计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(5):126−132 [doi: 10.15888/j.cnki.csa.006326]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于用户联合相似度的推荐算法①

朱振国, 刘民康, 赵凯旋

(重庆交通大学 信息科学与工程学院, 重庆 400074)
通讯作者: 刘民康, E-mail: cuzbelieve22@gmail.com

摘　要: 基于用户的协同过滤推荐算法在进行近邻用户的筛选时以用户之间相似度的计算结果作为依据, 数据量的增大加剧了数据的稀疏程度, 导致了计算结果的准确性较差, 影响了推荐准确度.针对该问题本文提出了一种基于用户联合相似度的推荐算法. 用户联合相似度的计算分为用户对项目属性偏好的相似度和用户之间人口统计学信息的相似度两个部分. 用户的项目属性偏好引入了 LDA 模型来计算, 计算时评分数据仅作为筛选依据, 因而避免了对数据的直接使用, 减缓了稀疏数据对相似度计算结果的影响; 用户之间人口统计学信息的相似度则在数值化人口统计学信息之后通过海明距离进行度量. 实验结果表明, 本文提出的算法在推荐准确度上优于传统协同过滤推荐算法.

关键词: 协同过滤; 稀疏数据; LDA; 联合相似度; 海明距离

引用格式: 朱振国,刘民康,赵凯旋.基于用户联合相似度的推荐算法.计算机系统应用,2018,27(5):126–132. http://www.c-s-a.org.cn/1003-3254/6326.html

Recommendation Algorithm Based on Combined Similarity of Users

ZHU Zhen-Guo, LIU Min-Kang, ZHAO Kai-Xuan

(School of Information Science and Engineering, Chongqing Jiaotong University, Chongqing 400074, China)

Abstract: The user-based collaborative filtering recommendation algorithm is based on the calculation of the similarity between users when the neighbor user is screened, and the increase in the amount of data exacerbates the sparseness of the data, which leads to the poor accuracy of the results and affects the recommendation accuracy. Aiming at this problem, this study proposes a recommendation algorithm based on the combined similarity of users. The calculation of combined similarity of users is divided into two parts: the similarity of the user's preference for item attributes and the similarity of the demographic information between the users. The algorithm introduces the LDA model to calculate the preference for the user's item attribute, and the scoring data is only used as the screening basis when calculating so as to avoid using it directly as well as slow down the influence of sparse data on similarity calculation results. While the similarity between demographic information is measured by Hamming distance after the numerlization of demographic information. Experimental results show that the proposed algorithm is superior to the traditional collaborative filtering recommendation algorithm in recommendation accuracy.

Key words: collaborative filtering; sparse data; LDA; combine similarity; Hamming distance

1　概述

信息技术的迅猛发展使我们进入了大数据时代, 用户发生的各种行为都伴随着对应数据的产生, 如用户在购物网站中的购买记录和评论、电影评分网站的

① 收稿时间: 2017-08-21; 修改时间: 2017-09-06; 采用时间: 2017-09-18; csa 在线出版时间: 2018-04-23

评分信息等. 数据井喷式的增长和累积造成严重的信息过载, 个性化推荐[1]作为处理应对这些问题的工具应运而生.

协同过滤推荐算法是在个性化推荐领域获得最为广泛使用的算法之一, 其主要功能是预测和推荐[2]. 算法通过对用户历史行为数据的挖掘发现用户的偏好, 基于不同的偏好对用户进行群组划分并推荐品味相似的项目. 协同过滤推荐算法主要分为两类, 分别是基于用户的协同过滤推荐算法 (user-based collaborative filtering) 和基于项目的协同过滤推荐算法 (item-based collaborative filtering)[3,4]. 其核心是依据用户历史评同用户对项目属性的偏好值通过 LDA (Latent Dirichlet 分数据计算用户之间或项目之间的相似度, 进而锁定 Allocation) 模型求得, 所得结果使用余弦定理进行相近邻范围, 对目标用户未评分项目进行预测, 将预测值最高的前 N 个项目推荐给目标用户. 但是随着用户数量和项目数量的增加, 加剧了用户－项目评分数据的稀疏性. 传统推荐算法面对这一问题时, 相似度计算准确性下降, 难以保证良好的推荐质量.

