决策表的高效属性约简算法

邓春燕[1,2],吕跃进[2],李金海[2]
DENG Chun-yan[1],LV Yue-jin[2],LI Jin-hai[2]

1.广西河池学院 计算机与信息科学系,广西 宜州 546300
2.广西大学 数学与信息科学学院,南宁 530004
1.Department of Computer and Information Science,Hechi University,Yizhou,Guangxi 546300,China
2.Department of Mathematics and Information Science,Guangxi University,Nanning 530004,China
E-mail:dchy_hc@163.com

DENG Chun-yan,LV Yue-jin,LI Jin-hai.Efficient attribute reduction algorithm on decision table.Computer Engineering and Applications,2009,45(4):152-155.

Abstract: Rough set theory is a new mathematical tool to deal with vagueness and uncertainty.Authors first study the existed attribute reduction algorithms comparatively,based on which two relatively reasonable formulas measuring attribute significance are designed and the recursive calculating methods of them are provided as well.Taking the above formulas as heuristic information,a new and efficient attribute reduction algorithm based on decision table is developed.A real example and experimental results are used to illustrate that the algorithm proposed in this paper is more efficient than those existed algorihtms.
Key words: rough set;decision table;attribute reduction;time complexity

摘 要:粗糙集理论是一种新型的处理模糊和不确定知识的数学工具。对现有决策表的属性约简算法进行了比较研究,在此基础上设计了两个合理度量属性重要性的公式,并给出了该公式的递归计算方法,利用新公式作为启发式信息设计了一种新的基于决策表的高效属性约简算法。实例与实验表明,该约简算法在效率上较现有算法有显著的提高。
关键词:粗糙集;决策表;属性约简;时间复杂度
DOI:10.3778/j.issn.1002-8331.2009.04.043 **文章编号:**1002-8331(2009)04-0152-04 **文献标识码:**A **中图分类号:**TP18

1 引言

粗糙集理论是波兰学者 Pawlak 在 1982 年首次提出的[1],这是一种新的处理模糊和不确定知识的数学工具。经过二十多年的研究和发展,粗糙集理论已被成功地应用于人工智能、数据挖掘、模式识别与智能信息处理等领域[2-5]。属性约简是粗糙集理论中的核心内容之一,粗糙集的有效算法方面的研究主要集中在属性约简方面[6-12]。所谓属性约简,就是在保持信息系统分类能力不变的条件下,删除其中的冗余属性。

到目前为止,国内外学者分别提出了基于不可区分矩阵[6-7]、划分[8-9]、关系矩阵[10]、差别矩阵[11]、条件熵[12]、正区域[13]的决策系统属性约简方法[6-13]。本文对现有决策表的属性约简算法进行了比较研究,在此基础上设计了两个合理度量属性重要性的公式,并给出了该公式的递归计算方法,利用新公式作为启发式信息并结合文献[8]提出的快速计算划分的方法设计了一种新的基于决策表的高效属性约简算法。

2 粗糙集的基本概念

下面先简要介绍本文主要用到的粗糙集基本概念,详细内容请参见文献[1-4]。

定义 1 一个信息系统 S 可以表示为 $S=(U,A,V,f)$, 其中 $U=\{x_1,x_2,\cdots,x_n\}$表示对象的非空有限集合;A 表示属性的非空有限集合;$V=\bigcup_{a\in A}V_a$, 其中 V_a 是属性 a 的值域;$f:U\times A\rightarrow V$ 是一个信息函数, 它为每个对象的每个属性赋予了一个信息值,即对任意 $a\in A,x_i\in U,f(x_i,a)\in V_a$。

如果属性集 A 可以分为条件属性集 C 和决策属性集 D,即 $C\cup D=A,C\cap D=\varnothing$, 则该信息系统称为决策表或决策系统,其中 D 一般只含有一个属性。

A 的任意一个属性子集 P 决定一个二元不可区分关系 $IND(P)$:

$$IND(P)=\{(x_i,x_j)\in U\times U:\forall a\in P,f(x_i,a)=f(x_j,a)\}$$

显然 $IND(P)$是 U 上的一个等价关系,对象 x_i 在属性集 P 上的等价类定义为$[x_i]_P=\{x_j:(x_i,x_j)\in IND(P)\}$,那么等价类集$\{[x_i]_P:x_i\in U\}$构成 U 上的一个划分,记为 U/P。

