文章编号:1005－0523(2006)02－0103－04

序决策逻辑及基于启发式遗传算法的规则挖掘

凌仕勇,黄兆华

(华东交通大学 信息工程学院,江西 南昌 330013)

摘要:很多真实世界处理排序问题代替分类问题,例如由不同生产厂商生产的消费品,大学之间的排序等等.一般地,一个全局的信息表需要给定.本文采用了基于 Rough 集理论的方法来阐述排序规则的挖掘问题,且利用了基于遗传算法的启发式算法来达到算法的实现.

关　键　词:序决策逻辑;启发式遗传算法;Rough 集

中图分类号:TP391.41　　**文献标识码**:A

1　引言

归纳学习和数据挖掘的一个最基本的任务是从分类中学习知识,而且许多的研究也已经集中在了这个任务上面.特别是基于 Rough 集上的研究更是如此,因为这个理论是在研究分类的课题上发展起来的[1][2].然而,在真实的世界里,我们可能会面对许多不仅仅是分类的问题,例如,生产厂商之间的产品评比,大学之间的比较等等,这些问题可归于对象之间的排序问题.在诸于这种问题中,我们着重于观察哪些属性对于整个的排序规则起更大的作用,哪些属性对于整个的排序规则起更小的作用以及哪些属性的子集对于决定整个的排序规则更充分.基于此系列排序问题的规则挖掘就是在一个由属性集描述的对象集上,我们通过专家或由其它途径得来的信息,找出在这些全局或单个排序规则之间的联系.

通过遗传算法的面向问题的启发式控制策略结合了遗传算法和启发式算法的优点[3].在下面的内容里,我们将首先介绍关于序的决策逻辑与语言,以及基于序规则的挖掘的一些知识,接下来阐述启发式遗传算法用于序决策逻辑规则的挖掘,最后给出结论.

2　序决策逻辑

2.1　排序信息表

在真实世界中,不仅仅存在分类问题,还有许多对象的排列问题.典型情况下,一个全局的排序对象需要给定,在此,我们称之为排序信息表[4]

定义为 $IT=(U, At, \{V_a \mid a\in At\}, \{I_a \in At\})$

U:有限非空对象集

At:有限非空属性集

V_a:满足 $a\in At$ 的非空值的集合

I_a: $U\rightarrow V_a$ 是一个信息函数

例如:信息表 1

表1: 一个信息表的实例

	a	b	c	d	o
p_1	middle	3 years	$200	heavy	1
p_2	large	3 years	$300	very heavy	3
p_3	small	3 years	$300	light	3
p_4	small	3 years	$250	very light	2
p_5	small	2 years	$200	very light	3

我们可人为地给定它的排序顺序如下

$\succ a$: $small \succ middle \succ large$.

收稿日期:2005－11－09

基金项目:江西省自然科学基金(0411035)和华东交通大学科研基金资助项目

作者简介:凌仕勇(1975－),男,江西九江人,2004 级研究生..

$\succ_b$: 3 *years* $\succ$ 2 *years*.

$\succ_c$: $S200 \succ S250 \succ S300$,

$\succ_d$: *very light* $\succ$ *light* $\succ$ *heavy* $\succ$ *very heavy*.

$\succ_e$: $1 \succ 2 \succ 3$.

此信息表通过 $OIT=(IT, \{\succ_a \mid a \in At\})$定义

另外有: $x \succ_{\{a\}} y \Leftrightarrow I_a(x) \succ_a I_a(y)$, $\succ_{\{a\}}$, 表示通过属性 a 基于 U 的序化的联系.

对于包含于 At 的子集 A, 定义

$x \succ_A y \Leftrightarrow \forall_a \in A[I_a(x) \succ_a I_a(y)]$

$\Leftrightarrow \bigwedge_{a \in A} I_a(x) \succ_a I_a(y) \Leftrightarrow \bigcap_{a \in A} \succ_{\{a\}}$

此序化的联系包含两个属性: 自反性和反向传递性

自反性: $(x \succ y \Rightarrow \neg (y \succ x)$

反向传递性: $(\neg (x \succ y), \neg (y \succ z) \Rightarrow \neg (x \succ z)$

2.2 序决策逻辑(Ordered Decision Logic, ODL)

相似于 Zadeh 的决策逻辑[2][5][6], 设 $IS=(U, A)$是一个信息系统, 其中 U 是所讨论对象的全域, A 是属性集. (a, v)或 a_v 是指定义在 IS 上的一种描述, 其中 $a \in A$ 是属性集 A 上的一个属性, v 是 a 关于个体集 U 上的个体 $x \in U$ 的属性值, 也就是 $v=a(x)$. 于是(a, v)或 a_v 被视为 Rough 逻辑中的一个原子公式.

