文章编号: 1001-0920 (2012) 02-0313-04

一种基于属性序的决策规则挖掘算法

官礼和[1], 王国胤[2,3], 胡 峰[1,3]

(1. 西南交通大学 信息科学与技术学院，成都 610031；2. 中国科学院重庆绿色智能技术研究院 电子信息技术研究所，重庆 401122；3. 重庆邮电大学 计算智能重庆市重点实验室，重庆 400065)

摘 要: 针对面向领域用户的决策规则挖掘问题, 用属性序描述领域用户的需求和兴趣, 模拟人脑分辨事物的过程, 提出了一种属性序下的分层递阶决策规则挖掘算法. 该算法在给定属性序下输出的决策规则集不仅具有唯一性, 且对任意待识别样本不会作出矛盾的决策. 实例和仿真实验结果表明了算法的有效性和可行性.

关键词: 数据挖掘；粗糙集；属性序；决策规则

中图分类号: TP18 **文献标识码:** A

A decision rules mining algorithm based on attribute order

GUAN Li-he[1], *WANG Guo-yin*[2,3], *HU Feng*[1,3]

(1. School of Information Science and Technology，Southwest Jiaotong University，Chengdu 610031，China; 2. Institute of Electronic Information Technology，Chongqing Institute of Green and Intelligent Technology of Chinese Academy of Sciences，Chongqing 401122，China；3. Chongqing Key Laboratory of Computional Intelligence, Chongqing University of Posts and Telecommunications，Chongqing 400065，China. Correspondent：GUAN Li-he, E-mail：guanlihe@cqjut.edu.cn)

Abstract: For a decision rules mining problem of user's requirements, the attribute order is used as the description of user's requirements, and a hierarchical algorithm of decision rules mining based on the attribute order is proposed. For a given attribute order, the set of decision rules not only is unique, but also the inconsistent decision can not be done for any new object. Simulation experimental results show the effectiveness and feasibility of the algorithm.

Key words: data mining；rough set；attribute order；decision rule

1 引 言

粗糙集理论[1]能用确定方法处理不确定和不完整信息, 不需要先验知识, 直接从数据中获取知识, 该理论已经取得了许多研究成果[2]. 在基于粗糙集理论的知识获取研究中, 值约简是其中重要的研究内容之一, 许多学者已对值约简算法进行了大量的研究[2-7]. 但这些研究没有区分用户对数据集的需求和兴趣, 为所有用户给出的是一样的知识. 实际应用中, 一个数据集通常被多个用户共享, 不同用户对数据集往往有着不同的需求, 且希望从数据中获取自己感兴趣的知识. 因此, 许多研究者已经开始研究面向领域用户的知识获取方法[8-12].

在面向领域用户的决策规则挖掘中, 如何描述用户的需求是一个关键问题. 最近, 这一问题已经开始受到重视. 王珏等人[8]提出了属性序的概念, 并给出了一种基于属性序的属性约简算法, 该算法能得到唯一的 Pawlak 约简. 事实上, 属性序是领域用户需求的一种重要描述方式, 它体现了属性相对于用户的重要性, 在面向领域用户的数据挖掘中具有重要意义. 自属性序的概念被提出以来, 许多学者作了进一步的研究[9-12], 但这些研究都是针对属性约简的, 对于给定属性序下的值约简问题至今未过多涉及.

本文在给定属性序下, 模拟人脑分辨事物的过程, 提出一种决策规则挖掘算法. 该算法在给定属性序下输出的决策规则集不仅具有唯一性, 且对任意待识别样本不会作出矛盾的决策. 最后通过实例和仿真实验验证了算法的有效性和可行性.

收稿日期: 2010-09-15；**修回日期:** 2010-12-22.

基金项目: 国家自然科学基金项目(61073146)；重庆市杰出青年科学基金项目(2008BA2041)；重庆市教委科学技术研究项目(KJ090512)；重庆市/信息产业部"计算机网络与通信技术重点实验室" 开放基金项目(CY-CNCL-2010-4).