性质 1 在信息系统 S 中,任意 $P\subseteq A$,有

基金项目:广西省教育厅科研项目(the Research Project of Department of Education of Guangxi Province,China under Grant No.2006[26])。
作者简介:邓春燕(1971-),女,讲师,主要研究领域为:数据挖掘,粗糙集理论与方法;吕跃进(1958-),男,教授,硕士生导师,主要研究领域为:数据挖掘,运筹与控制;李金海(1984-),男,硕士生,主要研究领域为:数据挖掘,粗糙集与概念格。
收稿日期:2008-08-20 **修回日期:**2008-10-30

$$IND(P)=\bigcap_{a\in P}IND(\{a\})$$

定义 2 在信息系统 S 中，任意 $R\subseteq A$，$X\subseteq U$，记 $U/R=\{R_1,R_2,\cdots,R_k\}$，则称

$$\underline{R}X=\{R_i:R_i\in U/R,R_i\subseteq X\}$$

$$\overline{R}X=\{R_i:R_i\in U/R,R_i\cap X\neq\emptyset\}$$

分别为 X 关于 R 的下近似集与上近似集。

定义 3 在信息系统 S 中，若 $P,Q\subseteq A$，记 $U/Q=\{Q_1,Q_2,\cdots,Q_t\}$，则 Q 的 P-正区域 $POS_P(Q)$ 定义为 $POS_P(Q)=\bigcup_{Q_i\in U/Q}\underline{P}Q_i$。

性质 2 在信息系统 S 中，若 $P,Q\subseteq A$，对于任意 $X\in U/P$，$X\subseteq POS_P(Q)$ 的充分必要条件是 $\exists Y\in U/Q$，使得 $X\subseteq Y$。

3　属性重要性度量

为了研究刻画属性重要性的新指标以及后面表述的方面，先给出如下定义 4。

定义 4 设决策表 $T=(U,C\cup D,V,f)$，$P\subseteq C$，$X\subseteq U$，记

$$ring_P(X)=\bigcup_{Y\in U/P\wedge Y\cap X\neq\emptyset\wedge|Y/D|\neq 1}Y$$

其中 $|X/D|$ 表示不可区分关系 $IND(D)$ 在 X 上形成的等价类的个数。

为了研究 $ring_P(X)$ 的一些性质，以便设计出合理刻画属性重要性的新指标，引入以下这个引理，详细的证明请参见文[13]。

引理 1[13] 设 $T=(U,C\cup D,V,f)$ 为一决策表，且 $R\subseteq P\subseteq C$，则 $U/P=\bigcup_{X\in U/R}X/(P-R)$。

性质 3 决策表，$R\subseteq P\subseteq C$，$X\subseteq Y\subseteq U$，则

（1）$ring_P(pos_P(D))=\emptyset$；

（2）$ring_P(X)\subseteq ring_P(Y)$；

（3）$ring_P(X)\subseteq ring_R(X)$。

证明

（1）对任意 $X\in U/P$，若 $X\cap pos_P(D)\neq\emptyset$，则 $X\subseteq pos_P(D)$。由性质 2 知，存在 $Y\in U/D$ 使得 $X\subseteq Y$，因此 $|X/D|=1$，所以 $ring_P(pos_P(D))=\emptyset$。

（2）显然。

（3）由引理 1 有

$$ring_P(X)=\bigcup_{Y\in U/P\wedge Y\cap X\neq\emptyset\wedge|Y/D|\neq 1}Y=\bigcup_{Y\in U/R}\{\bigcup_{Z\in Y/(P-R)\wedge Z\cap X\neq\emptyset\wedge|Z/D|\neq 1}Z\}=$$

$$\bigcup_{Y\in U/R\wedge Y\cap X\neq\emptyset\wedge|Y/D|\neq 1}\{\bigcup_{Z\in Y/(P-R)\wedge Z\cap X\neq\emptyset\wedge|Z/D|\neq 1}Z\}\subseteq$$

$$\bigcup_{Y\in U/R\wedge Y\cap X\neq\emptyset\wedge|Y/D|\neq 1}Y=ring_R(X)$$

推论 1 $|ring_R(pos_C(D))|-|ring_{R\cup\{a\}}(pos_C(D))|\geqslant 0$。

定理 1 设决策表 $T=(U,C\cup D,V,f)$，$P\subseteq C$，则 $pos_P(D)=pos_C(D)$ 当且仅当 $ring_P(pos_C(D))=\emptyset$。

