recommendation. However, the traditional similarity calculation method does not consider the difference in the number of items scored by users, so the constructed similarity matrix is symmetric. In response to this problem, a new calculation method called user asymmetric similarity is proposed by combining Pearson correlation coefficient. While considering users'ratings of the same item, calculate the ratio of the number of items with the same rating among users to the number of all scoring items of users, so as to describe the similarity between two users, and get the asymmetric relation between users. Secondly, use the user asymmetric similarity method to calculate the similarity matrix between users, and the similarity matrix and user rating matrix are integrated into the probability matrix factorization framework to achieve the social recommendation of users. Tested on public datasets, the results showed that the improved asymmetric similarity formula, compared to the traditional similarity calculation methods, can achieve more accurate recommendation results through socialized recommendations on sparse datasets.

Key words: social recommendation; asymmetric similarity; probabilistic matrix factorization

(责任编辑　段　鹏)