作者简介: 官礼和(1975−), 男, 讲师, 博士生, 从事粗糙集理论及其应用研究；王国胤(1970−), 男, 教授, 博士生导师, 从事粗糙集理论、智能信息处理等研究.

2 基本概念

下面简要介绍本文主要用到的一些粗糙集的基本概念, 详细内容参见文献[2,8].

定义 1 决策表 $S = \langle U, A, V, f\rangle$, 其中 U 为对象的集合, 也称为论域, $A = C \bigcup D$ 为属性集合, C 和 D 分别称为条件属性集和决策属性集, $D \neq \varnothing$, $V = \bigcup\limits_{a\in A} V_a$ 为属性值的集合, $f : U \times A \to V$ 为信息函数, 它指定 U 中每个对象 x 的属性值.

定义 2 在决策表 S 中, 任一决策规则可表示为 $t \to s|\alpha$, 其中: 前件 $t = \bigwedge(c, v), c \in B \subseteq C, v \in V_c$, 后件 $s = (d, w), w \in V_d, \alpha$ 为可信度. 特别地, 若 $\alpha = 1$, 则称该规则为确定规则; 否则, 为不确定规则.

定义 3 在决策表 S 中, 在 C 上定义一个完整的序关系"$\succ$", 将 C 中的所有属性分别标上 1 到 $|C|$. 这样, 在 C 上就得到了一个关于属性的序列, 称为"属性序", 记为 $AO : c_1 \succ c_2 \succ \cdots \succ c_{|C|}$, 且属性的编号越小, 该属性对用户越重要.

3 基于属性序的决策规则挖掘算法

3.1 算法描述

由于人脑中的知识结构是一个层次结构, 决定了人脑对事物的认识是多层次、多粒度的, 并且是一个有序的、层次递进的过程. 简而言之, 人脑在分辨事物时, 总是先利用最容易观察的现象或最有价值的信息, 若不能正确分辨, 则进一步利用次重要的信息, 逐次递进, 直到能正确分辨事物为止. 人脑分辨事物的过程可以在较浅的层次上利用较小的代价对事物进行正确分辨, 避免了在较深层次上的复杂性, 高效地实现信息处理.

事实上, 数据中隐藏的知识也是有层次、有粒度的, 并且知识的层次越深粒度越细. 在面向领域用户的决策规则挖掘问题中, 属性序作为领域用户需求的一种描述方式, 体现了一种层次性. 因此, 依据给定的属性序, 完全可以模拟人脑分辨事物的过程进行决策规则挖掘. 假定决策表 S 中 C 上的属性序为 $AO : c_1 \succ c_2 \succ \cdots \succ c_{|C|}$. 依据 AO, 首先在最重要属性 c_1 上区分对象, 若不能区分, 则进一步在次重要属性 c_2 上进行区分, 逐次递推, 直至能准确区分对象或所有条件属性都用完为止. 这一对象区分过程, 体现了逐步利用已有知识缩小问题求解范围, 直至逼近或得到最终结果的思想, 是一个分层逐步求精的决策规则挖掘过程. 于是, 一种基于属性序的决策规则挖掘算法可描述如下:

算法 1 基于属性序的决策规则挖掘算法

输入: 决策表 S 和 $AO : c_1 \succ c_2 \succ \cdots \succ c_{|C|}$.

输出: 决策规则集 RUL.

Step 1: $\mathrm{RUL} = \varnothing$.

Step 2: 采用文献 [11] 中基于属性序和分治法的属性约简算法对 S 进行属性约简, 得到唯一的属性约简 B.