证明 假设 $pos_P(D)=pos_C(D)$，对于任意 $X\in U/P$，若 $X\cap pos_C(D)\neq\emptyset$，则 $X\cap pos_P(D)\neq\emptyset$，因此 $X\subseteq pos_P(D)$，由性质 2 可知存在 $Y\in U/D$，使得 $X\subseteq Y$，即 $|X/D|=1$，所以 $ring_P(pos_C(D))=\emptyset$。

反之，若 $ring_P(pos_C(D))=\bigcup_{X\in U/P\wedge X\cap pos_C(D)\neq\emptyset\wedge|X/D|\neq 1}X=\emptyset$，因为 $pos_P(D)\subseteq pos_C(D)$，假设 $pos_P(D)\neq pos_C(D)$，则存在 $x\in U$ 使得 $x\in pos_C(D)$，且 $x\notin pos_P(D)$，故存在 $X\in U/P$，$x\in X$ 使得 $pos_C(D)\cap X\neq\emptyset$，且 $X\not\subset pos_P(D)$，由此可以推出 $X\cap pos_C(D)\neq\emptyset$，$|X/D|\neq 1$，这与 $ring_P(pos_C(D))=\emptyset$ 矛盾，所以 $pos_P(D)=pos_C(D)$。

定义 5 设决策表 $T=(U,C\cup D,V,f)$，属性 $a\in C$ 在 C 中相对于 D 的重要性定义为 $SIG(a,C,D)=|ring_{C-\{a\}}(pos_C(D))|-|ring_C(pos_C(D))|$。

由定义 5 和定理 1 易证以下定理。

定理 2 $a\in C$ 在 C 中相对于 D 是必要的当且仅当 $SIG(a,C,D)>0$。

推论 2 $core_C(D)=\{a\in C|SIG(a,C,D)>0\}$。

定义 6 设决策表 $T=(U,C\cup D,V,f)$，$R\subseteq C$，任意 $a\in C-R$ 在 R 中相对于 D 的重要性定义为：

$$SIG(a,R\cup\{a\},D)=|ring_R(pos_C(D))|-|ring_{R\cup\{a\}}(pos_C(D))|$$

定义 6 表明 $SIG(a,R\cup\{a\},D)$ 的值越大，属性 $a\in C-R$ 在 R 中相对于 D 越重要，文中将把 $SIG(a,R\cup\{a\},D)$ 作为寻找最小属性约简的启发式信息，以减少搜索空间。

由定义 6 和定理 1 易证以下定理。

定理 3 设决策表 $T=(U,C\cup D,V,f)$，$P\subseteq C$，若 $ring_P(pos_C(D))=\emptyset$ 且对任意 $a\in P$ 有 $SIG(a,P,D)>0$，则 P 为 C 相对于 D 的一个约简。

定理 3 是下一章给出属性约简算法的基础。

4　属性约简算法

为了使得所设计的属性约简算法具有更好的搜索效率，下面给出度量属性重要性公式的递归计算方法。

定理 4 设决策表 $T=(U,C\cup D,V,f)$，$R\subseteq C$，$a\in C-R$，则

$$SIG(a,R\cup\{a\},D)=$$

$$\left|\bigcup_{X\in ring_R(pos_C(D))/R}X\right|-\left|\bigcup_{X\in ring_R(pos_C(D))/R}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\right|$$

证明

$$|ring_{R\cup\{a\}}(pos_C(D))|=\left|\bigcup_{Y\in U/(R\cup\{a\})\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\right|=$$

$$\left|\bigcup_{X\in U/R}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\right|=$$

$$\left|\{\bigcup_{X\in U/R\wedge X\cap pos_C(D)\neq\emptyset\wedge|X/D|\neq 1}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\}\cup\right.$$

$$\{\bigcup_{X\in U/R\wedge X\cap pos_C(D)\neq\emptyset\wedge|X/D|=1}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\}\cup$$

$$\{\bigcup_{X\in U/R\wedge X\cap pos_C(D)=\emptyset\wedge|X/D|\neq 1}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\}\cup$$

$$\left.\{\bigcup_{X\in U/R\wedge X\cap pos_C(D)=\emptyset\wedge|X/D|=1}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\}\right|=$$

$$\left|\bigcup_{X\in U/R\wedge X\cap pos_C(D)\neq\emptyset\wedge|X/D|\neq 1}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\right|=$$

$$\left|\bigcup_{X\in ring_R(pos_C(D))/R}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\right|$$