$m(\varphi)=\{x \in U: x \mid \approx_{IS} \varphi$ 被称做 IS 上公式 φ 的意义集, 其中符号$\mid \approx IS$ 是满足或者可能满足符, 即所有满足 φ 或者可能满足 φ 的 U 上个体的集合, 其中 m 是意义函数符. 因此$(\varphi, m(\varphi))$是 IS 上的一个关于 φ 的基本粒. 原子公式 a_v 的基本粒记成$(a_v, m(a_v))$, 被称做基本粒.

在 ODL 语言中, 原子表达通过(a, ϕ)或(a, π)来给定, 有如下类似的性质:

以及:

$(x, y) \mid = (a, \succ)$ iff $x \succ_{\{a\}} y$

$(x, y) \mid = (a, \leq$ iff $x \leq_{\{a\}} y$

$(x, y) \mid = \neg \phi$ iff $not(x, y) \mid = \phi$

$(x, y) \mid = \phi \wedge \Psi$ iff $(x, y) \mid = \phi$ and $(x, y) \mid = \Psi$

$(x, y) \mid = \phi \vee \Psi$ iff $(x, y) \mid = \phi$ *or* $(x, y) \mid = \Psi$

$(x, y) \mid = \phi \rightarrow \Psi$ iff $(x, y) \mid = \neg \phi \vee \Psi$

$(x, y) \mid = \phi \equiv \Psi$ iff $(x, y) = \phi \rightarrow \Psi$ and $(x, y) \mid = \Psi \rightarrow \phi$

以及

$m(a, \phi)=\{(x, y) \in U \times U \mid x \succ_{\{a\}} y\}$

$m(a, \pi)=\{(x, y) \in U \times U \mid x \leq_a y\}$

$m(\neg \varphi)=\neg m(\varphi)$

$m(\varphi \wedge \Psi)=m(\varphi) \cap m(\Psi)$

$m(\varphi \vee \Psi)=m(\varphi) \cup m(\Psi)$

$m(\varphi \rightarrow \Psi)=\neg m(\varphi) \cap m(\Psi)$

$m(\varphi \equiv \Psi)=(m(\varphi) \cap m(\Psi)) \cup (\neg m(\varphi) \cap \neg m(\Psi))$

$\varphi \Rightarrow \Psi$ 的准确度同样类似定义为

$\text{accuracy}(\varphi \Rightarrow \Psi)=\mid m(\varphi \wedge \Psi) \mid / \mid m(\varphi) \mid = \mid m(\varphi) \cap m(\Psi) \mid / \mid m(\varphi) \mid$

例如: 对于信息表 1, 我们有下列结果:

$m((a, \phi))=\{(p_3, p_1), (p_4, p_1), (p_5, p_1), (p_3, p_2), (p_4, p_2), (p_5, p_2), (p_1, p_2)\}$,

$m((0, \phi))=\{(p_1, p_4), (p_1, p_2), (p_1, p_3), (p_1, p_5), (p_4, p_2), (p_4, p_3), (p_4, p_5)\}$,

$m((a, \phi) \rightarrow (0, \phi))=U \times U - \{(p_3, p_2), (p_3, p_1), (p_4, p_1), (p_5, p_1), (p_5, p_2)\}$

3 序规则的挖掘

从一个排序后的信息表, 我们可构造一个二进制信息表. 在这个二进制信息表中, 考虑 $U \times U$ 上的所有对象对, 信息函数定义如下:

$I_a((x, y))=1, x \succ_{\{a\}} y; 0, x \leq_{\{a\}} y$

在这个二进制信息表中, 定义为所有包含于集合 At 的子集 A 定义等价关系 E_A:

$(x, y) E_A (x, y) \Leftrightarrow (\forall a \in A) I_a((x, y))=I_a((x, y))$, 全部序化后的属性划分所有对象对成两个不相交的集合命名为 Cl_0 和 Cl_1. 于是用 Rough 集的上下近似描述如下:

$\underline{apr}=Y\{[(x, y)]_A \mid [(x, y)]_A \subseteq Cl_i\}$

$\overline{apr} Y\{[(x, y)]_A \mid [(x, y)]_A \cap Cl_i \neq \phi\}$

其中[(x, y)]是通过等价关系 E 定义的包含(x, y)的等价类.

