

概率粗糙集模型在推荐算法中的应用

陈功平,王 红

六安职业技术学院 信息与工程学院, 安徽 六安 237158

摘要: 推荐算法能够挖掘用户的潜在兴趣将项目自动地推荐给客户, 是解决信息过载的智能手段之一。由于网络中的用户数和项目数较多, 评分矩阵的稀疏性严重影响了推荐效果, 推荐的先验知识缺失严重。粗糙集是一种可以使用不完备的知识实施推理的有效方法, 使用概率粗糙集的 α 、 β 阈值合理划分边界域, 生成推荐策略, 降低评分矩阵稀疏性对推荐结果的影响。实验结果表明: 概率粗糙集模型能够有效提高在评分矩阵非常稀疏的情况下的推荐准确率, 其在 Movie Lens 数据集上的推荐准确率最高达到 92.80%, 覆盖率最高达 100%。

关键词: 概率粗糙集; 推荐算法; 参数学习

中图分类号: TP301

文献标识码: A

文章编号: 1000-2324(2017)02-0192-07

Application of Probabilistic Rough Set Model in Recommendation Algorithm

CHEN Gong-ping, WANG Hong

College of information and Electronic Engineering/Lu'an Vocation Technology College, Lu'an 237158, China

Abstract: Recommendation algorithm can dig into the potential interest of users and then automatically recommends the project to the users. It is one of the intellectual approaches solving the information overload. As the number of users and number of items are more, the sparseness of rating matrix has a strong impact on the effect of recommendation. The priori knowledge of recommendation is missing seriously. Rough set is an effective method which can adopt an incomplete knowledge to carry out ratiocination. It proposes dividing the boundary region by using probabilistic rough set threshold α and β , creating recommendation strategy and reducing the effect of the sparseness of score matrix to the recommendation result. The experimental results indicated that the model of probabilistic rough set could improve the recommendation accuracy rate under the circumstance of the high sparseness of score matrix. The recommendation accuracy rate could be up to 92.80% in the Movie Lens data sets and the highest fraction of coverage could be up to 100%.

Keywords: Probabilistic Rough Set; recommendation algorithm; parameter learning

网络的普及应用让人们习惯于从网络上获取信息, 而随着网络资源的爆炸式增长, 使得用户耗时间在无用信息上的时间和精力越来越多。推荐系统^[1]能够从用户的历史行为中挖掘用户兴趣, 将和用户兴趣相关的项目主动地推荐给用户, 减少检索时间。近年推荐算法得到了学者的广泛关注和研究, 其应用也得到各大网站的支持。

基于内容的推荐^[2] (Content-Based Recommendation)、基于知识的推荐^[3] (Knowledge-Based recommendation)、基于协同过滤的推荐^[4] (Collaborative Filtering-Based Recommendation) 等是常用的推荐算法, 协同过滤推荐算法是一种简单且常用的算法, 能够很好的与其他算法融合, 是亚马逊等商业网站推荐系统的核心算法。协同过滤推荐算法从用户的历史行为中学习规律, 挖掘用户喜好, 然后预测用户对未关联的项目的喜好程度, 将用户最喜爱的 Top-n 项目生成推荐列表提供给用户。

用户对项目的评分能很好的反映用户对项目的感兴趣程度, 由于网络中的用户和项目数量非常多, 导致评分矩阵十分稀疏, 降低推荐效果。粗糙集^[5]是一种可以利用不完备知识实施推理的有效方法, 将稀疏的评分矩阵作为粗糙集的属性库, 使用粗糙集思想合理划分等价类, 从评分矩阵中挖掘规则实施推荐, 提高推荐的效果。

实验表明, 推荐模型在 100 k Movie Lens 数据集上的准确率和覆盖率最高达到 92.80%和 100%, 比 Pawlak 粗糙集模型的性能更优。

收稿日期: 2015-05-22

修回日期: 2015-11-20

基金项目: 2015 年度安徽高校自然科学研究重点项目(KJ2015A435);安徽省 2016 年高校优秀青年人才支持计划重点项目(gxyqZD2016570);安徽省 2014 年高校优秀青年人才支持计划

