

基于数据库技术的属性约简算法

王熊彬¹ 郑雪峰¹ 徐章艳²

(北京科技大学信息工程学院 北京 100083)¹ (广西师范大学计算机系 桂林 541004)²

摘 要 为弥补传统属性约简模型的不足,一些学者提出了基于系统熵的属性约简和基于数据库模型的属性约简定义。基于数据库模型的属性约简的主要优点是能结合高效的数据库技术,使得相应的属性约简算法效率较高。为能利用高效的数据库技术来设计基于系统熵的属性约简算法,证明了两种属性约简的等价性。然后利用数据库中的相关操作设计了一个基于系统熵的高效属性约简算法,并用实例说明了新算法。

关键词 粗糙集,属性约简,系统熵,数据库模型,算法复杂度

中图法分类号 TP18 **文献标识码** A

Algorithm of Attribute Reduction Based on Database Technology

WANG Xiong-bin¹ ZHENG Xue-feng¹ XU Zhang-yan²

(School of Information Engineering, University of Science and Technology Beijing, Beijing 100083, China)¹

(Department of Computer, Guangxi Normal University, Guilin 541004, China)²

Abstract For improving the shortcoming of the model of traditional attribute reductions, some researchers proposed the definitions of attribute reduction based on the system entropy and database model. The main merit of attribute reduction based on databased model is that it can use the efficient database technology to design the algorithm of attribute reduction. So the corresponding algorithm of attribute reduction is efficient. For using the efficient database technology to design the algorithm of attribute reduction based on the system entropy, it was proved that the attribute reduction based on the system entropy are equal to that based on database model. Then the algorithm of attribute reduction based on system entropy with some operations of the database technology was provided. And an example is used to illustrate the new algorithm.

Keywords Rough set, Attribute reduction, System entropy, Database model, Algorithm complexity*

1 引言

粗糙集理论^[1,2]是一种新的处理不精确、不完全与不相容知识的数学理论,已被广泛应用于人工智能、模式识别、数据挖掘和智能决策等领域。在粗糙集理论中,属性约简是重要研究内容之一。目前,由于分析问题的出发点不同,已出现了各种不同的属性约简定义。例如:基于正区域的属性约简^[1,2]、上近似约简^[3-5]、可能性约简^[3-5]等等。在这些模型中,通常用严格包含去定义下近似集,这样下近似集就不能用等价关系很好地处理具有统计特征或噪声的数据。这一缺陷限制了传统粗糙集理论在带有噪声或脏数据领域中的广泛应用^[6]。为了弥补这一缺陷,一些新的模型 GRS(一般粗糙集模型)和 VPRS(变精度粗糙集模型)被一些学者提出^[6]。这些属性约简模型的另一个不足是:计算属性约简的算法的效率不高,算法的可扩展性也较差^[6]。为了弥补上述不足,文献^[6]提出了一种基于数据库模型的属性约简定义,该模型的主要优点是结合了高效数据库中相关的运算进行属性约简,且能处理带有噪声的数据集。由于结合了高效的数据库技术,

从而使得该模型在处理大数据集时成为可能。

另外,一些学者提出的基于 Skowron 差别矩阵的属性约简^[7,8]和基于信息熵的属性约简^[9]也能处理一些带噪声的数据,但是这些模型都是从条件属性出发,根据条件属性的分类能力进行条件属性约简,它们的出发点是一样的,只是采用的约简标准有所不同。最近有一些学者认为只关心条件属性的分类能力是不够的,应该加上决策属性的分类能力,因此提出了一种新的属性约简定义:基于系统熵的属性约简定义^[10],并给出了相应的属性约简算法。但该算法也像传统算法一样,没有利用高效的数据库技术,因此算法的效率不高。其时间复杂度为 $O(|C|^3|U|\log|U|)$ 。为了能利用高效的数据库中的相关操作,本文证明了基于数据库模型的属性约简与基于系统熵的属性约简模型是等价的。因此,也可以用高效的数据库技术来设计基于系统熵的属性约简算法。

2 两种不同的属性约简模型

为讨论方便,给出了基于系统熵模型的属性约简和基于数据库模型的属性约简的相关定义。

到稿日期:2008-10-28 返修日期:2009-01-11 本文受国家自然科学基金重点项目(69835001),广西教育厅基金(2006026)资助。

王熊彬(1971—),男,博士研究生,研究方向为粗糙集理论与应用、决策树、数据挖掘及模式识别, E-mail: bearwxb@163.com; 郑雪峰(1954—),男,教授,博士生导师,研究方向为数据挖掘、入侵检测; 徐章艳(1972—),男,博士,副教授,研究方向为粗糙集理论与应用、数据挖掘。