近年来为解决传统协同过滤推荐算法面临的困境, 学者们提出了不同的新方法尝试在推荐算法中融合. 如陈伶红等[5]提出使用在信息检索和数据挖掘中常用的加权技术 TF-IDF(Term Frequency-Inverse Document Frequency) 和信息熵得到用户对项目属性的偏好模型, 并以此为基础进行用户聚类、相似度计算和最近邻查询, 进而对用户未评价的项目预测评分, 给出推荐; Cheng-kang Hsieh 等[6]提出使用度量学习结合协同过滤提升推荐结果, 通过度量学习得到候选集项目与目标用户的距离, 令用户偏好度低的项目远离用户, 反之则靠近用户, 将稀疏数据的影响降到了较小的程度; 于波等[7]提出了一种结合项目属性的混合推荐算法, 通过将项目之间相似度的计算与传统协同过滤推荐算法通过动态加权的方式相结合, 用来解决数据的稀疏性问题. 上述提出的算法虽然缓解了稀疏数据对相似度计算结果准确性的影响, 但在计算过程中却并未提出对稀疏数据的有效处理方法, 其仍是相似度计算的直接数据来源.

用户联合相似度的计算是对用户之间相似度计算方法的一种提升. 它在用户个人历史行为相似性计算分析的基础上增加了对用户个人信息相似性的计算, 并将两部分的计算值进行线性组合作为最终的相似度计算结果. 用户联合相似度对原有用户相似度的计算范围进行了扩充, 用户的人口统计学数据不具有稀疏性, 因此相似度计算结果的准确性得到了提升, 而在用户行为不足的使用情境中, 也能对冷启动问题起到一定的缓解作用. 用户联合相似度提供了用户之间相似度更多维度和更全面、准确的计算方式, 因此本文在用户之间相似度的计算方式上使用了用户联合相似度.

在使用用户联合相似度计算用户相似度的基础上, 本文提出了一种基于用户联合相似度的推荐算法. 用户联合相似度将用户之间的相似度分成两个部分计算. 一是不同用户对项目属性偏好分布的相似程度; 二是用户之间人口统计学信息的相似程度, 最终线性组合两部分的计算结果作为用户之间的联合相似度. 不似度的计算; 不同用户的人口统计学信息的相似度使用海明距离度量信息的差异值, 所得结果使用反比例函数进行相似度的计算. 本文提出的联合相似度避免了对用户-项目评分数据的直接使用, 降低了稀疏数据对相似度计算结果准确性的影响.

2 理论基础

2.1 传统协同过滤推荐相关理论

协同过滤推荐方法的主要思想是利用已有用户群过去的行为或意见预测当前用户最可能喜欢或感兴趣的项目. 通过对用户-项目评分矩阵的处理来预测用户的喜好.

在基于用户的协同过滤推荐中, 相似度计算通常使用皮尔逊系数计算. $r_{i,j}$ 表示用户的评分项, $i\in 1,\cdots,n$, $j\in 1,\cdots,m$. $p=\{p_1,p_2,\cdots,p_m\}$代表项目集, $\overline{r_a}$、$\overline{r_b}$表示用户 a、b 平均评分, a、b 间相似度可通过式 (1) 计算:

$$sim(a,b)=\frac{\sum_{p\in P}\left(r_{a,p}-\overline{r_a}\right)\left(r_{b,p}-\overline{r_b}\right)}{\sqrt{\sum_{p\in P}\left(r_{a,p}-\overline{r_a}\right)^2}\sqrt{\sum_{p\in P}\left(r_{b,p}-\overline{r_b}\right)^2}} \quad (1)$$

用户 a 对项目 p 的预测评分则可以通过式 (2) 求得 (N 代表 a 与相似程度高的近邻集合):

$$pred(a,p)=\overline{r_a}+\frac{\sum_{b\in N}sim(a,b)*\left(r_{b,p}-\overline{r_b}\right)}{\sum_{b\in N}sim(a,b)} \quad (2)$$

在基于项目的协同过滤推荐中, 算法的主要思想是利用项目之间的相似度来计算预测值. $U=\{u_1,u_2,\cdots,$

u_m}代表对项目作出评价的用户集, $\overline{r_u}$表示每个用户的平均打分, 项目 a、b 的相似度则可以使用式 (3) 改进的余弦相似度计算如下:

$$sim(a,b)=\frac{\sum_{u\in U}(r_{u,a}-\overline{r_u})(r_{u,b}-\overline{r_u})}{\sqrt{\sum_{u\in U}(r_{u,a}-\overline{r_u})^2}\sqrt{\sum_{u\in U}(r_{u,b}-\overline{r_u})^2}} \quad (3)$$

