文章编号: 1001-0920(2004)09-0994-05

个性化决策规则的发现: 一种基于 Rough Set 的方法

蒙祖强, 蔡自兴
(中南大学 信息科学与工程学院, 湖南 长沙 410083)

摘 要: 为发现用户真正感兴趣的决策规则, 利用 RS 理论和方法设计了个性化决策规则发掘算法. 算法分为两步: 首先在属性约简中通过提出的理论尽可能去除用户不感兴趣的属性的方法来找出最佳约简; 然后在属性值约简中进一步去除与用户无关的属性, 从而抽取个性化决策规则. 从理论上论证了算法的有效性, 给出了实验分析, 证实了算法的可行性.

关键词: 个性化决策规则; 粗糙集; 约简; 知识发现
中图分类号: TP18 **文献标识码**: A

Discovery of personalized decision rule: A rough-set-based approach

MENG Zu-qiang, CAI Zi-xing
(College of Information Science and Engineering, Central South University, Changsha 410083, China. Correspondent: MENG Zu-qiang, E-mail: mengzuqiang@sohu.com)

Abstract: Based on rough set theory and method, an algorithm for discovering personalized decision rules is designed in order to find the rules which users really want. The expounded algorithm consists of two steps. First, the attributes with less interest of user are eliminated as possible during reduction of attributes, and in this way the best reduct can be found. Secondly, during reduction of attribute values, some irrelevant attributes are further eliminated, then abstractation of personalized decision rules is accomplished. The algorithm is proved efficient in theory, and the results of experiment show the feasibility of this algorithm.

Key words: personalized decision rule; rough set; reduct; knowledge discovery

1 引 言

现实中的数据库基本上都是面向多用户且为用户所共享的, 所以在数据采集阶段一般是尽可能考虑各种用户的需要, 使之包含与所有用户都有关的信息. 这样, 对于某一个特定用户, 这些数据蕴涵的知识中, 很多与该用户是无关的, 而用户真正需要的是感兴趣、对他有实际指导价值的知识, 本文称之为个性化知识, 挖掘个性化知识, 去掉不必要的多余知识, 是知识发现研究的一个重要内容和热点问题[1, 2].

从信息系统中获取知识是知识发现的一条重要途径, 知识多以决策规则的形式表现出来[3]. 这时相应的个性化知识称为个性化决策规则. 约简(包括属性约简和属性值约简)有多个, 利用不同的约简得到的决策规则是不同的, 所以找出与既定用户有关的约简是发现个性化决策规则的关键. 文献[4]在这方面作了一些研究, 它按属性的重要程序对其进行降序排序, 然后按属性在该序列中的先后关系对

收稿日期: 2003-09-28; **修回日期**: 2003-12-01.
基金项目: 国家自然科学基金重点资助项目(60234030).
作者简介: 蒙祖强(1974—), 男, 广西罗城人, 博士生, 从事机器学习与数据挖掘、多 agent 等研究; 蔡自兴(1938—), 男, 福建莆田人, 教授, 博士生导师, 从事人工智能、智能机器人、机器学习和自动控制等研究

约简进行"字典"排序. 其设计的效果相当于找出约简的"字典有序"序列中的第一个约简, 作为用户最感兴趣的约简. 该算法虽然可保证找到的约简中尽可能包含最重要的属性, 但也可能包含一些不重要的属性, 即"有最好的也可能有最坏的属性", 而最终影响了规则对问题本质的决策能力.

为此, 本文提出另一种算法, 使得约简中尽可能地去掉不重要的属性, 提高约简的"整体水平", 并探讨了信息系统被简化后, 在规则提取过程中如何进一步去掉不重要的属性, 进一步去掉用户不感兴趣的信息.

2　有关概念

2.1　信息系统和决策系统

一般地, 信息系统可以表示为$\langle U, A, \{V_a\}, f_a\rangle_{a\in A}$. 其中: U 是有限对象的集合(论域), A 是有限属性的集合, V_a 是对象在属性 $a \in A$ 上所有可能取值的集合, $f_a: U \to V_a$ 是论域 U 到值 V_a 上的映射, 称为信息函数. 利用信息函数 f_a, 在 U 上构造一个关于属性集 $B \subseteq A$ 的关系 R_B, 定义为: $R_B = \{\langle s_i, s_j\rangle \mid f_a(s_i) = f_a(s_j)$, 对于所有 $a \in B, s_i, s_j \in U\}$. 显然, R_B 是一个等价关系, 它对应于 U 的一个划分, 记为 P_B, 或者商集 U/R_B. 令 $A = C \cup D$, 并且 C 是条件属性集, D 是决策属性集, 则信息系统可表示为 $\langle U, C \cup D, \{V_a\}, f_a\rangle_{A\in a}$, 称为决策系统, 简记为 $\langle U, C \cup D\rangle$, 用 DS 表示.