定义 1^[1,2] 五元组 $S=(U, C, D, V, f)$ 是一个决策表, 其中 $U=\{x_1, x_2, \dots, x_n\}$ 表示对象的非空有限集, 称为论域; $C=\{c_1, c_2, \dots, c_r\}$ 表示条件属性的非空有限集; D 表示决策属性的非空有限集且 $C \cap D = \emptyset$; $V = \bigcup_{a \in C \cup D} V_a$, V_a 是属性 a 的值域, $f: U \times C \cup D \rightarrow V$ 是一个信息函数, 它对一个对象的每一个属性赋予一个信息值, 即 $\forall a \in C \cup D, x \in U$, 有 $f(x, a) \in V_a$; 每一个属性子集 $P \subseteq (C \cup D)$ 决定了一个二元不可区分关系 $IND(P)$:

$$IND(P) = \{(x, y) \in U \times U \mid \forall a \in P, f(x, a) = f(y, a)\}$$

关系 $IND(P)$ 构成了 U 的一个划分, 用 $U/IND(P)$ 表示, 简记为 U/P , U/P 中的任何元素 $[x]_P = \{y \mid \forall a \in P, f(x, a) = f(y, a)\}$ 称为等价类。

定义 2^[10] 设五元组 $S=(U, C, D, V, f)$ 是一个决策表, $\forall P \subseteq C$ ($U/P = \{P_1, P_2, \dots, P_t\}$), $U/D = \{D_1, D_2, \dots, D_k\}$), 决策表 S 的系统熵 $H_S(D|P)$ 定义为:

$$H_S(D|P) = - \sum_{i=1}^t \frac{|D_i|}{|U|} \sum_{j=1}^k \frac{|P_i \cap D_j|}{|D_j|} \log \frac{|P_i \cap D_j|}{|D_j|}.$$

引理 1^[10] 设五元组 $S=(U, C, D, V, f)$ 是一个决策表, 对 $\forall B \subseteq C$, 则有 $H_S(D|C) \geq H_S(D|B)$ 。

定义 3^[10] 设五元组 $S=(U, C, D, V, f)$ 是一个决策表, 对 $\forall B \subseteq C$, 若 $H_S(D|C) = H_S(D|B)$, 并对 $\forall b \in B$, 均有 $H_S(D|(B-b)) < H_S(D|C)$, 则称 B 为条件属性集 C 关于决策表属性集 D 的一个基于系统熵的属性约简, 简称 B 为决策表的一个基于系统熵的属性约简。记 $SERED_b(C)$ 为决策表的所有基于系统熵的属性约简集。

定义 4^[6] 设 $S=(U, C, D, V, f)$ 为一个决策表, 对 $\forall B \subseteq C$, 若 $U/(B \cup D) = U/(C \cup D)$, 且对 $\forall b \in B$, 均有 $(B - \{b\}) \cup D \neq U/(C \cup D)$, 则称 B 为 C 的一个关于数据库模型的属性约简, 简称 B 为基于数据库模型的属性约简。令 $DSRED_b(C)$ 为所有基于数据库模型的属性约简的集合。核 $Core(C) = \bigcap_{B \in DSRED_b(C)} B$ 。

定义 5 在决策表 $S=(U, C, D, V, f)$ 中, 任意属性子集 $P, Q \subseteq (C \cup D)$, 记 $U/P = \{P_1, P_2, \dots, P_t\}$, $U/Q = \{Q_1, Q_2, \dots, Q_s\}$, 若 $\forall P_i \in U/P \Rightarrow \exists Q_j \in U/Q$ 使得 $P_i \subseteq Q_j$, 则称 U/P 为 U/Q 的加细, 记为 $U/P < U/Q$ 。若 $U/P < U/Q$ 且 $U/Q < U/P$, 则称 $U/P = U/Q$ 。

引理 2^[5] 在决策表 $S=(U, C, D, V, f)$ 中, $\forall Q \subseteq P \subseteq (C \cup D)$, 则有 $U/P < U/Q$ 。

