

一种基于信息粒度的动态属性约简求解算法

王永生 郑雪峰 锁延锋

(北京科技大学计算机与通信工程学院 北京 100083)

(北京科技大学材料领域知识工程北京市重点实验室 北京 100083)

摘 要 动态属性约简是粗糙集理论的重要研究内容之一。针对动态决策表构造了一种基于信息粒度的动态属性约简模型,详细分析了决策表中出现新属性动态增加时信息粒度的增量式计算方法;在此基础上,以信息粒度作为启发信息,设计了一种动态属性约简求解算法,该算法能有效利用原决策表的属性约简结果和信息粒度来降低算法的计算复杂度,并使得约简结果具有较好传承性;最后通过算例分析和实验比较进一步验证了本算法的可行性和有效性。

关键词 信息粒度,动态属性约简,动态决策表,正区域,粗糙集理论

中图法分类号 TP18 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2015.4.043

Dynamic Algorithm for Computing Attribute Reduction Based on Information Granularity

WANG Yong-sheng ZHENG Xue-feng SUO Yan-feng

(School of Computer and Communication Engineering, University of Science and Technology Beijing, Beijing 100083, China)

(Beijing Key Laboratory of Knowledge Engineering for Materials Science, University of Science and Technology Beijing, Beijing 100083, China)

Abstract Dynamic attribute reduction is one of the important issues in rough set theory. A dynamic attribute reduction model based on information granularity was constructed in dynamic decision table, and an incremental approach for computing information granularity was discussed in detail when some new attribute set is added into decision table. On this basis, a dynamic attribute reduction algorithm was proposed by using information granularity as the heuristic information. The proposed algorithm can use attribute reduction and information granularity of original decision table, which can effectively reduce the computational complexity, so that the attribute reduction has better inheritance. Finally, the example and experimental comparison indicate the feasibility and validity of the proposed algorithm.

Keywords Information granularity, Dynamic attribute reduction, Dynamic decision table, Positive region, Rough set theory

1 引言

粗糙集是一种能有效处理不精确、不完备和不确定信息的数学工具^[1,2],已成功应用于数据挖掘、机器学习和模式识别等领域^[3-5]。属性约简是粗糙集的一个重要研究课题,属性约简在无需提供所需处理数据集之外的任何先验信息的情况下,从中选择对数据分类学习的重要属性,即在保持数据分类能力不变的前提下,从数据集中删除冗余和不必要的属性。

近些年来,基于粗糙集的属性约简研究已引起了研究人员的高度重视,已有许多有效的属性约简算法被提出。文献[6]提出了采用链式基数排序方法求解等价类的优化策略,并在此基础上给出了一种近似质量计算方法,设计的属性约简算法时间复杂度为 $\max\{O(|C||U|), O(|C|^2|U/C|)\}$ 。由于在粒计算理论中,等价类也称信息粒,文献[7]基于粒度计算模型,给出了相容决策表的约简定义,并利用信息熵作为启发信息设计了一种基于粒度计算的属性约简算法,计算复杂度较高。文献[8]针对代数观点下差别矩阵的缺陷,给出了一种基于分布属性约简的差别矩阵方法,但其并不适用于不

致决策表。文献[9]中给出了基于属性值分类的多层次粗糙集模型,基于此模型,采用自顶向下、逐步细化的搜索策略构造了一种启发式泛化约简算法。文献[10]面向不协调信息系统设计了一个属性分布约简方法,使算法的计算效率得到了提高。为了进一步提高算法的效率,文献[11]在Map-Reduce框架下构造了并行知识约简算法模型,并给出了一个快速的知识约简算法。

上述大多数算法都是面向静态决策表,而对于动态决策表的属性约简研究相对较少。文献[12]提出了基于改进差别矩阵的属性约简更新算法,当对象动态增加时,通过差别矩阵的更新实现了对属性约简的更新,且每次仅处理一个对象的增加。文献[13]提出了在并行环境下基于正区域模型的动态属性约简算法,当对象动态增加时,算法能快速地对属性约简进行动态更新。文献[14]从信息观下提出了基于信息熵模型的动态属性约简,当多个对象同时加入到决策表中时,算法能一次性对其进行处理。文献[15]通过引入0-1整数规划模型,提出了一种面向单个对象变化的属性约简更新算法,当多个对象同时变化时算法相对耗时。基于差别矩阵视角,文献

到稿日期:2014-04-20 返修日期:2014-08-14 本文受国家自然科学基金项目(61163025),材料领域知识工程北京市重点实验室2012年度阶梯计划项目(Z121101002812005)资助。