2.2　分辨矩阵

DS 的分辨矩阵定义为 $M = (m_{ij})_{n\times n}$, 其中

$$m_{ij} = \begin{cases} \{a \in C \mid f_a(s_i) \neq f_a(s_j), s_i, s_j \in U\}, \\ \quad f_d(s_i) \neq f_d(s_j), i < j; \\ \varnothing, \text{else} \end{cases}$$

$i, j = 1, 2, \cdots, n$, $n = |U|$. M 包含了一个信息系统中为区别所有对象所需要的信息. 令 $g(\mathrm{DS}) = \bigwedge_{1\le i,j\le n, i<j} \bigvee m_{ij}$, 称为 DS 的分辨函数. 它是一种合取范式, 只要把 $g(\mathrm{DS})$ 化为最简的析取范式, 则每一个析取项即构成了一个约简.

3　个性化属性约简

3.1　决策系统的分辨空间及其性质

定义 1　对于分辨矩阵 $M = (m_{ij})_{n\times n}$, 称集合 $\tilde{\omega}$ 为 M 上的一个空间, 如果对任意 $c \in \tilde{\omega}$, 存在分辨矩阵 M 中的项 m_{ij}, 使得 $c \subseteq m_{ij}$; 令 $\mathrm{SP} = \{m_{ij} \mid m_{ij} \neq \varnothing, i < j, i, j = 1, 2, \cdots, n\}$, 称为分辨矩阵 M 上的全空间; 令 $\tilde{\omega}_{\mathrm{sub}} = \{c' \subseteq c \mid c \in \tilde{\omega}, c' \neq \varnothing\}$, 称为空间 $\tilde{\omega}$ 的子空间. 显然, $|\tilde{\omega}| = |\tilde{\omega}_{\mathrm{sub}}|$, 这与普通集合论中子集的定义不同.

定义 2　对任意 $c, c' \in \tilde{\omega}$, 如果 $c' \subseteq c$, 定义运算 Θ: $c \Theta c' = c$ (吸收律). 对 $\tilde{\omega}$ 中的元素两两进行 Θ 运算(若此运算存在), 称为对 $\tilde{\omega}$ 的 Θ 简化运算; 简化后所得到的空间称为 $\tilde{\omega}$ 的 Θ 空间.

易知, $\tilde{\omega}$ 的 Θ 空间一般不是 $\tilde{\omega}$ 的子空间, 除非 $\tilde{\omega}$ 的 Θ 空间还是 $\tilde{\omega}$ 本身.

定义 3　取属性 $x \in C$, 对所有满足 $x \in c \in \tilde{\omega}$ 且 $c - \{x\} \neq \varnothing$ 的 c, 代之以 $c - \{x\}$, 从而得到 $\tilde{\omega}$ 的一个子空间 $\tilde{\omega}'$, 即 $\tilde{\omega}' = \{c - \{x\} \mid x \in c \in \tilde{\omega}, c - \{x\} \neq \varnothing\}$. 把这种替代操作称为对空间 $\tilde{\omega}$ 的 x - 子化, 子空间 $\tilde{\omega}'$ 称为 $\tilde{\omega}$ 的 x - 子化空间, 记为 $\tilde{\omega}_x$. 如果不存在满足 $x \in c \in \tilde{\omega}$ 的 c, 或者 $c - \{x\} = \varnothing$, 则不能对空间 $\tilde{\omega}$ 进行 x - 子化.

定义 4　$\Gamma \subseteq C$ 称为空间 $\tilde{\omega}$ 的一个覆盖, 如果对任意 $c \in \tilde{\omega}$, $\Gamma \cap c \neq \varnothing$; 如果 $\Gamma \subseteq C$ 是 $\tilde{\omega}$ 的一个覆盖, 且对任意 $a \in \Gamma$, $\Gamma - \{a\}$ 都不是 $\tilde{\omega}$ 的覆盖, 则称 Γ 是空间 $\tilde{\omega}$ 的一个最小覆盖.

定理 1　如果 Γ 是全空间 SP 的一个覆盖, 则必然存在 reduct $\subseteq \Gamma$, 使得 reduct $\in$ red(DS), 即 reduct 为 DS 的一个约简; 且 SP 的任一最小覆盖 $\Gamma_{\min}$ 都是 DS 的一个约简, 即 $\Gamma_{\min} \in$ red(DS).