确定项目间的相似度之后, 用户 u 对项目 p 的评分预测则按照式 (4) 计算 ($rateditem(u)$ 表示用户 u 评价过的项目集合):

$$pred(u,p)=\frac{\sum_{i\in rateditem(u)} sim(i,p)*r_{u,j}}{\sum_{i\in rateditem(u)} sim(i,p)} \quad (4)$$

两种传统协同过滤推荐算法虽然计算复杂性较低, 当用户或项目数量较多时, 其评分矩阵十分稀疏, 此时, 传统方法推荐效果不佳.

2.2 LDA 模型相关理论

LDA 是 David Blei 等人[8]于 2003 年提出的基于概率模型的主题模型算法, 它是一种非监督机器学习技术, 可用来识别大规模文档集或语料库中的潜在隐藏的主题信息[9]. LDA 的图模型如图 1 所示.

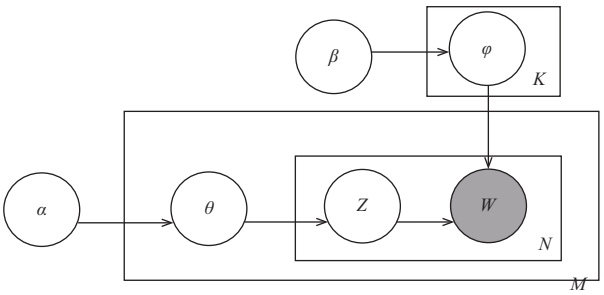

图 1 LDA 概率图模型表示

这是一个三层的贝叶斯概率模型. 图中的阴影圆和非阴影圆分别表示可观测变量和潜在变量, 箭头表示两变量间的条件依赖, 方框表示重复抽样, 重复次数在方框的右下角. M 代表语料中文档的数量, K 代表设置的主题数, N 代表训练语料库中出现的所有词, Z 代表隐藏的主题. θ 是语料库中所有文档在各个主题上的概率分布矩阵, $\vec{\theta}_m$代表第 m 篇文档的主题分布; φ 是所有主题在其对应词上的概率分布矩阵, $\vec{\varphi}_k$代表编号为 k 的主题之上的词分布. α 代表每篇文档主题分布的先验分布 Dirichlet 分布的参数, β 代表每个主题对应词分布的先验分布 Dirichlet 分布的参数 (α、β 也称为超参数), w 是可观测词[8].

LDA 作为一种生成模型, 以分词后的文档集 (通常为一篇文档一行) 和主题数 K 及超参数 α、β 作为输入, 其生成过程的核心可通过式 (5) 表示:

$$p(word|document)=\sum_{topic} p(word|topic)p(topic|document) \quad (5)$$

矩阵表示形式如图 2.

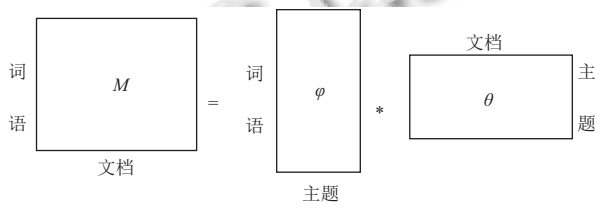

图 2 LDA 生成过程的矩阵表示

LDA 模型的标准生成过程可描述如下:

1) 从狄利克雷分布 Dir(α) 中抽样选择$\vec{\theta}_m$, m 为文档编号 (文档总数为 M), $\vec{\theta}_m$代表这篇文档下主题的分布;

2) 从狄利克雷分布 Dir(β) 中抽样选择$\vec{\varphi}_k$, k 为主题编号 (主题总数为 K), $\vec{\varphi}_k$代表每个主题下词的分布;

3) 对于每个单词的位置 $w_{i,j}$, $j\in(1, N_i)$, $i\in(1, M)$;

4) 从多项式分布 Multinomial(θ_i) 中抽样选择一个主题 $z_{i,j}$;

5) 从多项式分布 Multinomial($\varphi_{zi,j}$) 中抽样选择一个词 $w_{i,j}$.