Step 3: 令 $B = \{c_1^{'}, c_2^{'}, \cdots, c_{|B|}^{'}\}$, 且 $c_1^{'} \succ c_2^{'} \succ \cdots \succ c_{|B|}^{'}$. 先计算决策属性 d 对论域 U 的划分 $U/\{d\} = \{Y_1, Y_2, \cdots, Y_t\}$; 再调用递归函数 $\mathrm{DiguiFun}(U, 1)$ 获取决策规则.

Step 4: 输出决策规则集 RUL.

递归函数 $\mathrm{DiguiFun}(U, 1)$ 描述如下:

Void DiguiFun(ObjectSet Oset, Int r)

{

令 $V_r^{'} = \{c_r^{'}(x) : x \in O\mathrm{set}\}$, 先在 $c_r^{'}$ 上将 Oset 分解为 $|V_r^{'}|$ 部分: $O\mathrm{set}_1, O\mathrm{set}_2, \cdots, O\mathrm{set}_{|V_r^{'}|}$; 再对每一部分 $O\mathrm{set}_k (1 \leqslant k \leqslant |V_r^{'}|)$ 分 3 种情况进行处理:

1) 存在 $Y_i \in U/\{d\}$ 满足 $O\mathrm{set}_k \subseteq Y_i$. 此时由 $x \in O\mathrm{set}_k$ 在 $c^{'}_1, c^{'}_2, \cdots, c^{'}_r$ 和 d 上的取值构建一条确定规则, 并将其并入 RUL 中.

2) 不存在 $Y_i \in U/\{d\}$ 满足 $O\mathrm{set}_k \subseteq Y_i$, 且 $r < |B|$. 此时, 若存在 $s (r < s \leqslant |B|)$ 使得 $|\{c_s^{'}(x) : x \in O\mathrm{set}_k\}| > 1$, 则调用函数 $\mathrm{DiguiFun}(O\mathrm{set}_k, s)$; 否则, 若存在 $Y_j \in U/\{d\}$ 使得 $\alpha = |Y_j \bigcap O\mathrm{set}_k|/|O\mathrm{set}_k| > 0.5$, 则由 $y \in Y_j \bigcap O\mathrm{set}_k$ 在 $c^{'}_1, c^{'}_2, \cdots, c^{'}_r$ 和 d 上的取值构建一条可信度为 α 的不确定规则, 并将其并入 RUL 中.

3) 不存在 $Y_i \in U/\{d\}$ 满足 $O\mathrm{set}_k \subseteq Y_i$, 且 $r \geqslant |B|$. 此时, 若存在 $Y_j \in U/\{d\}$ 使得 $|Y_j \bigcap O\mathrm{set}_k|/|O\mathrm{set}_k| > 0.5$, 则由 $y \in Y_j \bigcap O\mathrm{set}_k$ 在 $c_1^{'}, c_2^{'}, \cdots, c_{|B|}^{'}$ 和 d 上的取值构建一条可信度为 $|Y_j \bigcap O\mathrm{set}_k|/|O\mathrm{set}_k|$ 的不确定规则, 并将其并入 RUL 中.

}

3.2 算法性质

在算法 1 中, 事先确定的属性序是算法成功获得用户感兴趣的决策规则的关键. 在实际应用中, 条件属性集上的属性序可依据对象属性值获取的难易程度、成本代价、实时性以及领域用户的需求等来确定. 此外, 由算法 1 获得的决策规则集具有如下 3 个性质.

命题 1 算法 1 可获得给定属性序下的所有确定规则和可信度 >0.5 的不确定规则.

由算法 1 中 Step 3 直接可证命题成立, 证明略.

命题 2 算法 1 在给定属性序下获得的决策规则集具有唯一性.