在文献[6]中给出了利用高效的数据库技术中的相关操作设计的基于数据库属性约简算法, 这样使得属性约简算法的效率大大提高。为了提高基于系统熵的属性约简算法的效率, 下面证明上述两种不同的属性约简模型是等价的, 这样, 也可以利用高效的数据库技术设计基于系统熵的属性约简算法。

3 两种属性约简模型的等价性

定理 1 设决策表 $S=(U, C, D, V, f)$, 对 $\forall B \subseteq C$, 若 $U/(B \cup D) = U/(C \cup D)$, 则有 $H_S(D|B) = H_S(D|C)$ 。

证明: 设 $U/B = \{B_1, B_2, \dots, B_p\}$, $U/C = \{C_1, C_2, \dots, C_q\}$, $U/D = \{D_1, D_2, \dots, D_k\}$ 。由于 $B \subseteq C$, 根据引理 2 有: $U/(B \cup D) < U/(C \cup D)$ 。又因 $U/(B \cup D) = U/(C \cup D)$, 故对 $\forall D_i \in U/D$, 均有 $D_i/B = D_i/C$ 。也就是说, 对任意 $X \in D_i/B$, 一定

存在 $Y \in D_i/C$ 使得 $X=Y$, 且对任意 $X \in D_i/C$ 一定存在 $X \in D_i/B$ 使得 $X=Y$ 。故有:

$$\sum_{j=1}^k \frac{|B_j \cap D_i|}{|D_i|} \log \frac{|B_j \cap D_i|}{|D_i|} = \sum_{j=1}^q \frac{|C_j \cap D_i|}{|D_i|} \log \frac{|C_j \cap D_i|}{|D_i|}$$

另一方面, 显然有:

$$\begin{aligned} & - \sum_{i=1}^k \frac{|D_i|}{|U|} \sum_{j=1}^p \frac{|B_j \cap D_i|}{|D_i|} \log \frac{|B_j \cap D_i|}{|D_i|} \\ & = - \sum_{i=1}^k \frac{|D_i|}{|U|} \sum_{j=1}^q \frac{|C_j \cap D_i|}{|D_i|} \log \frac{|C_j \cap D_i|}{|D_i|} \end{aligned}$$

即有 $H_S(D|B) = H_S(D|C)$, 故命题成立。

定理 2 设在决策表 $S=(U, C, D, V, f)$ 中, 对 $\forall B \subseteq C$, 若 $H_S(D|B) = H_S(D|C)$, 则有 $U/(B \cup D) = U/(C \cup D)$ 。

证明: 设 $U/B = \{B_1, B_2, \dots, B_p\}$, $U/C = \{C_1, C_2, \dots, C_q\}$, $U/D = \{D_1, D_2, \dots, D_k\}$ 。由于 $B \subseteq C$, 根据引理 2 有 $U/(B \cup D) < U/(C \cup D)$ 。假设 $U/(B \cup D) \neq U/(C \cup D)$, 则一定存在 $D_i \in U/D$ 使得 $D_i/B \neq D_i/C$ 。由于 $B \subseteq C$, 根据引理 2 有, 则一定存在 $X \in D_i/B$ 使得 $X = Y_1 \cup Y_2 \cup \dots \cup Y_h$ (这里 $h \geq 2$ 且 $Y_1, Y_2, \dots, Y_h \in D_i/C$)。设 $X = \frac{|B_{j_0} \cap D_i|}{|D_i|}$, $Y_r = \frac{|C_{j_r} \cap D_i|}{|D_i|}$ ($r=1, 2, \dots, h$), 则有:

$$\begin{aligned} & - \frac{|B_j \cap D_i|}{|D_i|} \log \frac{|B_j \cap D_i|}{|D_i|} \\ & = - \sum_{r=1}^h \frac{|C_{j_r} \cap D_i|}{|D_i|} \log \sum_{r=1}^h \frac{|C_{j_r} \cap D_i|}{|D_i|} \end{aligned}$$

另一方面有:

$$\begin{aligned} & - \log \sum_{r=1}^h \frac{|C_{j_r} \cap D_i|}{|D_i|} < \\ & - \log \frac{|C_{j_r} \cap D_i|}{|D_i|} \quad (\text{因为 } r > 1) \end{aligned}$$