3 用户的联合相似度及推荐过程

传统的协同过滤算法虽已获得广泛应用, 但其推荐的准确性受限于相对稠密的数据[10]. 为减缓稀疏数据对相似度计算产生影响, 本文提出了基于用户联合相似度的方法. 用户的相似度从用户对影片属性偏好和用户人口统计学信息两个方面计算.

3.1 用户对影片属性偏好的相似度

在用户对影片属性偏好的计算上, 前提是数据的筛选和整合. 数据集中包括用户对电影的评分, 评分区间为 1–5, 分值的大小与用户对电影的喜好程度成正比. 同时电影是一种多属性项目, 因而被评分过的电影均有其属性描述数据. 如表 1 所示.

表 1 项目属性描述

项目	项目属性				
	属性1	属性2	属性3	…	属性n
I_1	f_{11}	f_{12}	f_{13}	…	f_{1n}
I_2	f_{21}	f_{22}	f_{23}	…	f_{2n}
I_3	f_{31}	f_{32}	f_{33}	…	f_{3n}
…	…	…	…	…	…
I_n	f_{n1}	f_{n2}	f_{n3}	…	f_{nn}

每个项目对应的属性标注 f_{nn} 均有其对应值, 若对应值为 1, 则表明项目具有该属性, 否则没有.

用户的评分数据使用矩阵进行存储, 得到用户-项目评分矩阵, 如表 2 所示. 依照评分矩阵计算每个用户的打分平均值. 为得到用户对于项目各个属性的偏好程度, 使用用户的打分平均值对被评价项目筛选. 由于评分大小与用户对于影片的喜好程度成正比, 所以使用每个用户的打分平均值作为依据, 将评分项目中高于均值的保留, 并结合影片属性描述数据, 形成每个用户对应的高分评价电影列表. 如 U_1 形成的高分评价电影列表如表 3 所示.

表 2 用户-项目评分矩阵

	项目1	项目2	项目3	项目4	项目5
用户1	5	3	4	3	3
用户2	4	-	-	-	2
用户3	-	-	2	-	2
用户4	-	-	1	-	-
用户5	4	3	-	-	-

表 3 用户 U_1 高分评价电影列表

用户	项目	项目属性				
		属性1	属性2	属性3	…	属性n
U_1	I_1	f_{11}	f_{12}	f_{13}	…	f_{1n}
	I_3	f_{31}	f_{32}	f_{33}	…	f_{3n}
	I_4	f_{41}	f_{42}	f_{43}	…	f_{4n}
	…	…	…	…	…	…
	I_m	f_{m1}	f_{m2}	f_{m3}	…	f_{mn}

按照用户 ID, 结合每个 ID 下得到的高分电影评价列表, 对每个用户对应的所有高分评价项目中各个属性的出现次数进行统计. 如对于 U_1, 其所对应的高分评价项目包括 I_1、I_3、I_4 等, 每部电影有其不同的属性描述, 对于每个属性我们可以统计其在这些高分影片中出现的总次数. 如对于不同用户来说, 根据其高分评价项目可以得到如表 4 统计信息.

表 4 用户偏好项目属性出现次数统计

用户	偏好项目各个属性出现次数				
	属性1	属性2	属性3	…	属性n
U_1	n_{11}	n_{12}	n_{13}	…	n_{1n}
U_2	n_{21}	n_{22}	n_{23}	…	n_{2n}
…	…	…	…	…	…
Un	n_{n1}	n_{n2}	n_{n3}	…	n_{nn}

在数据集中, 电影对应属性共有 18 种, 如表 5 所示.

表 5 电影属性描述词

编号	属性	编号	属性	编号	属性
1	Action	7	Documentary	13	Mystery
2	Adventure	8	Drama	14	Romance
3	Animation	9	Fantasy	15	Sci-Fi
4	Children's	10	Film-Noir	16	Thriller
5	Comedy	11	Horror	17	War
6	Crime	12	Musical	18	Western

每个属性词在以用户 ID 为单位的偏好属性统计中对应有其出现频数, 如表 3 所示, 将属性编号使用对应的属性描述词替换, 结合其出现频数生成每个属性词对应的长词语串, 组合每个属性词的词语串得到一个用户的偏好文档, 如图 3(a) 所示, 图 3(b) 为全部用户的属性偏好文档集合.