从$[(x, y)]_A \in \underline{apr}(Cl_i)$的每一个等价类, 可得出一定的排序规则: $Des([(x, y)]_A) \Rightarrow Dex(Cl_i)$

从$[(x, y)]_A \in \overline{apr}(Cl_i)$的每一个等价类, 可通过准确度(accuracy)和覆盖度(coverage)得出某个可能的排序规则: $Des([(x, y)]_A \Rightarrow Dex(Cl_i)$

其中: $accuracy = \frac{\mid m(Des([(x, y)]_A \wedge Des(Cl_i))] \mid}{\mid m(Des)[(x, y)]_A)) \mid}$

$$coverage=\frac{|m(Des([(x,y)]_A Des(Cl_i))|}{|m(Des(Cl_i))|}$$

由 Rough 理论得出最小约简 $A=\{b,c\}$，于是有

$\{[b=0,c=1]\}$的等价类：(1,2)(1,3)(1,4)(4,2)(4,3)　$o=1$

$\{[b=1,c=0]\}$的等价类：(1,5)(4,5)　$o=1$

$\{[b=0,c=0]\}$的等价类：(2,1)(2,3)(2,4)(3,1)(3,2)(3,4)(4,1)(5.1)　$o=0$

$\{[b=1,c=0]\}$的等价类：(2,5)(3,5)　$o=0$

$\{[b=0,c=1]\}$的等价类：(5,2)(5,3)(5,4)　$o=0$

$\underline{arp}(Cl_1)=\phi$,

$\underline{arp}(Cl_0)=\{[b=0,c=0]\}$,

$\overline{arp}(Cl_1)=\{[b=1,c=0],[b=0,c=1]\}$,

$\overline{arp}(Cl_0)=\{[b=1,c=0],[b=0,c=1],[b=0,c=0]\}$

产生排序规则

$R1:(b,\underline{\pi})\wedge(c,\underline{\pi})\Rightarrow(o,\underline{\pi})$　accuracy=1 coverage=0.615

$R2:(c,\phi)\Rightarrow(0,\phi)$　accuracy=0.625 coverage=0.714

$R3:(b,\phi)\Rightarrow(0,\phi)$　accuracy=0.5 coverage=0.286

4　启发式遗传算法用于序规则的挖掘

4.1　基于序的遗传算法理论基础

遗传算法是一个以适应度函数为目标函数，对种群中的各个个体施加遗传操作，实现群体结构重组，经过迭代处理，达到总体优化，最后求得逼近的最优解的过程．其主要有参数编码，初始种群设定，适应度函数设定，选择操作，交换操作，变异操作，算法控制参数设定等步骤组成．

对于交叉算子在此用 MOX（Modified Order Crossover）：首先，在第一个父染色体中随机选择一个基因，这个基因将是这个染色体适配区域的结束位置，同样这个位置也是第二个染色体适配区域的结束位置（适配区域的开始位置是从染色体的起始部分）．然后，保持适配区域不变，把其它位置的基因设置为在另一父染色体中出现的顺序．例如：

1 2 3 | 4 5 6　　MOX　　1 2 3 | 4 5 6

4 2 1 | 3 6 5　　⟶　　4 2 1 | 3 6 5

MOX 算子在某种程度上是严格传递性的：如子染色体中值 x 先于值 y，则在它的至少一个父染色体上可得到同样的顺序．

定义 $l=o(S)$为一个给定模式 S 的阶，假设第一个父染色体符合模式 S，那么有下面两种情况：至少有一个子染色体适合模式 S

(a) 所有来源于 S 的值在第一个父染色体的适配区域

(b) 所有来源于 S 的值不在第二个父染色体的适配区域

这两个事件是独立的，

对于事件(a)，其破坏概率$\binom{k}{l}/\binom{n}{l}$

对于事件(b)，其破坏概率$\binom{n-k}{l}/\binom{n}{l}$

于是 $P_c\leq\frac{1}{n+1}\sum_{k=0}^{n}P_1P_2=\frac{1}{n+1}\sum_{k=0}^{n}\frac{\binom{k}{l}}{\binom{n}{l}}\frac{\binom{n-k}{l}}{\binom{n}{l}}$

类似如遗传算法的模式理论，有[7−9]

$$E(N_{i+1})\geq N_t\frac{f}{f}(1-p_{cross}\cdot P_c-p_{mut}\cdot P_m)$$

其中 P_{cross}为交换概率，P_{mut}为变异概率，P_c 为通过交换的模式 S 的破坏概率，P_m 为通过变异的模式 S 的破坏概率．N_t 和 N_{t+1}分别为 t 和 $t+1$ 代适合模式 S 的个体数目．

4.2　算法过程

遗传算法部分：

(1)找出一个策略(用启发式)以给出一个近似结果

(2)用某个控制序列修改(参数化)这个策略，因此结果依赖于此序列．

(3)将这个控制序列译码成一个染色体．

(4)用遗传算法产生这个控制序列；处理这个被此序列控制的启发式算法，通过此启发式算法评估这个控制序列的适应度函数．

(5)进化出最好的控制序列，且这个序列产生最好的对象，输出这个对象．

启发式部分：

(1)设 R 是所有属性集，$(b_1\cdots b_n)=\tau(a_1\cdots$

a_n)是排序后的属性表，τ 是一个排列函数.