王永生(1978—),男,博士生,主要研究方向为粗糙集理论及应用、知识发现, E-mail: bjnetbee@163.com; 郑雪峰(1951—),男,教授,博士生导师,主要研究方向为数据挖掘、知识发现和信息安全; 锁延锋(1977—),男,博士生,主要研究方向为数据挖掘、信息安全。

[16]给出了一种面向集值系统的动态属性约简算法;文献[17]分析了新增对象的变化情况,提出了一种基于差别矩阵元素集的属性约简动态更新算法。当决策表中属性动态变化时,文献[18]提出了基于一种动态约简算法求解不同信息熵下的属性约简。同时,文献[19]在粗糙模糊模型下,设计了两种增量式方法来动态求解下近似和上近似,该方法能有效提高约简算法的计算效率。文献[20]提出一种针对动态决策表的知识获取方法,实现了知识的动态更新。而决策表中属性的动态变化现象在实际的应用领域很常见,例如在医疗决策支持系统中,随着医疗设备的进步,先前测试成本过高或不能测试某些疾病的特征,现在可通过技术获取,从而使得医生的诊疗方案更加准确;在工业生产等方面也同样面临这种情况,人们可获得的数据属性更加丰富。

为此,如何有效更新此类决策表的属性约简已成为人们的迫切需求。本文利用粒度计算的思想,研究了当决策表中多个属性动态增加时,信息粒度的增量式计算方法,并构造了基于信息粒度的动态求解属性约简算法,避免了把动态更新后的决策表看成全新的决策表重新处理,有效利用了原决策表的属性约简结果和信息粒度。算例分析和实验结果比较表明了动态属性约简算法是有效可行的,为粒度计算下动态求解属性约简提供了一种新的解决途径。

2 相关知识

五元组 $S=(U, C, D, V, f)$ 为决策信息系统,简称决策表,其中 U 为非空的对象集, C 为条件属性集, D 为决策属性集,且 $C \cap D = \emptyset$, 令 $A = C \cup D$, 则 V 表示属性集 A 的值域,即有 $f: U \times A \rightarrow V$ 为信息映射函数,该函数为每个属性赋予相应的属性值。

定义 1^[2] 决策表 $S=(U, C, D, V, f)$, 对于任意的条件属性 $b \in B \subseteq C$, 则属性集 B 所对应的不可分辨关系(也叫等价关系) $IND(B) = \{(x, y) \in U \times U | b \in P, f(x, b) = f(y, b)\}$, 在不可分辨关系下可得到属性集 B 所对应的划分为: $U/IND(B) = \{B_1, B_2, \dots, B_v\}$ 。

定义 2^[2] 决策表 $S=(U, C, D, V, f)$, 对于任意的属性子集 $B \subseteq C$, 则对应的划分(条件等价类)为 $U/IND(B) = \{B_1, B_2, \dots, B_v\}$, 属性 D 所对应的划分为 $U/IND(D) = \{D_1, D_2, \dots, D_w\}$, 则决策表 S 中 B 相对于 D 的正区域为 $Pos_B(D) = \bigcup_{D_i \in U/D} B_i - (D_i)$, 负区域为 $Neg_B(D) = U - POS_B(D)$ 。

定义 3 决策表 $S=(U, C, D, V, f)$, 对于非空属性子集 $B \subseteq C$, 则 B 相对于决策属性 D 的信息粒度定义为:

$$IG(B) = \sum_{B_i \subseteq U/B \wedge B_i \subseteq Pos_B(D)} \frac{|B_i|^2}{|U|^2}$$

其中, B_i 为属性集 B 在 U 上的等价类, 且 B_i 是正区域 $POS_B(D)$ 中的对象集合。

定义 4 决策表 $S=(U, C, D, V, f)$, 对于 $\forall c_i \in C$, 当 $IG(C - \{c_i\}) \neq IG(C)$ 时, 则条件属性 c_i 为核属性, 同时记决策表 S 的核属性集为 $Core$ 。

定义 5 决策表 $S=(U, C, D, V, f)$, 若 $Red(C)$ 为决策表的属性约简, 则 Red 满足:

- (1) $IG(Red) = IG(C)$;
- (2) 对于 $\forall c_i \in Red$, 使得 $IG(Red - \{c_i\}) \neq IG(C)$ 。