因此有

$$\begin{aligned} & - \frac{|B_j \cap D_i|}{|D_i|} \log \frac{|B_j \cap D_i|}{|D_i|} < \\ & - \sum_{r=1}^h \frac{|C_{j_r} \cap D_i|}{|D_i|} \log \frac{|C_{j_r} \cap D_i|}{|D_i|} \end{aligned}$$

从而有:

$$\begin{aligned} & - \sum_{j=1}^p \frac{|B_j \cap D_i|}{|D_i|} \log \frac{|B_j \cap D_i|}{|D_i|} < \\ & - \sum_{j=1}^q \frac{|C_j \cap D_i|}{|D_i|} \log \frac{|C_j \cap D_i|}{|D_i|} \end{aligned}$$

根据引理 1 有:

$$\begin{aligned} & - \sum_{i=1}^k \frac{|D_i|}{|U|} \sum_{j=1}^p \frac{|B_j \cap D_i|}{|D_i|} \log \frac{|B_j \cap D_i|}{|D_i|} < \\ & - \sum_{i=1}^k \frac{|D_i|}{|U|} \sum_{j=1}^q \frac{|C_j \cap D_i|}{|D_i|} \log \frac{|C_j \cap D_i|}{|D_i|} \end{aligned}$$

即

$$H_S(D|B) < H_S(D|C).$$

这与条件矛盾, 故假设不成立, 从而命题成立。即有 $U/(B \cup D) = U/(C \cup D)$ 。

综上所述, 命题成立。

定理 3 设 $S=(U, C, D, V, f)$ 为决策表, 对 $\forall B \subseteq C$, 若 $U/(B \cup D) \neq U/(C \cup D)$, 则有 $H_S(D|B) \neq H_S(D|C)$ 。

证明: 假设 $H_S(D|B) \neq H_S(D|C)$ 不成立, 则有 $H_S(D|B) = H_S(D|C)$ 。由定理 2 可知有 $U/(B \cup D) = U/(C \cup D)$ 。这与条件矛盾, 故假设不成立, 从而命题成立。

定理 4 设 $S=(U, C, D, V, f)$ 为决策表, 对 $\forall B \subseteq C$, 若

$H_S(D|B) \neq H_S(D|C)$, 则有 $U/(B \cup D) \neq U/(C \cup D)$ 。

证明: 假设 $U/(B \cup D) \neq U/(C \cup D)$ 不成立, 则有 $U/(B \cup D) = U/(C \cup D)$ 。根据定理 1 可知有: $H_S(D|B) = H_S(D|C)$ 。这与条件矛盾, 故假设不成立, 从而命题成立。

定理 5 设 $S = (U, C, D, V, f)$ 为决策表, 对 $\forall B \in DSRED_b(C)$, 则有 $B \in SERED_b(C)$ 。

证明: 由于 $B \in SERED_b(C)$, 根据基于数据库模型的属性约简定义可知有: (1) $U/(B \cup D) = U/(C \cup D)$; (2) 对 $\forall b \in B$ 有 $U/((B - \{b\}) \cup D) \neq U/(C \cup D)$ 。

当 (1) 成立时, 根据定理 1 可知有: $H_S(D|B) = H_S(D|C)$ 。当 (2) 成立时, 根据定理 3 有: $H_S(D|(B - \{b\})) \neq H_S(D|C)$ 。根据系统熵的属性约简定义可知有: $B \in SERED_b(C)$ 。

定理 6 设 $S = (U, C, D, V, f)$ 为一决策表, 对 $B \in SERED_b(C)$, 则有: $B \in DSRED_b(C)$ 。

证明: 由于 $B \in SERED_b(C)$ 根据系统熵的属性约简定义可知有: (1) $H_S(D|B) = H_S(D|C)$; (2) 对 $\forall b \in B$ 有 $H_S(D|(B - \{b\})) \neq H_S(D|C)$ 。

当 (1) 成立时, 根据定理 2 有: $U/(B \cup D) = U/(C \cup D)$ 。当 (2) 成立时, 根据定理 4 有: $U/((B - \{b\}) \cup D) \neq U/(C \cup D)$ 。根据基于数据库模型的属性约简定义可知有: $B \in DSRED_b(C)$ 。

定理 7 设 $S = (U, C, D, V, f)$ 为一决策表, 则有 $DSRED_b(C) = SERED_b(C)$ 。

证明: 根据定理 5 和 6 易知命题成立。

定理 7 表明: 基于系统熵的属性约简定义与基于数据库属性约简定义是等价的, 因此可以用设计其中一种的属性约简算法的方法去设计另外一种属性约简算法。它为设计高效的属性约简算法提供了理论基础。