根据分辨矩阵和全空间的构造方法, 容易证明定理 1.

性质 1　如果 $\Gamma \subseteq C$ 是包含一个约简的属性集, 则 Γ 是全空间 SP 的覆盖.

证明　Γ 包含约简 reduct, 即 reduct $\subseteq \Gamma$. 由约简和分辨矩阵的定义可知, 对任意 M 中的项 m_{ij}, reduct $\cap\, m_{ij} \neq \varnothing$, 所以 $\Gamma \cap m_{ij} \neq \varnothing$, 从而 Γ 是全空间 SP 的覆盖. □

定理 2　假设 $\tilde{\omega}_{\mathrm{sub}}$ 为空间 $\tilde{\omega}$ 的一个子空间, $\Gamma \subseteq C$ 为子空间 $\tilde{\omega}_{\mathrm{sub}}$ 的一个覆盖, 则 Γ 也是空间 $\tilde{\omega}$ 的一个覆盖.

证明　对任意 $c \in \tilde{\omega}$, 由于 $\tilde{\omega}_{\mathrm{sub}}$ 为 $\tilde{\omega}$ 的一个子空间, 故必存在 $c' \in \tilde{\omega}_{\mathrm{sub}}$, 使得 $c' \subseteq c$. 由于 $\Gamma \subseteq C$ 为子空间 $\tilde{\omega}_{\mathrm{sub}}$ 的一个覆盖, 由定义 4, $\Gamma \cap c' \neq \varnothing$, 从而 $\Gamma \cap c \neq \varnothing$, 所以 Γ 也是空间 $\tilde{\omega}$ 的一个覆盖. □

定理 3　假设 $\tilde{\omega}'$ 为空间 $\tilde{\omega}$ 的 Θ 空间, $\Gamma \subseteq C$ 为 $\tilde{\omega}'$ 的一个覆盖, 则 Γ 也是空间 $\tilde{\omega}$ 的一个覆盖.

该定理的证明是显然的, 在此从略.

定理 2′　假设 $\tilde{\omega}_{\mathrm{sub}}$ 为空间 $\tilde{\omega}$ 的一个子空间, $\Gamma_{\min} \subseteq C$ 为子空间 $\tilde{\omega}_{\mathrm{sub}}$ 的一个最小覆盖, 则 $\Gamma_{\min}$ 也是空间 $\tilde{\omega}$ 的一个最小覆盖.

证明　$\Gamma_{\min}$ 为子空间 $\tilde{\omega}_{\mathrm{sub}}$ 的一个最小覆盖, 显

然也是$\tilde{\omega}_{sub}$的一个覆盖. 由定理2, Γ_{min}是空间$\tilde{\omega}$的一个覆盖.

假设$\tilde{\omega}_{sub}$是$\tilde{\omega}$的x_-子化空间, 由定义3知$x \notin \tilde{\omega}_{sub}$, 而且$x \notin \Gamma_{min}$(若$x \in \Gamma_{min}$, 因为$x \notin \tilde{\omega}_{sub}$, 故$\Gamma_{min} - \{x\}$也是$\tilde{\omega}_{sub}$的覆盖. 这与题设"$\Gamma_{min}$为$\tilde{\omega}_{sub}$的一个最小覆盖"相矛盾). 对任意$a \in \Gamma_{min}$, 由于$\Gamma_{min} - \{a\}$不是$\tilde{\omega}_{sub}$的覆盖, 所以存在$c \in \tilde{\omega}_{sub}$, 使得$\Gamma_{min} - \{a\} \cap c = \varnothing$. 如果$c \in \tilde{\omega}$, 则知$\Gamma_{min} - \{a\}$不是$\tilde{\omega}$的覆盖; 如果$c \notin \tilde{\omega}$, 但$c \in \tilde{\omega}_{sub}$, 则必然存在$c' \in \tilde{\omega}$, 使得$c = c' - \{x\}$. 而$x \notin \Gamma_{min}$, 故

$$\Gamma_{min} - \{a\} \cap c' = \Gamma_{min} - \{a\} \cap (c \cup \{x\}) =$$
$$(\Gamma_{min} - \{a\} \cap c) \cup (\Gamma_{min} - \{a\} \cap \{x\}) =$$
$$\varnothing \cup \varnothing = \varnothing,$$