性质 1 决策表 $S=(U, C, D, V, f)$, 若 $\forall P, Q \subseteq C$, 假设

$P \subseteq Q$, 则有 $IG(P) \leq IG(Q)$ 。

证明: 从定义 3 的信息粒度定义易证性质 1 成立。

定理 1 决策表 $S=(U, C, D, V, f)$, 若属性集 Red 满足 $IG(Red) = IG(C)$ 且对于 $\forall c_i \in Red$, $IG(Red - \{c_i\}) \neq IG(C)$, 则 Red 是决策表的一个约简。

证明: (反证法) 假设 Red 不是决策表的属性约简, 即存在另外一个属性约简 $R' \subset Red$, 使得 $IG(R') = IG(C)$, 则必定有 $Red - R' \neq \emptyset$ 成立, 不失一般性, 若设 $c_i \in Red - R'$, 则存在 $R' \subseteq Red - \{c_i\} \subset Red$ 。然后根据性质 1 信息粒度下的偏序关系, 可知 $IG(R') \leq IG(Red - \{c_i\}) \leq IG(Red)$, 因而可得 $IG(Red - \{c_i\}) \leq IG(Red)$, 且显然 $IG(Red - \{c_i\}) = IG(C)$ 成立, 最后根据定义 4 的核属性定义, 可得属性 $c_i \notin Core$ 。因为属性 $c_i \in Red - R'$, 若 $Red = \{c_1', \dots, c_i, \dots, c_p'\}$, 由于属性 $c_i \in Red$, 则属性 c_i 不能从约简结果 Red 中删除, 可知 $IG(Red - \{c_i\}) \neq IG(C)$ 成立, 再根据定义 4 可得属性 $c_i \in Core$, 这与 $c_i \notin Core$ 相矛盾, 所以假设不成立, 即不存在另外一个属性约简结果 $R' \subset Red$, 使得 $IG(R') = IG(C)$ 成立。因而定理 1 得证, 即 Red 是决策表的一个约简。

定义 6 决策表 $S=(U, C, D, V, f)$, 对于 $\forall b \in B \subseteq C$, 则属性 c_i 的属性重要性度量为 $Sig(b, R, D) = IG(B \cup \{b\}) - IG(B)$ 。

3 动态属性约简算法

3.1 增量式计算方法

在实际工程应用中决策表的属性往往是动态变化的, 即系统中存在新属性的增加或原属性的删除。属性的增加和删除对信息粒度的计算方法是类似的。本节将主要研究当有新属性增加到决策表时, 信息粒度的增量式更新计算方法。假定原决策表 $S=(U, C, D, V, f)$ 中, $U/C = \{X_1, X_2, \dots, X_m\}$, $U/D = \{D_1, D_2, \dots, D_n\}$, 下面详细分析当多个属性增加到决策表时, 信息粒度的增量式计算方法。

当在决策表中增加一个属性集 P 时, 首先根据定义 4 的信息粒度的计算方法, 需计算其对应的 $U/(C \cup P) = \{X_1/P, X_2/P, \dots, X_m/P\}$, 并计算其对应的信息粒度 $IG(C)$ 。然后根据定义 5 可知决策表中的核属性是必可少的, 为此需计算属性集 P 中是否有核属性存在, 如有核属性集 $PCore$, 则需将其加入属性约简 Red 中, 计算其对应的划分 $U/(Red \cup PCore) = \{R_1/PCore, R_2/PCore, \dots, R_t/PCore\} = \{R_1', R_2', \dots, R_v'\}$, 易知 $v \geq t$ 。若此时的信息粒度并不满足属性集 C 下的粒度度量, 则需分别计算其余属性在核属性加入的前提下的信息粒度, 即对于 $\forall a \in Red - PCore$, 计算其决策表上对应的划分 $U/(Red \cup PCore \cup \{a\}) = \{R_1'/\{a\}, R_2'/\{a\}, \dots, R_v'/\{a\}\}$, 由此可计算其对应的信息粒度, 直到新加入的属性满足属性约简定义。

3.2 算法描述

通过上述增量式计算方法, 我们可有效利用在条件属性 C 下对样本集的划分和在原属性约简 Red 下对样本集的划分, 无需重新求解其在决策表下的划分, 有效节省了计算时间, 提高了算法的计算效率。

算法 基于信息粒度的动态属性约简算法

输入: 决策表 $S=(U, C, D, V, f)$, 原属性约简 Red , 增加的新属性集 P , 原决策表的划分 U/C 和 U/D ;

输出:新的属性约简 Red' ;

Begin;

Step 1 初始化属性集 P 中的核属性集 $PCore = \emptyset$, 新的属性约简 $Red' = \emptyset$;