作者简介: 陈功平(1980-),男,硕士研究生,讲师.研究方向:个性化推荐,网络技术. E-mail:chgp06@126.com

***通讯作者:** Author for correspondence. E-mail:wh0115140@126.com

1 概率粗糙集模型

概率粗糙集模型是 Pawlak 粗糙集模型的扩展, 使用 α 、 β 参数细化 Pawlak 粗糙集模型的正域、负域和边界区域而得到的一种粗糙集模型^[6]。

1.1 Pawlak 粗糙集模型

Pawlak 模型定义了上、下近似集合, 设 U 表示论域, R 是 U 的一个等价关系, 即将 U 划分为多个不相交的 R 子集, 可表示为 $U/R = \{R_1, R_2, \dots, R_n\}$, 现有 X 集合为 U 的子集, 即, $X \subseteq U$, 使用等价关系 R 来近似 X , X 的上近似和下近似^[7]被定义如公式 (1) ~ (2)。

$$R^*(X) = \{x | x \in U, [x] \subseteq X, [x] \cap R_i \neq \emptyset\} \tag{1}$$

$$R_*(X) = \{x | x \in U, [x] \subseteq X, [x] \subseteq R_i\} \tag{2}$$

其中, $[x]$ 是 X 的一个划分, R_i 是 R 集合中的一个子集, 即, $R_i \subseteq R$ 。 X 集合下近似与上近似的比值称为集合 X 的粗糙程度。

$$P(X|R) = \frac{|R_*(X)|}{|R^*(X)|} \tag{3}$$

显然 $0 \leq P(X|R) \leq 1$, 当 $P(X|R)=1$ 时, 集合 X 相对于等价关系 R 是精确的, 当 $P(X|R) < 1$ 时, 集合 X 相对于等价关系 R 是粗糙的。

根据 X 的上下近似, 可以将 U 划分为不相交的正域、负域和边界域, 定义公式^[8]见 (4) ~ (6)。

$$POS(X) = R_*(X) \tag{4}$$

$$NEG(X) = U - R^*(X) \tag{5}$$

$$BND(X) = R^*(X) - R_*(X) \tag{6}$$

正域、负域和边界域的关系如图 1 所示。整个区域表示论域 U , 圆表示要近似的集合 X , 矩形表示等价关系 R , 粗线实心矩形区域是 X 的下近似, 即正域, 虚线区域 (去掉正域) 是 X 的边界域, 虚线外的区域是 X 的负域。

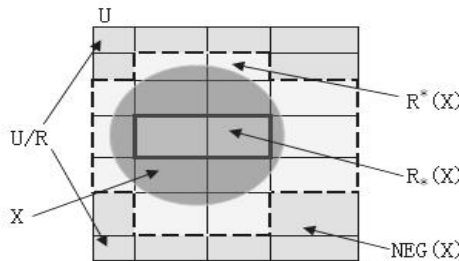


图 1 正、负和边界区域的关系

Fig.1 The relation of positive、negative and boundary regions

1.2 概率粗糙集模型

从图 1 可以看出, 正域、负域是相互独立的, 而边界域中, 有的相交部分多, 有的相交部分少, 定义 $P(X/R_i)$ 表示集合 X 与 R_i 等价关系的重叠程度^[9]。

$$P(X/R_i) = \frac{|X \cap R_i|}{|R_i|} \tag{7}$$

因此, 可以使用 $P(X/R_i)$ 来进一步细化 X 的上下近似以及正域、负域和边界域, 设置两个参数 α 、 β 作为细化指标, 定义^[10]如公式 (8) ~ (12) 所示。

$$R_{*\alpha\beta}(X) = \{X | X \in U, P(X)[X] \geq \alpha\} \tag{8}$$

$$R^*_{\alpha,\beta}(X) = \{x | x \in U, P(X|[x]) > \beta\} \tag{9}$$

$$POS_{\alpha,\beta}(X) = R_{*\alpha\beta}(X) \tag{10}$$

$$NEG_{\alpha,\beta}(X) = U - R^*_{\alpha,\beta}(X) = \{x | x \in U, P(X|[x]) \leq \beta\} \tag{11}$$

$$BND_{\alpha,\beta}(X) = R^*_{\alpha,\beta}(X) - R_{*\alpha\beta}(X) = \{x | x \in U, \alpha > P(X|[x]) \geq \beta\} \tag{12}$$