为了更好、更直观地说明两种不同属性约简模型的等价性, 用如下的决策表说明之。

例 1 决策表 1 (表 1)

在决策表 1 中, 有 12 对象元素, 条件属性集为 $C = \{a, b, c, d, e\}$, 决策属性集为 D 。根据 $U/(C \cup D)$ 的定义可得: $U/(C \cup D) = \{\{1, 3\}, \{2\}, \{4, 5\}, \{6, 7\}, \{8\}, \{9\}, \{10, 11\}, \{12\}\}$ 。同理可得: $U/C = \{\{1, 2, 3\}, \{4, 5\}, \{6, 7\}, \{8, 9\}, \{10, 11\}, \{12\}\}$; $U/D = \{\{1, 3, 4, 5, 8\}, \{2, 6, 7, 9, 10, 11, 12\}\}$ 。故有:

$$\begin{aligned} H_S(D|C) &= -\frac{5}{12} \left(\frac{2}{5} \log \frac{2}{5} + \frac{2}{5} \log \frac{2}{5} + \frac{1}{5} \log \frac{1}{5} \right) - \\ &\quad \frac{7}{12} \left(\frac{1}{7} \log \frac{1}{7} + \frac{2}{7} \log \frac{2}{7} + \frac{1}{7} \log \frac{1}{7} + \frac{2}{7} \log \frac{2}{7} + \frac{1}{7} \log \frac{1}{7} \right) \\ &= -\frac{5}{12} \left(\frac{4}{5} \log \frac{2}{5} + \frac{1}{5} \log \frac{1}{5} \right) - \frac{7}{12} \left(\frac{3}{7} \log \frac{1}{7} + \frac{4}{7} \log \frac{2}{7} \right) \end{aligned}$$

表 1 决策表 1

U	a	b	c	d	e	D
1	1	0	2	1	0	0
2	1	0	2	1	0	1
3	1	0	2	1	0	0
4	0	1	1	2	1	0
5	0	1	1	2	1	0

6	2	2	0	0	1	1
7	2	2	0	0	1	1
8	0	1	2	0	2	0
9	0	1	2	0	2	1
10	2	0	2	2	0	1
11	2	0	2	2	0	1
12	1	1	0	2	2	1

另一方面, 易知:

$$\begin{aligned} U/(C \cup D) &= U/(\{a, c\} \cup D) = U/(\{a, d\} \cup D) = U/(\{a, e\} \cup D) \\ &= U/(\{b, d\} \cup D) = U/(\{c, d\} \cup D) = U/(\{d, e\} \cup D) \\ U/(C \cup D) &\neq U/(\{a\} \cup D) \neq U/(\{c\} \cup D) = U/(\{d\} \cup D) \\ &= U/(\{b\} \cup D) \neq U/(\{e\} \cup D) \end{aligned}$$

因此有: $DSRED_b(C) = \{\{ac\}, \{ad\}, \{ae\}, \{bd\}, \{cd\}, \{de\}\}$ 。

另一方面, 有:

$$\begin{aligned} H_S(D|C) &= H_S(D|(\{a, c\})) = H_S(D|(\{a, d\})) \\ &= H_S(D|(\{a, e\})) \\ &= H_S(D|(\{b, d\})) = H_S(D|(\{c, d\})) \\ &= H_S(D|(\{d, e\})) \\ H_S(D|C) &\neq H_S(D|(\{a\})) \neq H_S(D|(\{b\})) \neq H_S(D|(\{c\})) \\ &\neq H_S(D|(\{d\})) \neq H_S(D|(\{e\})) \end{aligned}$$

因此有: $SERED_b(C) = \{\{ac\}, \{ad\}, \{ae\}, \{bd\}, \{cd\}, \{de\}\}$ 。

根据上述计算可知: $SERED_b(C) = DSRED_b(C)$ 。

4 基于数据库技术的属性约简算法

在上一节中, 证明了两种属性约简等价性, 本节利用数据库技术设计属性约简算法。

算法 1 计算决策表中基于系统熵的核

输入: 决策表 $S = (U, C, D, V, f)$;

输出: 决策表的核 $Core(C)$;

1) 令 $Core(C) = \emptyset$;