(2)*for* $i=1$ *to* n *do*(ⅲ，ⅳ)

(3) *set* $R \leftarrow R-b_i$

(4)若 R 不满足所定义的约简条件，转(ⅲ)

算法的结果将总是一个约简，结果依赖于属性的排序.为计算给定排列的适应度函数，我们必须运行一次这个算法以得到约简的长度，适应度函数定义如下：$F(\tau)=1/L_\tau$ 或 $F(\tau)=n-L_\tau+1$，或另一种优化 $F(\tau)=m-R_\tau+(n-L_\tau+1)/n$，其中 R_τ 是约简产生的数目.

4.3 遗传算子的选择

交叉算子：设染色体 $S_1=\{S_{11},\ldots,S_{1_n}\}$，$S_2=\{S_{21},\ldots,S_{2_n}\}$，选择一个随机整数 $0<r<n$，S_3，S_4 是(S_1，S_2 的后代，则：

$S_3=\{S_i \mid if \quad i\leq r, S_1=S_1\cup s_i, else \quad S_2=S_2\cup s_i\}$

$S_4=\{S_i \mid if \quad i\leq r, S_2=S_2\cup s_i, else \quad S_1=S_1\cup s_i\}$

变异算子：设染色体 $S_1=\{S_{11},\ldots,S_{1_n}\}$选择一个随机整数 $0<r<n$，S_3 是 S_1 的变异，则：

$S_3=\{s_i \mid if \quad i\neq r, s_i=s_{1i}, else \quad s_i=1-s_{1_t}\}$

选择算子：采用赌轮法进行选择.

5 结论

对象的排序是决策系统的的一个基本课题，且可以在智能信息系统中发挥显著的作用.挖掘排序规则在此公式化为发现属性值和全局排序对象的联系的过程，本文中，采用了基于 *Rough* 集理论的方法来阐述这个思想.依赖于不同的实际问题，我们可以用很多现成的算法来进行最小排序规则的挖掘.本文采用基于遗传算法的启发式算法，以解决陷于局部最优和算法效率问题.且遗传算法具有并行计算的特点，在接下来的分布环境下的研究提供了一个很好的方法.

参考文献：

[1] Pawlak，Z.RoughClassfication[J].InternationalJournalofMan－Machine Studies，20，469－483，1984.

[2] Pawlak，Z.Rough Sets，Theoretical Aspects of Reasoning about Data[J].Kluwer Academic Publishers，Dordrecth，1991.

[3] Ivo Düntsch，Günther Gediga.Rough Set Data Analysis：Rough Relation Algebras [J] Fundamentation Information. Vol. 21 (1997)321－331.

[4] Y.Y.Yao，Ying Sai.Mining Ordering Rules Using Rough Set Theory[J]. Bulletin of International Rough Set Society.2001，5：99－106

[5] 刘 清.Rough 集及 Rough 推理[M].北京：科学出版社，2003(第二版).

[6] 刘 清，黄兆华.G－逻辑及其归结推理[J].计算机学报.2004.24(7)：853－873.

[7] Jakub wroblewski.Theoretical Foundation of Order－based Genetic Algorithms[J]. Fundamenta Informaticae.vol.28(3,4)，pp：423－430.IOS Press，1996.

[8] Wroblewski.Genetic Algorithms in Decomposition and Classification Problems[J]. Fuzzy Sets and Systems，Vol.19(1997)，111－127.

[9] Wroblewski J.Finding minimal reducts using genetic algorithms.Proc. of the Second Annual Join Conference on Information Sciences，pp. 186－189，September 28－October 1，195，Wrightsville Beach，NC.

Order-based Decision Logic and Mining Rules Based on Heuristic Genetic Algorithm

LING Shi-yong，HUANG Zhao-hua

(Scholl of Information Eng.， East China Jiaotong University， ，Nanchang 330013，China)

Abstract：Many real world problems deal with ordering of objects instead of classifying objects，Examples of such problems and the consumer products produced by different manufactures，ranking of universities，and so on. In general，a overall information table of the object is given. In this paper ，we introduce the mining rules based on Rough Set theory，and finish the algorithm with heuristic genetic algorithm.

Key words：order-based decision logic；heuristic genetic algorithm；rough set