当参数 $\alpha=1, \beta=0$ 时, 概率粗糙集等价于 Pawlak 粗糙集, 当 $\alpha<1, \beta>0$ 时, Pawlak 粗糙集的部分边界域将划分到正域和负域中, 当参数 $\alpha=\beta$ 时, 边界域将全部划分到正域或负域中。

2 基于粗糙集的推荐模型

基于粗糙集的推理需要从信息表中学习推理策略, 表 1 可以看成是一个基于评价矩阵的信息表, 这里将评价 1~3 作为条件属性, 总体评价作为决策属性, 即, 通过分析评价 1、评价 2、评价 3 和总体评价之间的关系, 预测新用户对总体评价的结果。

表 1 满意度调查表
Table 1 Table of satisfaction survey

用户 User	评价 1 Assessment 1	评价 2 Assessment 2	评价 3 Assessment 3	总体评价 Overall assessment
1	+	+	+	+
2	+	-	-	-
3	+	+	-	+
4	+	+	+	+
5	+	-	+	+
6	+	-	-	-
7	+	+	-	+
8	+	-	+	-
9	+	+	-	+
10	+	+	-	-

2.1 构造等价类

将表 1 条件属性 (评价 1~3) 取值相同的用户归类到同一个关系 (等价类), 等价类结果如表 2 的第 1 列所示, $R_1\sim R_4$ 表示等价类的名称, 花括号中的阿拉伯数字表示用户编号。

表 2 基于表 1 的等价类构造及条件概率计算

Table 2 Equivalence class construction and conditional probability calculation on Table 1

等价类 Equivalence class	$P(\text{总体评价}=+ R_i)$ $P(\text{Overall assessment}=+ R_i)$
$R_1=\{1,4\}$	1
$R_2=\{2,6\}$	0
$R_3=\{3,7,9,10\}$	0.75
$R_4=\{5,8\}$	0.5

2.2 计算决策概率

从表 1~2 可以得到, R_1 等价类的总体评价均为+, 按照粗糙集推理策略, 若有新用户划分到 R_1 类, 新用户总体评价为+的概率为 100%; R_2 类中, 总体评价全部为-, 因此 R_2 类总体评价为+的概率为 0; R_3 类中 3、7、9 用户的总体评价为+, 10 用户的总体评价为-, 所以 R_3 类的总体评价为+的概率为 $3/4=75\%$ 。以“总体评价”为+作为条件, 每个等价类决策属性为+的概率值结果见表 2 的第 2 列。

经过等价类构造和决策概率计算, 总体评价属性的决策表如表 3 所示。

表 3 总体评价的决策表

Table 3 Decision of overall assessment

等价类 Equivalence class	属性 1 Attribute 1	属性 2 Attribute 2	属性 3 Attribute 3	条件概率 Conditional probability
R_1	+	+	+	1
R_2	+	-	-	0
R_3	+	+	-	0.75
R_4	+	-	+	0.5

2.3 评价指标

推荐时, 找不到等价类的样本, 即无先验知识的样本无法完成推荐; 概率粗糙集推荐模型中的边界域样本虽然能够找到等价类, 但因位于边界域所以也无法完成推荐。使用准确率 Precision、覆盖率 Coverage 和二者的 F-Score 值作为推荐效果的评价指标, 三个评价指标的计算公式如 (13)~

(15) 所示。

$$Precision = \frac{|(POS(X) \cap +) \cup (NEG(X) \cap -)|}{|POS(X) \cup NEG(X)|} \quad (13)$$

$$Coverage = \frac{|POS(X) \cup NEG(X)|}{|U|} \quad (14)$$

$$F - Score = \frac{2 \times Precision \times Coverage}{Precision + Coverage} \quad (15)$$

Precision 是正域样本推荐为+、负域样本推荐为-的数量之和与正域、负域样本总数的比值, 即推荐正确样本数和能够完成推荐的样本数的比值; *Coverage* 是正域、负域样本数之和与总样本数的比值, 即能够完成推荐的样本数与参加推荐的总样本数的比值; 通常情况下这两个比值是互相制约的, 因此使用 F-Score 值考察推荐结果的优劣。