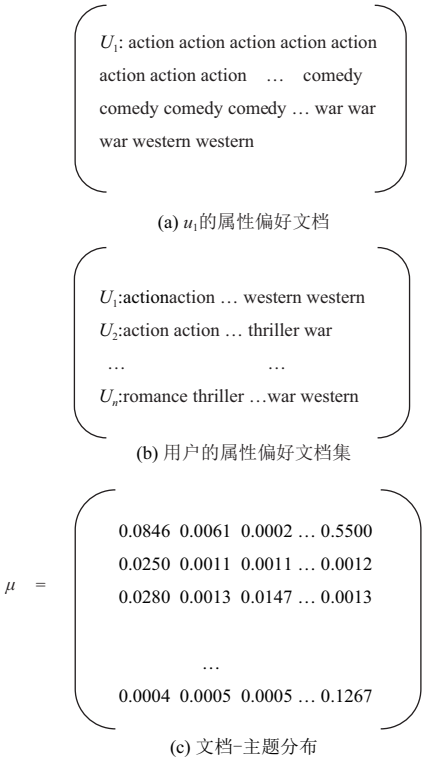

图 3 属性偏好文档及文档主题分布图示

所有用户属性的偏好文档看做待处理文档集合, 将该集合同给定的先验超参数 α、β 和主题数作为 LDA 模型的输入 (影片属性分 18 个类, 主题数设为 18), 未知参数 θ、φ 的估计使用收缩吉布斯采样求得. 得到未知参数的估计后, 进而得到文档-主题概率分布矩阵 (用户在 18 个属性上的偏好值) 和主题-词语概率分布矩阵, 我们对文档-主题概率分布矩阵做进一步的用户属性偏好相似度计算. 图 3(c) 代表每个文档在各个主题上的偏好值.

将每一位用户得到的主题偏好结果作为一个向量, 即矩阵 μ 中的一行. 使用余弦定理对不同用户的属性偏好情况进行衡量, 用户 a、b 间属性偏好的相似度 $sim_{pre(a,b)}$ 通过式 (6) 计算:

$$sim_{pre(a,b)} = \cos(v_a, v_b) = \frac{v_a * v_b}{|v_a| * |v_b|} \tag{6}$$

3.2 用户间人口统计学信息的相似度

数据中对于每个用户的人口统计学信息从年龄、性别、职业三个维度来描述. 年龄范围: 7–73, 职业共 21 种, 性别分为男女两类. 用户个人信息和职业分类情况如表 6、表 7 所示.

表 6 用户人口统计学信息

用户ID	年龄	性别	职业
1	24	M	Technician
2	53	F	Other
3	23	M	Writer
…	…	…	…
943	22	M	Student

表 7 用户职业信息列表

编号	职业	编号	职业	编号	职业
1	Administrator	8	Healthcare	15	programmer
2	Artist	9	Homemaker	16	Retired
3	Doctor	10	Lawyer	17	Salesman
4	Educator	11	Librarian	18	Scientist
5	Engineer	12	Marketing	19	Student
6	Entertainment	13	None	20	Technician
7	Executive	14	Other	21	Writer

为了计算用户之间个人信息的相似程度, 需要对这些描述数据进行数值化处理. 对于职业和年龄的处理分别按照年龄分段标准和国家职业大类划分标准进行. 用 a 代表年龄, 7–73 的年龄可划分为: a<18、18≤a≤24、25≤a≤34、35≤a≤44、45≤a≤49、50≤a≤55 和 a≥56 七类, 以数字 1–7 代替; 职业大类的划分以国家职业分类标准作为依据, 按照企事业负责人、专业技术人员、服务业商业、文娱从事着、教育行业、家政以及其他分为七类, 使用数字 1–7 代替; 性别数据按照男女作为划分, 使用数字 1、0 代替. 按照上述方法, 每位用户的人口统计学信息可以用一个三位数字的字符串来表示. 如表中 1–3 的用户的信息可以分别表示为: “212”、“607”、“604”. 他们之间的相似程度比较则可以通过等长字符串的差异程度来比较. 我们使用距离度量方法中的海明距离作为工具来对字符串的差异来进行计算.