所以在这种情况下, $\Gamma_{min} - \{a\}$不是$\tilde{\omega}$的覆盖. 总之, 对任意$a \in \Gamma_{min}$, $\Gamma_{min} - \{a\}$都不是$\tilde{\omega}$的覆盖, 所以Γ_{min}是$\tilde{\omega}$的最小覆盖 □

定理3′ 假设$\tilde{\omega}'$为空间$\tilde{\omega}$的Θ空间, $\Gamma_{min} \subseteq C$为$\tilde{\omega}'$的一个最小覆盖, 则Γ_{min}也是空间$\tilde{\omega}$的一个最小覆盖

定义5 由全空间SP经过有限次Θ简化运算和x_-子化操作所得到的空间, 统称为SP的导出空间.

定理4 假设$\Gamma \subseteq C$为SP的任一导出空间$\tilde{\omega}'$的一个覆盖, 则Γ也是SP的一个覆盖.

由定理2和定理3易知, 定理4是成立的.

定理5 假设$\Gamma_{min} \subseteq C$为SP的导出空间$\tilde{\omega}'$的一个最小覆盖, 则Γ_{min}也是SP的一个最小覆盖.

由定理4, 定理2′和定理3′可知, 定理5是成立的.

定理6 假设$\tilde{\omega}'$为SP的任一导出空间, $\Gamma_{min} \subseteq C$为$\tilde{\omega}'$的一个最小覆盖, 则Γ_{min}是决策系统DS的一个约简, 即$\Gamma_{min} \in red(DS)$.

由定理5知, Γ_{min}为SP的一个最小覆盖, 再根据定理1, Γ_{min}是决策系统DS的一个约简, 故定理6成立.

由定理6可知, 由任意的导出空间都可以构造决策系统DS的一个约简. 从这个角度讲, 就分辨能力而言, SP的任一导出空间均与初始系统的条件属性集C具有相同的分辨能力, 所以又把任一导出空间统称为DS上的分辨空间.

定义6 假如对任意$c \in \tilde{\omega}$, 均有$|c| = 1$, 则$\tilde{\omega}$称为M上的单目空间.

单目空间不一定是SP的导出空间, 但是导出空间和单目空间具有下列重要关系:

定理7 假设全空间SP在经过Θ简化运算得到导出空间$\tilde{\omega}_1$, $\tilde{\omega}_1$经过子化操作得到$\tilde{\omega}_2$, $\tilde{\omega}_2$又经过Θ简化运算得到$\tilde{\omega}_3$, …. 在Θ简化运算和子化操作交互进行过程中, 产生了一个导出空间序列$\tilde{\omega}$(=SP), $\tilde{\omega}_2$, $\tilde{\omega}_3$, …, 则该序列在经过有限次的Θ简化运算和子化操作后会收敛于一个单目空间.

证明 令$width(\tilde{\omega}) = max\{|c| \mid c \in \tilde{\omega}\}$, $i = 1, 2, \cdots$. 对序列中的任意$\tilde{\omega}$, 如果它是子化空间, 则可对其运用一次Θ简化运算, 使之变成Θ简化空间, 所以不防假设$\tilde{\omega}$是一个Θ简化空间.

如果$width(\tilde{\omega}) \geq 2$, 则假定$|c| = width(\tilde{\omega})$, $c \in \tilde{\omega}$. 进一步假定$c = \{a_1, a_2, \cdots, a_m\}$, 显然, $m = width(\tilde{\omega})$. 如果对$j = 1, 2, \cdots, m$, 有$\{a_j\} \in \tilde{\omega}$, 则根据定义3可知, 不能对$\tilde{\omega}$进行$a_{j-}$子化操作. 但是这种情况是不可能出现的, 因为$\tilde{\omega}$是一个$\Theta$简化空间, 上述的$\{a_j\}$和$c$不可能共存于$\tilde{\omega}$中. 这样, 必存在$a_j \in c$, 使得$\{a_j\} \notin \tilde{\omega}$, 于是至少可对$\tilde{\omega}$进行$a_{j-}$子化而得到$a_{j-}$子化空间$\tilde{\omega}_{+1}$. 所以, $width(\tilde{\omega}) > width(\tilde{\omega}_{+1})$, 即序列中导出空间的宽度是呈严格递减的, 最终必然达到宽度为1的一个导出空间, 这个空间就是一个单目空间. □

定理8 假设单目空间$\tilde{\omega}$是SP的一个导出空间, 则$\Gamma = \{a \in c \mid c \in \tilde{\omega}\}$为单目空间$\tilde{\omega}$的一个最小覆盖, 因而是SP的一个约简.