按照粗糙集推理规则, 将划分到正域中的等价类的总体评价预测为+, 划分到负域中的等价类的总体评价预测为-, 当 $\alpha=1, \beta=0$ 时, 由表 3 的决策表可得到概率粗糙集的三个区域分别为: $POS_{1,0}(X)=R_1, NEG_{1,0}(X)=R_2, BND_{1,0}(X)=R_3 \cup R_4$, 边界域无法预测, 即可以为 R_1 和 R_2 等价类预测, 无法为 R_3 和 R_4 等价类预测, R_1 类预测为+、 R_2 类预测为-, 所以预测的准确率 $Precision=100\%$, 但只能为 1、2、4、6 号共 4 个用户预测, 预测的覆盖率 $Coverage=4/10=40\%$, $F-Score=57.1\%$ 。

当 $\alpha=0.6, \beta=0$ 时, 三个区域分别为: $POS_{0.6,0}(X)=R_1 \cup R_3, NEG_{0.6,0}(X)=R_2, BND_{0.6,0}(X)=R_4$, 此时预测的准确率 $Precision=7/8=87.5\%$, 覆盖率 $Coverage=8/10=80\%$, 牺牲了 12.5% 的准确率, 提高了 40% 的覆盖率, $F-Score=83.6\%$ 。

若 α, β 都取 0.5, 则边界域为空, 即 $Coverage=100\%, Precision=8/10=80\%, F-Score=88.9\%$ 。在实际应用中, α, β 参数通过学习获得具体取值。第 3 节介绍概率粗糙集模型中决策表训练和参数学习算法。

3 决策表构造和参数学习算法

将评分矩阵中的积极评分 (+) 标记为 1, 消极评分 (-) 标记为 -1, 无关项标记为 0, 将 *Precision*、*Coverage* 作为评价标准, 使用训练数据集 Train 构造基于概率粗糙集推荐模型的决策表、学习最优的参数 α, β , 然后通过测试数据集 Test 验证推荐模型的推荐效果。

3.1 属性约简

定义 $U=\{u_i | i=1, \dots, n\}$ 表示用户集, $I=\{i_j | j=1, \dots, m\}$ 表示项目集, 项目可以是商品、电影、新闻等, 评分矩阵 $R=\{r_{ij} | i=1, \dots, n, j=1, \dots, m\}$ 表示用户与项目之间的历史行为。为了使得算法能够适用于全数据集的推荐, 因此要为所有的项目都构造出一份有用的决策表, 当学习项目 i_j 的决策表时, 项目 i_j 的有效评分作为决策属性, 项目 i_j 以外的项目评分作为条件属性, 由于评分矩阵中的项目数量多, 因此初始时条件属性有 $m-1$ 个, 条件属性数量过多不利于等价类的构造和测试时的匹配搜索, 因此要减少条件属性的数量, 即属性约简。

将初始的 $m-1$ 个条件属性中有效评分最多的前 N 项属性作为条件属性, 对于不同的数据集, N 的取值不同, 需要通过训练学习得到。

3.2 决策表构造和参数学习

基于全数据集的决策表构造和参数学习算法见算法 1 所示。

算法 1 决策表构造和参数学习

输入: 训练集评分矩阵 Train, N, α', β'

输出: 等价类 Equi_Class 及对应参数 α, β

1. 从 Train 中提取项目 i 的非 0 评价矩阵 Train (i)
2. 从 Train (i) 中提取除项目 i 外、有效评价数量最多的前 N 个项目的的评价数据及项目 i 的评价数据, 存入 Info_Table(i)
3. 从 Info_Table(i) 中提取项目 i 的等价类表 Equi_Class(i)

- 4. $\alpha(i) = 1, \beta(i) = 0$
- 5. $\alpha(i) = 0.01$, 直到 $\alpha(i) < \alpha'$ 或 $Precision(i) < 88\%$ 和 $Coverage(i) < 88\%$
- 6. $\beta(i) = 0.01$, 直到 $\beta(i) > \beta'$ 或 $Precision(i) < 88\%$ 和 $Coverage(i) < 88\%$