海明距离定义为两个等长字符串之间对应位置的不同字符的个数[11]. 即一个字符串变换成另外一个字符串所需替换的字符个数, 如“10111”和“10010”的海明距离是 2. 用户的人口统计学信息经过数值化处理之后均被表示为三位数字组成的等长字符串, 可以使用海明距离对他们之间的差异进行衡量, 距离的值越大则相似性越小, 否则相似性越大. 针对数据, 用户之间的海明距离的值为: 0–3. 距离为 0 的时候, 用户之间的统计学信息相似度最高, 距离为 3 的时候, 相似性最低. 我们通过式 (7) 对得到的海明距离进行处理, 得到个人信息相似度 $sim_{fea(a,b)}$:

$$sim_{fea(a,b)} = \frac{1}{dis_{hamming(a,b)}{}^{n} + 1} \tag{7}$$

$dis_{hamming(a,b)}$ 表示用户 a, b 之间的海明距离, 使用幂指数 n 对海明距离进行放大, 幂指数取距离的最大值 3. 分母上的 1 起到如下两个作用: 1) 为了保证海明距离为零时可求得计算结果; 2) 保证 $sim_{fea(a,b)} \in (0, 1]$. 海明距离为 0 时, 则表明两个用户的人口统计学信息相似度最高, 分母上的 1 可以使个人信息相似度取到 1; 海明距离为 1 时, 分母上的 1 可以作为对距离值的放大, 避免了海明距离在非 0 时取到最大的相似度 1. 又由于 $dis_{hamming(a,b)} \in [1, 3]$, 且 2^n 和 3^n 都是数值上远大于 1 的值, 因此分母上的 1 海明距离取值非 0、1 的情况下, 不会对海明距离计算值过度放大, 从而影响个人信息相似度计算值的准确度.

3.3 用户联合相似度的形成

联合相似度的构成是用户影片属性偏好相似度和用户人口统计学信息相似度的线性组合. a, b 间用户联合相似度 $sim_{combine(a,b)}$ 可表示为:

$$sim_{combine(a,b)} = \lambda sim_{pre(a,b)} + (1-\lambda)\, sim_{fea(a,b)} \tag{8}$$

联合相似度的计算使用评分数据作为筛选依据, 而没有直接参与计算, 避免了其高稀疏程度对计算结

果的影响, 对于用户之间的相似度计算针对性更强、准确度更高.

3.4 推荐过程

计算出不同用户对影片属性偏好的相似程度和用户人口统计学信息相似度后, 结合线性组合系数 λ 对两者加权, 所得结果即为用户之间的联合相似度. 将相似度计算应用到推荐算法中, 形成基于用户联合相似度的协同过滤推荐算法.

在求得近邻集合和用户间的联合相似度之后, 根据如下公式计算目标用户 a 对目标项目 p 的预测评分:

$$pred(a,p)=\overline{r_a}+\frac{\sum\limits_{b\in N} sim_{combine}(a,b)*\left(r_{b,p}-\overline{r_b}\right)}{\sum\limits_{b\in N} sim_{combine}(a,b)} \quad (9)$$

其中, $\overline{r_a}$ 表示用户 a 对已评论项目的平均评分, $\overline{r_b}$ 为用户 b 对已评论项目的平均评分, N 表示近邻集合, $r_{b,p}$ 表示 b 对 p 的评分值, $pred(a, p)$ 表示 a 对 p 的预测评分值, $sim_{combine(a,b)}$ 表示 a 和其近邻 b 的用户联合相似度.

得到目标用户对未选择项目的评分后, 根据评分降序排列, 将评分最高的前 n 个项目推荐给该目标用户.

4 实验与结果分析

4.1 实验使用数据集

本文实验使用的数据集来自美国明尼苏达州立大学 GroupLens 研究小组提供的 MovieLens (ml-100k). 该数据集中包含了 943 位用户对 1682 部电影的 10 万条评分, 每位用户评分数不少于 20 条, 评分范围: 1–5. 数据集的原始用户-项目评分矩阵的稀疏度为 93.7%[12]. 实验过程中将数据集按照 4:1 的比例划分训练集和测试集.