证明 取任意$a \in \Gamma$, 假设$a \in c \in \tilde{\omega}$, 由于$\tilde{\omega}$是单目空间, 所以$c = \{a\}$. 这样, $\Gamma - \{a\} \cap c = \Gamma - \{a\} \cap \{a\} = \varnothing$. 因此, 由定义4, Γ是$\tilde{\omega}$一个最小覆盖. 由于$\tilde{\omega}$是SP的一个导出空间, 根据定理6, Γ是SP的一个约简. □

3.2 约简算法

在决策系统$DS = \langle U, C \cup D \rangle$中, 假设$C = \{a_1, a_2, \cdots, a_h\}$, 某一用户相应的属性权值表$weight = \{t_1, t_2, \cdots, t_h\}$, 其中属性$a_i$与其权值$t_i$的对应关系用函数$w(a_i)$来表示, 即$t_i = w(a_i)$, $t_i \in [0, 1]$, $i = 1, 2, \cdots, h$.

对任意$\Gamma_1, \Gamma_2 \in red(DS)$, 按其属性权值的大小分别对$\Gamma_1$和$\Gamma_2$中的属性进行升序排列, 假设得到的有序序列分别为$\Gamma_1'$和$\Gamma_2'$, 并用$\Gamma_i'(j)$表示有序序列$\Gamma_i'$中的第$j$个属性; 如果$\Gamma_1'$和$\Gamma_2'$长度不等, 则在较短的属性序列的末尾补$\varnothing$(空集合), 并定义$w(\varnothing) = 0$. 这样, 如果"$\Gamma_1$比$\Gamma_2$更优(更能令用户感兴趣)", 则记为$Tail(\Gamma_1) > Tail(\Gamma_2)$. 其评判方法如下(即函数Tail()定义): 如果$w(\Gamma_1'(1)) > w(\Gamma_2'(1))$, 则$Tail(\Gamma_1) > Tail(\Gamma_2)$; 如果

$w(\Gamma_1'(1)) < w(\Gamma_2'(1))$, 则 $\mathrm{Tail}(\Gamma_1) < \mathrm{Tail}(\Gamma_2)$; 如果 $w(\Gamma_1'(1)) = w(\Gamma_2'(1))$, 则用同样的方法比较下一属性 $\Gamma_1'(2)$ 和 $\Gamma_2'(2)$, …, 直到最后一位; 如果每一位的权值都相等, 则 $\mathrm{Tail}(\Gamma_1) = \mathrm{Tail}(\Gamma_2)$. 不失一般性, 假设下文提到的覆盖(包括约简) 都是有序的, 另外把 Tail 函数值最大的约简作为最佳约简, 记为 Γ_{best}. 因为它是符合用户需要的, 所以称为个性化(属性) 约简. Γ_{best} 的求解算法可描述如下:

算法 1:

输入: 信息表(U 和 $C \cup D$ 刻画), 对应于 C 的权值表 $\mathrm{weight} = \{t_1, t_2, \cdots, t_h\}$

输出: 最佳约简 Γ_{best}

1) 扫描信息表, 建立分辨矩阵 $M = (m_{ij})_{n\times n}$;

2) 由 M 构造全空间 SP;

3) 令 $\tilde{\omega} =$ SP; //$\tilde{\omega}$ 为空间变量, 初值等于 SP;

4) 对空间 $\tilde{\omega}$ 进行 Θ 简化运算, 得到 $\tilde{\omega}$ 的 Θ 空间, 记为 $\tilde{\omega}'$;

5) 如果 $\tilde{\omega}'$ 是单目空间, 则令 $\Gamma_{best} = \bigcup_{a\in\tilde{\omega}'} a$, 算法终止; 否则转 6);

6) 令 $t = \min(\mathrm{weight})$, 并令 $\mathrm{weight} = \mathrm{weight} - \{t\}$;

7) 假设 $w(a) = t$, 如果能对空间 $\tilde{\omega}'$ 进行 a- 子化, 则令 $\tilde{\omega}_a =$ a- 子化空间; 否则转 6);

8) 令 $\tilde{\omega} = \tilde{\omega}_a$, 转 4).

由定理 7 可知, 该算法是收敛的.

定理 9　算法 1 产生的约简 Γ_{best} 是最佳约简, 即 $\mathrm{Tail}(\Gamma_{best})$ 最大.

由上面定理的证明过程和有关结论可知, 该定理是成立的, 证明略.