所以

$$SIG(a,R\cup\{a\},D)=$$

$$\left|\bigcup_{X\in ring_R(pos_C(D))/R}X\right|-\left|\bigcup_{X\in ring_R(pos_C(D))/R}\{\bigcup_{Y\in X/\{a\}\wedge Y\cap pos_C(D)\neq\emptyset\wedge|Y/D|\neq 1}Y\}\right|$$

4.1　计算 $SIG(a,C,D)$ 和 $SIG(a,R\cup\{a\},D)$ 的算法

为了表述的方便，记文献[8]中计算划分的算法为**算法** 1。

算法 2 计算 $SIG(a,C,D)$。

输入：$a,C,D,pos_C(D)$

输出：（1）$SIG(a,C,D)$；（2）$ring_{C-\{a\}}(pos_C(D))/(C-\{a\})$

1.$SIG(a,C,D)=0$

$ring_{C-\{a\}}(pos_C(D))=\emptyset$

$ring_{C-\{a\}}(pos_C(D))/(C-\{a\})=\emptyset$

2.调用算法 1 计算 $U/(C-\{a\})$，记 $U/(C-\{a\})=\{R_1,R_2,\cdots,R_k\}$

3. For($i=1,i\leqslant k,i$++)

如果 $R_i\cap pos_C(D)\neq\emptyset$ 且 $|R_i/D|\neq 1$，执行

$ring_{C-\{a\}}(pos_C(D))=ring_{C-\{a\}}(pos_C(D))\cup R_i$;

$ring_{C-\{a\}}(pos_C(D))/(C-\{a\})=ring_{C-\{a\}}(pos_C(D))/(C-\{a\})\cup\{R_i\}$

4.$SIG(a,C,D)=|ring_{C-\{a\}}(pos_C(D))|$

算法 2 的时间复杂度为 $O((|C|+|D|)|U|)$。

算法 3 计算 $SIG(a,R\cup\{a\},D)$。

输入：$ring_R(pos_C(D))/R, a\in C-R, D, pos_C(D)$

输出：(1)$SIG(a,R\cup\{a\},D)$；(2)$ring_{R\cup\{a\}}(pos_C(D))/(R\cup\{a\})$

1.$SIG(a,R\cup\{a\},D)=0$

$ring_R(pos_C(D))=\emptyset$

$ring_{R\cup\{a\}}(pos_C(D))=\emptyset$

$ring_{R\cup\{a\}}(pos_C(D))/(R\cup\{a\})=\emptyset$

2.对于任意 $X\in ring_R(pos_C(D))/R$，执行

2.1 $ring_R(pos_C(D))=ring_R(pos_C(D))\cup X$

2.2 调用算法 1 计算 $X/\{a\}$，记 $X/\{a\}=\{Y_1,Y_2,\cdots,Y_t\}$

2.3 对任意 $Y_i\in X/\{a\}$

若 $Y_i\cap pos_C(D)\neq\emptyset$ 且 $|Y_i/D|\neq 1$，则执行

$ring_{R\cup\{a\}}(pos_C(D))=ring_{R\cup\{a\}}(pos_C(D))\cup Y_i$

$ring_{R\cup\{a\}}(pos_C(D))/(R\cup\{a\})=ring_{R\cup\{a\}}(pos_C(D))/(R\cup\{a\})\cup\{Y_i\}$

3. $SIG(a,R\cup\{a\},D)=|ring_R(pos_C(D))|-|ring_{R\cup\{a\}}(pos_C(D))|$

算法 3 的时间复杂度为 $O(|ring_R(pos_C(D))|)$。

4.2 属性约简算法

算法 4 计算条件属性集的约简。

输入：决策表 $T=(U,C\cup D,V,f)$，其中 $C=\{a_1,a_2,\cdots,a_k\}$

输出：条件属性集的一个约简

1.计算 $pos_C(D)$

2.令 $core_C(D)=\emptyset$

3.For($i=1,i\leqslant k,i$++)