4 实验结果分析

使用 100 k 的 Movie Lens^[11,12]数据集验证文中提出的推荐算法, 数据集中存储电影评分记录 10 万条, 每条记录由用户 id、电影 id 和评分值三项构成, 评分值为 1~5 的整数, 评分记录情况如表 4。

表 4 评分表简介
Table 4 Introduction of rating

项目 Item	简介 Introduction
用户数	共 943 个, id 从 1 到 943
项目数	共 1682 个, id 从 1 到 1682
稀疏度	100 k/(943*1682)≈0.063

将 100 k 的数据按照 8:2 的比例划分为训练集 Train 和测试集 Test, 因部分电影的评分记录较少, 所以有些电影只出现在 Train 或 Test 中。

4.1 N 值的训练

以 Train 中有效评分较多的 1 号和 50 号电影为例, 表 5 列出的是当 N 值分别设置为 5、8、10、15 时, 得到的等价类数量与条件属性 N 值的关系。

表 5 N 与等价类数量的关系
Table 5 Relationship between N and the number of equivalent classes

N	电影 1 Movie 1	电影 50 Movie 50
5	72	79
8	224	278
10	305	389
15	342	448

N=15 时, 等价类个数与 Train (i) 的行数差值很小, 即能够被划分到同一等价类的用户量非常少, 则划分的集合粗糙度小, 失去粗糙集推荐的意义; N=10, 等价类个数降低不明显; N=5, 等价类数量减少迅速, 用来划分集合, 粗糙度相对较高; 若 N 值再降低, 将导致粗糙程度加大, 不利于推荐, 因此 N 可设置为 5 到 10 之间的整数。

4.2 参数学习结果

评分表中的实际评分值为 1~5 的整数, 为了适应粗糙集推荐模型, 将 5 级评分简化为 2 级评分, 设置划分值 γ , $\gamma=1$, 表示评分值 1 标记为 -1, 其余有效评分值标记为 1; $\gamma=2$, 表示 1、2 分标记为 -1, 其余有效评分值标记为 1, 以此类推。

当 $\alpha'=\beta'=0.5$ (后文无特别说明, α' 、 β' 均设置为 0.5)、 $\gamma=2$ 时, 参数学习后, α 、 β 、准确率、覆盖率、F-Score 的平均值如表 6 所示。

表 6 训练集训练指标
Table 6 Training index of Train

项目 Item	N=5	N=8	N=10
α	0.68	0.76	0.84
β	0.20	0.16	0.12
Precision	91%	96 %	97%
Coverage	97%	98%	98%
F-Score	94%	97%	97%

从表 6 可见, 训练集的平均准确率和覆盖率指标都在 90% 以上。

4.3 测试结果与分析

4.3.1 测试结果分析 表 7 是划分值为 1 时概率粗糙集模型的推荐效果。

表 7 $N=1$ 时的推荐效果

Table 7 Recommended effect of $N=1$

N	数据集	非等价电影	正确分类电影	边界域电影	错误分类电影	精度	覆盖率	F 值
	Data set	NF	CCI	BI	ECI	Precision	Coverage	F-Score
5	Test	11	18484	30	1434	92.80%	99.79%	96.17%
	Train	0	77066	123	2811	96.48%	99.85%	98.14%
8	Test	68	18234	2	1655	91.68%	99.65%	95.50%
	Train	0	78663	25	1312	98.36%	99.97%	99.16%
10	Test	239	17938	0	1782	90.96%	98.80%	94.72%
	Train	0	79382	6	612	99.23%	99.99%	99.61%

表中 $Precision=CCI/(CCI+ECI)$, $Coverage=(CCI+ECI)/(NF+CCI+BI+ECI)$ 。由表 7 可见, N 值越大, 找不到等价类的记录越多, 当 $N=5$ 时, Test 集合的总体评价价值最优, 经验证, 对其他划分值 N 取 5 时评价价值都是最优, 后文无特别说明 N 值取 5。