就个性化知识发现而言, 算法 1 和算法 2(文献[4] 提出的算法) 是针对不同应用目标、不同应用要求而提出的, 都是对个性化知识发现的完善与补充. 实际上, 它们的优缺点是互补的, 即算法 1 的优点正是算法 2 的缺点, 而算法 2 的优点又是算法 1 的缺点. 具体讲, 算法 1 是为了尽可能去掉一些最差的属性, 以保证约简的整体水平, 这可能丢掉一些最优属性; 而算法 2 则要保留一些最优的属性, 这可能使得一些很差的属性也在约简中出现, 使得约简的整体水平下降. 在具体应用时, 究竟选取什么样的算法, 则要根据实际问题、实际应用而定.

4　个性化决策规则的提取

用 $\mathrm{red} = \{b_1, b_2, \cdots, b_m\}$ 表示既得的约简, 相应简化的决策系统 $\mathrm{DS} = \langle U, \mathrm{red} \cup D\rangle$, 定义一种改进的分辨矩阵 $MV = (mv_{ij})_{n\times n}$, $n = |U|$, 其中

$$mv_{ij} = \begin{cases} \{a \in \mathrm{red}: f_a(s_i) \neq f_a(s_j), s_i, s_j \in U\}, \\ \quad f_d(s_i) \neq f_d(s_j), i \neq j; \\ \varnothing, \text{else.} \end{cases}$$

定义 7　$MV = (mv_{ij})_{n\times n}$, 对任意 $s_k \in U$, 令 $gv(s_k) = \bigwedge_{1\le i\le n} \bigvee mv_{ik}$, 称为对象 s_k 的分辨函数.

决策规则的提取可通过对各对象的分辨函数进行简化来实现, 不同的简化方法会产生不同的决策规则. 在特定领域中, 按照既定的能满足用户需要程度为标准的评价准则, 评价值高的规则称为个性化规则. 本文中 Tail 函数值大的决策规则就是个性化决策规则.

实际上, 属性值的约简也存在类似于属性约简的问题, 即是在规则中尽可能保存最好的属性, 还是尽可能去掉较差的属性. 对于后者, 分别定义分辨矩阵 MV 的每一个列的全空间 VSP_j (或称对象 s_j 的全空间), 即 $\mathrm{VSP}_j = \{mv_{ij} \mid mv_{ij} \neq \varnothing, i = 1, 2, \cdots, n\}$, 然后在算法 1 中从步骤 3) 起, 以 VSP_j 替代 SP 即可, 得到的算法记为算法 1′. 具体描述如下:

算法 1′

输入: 简化后的信息表(U 和 $\mathrm{red} \cup D$ 刻画), 用户相对于 C 的权值表 $\mathrm{weight} = \{t_1, t_2, \cdots, t_{|C|}\}$

输出: 个性化决策规则集 RS

Begin

扫描简化后的信息表, 建立 $MV = (mv_{ij})_{n\times n}$;

For $j = 1$ to $|U|$ {

1) 由 MV 构造对象 s_j 的全空间 VSP_j;

2) 令 $\tilde{\omega} =$ SP, temp = weight;

3) 对 $\tilde{\omega}$ 作 Θ 简化运算, 得到 $\tilde{\omega}$ 的 Θ 空间, 记为 $\tilde{\omega}'$;

4) If($\tilde{\omega}'$ 是单目空间) then {TempResult[j] = $\bigcup_{a\in\tilde{\omega}'} a$; continue; }

/* 如果条件成立, 则 $\bigcup_{a\in\tilde{\omega}'} a$ 是对象 s_i 的分辨函数 $gv(s_j)$ 的值, 暂存在 TempResult[j] 中, 算法去处理下一对象 */

5) If(temp $\neq \varnothing$), 令 $t = \min(\mathrm{temp})$, 并令 temp = temp - $\{t\}$; else continue;

6) 假设 $w(a) = t$, 如果能对空间 $\tilde{\omega}'$ 进行 a- 子化, 则令 $\tilde{\omega} = \tilde{\omega}_a$, 转 3); 否则转 5);

}

For $j = 1$ to $|U|$ {

For each $a \in$ red{

If($a \notin$ TempResult[J]) $T(s_j, a)$ = ' * ';
/* $T(s_j, a)$ 表示在简化的信息表中 a 所在的列与 s_j 所在的行的交叉处,符号 * 表示不重要或可以不关心 */
else $T(s_j, a)$ 保持原值不变;
}}
在当前简化的信息表中删除重复的行(仅保留一个)和被蕴涵的行,得到一个压缩的信息表;
把压缩的信息表转化为决策规则集 RS;
return RS; }
End.