Step 2 对于属性集 P , 利用计数排序方法求解 $U/(C \cup P)$ 下的条件等价类;

Step 3 根据 Step 2 求解出的等价类, 计算决策表的正区域 $Pos_{C \cup P}(D)$ 和负区域 $Neg_{C \cup P}(D)$;

Step 4 计算决策表在 $C \cup P$ 下的信息粒度 $IG(C \cup P)$;

Step 5 对于 $\forall a \in P$, 分别计算信息粒度 $IG(C \cup P - \{a\})$; 如果 $IG(C \cup P - \{a\}) \neq IG(C \cup P)$, 则新增的属性 a 为核属性, 即 $PCore = PCore \cup \{a\}$;

Step 6 对于新的核属性集 $PCore$, 计算其在决策表上的等价类 $U/(Red \cup PCore) = \{R_1/PCore, R_2/PCore, \dots, R_t/PCore\} = \{R'_1, R'_2, \dots, R'_t\}$, 并计算属性集 $Red \cup PCore$ 的信息粒度 $IG(Red \cup PCore)$;

Step 7 如果 $IG(Red \cup PCore) = IG(C \cup P)$, 则 $Red' = Red \cup PCore$, 输出新的属性约简 Red' , 算法结束; 否则跳转至 Step 8;

Step 8 对于 $\forall a \in Red' - PCore$, 依次计算其等价类 $U/(Red' \cup \{a\}) = \{R'_1/\{a\}, R'_2/\{a\}, \dots, R'_t/\{a\}\}$, 由此计算其对应的信息粒度为 $IG(Red' \cup \{a\})$; 然后根据条件属性的重要性度量 $Sig(a, R, D) = IG(R \cup \{a\}) - IG(R)$, 选择信息粒度的最大值所对应的属性 c_i 存入 Red' 中, 即 $Red' = Red' \cup \{a_i\}$ 。

Step 9 如果信息粒度 $IG(Red') = IG(C \cup P)$, 则输出新的属性约简 Red' , 算法结束; 否则跳转至 Step 8。

End

算法的计算复杂性分析:

通过上述算法可知, 算法的 Step 2 仅需计算新属性集 P 在 U/C 和 U/D 的划分, 利用计数排序方法其复杂度为 $O(|P| \log |U/C|)$, 由于 Step 2 已计算出 $U/(C \cup P)$ 的划分, 因此 Step 3 求解决策表的正区域 $Pos_{C \cup P}(D)$ 和负区域 $Neg_{C \cup P}(D)$ 时仅需扫描一遍 $U/(C \cup P)$ 中的对象在决策属性上的值分布, 则算法 Step 3 的时间复杂度为 $O(|U/(C \cup P)| |D|)$; 算法的 Step 4 计算属性集 $C \cup P$ 下的信息粒度, 其时间复杂度为 $O(|U/(C \cup P)|^2)$; 算法的 Step 5 的计算耗时主要是用于计算新属性 P 中每个属性在决策表中条件属性 C 下的信息粒度情况, 其时间复杂度为 $O(|U/(C \cup \{a\})|^2)$; 算法的 Step 6 是计算属性集 $C \cup PCore$ 下的信息粒度, 时间复杂度为 $O(|U/(C \cup PCore)|^2)$; 算法的 Step 8 用于计算每个候选属性的重要性, 其最坏时间复杂度为 $O(|U/(Red \cup P)|^2)$; 可知算法空间复杂度为 $O(|U| |C|)$ 。当把新的属性集增加到原决策表中时, 文献[7]中的属性约简算法的时间复杂度为 $O(|U|^2 |C \cup P|)$, 文献[8]的时间复杂度为 $O(|U|^2 |C \cup P|^2)$, 为此本文算法的计算复杂性得到明显改善。

4 算例分析及实验比较

4.1 算例分析

为了详细分析本文算法的可行性, 下面将以一个决策表为例(见表 1)来详细分析算法的计算过程。表 1 为某炼钢厂炼炉故障分析系统的数据, 对数据进行离散化, 其中, 决策表共有 9 个故障分析实例集 $U, U = \{1, 2, \dots, 8, 9\}$, 原有 4 个条件属性为故障征兆集 $C, C = \{c_1, c_2, c_3, c_4\}$, 1 个决策属性 D 为故障类型。由于测量监测技术的提高, 工程人员发现了其他故障特征, 为此将其纳入决策表中用于故障分析, 设此时增