证明 首先, 由文献 [8] 中的命题 3.2 可知, 算法 1 中 Stept 2 对给定属性序可得决策表 S 的唯一属性约简. 其次, Stept 3 依据给定的属性序调用递归函数, 从编号最小的属性开始对对象集作划分, 对该划分中的

每个等价类作如下处理: 若该等价类是某个决策类的子集, 则由该等价类产生一条确定规则; 否则需调用递归函数对该等价类继续在编号次小的属性上继续作同样的处理, 逐次递推, 直至获得确定规则或可信度大于0.5的不确定规则为止. 可见, 这一决策规则的产生过程是确定的. 所以, 算法1得到的决策规则集是唯一的. □

命题3　假定RUL是由算法1获得的决策规则集, x 为一个待识别样本, 则不存在两条不同的决策规则 $\mathrm{DR}_1, \mathrm{DR}_2 \in \mathrm{RUL}$, 使得 x 在 DR_1 和 DR_2 上冲突.

证明　反证法. 假设存在两条不同的决策规则 $\mathrm{DR}_1, \mathrm{DR}_2 \in \mathrm{RUL}$, 使得 x 在 DR_1 和 DR_2 上冲突, 即 x 满足 DR_1 和 DR_2 的前件, 但 DR_1 和 DR_2 的决策值不同. 设 B 为属性约简, B_1 和 B_2 分别为 DR_1 和 DR_2 前件中的条件属性集, 则由算法1中步骤Step 3可知 B_1 与 B_2 之间必为包含关系. 不妨假定 $B_1 \subseteq B_2$, 且令 DR_1 和 DR_2 分别是由对象 x_1 和 x_2 产生的, 则 x_1 和 x_2 在 B_1 上不可区分, 必有 $B_1 = B$ (不然, 将继续调用递归函数, 依据 $B - B_1$ 中的属性编号从小到大对 x_1 和 x_2 进行区分, 而不是就此产生规则 DR_1). 由于 $B_2 \subseteq B$, 必有 $B_1 = B_2 = B$, 从而表明 x_1 和 x_2 在 B 上不可区分. 所以, 由 x_1 和 x_2 至多产生一条不确定规则, 这与 DR_1 和 DR_2 是规则集中两条不同的决策规则矛盾. □

3.3　算法复杂度

设 $|C| = m, |U| = n, p = \max\{|V_c| : c \in C\}$, 在最坏情况下(论域 U 中的所有对象均是不一致的), 算法1的平均时间复杂度为 $\max(O(nm(m + \log n)), O(p^{|B|}))$, 空间复杂度为 $O(nm)$.

4　实例说明

表1是一个决策表 S. 假定属性序为 $c_1 \succ c_2 \succ c_3 \succ c_4$, 现利用算法1来计算 S 的决策规则集:

Step 1: $\mathrm{RUL} = \varnothing$.

Step 2: 采用文献[11]中的属性约简算法计算得到 S 的唯一属性约简 $B = \{c_2, c_4\}$, 且有 $c_2 \succ c_4$.

Step 3: 决策类为: $Y_0 = \{x_1, x_2, x_5, x_6\}$ 和 $Y_1 = \{x_3, x_4\}$; $U/\{c_2\} = \{\{x_1, x_3\}, \{x_2, x_4, x_5, x_6\}\}$.

表1　决策表 S

U	C				d
	c_1	c_2	c_3	c_4	
x_1	0	1	1	0	0
x_2	1	0	2	1	0
x_3	0	1	2	1	1
x_4	1	0	1	2	1
x_5	1	0	1	2	0
x_6	1	0	1	2	0

$\{x_1, x_3\}$ 不是任何决策类的子集, 故调用递归函数进一步计算 $\{x_1, x_3\}/\{c_4\} = \{\{x_1\}, \{x_3\}\}$. 此时, $\{x_1\} \subseteq Y_0$ 且 $\{x_3\} \subseteq Y_1$, 从而由 x_1 和 x_3 分别得到两条确定规则 $d_1 : (c_2, 1) \bigwedge (c_4, 0) \rightarrow (d, 0)|1$ 和 $d_2 : (c_2, 1) \bigwedge (c_4, 1) \rightarrow (d, 1)|1$.