5 实验分析

为了便于比较,实验中选用文献[5]给出的一组汽车数据. 假设用户 user 出于某种需要,欲了解车型(size)、汽缸数(cyl)和位移(displace)对汽车行驶总里程(mileage)可能存在的影响,然后依次是涡轮机(turbo),功率(power),压缩率(comp),重量(weight),燃料(fuelsys),挂档(trans)等. user 对于属性集的权值表如表 1 所示.

表 1 属性和权值的对应关系

属性	size	cyl	turbo	fuelsys	displace	comp	power	trans	weight
权值	1.00	0.95	0.100	0.081	0.940	0.085	0.090	0.01	0.083

用算法 1 求得最佳约简{size, fuelsys, displace, weight}(该约简记为 Rt1),而采用算法 2 求得的最佳约简为{size, cyl, turbo, fuelsys, trans, weight}(该约简记为 Rt2). 把这两个约简分别按它们的权值进行升序排列,结果如表 2 所示.

表 2 按权值升序排列的约简

属性	trans	fuelsys	weight	comp	power	turbo	displace	cyl	size
权值	0.01	0.081	0.083	0.085	0.09	0.10	0.94	0.95	1.00
Rt1	-	fuelsys	weight	-	-	-	displace	-	size
Rt2	trans	fuelsys	weight	-	-	turbo	-	cyl	size

注: 符号"—"表示相应的约简中不存在对应的属性

从表 2 可以看出,约简 Rt1 中的最差属性是 trans,而约简 Rt2 中的最差属性是 fuelsys. 实际上,{fuelsys, weight}是核属性,是每一个约简的公共部分,所有的约简算法都不能去除它们. 严格说,以上两个算法找到的约简中最差的属性分别是 displace 和 trans. 这样,约简 Rt2 包含了 trans 和 turbo 这两个 user 不怎么感兴趣的属性. 如果按照"少劣为优"的原则,那么约简 Rt1 较约简 Rt2 更能令用户感兴趣.

另外,user 不感兴趣的核属性{fuelsys, weight}在约简算法中是无法去除的,但在产生决策规则时,可用算法 1 尽可能去掉那些权值较低的属性(包括核属性). 同时对算法 2 进行相应的改造(得到的算法称为算法 2'),使之能够进行属性值的约简,以提取决策规则. 这里约定: 如果在属性约简阶段用算法 a,而在决策规则提出阶段(即属性值约简阶段)用算法 b,则说决策规则的提取采用了算法 a-b. 相应算法的运行结果如表 3 和表 4 所示. 从这两个表中可以看出,算法 1-1 产生的决策规则中所含属性的"质量"高一些,即属性权值大一些,多为用户关心的属性,如表 4 中的规则含有低权值属性 turbo 和 trans 等,而表 3 中则没有. 可见,算法 2-2 产生的决策规则中包含"质量"低一些的属性.

表 3 算法 1-1 产生的决策规则

R1: weight: = heavy⇒mileage: = low
R2: size: = compact∧displace: = medium⇒mileage: = medium
R3: size: = compact∧fuelsys: = 2-BBL∧displace: = small⇒mileage: = medium
R4: weight: = light⇒mileage: = high
R5: size: = subcompact⇒mileage: = high
R6: fuelsys: EFI∧displace: = small⇒mileage: = high

表 4 算法 2-2 产生的决策规则

R1: weight: = heavy⇒mileage: = low
R2: cyl: = 4∧trans: = low⇒mileage: = medium
R3: cyl: 6∧weight: = medium⇒mileage: = medium
R4: size: = compact∧cyl: = 4∧fuelsys: = 2-BBL⇒mileage: = medium
R5: size: = compact∧weight: = medium⇒mileage: = medium
R6: cyl: = 6∧turbo: = EFI⇒mileage: = medium
R7: weight: = low⇒mileage: = high
R8: size: = subcompact⇒mileage: = high
R9: cyl: = 4∧trbo: = *n*∧fuelsys: = EFI∧trans: = manual⇒mileage: = high

6 结 语

本文研究在信息系统中发现个性化决策规则的问题. 主要利用 Rough Set 理论和方法,提出分辨函数、分辨空间等重要概念,并以定理的形式给出了一系列的性质,进而得出相应的属性约简和属性值约简算法. 从理论上证明了这种算法对于发掘个性化决策规则是正确和有效的,从而使得挖掘系统避免产生那些包含用户不感兴趣的属性的规则;同时在实验中作了具体的对比分析,同样说明了算法的有效性.