调用算法 2 计算 $SIG(a_i,C,D)$。若 $SIG(a_i,C,D)>0$，则 $core_C(D)=core_C(D)\cup\{a_i\}$

4.若 $ring_{core_C(D)}(pos_C(D))=\emptyset$，则转步骤 8

5.令 $E=core_C(D)$

6.对属性集 $C-E$ 中的每个属性 a 调用算法 3 计算 $SIG(a,E\cup\{a\},D)$。并选择满足 $SIG(a',E\cup\{a'\},D)=\max\limits_{a\in(C-E)}\{SIG(a,E\cup\{a\},D)\}$的属性 a' 执行 $E=E\cup\{a'\}$（如果不只一个属性满足条件，可从中任选一个）。同时，令 $T=ring_{E\cup\{a'\}}(pos_C(D))/(E\cup\{a'\})$（为了方便下次调用算法 3），$SIG(a',E\cup\{a'\},D)$是调用算法 3 求 $SIG(a',E\cup\{a'\},D)$时得到的返回值

7.如果 $ring_E(pos_C(D))=\emptyset$，转步骤 8；否则返回步骤 6

8.结束

在算法 4 中，步骤 1 求 $pos_C(D)$，由文献[8]可知，执行步骤 1 的时间复杂度为 $O((|C|+|D|)|U|)$；步骤 3 需要计算$|C|$次 $SIG(a,C,D)$，由算法 2，计算每个 $SIG(a,C,D)$的时间复杂度为 $O((|C|+|D|)|U|)$，故执行步骤 3 的时间复杂度为 $O(|C|(|C|+|D|)|U|)$；步骤 6、7 在最坏的情况下，需要计算$|C|(|C|+1)/2$ 次 $SIG(a,E\cup\{a\},D)$，从 $E=core_C(D)$开始，步骤 6 每次都保存了调用算法 3 求 $SIG(a',E\cup\{a'\},D)$时得到的返回值 $ring_{E\cup\{a'\}}(pos_C(D))/(E\cup\{a'\})$（$a'$是使 $SIG(a,E\cup\{a\},D)$最大的属性），所以每次调用算法 3 求 $SIG(a,E\cup\{a\},D)$的时间复杂度为 $O(|ring_R(pos_C(D))|)$，因此执行步骤 6、7 的时间复杂度为 $O(|C|^2|ring_{core_C(D)}(pos_C(D))|)$；所以算法 4 的时间复杂度为 $O(|C|(|C|+|D|)|U|)$。一般地，决策属性 D 只有一个，所以算法 4 的时间复杂度为 $O(|C|^2|U|)$。

5 实例分析与实验

5.1 实例分析

下面以表 1 来说明算法 4 的可行性。

表 1 一个决策表

U	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	d
x_1	0	0	0	0	1	0	1	1	0	1
x_2	1	1	1	0	1	0	1	0	1	1
x_3	1	1	2	0	1	0	1	2	1	2
x_4	1	0	2	0	1	1	1	1	1	1
x_5	1	0	2	0	1	1	0	0	1	3
x_6	1	0	2	0	1	2	0	3	0	3
x_7	1	0	2	1	1	2	0	2	1	2
x_8	1	0	2	1	1	2	0	2	1	4
x_9	1	2	1	2	0	3	2	3	1	4
x_{10}	1	2	1	3	0	3	2	3	1	5

(1)计算 $pos_C(D)=\{x_1,x_2,x_3,x_4,x_5,x_6,x_9,x_{10}\}$；

(2)对于 C 中的每个属性 a_i 计算 $SIG(a_i,C,D)$；$SIG(a_4,C,D)=2>0$，$SIG(a_i,C,D)=0(i=1,2,3,5,6,7,8,9)$，所以 $core_C(D)=\{a_4\}$；

(3)$ring_{core_C(D)}(pos_C(D))=\{x_1,x_2,x_3,x_4,x_5,x_6\}$；

(4)$B=\{a_4\}$；

(5)对于 $C-B$ 中的每个属性 a_i 计算 $SIG(a_i,B\cup\{a_i\},D)$：

$SIG(a_1,B\cup\{a_1\},D)=1$ $SIG(a_2,B\cup\{a_2\},D)=0$

$SIG(a_3,B\cup\{a_3\},D)=2$ $SIG(a_5,B\cup\{a_5\},D)=0$

$SIG(a_6,B\cup\{a_6\},D)=1$ $SIG(a_7,B\cup\{a_7\},D)=2$

$SIG(a_8,B\cup\{a_8\},D)=4$ $SIG(a_9,B\cup\{a_9\},D)=0$

故 $a_8\in C-B$ 满足 $SIG(a_8,E\cup\{a_8\},D)=\max\limits_{a\in(C-E)}\{SIG(a,E\cup\{a\},D)\}$，执行 $B=B\cup\{a_8\}$；