2) 对任意 $a \in C$, 若 $|U/(C \cup D)| \neq |U/(C - \{a\} \cup D)|$, 则 $Core(C) = Core(C) \cup \{a\}$;

其中第 2) 步可以用数据库中的选择和计算记录两种运算。故若不考虑数据库基本运算的时间复杂度, 则该项算法的时间复杂度为 $O(|C||U|)$ 。

SQL 中的运算实现的 $|U/B|$ 如下:

SELECT DISTINCT COUNT(*) FROM (SELECT B FROM T)。

算法 2 用数据库技术实现的基于系统熵的属性约简算法

输入: 决策表 $S = (U, C, D, V, f)$;

输出: 决策表的核 $reduc(C)$;

1) 用算法 1 求出决策表 S 的核 $Core(C)$; $reduc(C) = Core(C)$;

2) 令 $B = C - reduc(C)$;

3) 对 $\forall a \in B$, 计算 $merit(a) = \max_{a \in B} \left\{ 1 - \frac{|U/((reduc(C)) \cup D)|}{|U/((reduc(C) \cup \{a\}) \cup D)|} \right\}$, 若

这样的属性 a 不只一个, 则任选一个;

4) 若 $merit(a) \neq 1$, 则 $reduc(C) = reduc(C) \cup \{a\}$, 并转第 2) 步; 否则算法终止, 并输出属性约简 $reduc(C)$ 。

算法的复杂度分析: 在不考虑数据库的基本运算所花的时间时, 算法的时间复杂度为 $O(|C|^2|U|)$, 空间复杂度为 $O(|U|)$ 。为了更好地说明新算法, 用决策表 1 说明之。

由算法 2 的第 1) 步得到, 计算 $|U/(C \cup D)|$ 的 SQL 操作语言如下:

SELECT DISTINCT COUNT(*) FROM DECISION;
其中 DECISION 就表示决策表 1, 它的条件属性和决策属性分别为: $\{a, b, c, d, e\}$ 和 $\{D\}$ 。

故 $|U/(C \cup D)| = 8$ 。同理可得: $|U/(C - \{a\} \cup D)| = 8$; $|U/(C - \{b\} \cup D)| = 8$; $|U/(C - \{c\} \cup D)| = 8$; $|U/(C - \{d\} \cup D)| = 8$; $|U/(C - \{e\} \cup D)| = 8$ 。故 $Core(C) = \emptyset$; $reduc(C) = \emptyset$ 。

由算法的第 2) 步得: $B = \{a, b, c, d, e\}$ 。

由算法 2 的第 3) 步得:

$$\begin{aligned} merit(a) &= \left\{ 1 - \frac{|U/((reduc(C)) \cup D)|}{|U/((reduc(C) \cup \{a\}) \cup D)|} \right\} \\ &= 1 - \frac{|U/D|}{|U/(\{a\} \cup D)|} = 1 - \frac{2}{5} = \frac{3}{5} \end{aligned}$$

同理可得: $merit(b) = 3/5$; $merit(c) = 2/4$; $merit(d) = 4/6$; $merit(e) = 4/6$ 。故值最大的有两个, 即 $\{d, e\}$ 。选择 $\{d\}$ 。

由算法 2 的第 4) 步得:

$reduc(C) = \{d\}$, 算法转第 2) 步, 得到 $B = \{a, b, c, e\}$ 。

由算法 2 的第 3) 步得:

$$\begin{aligned} merit(a) &= \left\{ 1 - \frac{|U/((reduc(C)) \cup D)|}{|U/((reduc(C) \cup \{a\}) \cup D)|} \right\} \\ &= 1 - \frac{|U/(\{d\} \cup D)|}{|U/(\{d, a\} \cup D)|} = 1 - \frac{6}{8} = \frac{2}{8} \end{aligned}$$

同理可得: $merit(b) = 2/8$; $merit(c) = 2/8$; $merit(e) = 2/8$ 。故值最大的有 4 个, 即 $\{a, b, c, e\}$ 。选择 $\{a\}$ 。

由算法 2 的第 4) 步得:

$reduc(C) = \{d, a\}$, 算法转第 2) 步, 得到 $B = \{b, c, e\}$ 。

由算法 2 的第 3) 步得:

$$\begin{aligned} merit(a) &= \left\{ 1 - \frac{|U/((reduc(C)) \cup D)|}{|U/((reduc(C) \cup \{a\}) \cup D)|} \right\} \\ &= 1 - \frac{|U/(\{d, a\} \cup D)|}{|U/(\{d, a, b\} \cup D)|} = 1 - \frac{8}{8} = 1 \end{aligned}$$