(下转第 1003 页)

表 3　FGA 选用参数表

函数	Δ	m	h	N	P_m	交叉率	Δp_m	Δ1	Δ2	Σ_m	Σ_s
$f_1(x)$	0.1	8	10	40	0.000 1	0.9	0.1	0.02	0.2	0.42	0.49
$f_2(x)$	0.15	8	10	40	0.000 1	0.9	0.1	0.02	0.2	0.42	0.49
$f_3(x,y)$	0.2	6	10	40	0.000 1	0.9	0.1	0.02	0.2	0.42	0.49
$f_4(x,y)$	0.1	8	8	40	0.000 1	0.9	0.1	0.02	0.2	0.42	0.49

5　结　　语

本文分析了影响遗传算法性能的因素, 在此基础上设计了一个新的家族遗传算法. 该算法的主要思想是改良选择和变异算子, 同时在优良解附近的微型空间构造优良家族, 并在其中寻找更优的个体, 实现"龙生龙, 凤生凤". 在对比的实验中, FGA 比传统的遗传算法收敛速度几乎有了数量级上的飞跃, 精度也提高很多, 说明该算法具有应用的潜力.

参考文献(References):

[1] Holland J H. *Adaptation in Nature and Artificial System* [M]. Ann Arbor: The University of Michigan Press, 1975.

[2] Radolph G. Convergence analysis of canonical genetic algorithms[J]. *IEEE Trans on Neural Network*, 1994, 5(1): 96-101.

[3] Qix F. Palmieri theoretical analysis of evolutionary algorithms with an infinite population size in continuous space [J]. *IEEE Trans on Neural Network*, 1994, 5(1): 102-129.

[4] 庄健, 王孙安. 自调节遗传算法的研究[J]. 西安交通大学学报, 2002, 36(11): 359-363.
(Zhuang J, Wang S A. Study on self-adjusting of gene migration genetic algorithm [J]. *J of Xi'an Jiaotong University*, 1994, 36(11): 359-363.)

[5] 陈国良, 王煦法, 庄镇泉, 等. 遗传算法及其应用[M]. 北京: 人民邮电出版社, 1996.

[6] 张铃, 张钹. 统计遗传算法[J]. 软件学报, 1997, 8(5): 335-344.
(Zhang L, Zhang B. The statistical genetic algorithm [J]. *J of Software*, 1997, 8(5): 335-344.)

[7] 张铃, 张钹. 遗传算法机理的研究[J]. 软件学报, 2000, 11(7): 945-952.
(Zhang L, Zhang B. Research on the mechanism of genetic algorithms [J]. *J of Software*, 2000, 11(7): 945-952.)

[8] Chen Wei, Chen Li, Ma Yao. The improvement of genetic algorithm performance [A]. *Proc of 2002 Int Conf On Machine Learning and Cybernetics* [C]. Beijing, 2002. 945-951.

[9] 张文修, 梁怡. 遗传算法的数学基础[M]. 西安: 西安交通大学出版社, 2001. 54-79.

[10] 李建华, 王孙安. 最优家族遗传算法[J]. 西安交通大学学报, 2004, 38(1): 77-80.
(Li J H, Wang S A. Optimum family genetic algorithm [J]. *J of Xi'an Jiaotong University*, 2004, 38(1): 77-80.)

(上接第 998 页)

参考文献(References):

[1] Perng Chang-Shing, Wang Haixun, Ma Sheng, et al. User-directed exploration of mining space with multiple attributes [A]. *In the 2nd IEEE Int Conf on Data Mining (ICDM)* [C]. Maebashi, 2002. 394-401.

[2] Bayardo R J, Agrawal R. Mining the most interseting rules [A]. *Proc of 5th Int ACM SIGKDD Int Conf Knowledge Discovery Data Mining* [C]. San Diego, 1999. 145-154.

[3] 韩祯祥, 张琦, 文福拴. 粗糙集理论及其应用综述[J]. 控制理论与应用, 1999, 16(2): 153-157.
(Han Z X, Zhang Q, Wen F S. A survey on rough set theory and its application [J]. *Control Theory and Applications*, 1999, 16(2): 153-157.)

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

[5] 王珏. Rough Set 约简与数据浓缩[J]. 高技术通讯, 1997, (11): 40-45.
(Wang J. Rough set reduction and data enriching [J]. *High Technology Letters*, 1997, (11): 40-45.)