(6)$ring_B(pos_C(D))=\{x_2,x_5\}\neq\emptyset$；

(7)对于 $C-B$ 中的每个属性 a_i 计算 $SIG(a_i,B\cup\{a_i\},D)$：

$SIG(a_1,B\cup\{a_1\},D)=0$ $SIG(a_2,B\cup\{a_2\},D)=2$

$SIG(a_3,B\cup\{a_3\},D)=2$ $SIG(a_6,B\cup\{a_6\},D)=2$

$SIG(a_7,B\cup\{a_7\},D)=2$ $SIG(a_9,B\cup\{a_9\},D)=0$

故 $a_2\in C-B$ 满足 $SIG(a_2,E\cup\{a_2\},D)=\max\limits_{a\in(C-E)}\{SIG(a,E\cup\{a\},D)\}$，执行 $B=B\cup\{a_2\}$；

(8)$ring_B(pos_C(D))=\emptyset$；

(9)结束，输出 $B=\{a_2,a_4,a_8\}$。

事实上，$B=\{a_2,a_4,a_8\}$恰好为决策表的最小约简。

5.2 实验

在这部分，选用表 1 中的决策表（记为决策表 1）与 UCI (University of California, Irvine)[14]中两个决策表 BUPA Liver Disorders（记为决策表 2）与 Chess End-Game（记为决策表 3）在个人电脑（Pentium® 4, CPU 1.60 GHz, 512 MB, Windows XP Professional）上进行实验，选用文献[12]提出的算法（记为算法 5）和本文中的算法 4 计算各决策表对应的属性约简，比较这两种算法的计算时间，实验结果如表 2 所示。

表 2　计算时间比较

决策表	对象个数	条件属性数	算法 5		算法 4	
			是否约简	运行时间/s	是否约简	运行时间/s
决策表 1	10	9	是	0.006	是	0.004
决策表 2	345	6	是	8.620	是	6.860
决策表 3	3 196	36	是	3 864.600	是	536.700

从表2 可以看出，当数据库比较大时，文中约简算法 4 在效率上较文[12]的约简算法 1 有显著的提高。原因是由于约简算法 4 采用递归式的搜索方式求属性约简，在搜索过程中，对象集不断迅速地收缩，从而使得搜索效率得到显著的提高。

6　结束语

本文对现有决策表的属性约简算法进行了比较研究，在此基础上设计了两个合理度量属性重要性的公式，并给出了该公式的递归计算方法，利用新公式作为启发式信息并结合文献[8]提出的快速计算划分的方法设计了一种新的基于决策表的高效属性约简算法，其时间复杂度为 $O(|C|^2|U|)$，通过实例和实验表明，该约简算法在效率上较现有算法有显著的提高。

参考文献：

[1] Pawlak Z.Rough sets[J].International Journal of Computer and Information Science，1982，11（5）：341-356.

[2] 张文修.粗糙集理论与方法[M].北京：科学出版社，2001：3-21.

[3] 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.

[4] 丁敬国，胡贤磊，焦景民，等.基于粗糙集的关联规则数据挖掘在层流冷却中的应用[J].东北大学学报：自然科学版，2007，28（11）：1583-1586.

[5] 张东波，王耀南，段峰，等.粗糙集在智能空瓶检测感兴趣区域提取中的应用[J].系统仿真学报，2007，19（5）：1021-1025.

[6] Skowron A，Rauszer C.The discernibility matrics and functions in information system[M]//Slowinski R.Intelligent Decision Support Handbook of Applications and Advances of the Rough Sets Theory.Dordrecht：Kluwer Academic Publishers，1992：331-362.

[7] Wang Jue，Wang Ju.Reduction algorithms based on discernibility matrix：The ordered attributes method[J].Journal of Computer & Science，2001，16（6）：489-504.

[8] 李金海，吕跃进.决策系统的快速属性约简算法[J].电子科技大学学报，2007，36（6）：1237-1240.