4.2 评价指标

评估指标是算法的性能优劣的体现, 为对基于用户联合相似度推荐算法的准确度进行评估, 采用广泛使用的平均绝对偏差 (Mean Absolute Error, MAE)[13]和均方根误差 (Root Mean Square Error, RMSE)[14]作为实验结果的评估标准.

$$MAE=\frac{\sum\limits_{i=1}^{N}|p_i-q_i|}{N} \quad (10)$$

$$RMSE=\sqrt{\frac{1}{n}\sum_{i=1}^{N}(p_i-q_i)^2} \quad (11)$$

其中 p_i 表示预测评分, q_i 表示实际评分.

4.3 实验结果

4.3.1 联合相似度的线性组合系数

对基于用户联合相似度的推荐算法, 通过固定不同规模的邻域大小确定使 *MAE* 最小的参数 λ 的值, 不同规模的近邻在 *MAE* 最小时对应不同的 λ. 我们在测试集上选择 30–150 的近邻规模进行实验, 以 30 作为区间间隔确定参数 λ, 实验结果如图 4 所示. 它描述了算法在不同近邻值下, *MAE* 最小时 λ 的取值. 根据图示可以看出五种不同的近邻规模分别在 0.8、0.9、0.9、0.7、0.7 上取得 *MAE* 的最小值, 我们计算这五个值的平均数作为联合相似度的组合系数, 求得该值为 0.8.

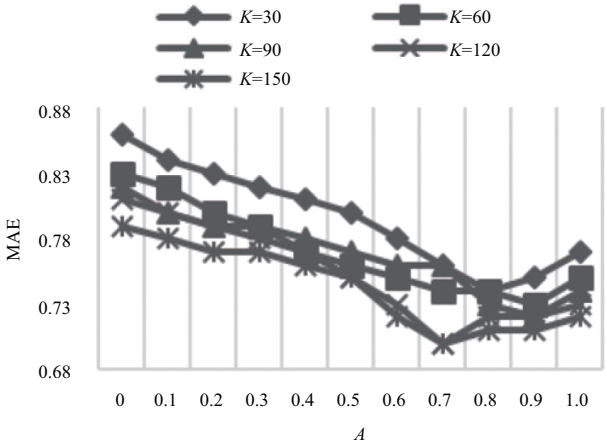

图 4 用户联合相似度中 λ 的取值确定

4.3.2 基于用户联合相似度的推荐

确定了 λ 的值之后, 引入传统的基于用户的协同过滤算法与本文基于用户联合相似度的算法在测试集上的实验结果进行对比. 通过 3 种相似度度量方法作为比较, 3 种相似度度量方法包括余弦夹角相似度、皮尔逊相关系数和杰卡德系数. 目标用户最近邻个数分别为 (10, 20, 30, 40, 50, 60, 70), 对应 *MAE* 值和 *RMSE* 值如图 5 和图 6 所示.

4.4 实验分析

从图示的实验结果可以看出本文提出的基于用户联合相似度的推荐算法在各个近邻规模上的 MAE 值和 RMSE 值相比使用其他方法计算相似度的传统推荐算法都有了不同程度的下降, 验证了提出算法的有效性和推荐准确度的提升.