同理, $\{x_2, x_4, x_5, x_6\}$ 也不是任何决策类的子集, 故调用递归函数进一步计算 $\{x_2, x_4, x_5, x_6\}/\ \{c_4\} = \{\{x_2\}, \{x_4, x_5, x_6\}\}$. 由于 $\{x_2\} \subseteq Y_0$, 由 x_2 得到一条确定规则 $d_3 : (c_2, 0) \bigwedge (c_4, 1) \rightarrow (d, 0)|1$. 而 $\{x_4, x_5, x_6\}$ 不是任何决策类的子集且 B 中所有属性已用完, 故由 $x_5 \in Y_0 \bigcap \{x_4, x_5, x_6\}$ 得到一条可信度为0.67的不确定规则 $d_4' : (c_2, 0) \bigwedge (c_4, 2) \rightarrow (d, 0)|0.67$.

Step 4: 输出 $\mathrm{RUL} = \{d_1, d_2, d_3, d_4'\}$.

实例结果分析: 不难发现, 规则 d_1 可进一步简化为 $(c_4, 0) \rightarrow (d, 0)|1$, 即RUL中存在冗余条件属性的决策规则. 主要原因是: 算法1是依据给定属性序分层递阶地调用递归函数产生决策规则的, 而某些对象完全可能直接由次重要属性便可区分, 最终导致规则中出现冗余属性. 虽然如此, 但算法1的输出结果却具有一个非常好的性质, 即对任何待识别样本不会作出矛盾的决策(见命题3). 如果对算法1所得规则进一步简化, 则简化后的决策规则集将不一定具有命题3中的性质. 所以, 本文不打算进一步简化规则.

5　实验测试

为了验证算法1的有效性和可行性, 从UCI数据库中选取4个离散值数据集作为实验数据, 如表2所示. 实验测试环境是: CPU: P4 2.6 GHz, 内存: 512 MB, 操作系统: Windows XP, 开发工具为 VC++6.0.

表2　数据集

序号	数据集	记录数	条件属性数	决策属性数
1	Soybean-Small	47	35	1
2	Tic-tac-toe	958	9	1
3	Chess	3 196	36	1
4	Letter	20 000	16	1

由于原始数据集中没有给出任何用户对数据集的需求信息, 实验中依据文献[2]中条件属性相对于决策属性重要性的定义及其值域大小来确定属性序, 并采用五折交叉法分别利用本文的算法1, 文献[4]中基于决策矩阵的值约简算法(算法a), 文献[6]中改进的启发式规则获取算法(算法b)和文献[7]中最简决策规则挖掘算法(算法c)进行对比实验. 实验结果如表3和表4所示, 其中 N 为规则数, CR, NR 和RR分别为正确、错误和拒绝识别率.

从表3和表4可以看出, 算法1得到的规则数与其他3种算法有一定差异, 在识别率方面具有较好的性能. 由于算法c只能获得确定规则, 导致其各指标值

表 3 实验结果 1

序号	算法 a				算法 b			
	N	CR	NR	RR	N	CR	NR	RR
1	13.4	0.71	0.12	0.17	13.4	0.71	0.12	0.17
2	142.2	0.77	0.09	0.14	396.2	0.82	0.11	0.07
3	985.4	0.83	0.10	0.07	1 124	0.83	0.09	0.08
4	12 856	0.36	0.26	0.38	11 771	0.35	0.26	0.39

表 4 实验结果 2

序号	算法 c				算法 1			
	N	CR	NR	RR	N	CR	NR	RR
1	10.4	0.51	0.28	0.21	14.6	0.81	0.08	0.11
2	102.2	0.57	0.29	0.14	260.2	0.83	0.09	0.08
3	685.6	0.63	0.22	0.15	782	0.88	0.06	0.06
4	6 756	0.34	0.26	0.40	12 653.4	0.47	0.15	0.38

不理想. 此外, 对于数据集Letter, 算法 1 的正确识别率虽然高, 但拒识别率仍然保持较高的水平. 主要原因是: 这个数据集中不一致的对象非常少, 而实验采用的是五折交叉法, 导致训练数据只包含了样本全集中的一部份, 由此获取的规则必然会导致大量测试样本不能被识别. 解决该问题的一种可行方法是让训练数据尽量包含所有可能的样本. 尽管如此, 算法 1 基于属性序的思想, 能根据不同用户的不同需求或兴趣, 为其从数据集中获得具有较高识别率的决策规则 (知识).