图 2 是划分值分别取 1、2、3、4, Test 集合推荐效果比较, 当划分值取 1 时效果最优, 取 3 时效果最差, 准确率只有 58.77%, 整个模型的覆盖率指标都非常高, 为了提高推荐精确度, 可以牺牲一定数量的覆盖率指标值。

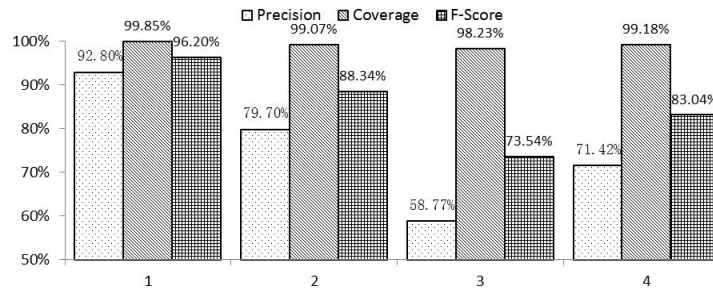


图 2 不同 N 的推荐效果比较

Fig.2 Recommended effect comparison of different N

4.3.2 与 Pawlak 粗糙集推荐模型的比较 Pawlak 粗糙集模型可以使训练集的推荐准确率达到 100%, 表 8 是 Test 和 Train 集合划分值分别取 1、2、3、4 时的三项指标值。

表 8 Pawlak 粗糙集模型与概率粗糙集模型性能比较

Table 8 Recommended effect comparison between Pawlak and probabilistic rough set

γ	模型	精度	覆盖率	F 值	数据集
	Model	Precision	Coverage	F-Score	Data set
1	A	93.18%	83.26%	87.94%	Test
	B	92.80%	99.85%	96.20%	
	A	79.97%	65.15%	71.80%	
	B	79.70%	99.07%	88.34%	
	A	57.74%	63.82%	60.63%	
	B	58.77%	98.23%	73.54%	
	A	74.06%	80.12%	76.97%	
	B	71.42%	99.18%	83.04%	
2	A	100.00%	81.63%	89.89%	Train
	B	96.48%	99.85%	98.14%	
	A	100.00%	65.44%	79.11%	
	B	91.72%	99.00%	95.22%	
	A	100.00%	65.03%	78.81%	
	B	88.08%	98.32%	92.92%	
	A	100.00%	78.91%	88.21%	
	B	92.32%	99.29%	95.68%	

Model 列中 A 表示“Pawlak 粗糙集模型”, B 表示“概率粗糙集模型”, 由数据可见, 在划分值相同时, 准确率差值较小, Test 集合的最大差值不到 3%, 覆盖率差值较大, Test 集合的最大差值达到 35%, 可见, 概率粗糙集模型和综合指标优于 Pawlak 粗糙集模型。

图 3 是 Test 集合不同划分值的各项指标比较。

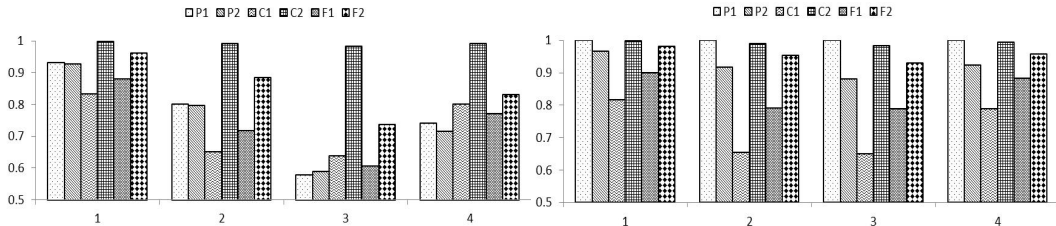


图 3 Test 集合不同 N 的评价指标比较 Fig.3 Recommended effect comparison of Test under different N
 图 4 Train 集合不同 N 值的评价指标比较 Fig.4 Recommended effect comparison of Train under different N

P1、C1、F1 和 P2、C2、F2 分别表示 Pawlak 模型和概率粗糙集模型下 Precision、Coverage、F-Score 指标的值得。Train 集合不同划分值的各项指标比较(图 4)。