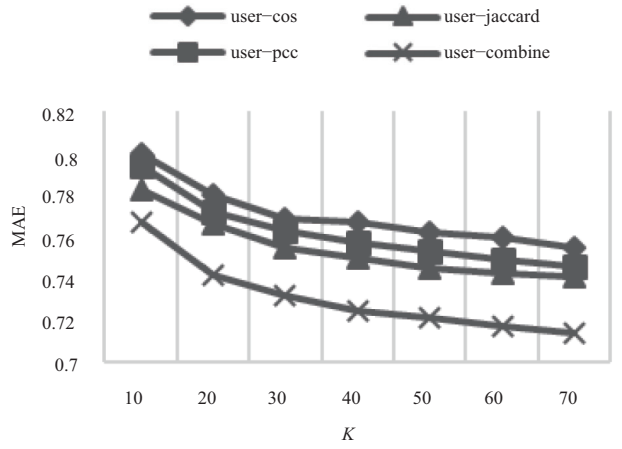

图 5 MAE 与传统协同过滤算法比较

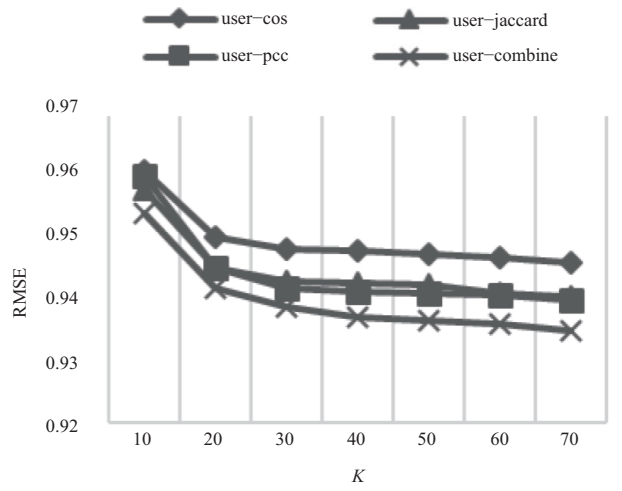

图 6 RMSE 与传统协同过滤算法比较

5 结束语

为了在高度稀疏数据的前提下提升推荐算法的推荐质量, 本文提出了基于用户联合相似度的推荐算法. 联合相似度将用户相似度的计算分为用户对影片属性偏好分布的相似度和用户人口统计学信息相似度两部分, 使用 λ 作为两部分相似度的线性组合系数, 得到最终相似度计算值. 用户对影片属性的偏好分布使用 LDA 模型对用户属性偏好文档集处理得到, 接着使用余弦定理对不同用户分布的相似程度进行评估; 用户人口统计学信息相似度通过数值化用户的个人信息, 求得不同用户间信息的海明距离计算相似度. 用户联合相似度相比传统基于用户的推荐算法得到了更加准确用户近邻范围. 最后, 将算法在 MovieLens (ml-100k) 数据集中进行实验. 结果表明, 本文提出的基于用户联合相似度的推荐算法比传统基于用户的协同过滤算法推荐准确率高, 在推荐效果上有所提升.

参考文献

1 Ricci F, Rokach L, Shapira B, 等. 推荐系统: 技术、评估及高效算法. 李艳民, 胡聪, 吴宾, 等译. 北京: 机械工业出版社, 2015.
2 Jannach D, Zanker M, Felfernig A, 等. 推荐系统. 蒋凡, 译. 北京: 人民邮电出版社, 2013.
3 叶柏龙, 徐静静, 严算. 基于评分和项目特征的群组推荐方法. 计算机应用研究, 2017, 34(4): 1032–1035, 1046.
4 荣辉桂, 火生旭, 胡春华, 等. 基于用户相似度的协同过滤推荐算法. 通信学报, 2014, 35(2): 16–24.
5 陈伶红, 徐华中, 李鲍, 等. 一种基于用户对项目属性偏好的推荐算法. 武汉理工大学学报 (信息与管理工程版), 2016, 38(5): 616–620.
6 Hsieh CK, Yang LQ, Cui Y, *et al*. Collaborative metric learning. Proceedings of the 26th International Conference on World Wide Web. Perth, Australia. 2017. 193–201.
7 于波, 陈庚午, 王爱玲, 等. 一种结合项目属性的混合推荐算法. 计算机系统应用, 2017, 26(1): 147–151. [doi: 10.15888/j.cnki.csa.005490]
8 Blei DM, Ng AY, Jordan MI. Latent dirichlet allocation. Journal of Machine Learning Research, 2003, (3): 993–1022.
9 马晨. LDA 漫游指南. 北京: 人民邮电出版社, 2015.
10 代金龙. 协同过滤算法中数据稀疏性问题研究[硕士学位论文]. 重庆: 重庆大学, 2013.
11 李青, 尹四清. 结合用户偏好和相似性的网络结构推荐算法. 计算机工程与设计, 2016, 37(3): 814–818.
12 Maxwell Harper F, Konstan JA. The movielens datasets: History and context. ACM Transactions on Interactive Intelligent Systems (TiiS)-Regular Articles and Special Issue on New Directions in Eye Gaze for Interactive Intelligent Systems (Part 1 of 2), 2016, 5(4): 19.
13 李伟霖, 王成良, 文俊浩. 基于评论与评分的协同过滤算法. 计算机应用研究, 2017, 34(2): 361–364, 412.
14 呼亚杰. 一种基于类别偏好协同过滤推荐算法的实现与优化[硕士学位论文]. 兰州: 兰州大学, 2016.