6 结 论

面向领域用户兴趣的知识获取是当前数据挖掘的一个重要研究内容. 本文将属性序用来描述领域用户的需求, 模拟人脑分辨事物的过程, 采用分层递阶的原则, 提出了一种基于属性序的决策规则挖掘算法. 算法体现了知识是有层次、有粒度的思想, 能以有限的信息和较少的代价尽可能获取高质量的知识, 从而达到用最少的信息量即可作出满意的决策, 具有明确的实际意义和实用价值..

参考文献(References)

[1] Pawlak Z. Rough sets[J]. Int J of Computer and Information Sciences, 1982, 11(1): 341-356.

[2] 王国胤. Rough 集理论与知识获取[M]. 西安: 西安交通大学出版社, 2001: 147-152.
(Wang G Y. Rough set theory and knowledge acquisition[M]. Xi'an: Xi'an Jiaotong University Press, 2001: 147-152.)

[3] Mollestad T, Skowron A. A rough set framework for data mining of propositional default rules[C]. Proc of the 9th Int Symposium on Foundations of Intelligent Systems. Berlin: Springer-Verlag, 1996: 448-457.

[4] Ziarko W, Cerone N, Hu X. Rule discovery from database with decision matrices[C]. Proc of the 9th Int Symposium on Foundation of Intelligent Systems, Berlin: Springer-Verlag, 1996: 653-662.

[5] Hong T P, Lin C E, Lin J H, et al. Learning cross-level certain and possible rules by rough sets[J]. Expert Systems with Applications, 2008, 34(3): 1698-1706.

[6] 张利, 卢秀颖, 吴华玉, 等. 基于粗糙集的启发式值约简的改进算法[J]. 仪器仪表学报, 2009, 30(1): 82-85.
(Zhang L, Lu X Y, Wu H Y, et al. Improved heuristic algorithm used in attribute value reduction of rough set[J]. Chinese J of Scientific Instrument, 2009, 30(1): 82-85.)

[7] 钱进, 孟祥萍, 刘大有, 等. 一种基于粗糙集理论的最简决策规则挖掘算法[J]. 控制与决策, 2007, 22(12): 1368-1372.
(Qian J, Meng X P, Liu D Y, et al. A mining algorithm for concise decision rules based on rough sets theory[J]. Control and Decision, 2007, 22(12): 1368-1372.)

[8] Wang Jue, Wang Ju. Reduction algorithms based on discernibility matrix: The order attributes method[J]. J of Computer Science and Technology, 2001, 16(6): 489-504.

[9] Zhao Kai, Wang Jue. A reduction algorithm meeting users' requirements[J]. J of Computer Science and Technology, 2002, 17(5): 578-593.

[10] Han Suqin, Wang Jue. Reduct and attribute order[J]. J of Computer Science and Technology, 2004, 19(4): 429-449.

[11] 胡峰, 王国胤. 属性序下的快速约简算法[J]. 计算机学报, 2007, 30(8): 1429-1435.
(Hu F, Wang G Y. Quick reduction algorithm based on attributes order[J]. Chinese J of Computer, 2007, 30(8): 1429-1435.)

[12] 韩素青, 赵岷. Reduct 理论[M]. 北京: 清华大学出版社, 2010: 65-264.
(Han S Q, Zhao M. Reduct Theory[M]. Beijing: Tsinghua University Press, 2010: 65-264.)