[9] 吕跃进，李金海.一种基于划分的信息系统属性约简算法[J].计算机工程与应用，2008，44（22）：149-151，174.

[10] 李金海，吕跃进.一种基于关系矩阵的信息系统属性约简算法[J].计算机工程与应用，2008，44（9）：147-149.

[11] 高学东，丁军.基于简化差别矩阵的属性约简算法[J].系统工程理论与实践，2006（6）：101-107.

[12] 苗夺谦，胡桂荣.知识约简的一种启发式算法[J].计算机研究与发展，1999，36（6）：681-684.

[13] 刘少辉，盛秋戬，吴斌，等.Rough 集高效算法研究[J].计算机学报，2003，26（5）：524-529.

[14] Bay S D.UCI KDD Archive[EB/OL].[1999].http：//kdd.ics.uci.edu.

（上接 22 页）

机、无线通信及控制技术，实现移动用户图形化的望远镜控制，自动化的消息通知，能够对 LAMOST 实现稳定的实时的控制，是对 LAMOST 现有控制系统的有益补充。随着移动通信技术、网络及 Java 技术的发展，3G 网络的运行，移动 LAMOST 网络控制系统一定会为 LAMOST 的运行带来越来越重要的作用。

参考文献：

[1] 中国科学院.大天区面积多目标光纤光谱天文望远镜（LAMOST）项目可行性研究报告[R].1997.

[2] 徐灵哲，徐欣圻.LAMOST 远程无线监控系统研究[J].天文学报，2008，49（1）.

[3] 帅小应，廉东本.基于 J2ME 的移动位置服务 3G 手机应用平台的设计[J].计算机应用，2004，24（11）：146-148.

[4] 王建光，段富.一种 UML 模型到 XML 模型的转方法[J].计算机技术与发展，2007，17（7）：123-126.

[5] LAMOST 关于 TCS 和力、位移促动器接口.2007.

[6] 王明文，朱清新.基于 UML 的 XML Schema 设计[J].电子科技大学学报，2006，35（3）：389-391.

[7] Routledge N，Bird L，Goodchild A.UML and XML schema[J].Australian Computer Science Communications，2002，24（2）：157-166.

[8] 万长胜.LAMOST 观测控制系统的命令解析[J].中国科学技术大学学报，2005，35（4）：480-485.

[9] Muchow J.Implementing push technology with J2ME and MIDP[EB/OL].[2003-06].http：//www-128.ibm.com/developerworks/edu/wi-dw-wi-midpreg-i.html.

（上接 139 页）

[3] Fumagalli M，Lancini R，Tubaro S.A novel error-concealment algorithm for an unbalanced multiple description coding architecture[C]//Proceedings of International Packet Video Workshop，US，2004，12.

[4] Fumagalli M，Lancini R，Tubaro S.A sequence-based error-concealment algorithm for an unbalanced multiple description video coding system[J].Signal Processing：Image Communication，2006：829-849.

[5] Li B，Huang F，Sun L F，et al.An unbalanced multiple description coding scheme for video transmission over wireless ad hoc networks[C]//IEEE International Conference on Multimedia and Expo，IEEE，2006，7：1409-1421.

[6] Vilei A，Convertino G，Oliva S，et al.A novel unbalanced multiple description scheme for video transmission over WLAN[C]//Proceedings of the 3rd ACM International Workshop on Wireless Mobile Applications and Services on WLAN Hotspots.Germany：ACM，2005，9：39-45.

[7] JVT-G050 H.264 standard[S].Thailand，Pattaya，2003.

[8] Li Q Ch，Xing X，Xin F.A visual attention model for adapting images on small displays[J].ACM Multimedia Systems Journal，2003，9（4）：353-364.

[9] Wang Y，Fan X，Li H Q.An attention based spatial adaptation scheme for H.264 videos on mobiles[C]//IEEE the International Multi-Media Modeling Conference Proceedings，Jan 2006.

[10] Laurent I，Christof K，Ernst N.A model of saliency-based visual attention for rapid scene analysis[J].IEEE Transactions on Pattern Analysis and Machine Intelligence，1998，20（11）：1254-1259.

[11] Rowley H A，Baluja S，Kanada T.Neural network-based face detection[C]//IEEE Computer Society Conference on Computer Vision and Pattern Recognition，18-20 June，1996：203-208.