4.3.3 α' 和 β' 对结果的影响 为了提高推荐准确率，应提高参数 α 的值，从表 8 可以看出，整个推荐的覆盖率指标较好，即参数 β 可以不调整。将 α' 和 β' 的值平滑变化，经实验验证，即使将 β' 的值设置较高，实际得到的 β 都不会超过 0.5，而 α' 的值对 α 影响较大，最后得到的 α 与 α' 接近。

划分值取 2，Test 集合的预测结果如表 9 所示。

表 9 α' 和 β' 对结果的影响
 Table 9 Recommended effect of α' and β'

α'	β'	正确分类电影 CCI	边界域电影 BI	错误分类电影 ECI	精度 Precision	覆盖率 Coverage	F 值 F-Score
1	0	10388	6949	2602	79.97%	65.15%	71.80%
0.8	0.8	13417	3381	3141	81.03%	83.04%	82.02%
0.7	0.7	14262	2280	3397	80.76%	88.57%	84.48%
0.6	0.6	15101	1091	3747	80.12%	94.53%	86.73%
0.5	0.5	15744	186	4009	79.70%	99.07%	88.34%
0.4	0.4	15439	686	3814	80.19%	96.56%	87.62%
0.3	0.3	14836	1606	3497	80.93%	91.95%	86.09%
0.2	0.2	13987	2739	3213	81.32%	86.26%	83.72%
0.1	0.1	11889	5249	2801	80.93%	73.67%	77.13%

当 α' 取 1、 β' 取 0 时，边界域的数量最多； α' 和 β' 都取 0.5 时，边界域数量最少，所以推荐的覆盖率指标较高。

5 结语

使用概率粗糙集理论中的 α 、 β 参数将粗糙集中的边界域分配到正域和负域中，可以降低评分矩阵稀疏性对推荐精度的影响，采用 α 、 β 参数逐步逼近、保证推荐准确率和覆盖率指标的参数学习算法，有效的提高了推荐效果，比 Pawlak 粗糙集模型的性能更优。粗糙集推荐模型的边界域是无法实施推荐的，可以结合其他推荐算法，进一步提高推荐效果。

参考文献

- [1] Guan Y, Cai SM, Shang MS. Recommendation algorithm based on item quality and user rating preferences[J]. Frontiers of Computer Science, 2014,8(2):289-297
- [2] Lops P, Gemmis MD, Semeraro G, et al. Content-based and collaborative techniques for tag recommendation: an empirical evaluation[J]. Journal of Intelligent Information Systems, 2013,40(1):41-61
- [3] Liu XC, Zhou JC. Research on Knowledge-based Personalized Recommendation Service System Retrieval Service[J]. Energy Procedia, 2011,13(4):10103-10108
- [4] 孙光福,吴乐,刘淇,等.基于时序行为的协同过滤推荐算法[J].软件学报,2013,24(11):2721-2733
- [5] Liu D, Li TR, Zhang JB. Incremental updating approximations in probabilistic rough sets under the variation of attributes[J]. Knowledge-Based Systems, 2015,73(1):81-96
- [6] Ma XA, Wang GY, Yu H, et al. Decision region distribution preservation reduction in decision-theoretic rough set model[J]. Information Sciences, 2014,278(10):614-640
- [7] Azam N, Yao JT. Game-theoretic rough sets for recommender systems[J]. Knowledge-Based Systems, 2014,72(1):96-107
- [8] 陈昊,杨俊安,庄镇泉.变精度粗糙集的属性核和最小属性约简算法[J].计算机学报,2012,35(5):1011-1017
- [9] 徐久成,李晓艳,孙林.一种基于概率粗糙集模型的图像语义检索方法[J].南京大学学报:自然科学版,2011,47(4):438-445
- [10] 王兆浩,舒兰,丁修勇.几种粗糙集模型的推广研究[J].计算机工程与应用,2011,47(36):68-72
- [11] 袁汉宁,周彤,韩言妮,等.基于 MI 聚类的协同推荐算法[J].武汉大学学报:信息科学版,2015,40(2):253-257
- [12] 田谦益,王小北.基于 Memetic 框架混合群智能算法的 NARMAX 模型参数辨识[J].山东科技大学学报:自然科学版,2013,32(1):88-94