同理可得: $merit(c) = 1$; $merit(e) = 1$ 。故值最大的有 3 个, 即 $\{b, c, e\}$ 。但它们的值均为 1, 故算法终止, 输出属性约简 $reduc(C) = \{d, a\}$ 。

结束语 由于由数据库模型的属性约简定义而设计的属性约简算法结合了数据库技术, 故该算法的效率较好, 其中经常使用的数据库运算有: 投影、选择和计数。使用数据库中相

关的运算是一种较好的方法。基于系统熵的属性约简算法的时间复杂度为 $O(|C|^3 |U| \log |U|)$ 。这一时间复杂度对处理大数据集时是不理想的。由于基于系统熵的属性约简与基于数据库模型的属性约简是相互等价的, 故可以用设计基于数据库模型的属性约简算法的方法去设计基于系统熵的属性约简算法, 即可以用数据库的相关操作设计基于系统熵的属性约简算法, 其时间复杂度为 $O(|C|^2 |U|)$ 。这样就可以将高效数据库的相关运算应用到属性约简算法中, 从而提高算法的运行效率, 使其能够处理大数据集。故这样的算法具有很好的可扩展性。

参考文献

- [1] Pawlak Z, Grzymala-Busse J, Slowinski R. Rough sets [J]. Communications of the ACM, 1995, 8(1): 89-95
- [2] Pawlak Z. Rough set theory and its application to data analysis [J]. Cybernetics and Systems, 1998, 9(5): 661-668
- [3] Xu Z Y, Yang B R, Song W. Comparative study of different attribute reduction based on decision table [J]. Chinese Journal of Electronics, 2006, 15(4A): 953-956
- [4] 邓大勇, 黄厚宽, 李向军. 不一致决策系统中属性约简的比较 [J]. 电子学报, 2007, 35(2): 252-255
- [5] 徐章艳, 杨炳儒, 宋威, 等. 几种不同属性约简的比较 [J]. 小型微型计算机系统, 2008, 29(5): 848-853
- [6] Hu X H, Lin T Y, Han J C. A new rough sets model based on database systems [J]. Fundamenta informaticae, 2004, 59(1): 135-152
- [7] Skowron R, Rauszer C. The discernibility matrices and functions in information systems [M]. R. Slowinski, ed. Intelligent decision support-handbook of applications and advances of the rough sets theory. Dordrecht: Kluwer, 1992: 316-362
- [8] Hu X H, Cercone N. Learning in relational databases: a rough set approach [J]. International Journal of computational intelligence, 1995, 11(3): 323-338
- [9] Wang G Y. Rough reduction in algebra view and information view [J]. International Journal of Intelligent System, 2003, 18(5): 679-688
- [10] Zhao J, Wu Z F, Li H. System entropy and its application in feature selection [J]. The Journal of China Universities of Posts and Telecommunications, 2004, 11(1): 100-105

(上接第 204 页)

- [11] Weber R, Schek H, Blott S. A quantitative analysis and performance study for similarity search methods in high dimensional spaces [C] // Proc. of the 24th Intl. Conf. on Very Large Data Bases (VLDB). 1998: 194-205
- [12] Guttman A. R-trees: A dynamic index structure for spatial searching [C] // Proc. of ACM Conf. on Management of Data (SIGMOD). 1984: 47-57
- [13] Bentley J L. K-D trees for semi-dynamic point sets [C] // Proc. of the 6th ACM Symposium on Computational Geometry (SCG). 1990: 187-197
- [14] Gionis A, Indyk P, Motwani R. Similarity search in high dimensions via hashing [C] // Proceedings of the 25th International Conference on Very Large Data Bases (VLDB). 1999

- [15] Bawa M, Condie T, Ganesan P. LSH Forest: SelfTuning [C] // Indexes for Similarity Search International World Wide Web Conference Proceedings of the 14th International Conference on World Wide Web. 2005
- [16] Stein B. Principles of hash-based text retrieval [C] // Annual ACM Conference on Research and Development in Information Retrieval Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. 2007
- [17] Athitsos V, Potamias M, Papapetrou P, et al. Nearest Neighbor Retrieval Using Distance-Based Hashing [C] // Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on. 2008