加的新故障条件特征集(属性集)为 $P = \{c_5, c_6, c_7\}$ 。

表 1 动态决策表

实例	条件属性 C				新属性集 P			决策属性 D
	c_1	c_2	c_3	c_4	c_5	c_6	c_7	
1	1	1	2	2	3	3	2	F
2	1	2	1	2	1	1	2	F
3	2	2	1	1	3	2	2	T
4	1	2	1	2	2	1	1	T
5	2	2	1	1	3	2	2	T
6	1	1	2	1	2	2	2	T
7	1	1	2	1	3	2	1	T
8	2	2	1	2	1	3	2	F
9	2	2	1	2	1	2	2	T

由此可知条件属性 C 在决策表上划分为 $U/C = \{\{1\}, \{2, 4\}, \{3, 5\}, \{6, 7\}, \{8, 9\}\}$, 根据属性约简的定义可知 $U/C = U/Red$, 决策属性 D 在决策表的划分 $U/D = \{\{1, 2, 4, 8\}, \{3, 5, 6, 7, 9\}\}$; 并由此可知, 决策表的正区域为 $Pos_C(D) = \{\{1\}, \{2, 4\}, \{3, 5\}, \{6, 7\}\}$, 负区域为 $Neg_C(D) = \{\{8, 9\}\}$; 根据属性约简的定义, 可知原决策表的属性约简结果为 $Red = \{c_1, c_3, c_4\}$ 。

根据本文的动态属性约简算法 Step 2, 需在 U/C 的划分下计算其在新决策表中的划分 $U/(C \cup P) = \{\{1\}, \{2\}, \{4\}, \{3, 5\}, \{6\}, \{7\}, \{8\}, \{9\}\}$, 由此可计算出新决策表的正区域 $Pos_{C \cup P}(D) = \{\{1\}, \{2\}, \{4\}, \{3, 5\}, \{6\}, \{7\}, \{8\}, \{9\}\}$, 负区域 $Neg_{C \cup P}(D) = \emptyset$, 即此时实例集均为正样本; 根据算法的 Step 4 计算条件属性 $C \cup P$ 的信息粒度 $IG(C \cup P) = 11/81$ 。根据算法的 Step 5, 首先需判断新增加的属性集有无核属性集, 即对于 $\forall a \in P$, 分别计算条件属性集 $C \cup P - \{a\}$ 的信息粒度 $IG(C \cup P - \{a\})$, 为了便于计算表示, 可令 $C' = C \cup P$ 。

由于 $U/(C' - \{c_5\}) = \{\{1\}, \{2\}, \{4\}, \{3, 5\}, \{6\}, \{7\}, \{8\}, \{9\}\}$, 则其信息粒度 $IG(C' - \{c_5\}) = 11/81$ 。

由于 $U/(C' - \{c_6\}) = \{\{1\}, \{2\}, \{4\}, \{3, 5\}, \{6\}, \{7\}, \{8\}, \{9\}\}$, 则其信息粒度 $IG(C' - \{c_6\}) = 9/81$ 。

由于 $U/(C' - \{c_7\}) = \{\{1\}, \{2\}, \{4\}, \{3, 5\}, \{6\}, \{7\}, \{8\}, \{9\}\}$, 则其信息粒度 $IG(C' - \{c_7\}) = 11/81$ 。

根据条件属性集 $C' - \{a\}$ 的信息粒度 $IG(C' - \{a\})$ 可知, 属性 c_6 为核属性集, 即 $PCore = \{c_6\}$ 。根据算法的 Step 6, 对于新的核属性集 $PCore$, 计算 $Red \cup PCore$ 在决策表上的划分 $U/(Red \cup PCore) = \{\{1\}, \{2, 4\}, \{3, 5\}, \{6, 7\}, \{8\}, \{9\}\}$, $Red' = Red \cup PCore$, 此时 $IG(Red') \neq IG(C \cup P)$, 则继续执行算法的 Step 8。对于 $\forall a \in Red - PCore$, 则依次计算划分 $U/(Red' \cup \{a\})$ 的信息粒度。

由于 $U/(Red' \cup \{c_5\}) = \{\{1\}, \{2\}, \{4\}, \{3, 5\}, \{6\}, \{7\}, \{8\}, \{9\}\}$, 则其信息粒度 $IG(Red' \cup \{c_5\}) = 11/81$ 。

由于 $U/(Red' \cup \{c_7\}) = \{\{1\}, \{2\}, \{4\}, \{3, 5\}, \{6\}, \{7\}, \{8\}, \{9\}\}$, 则其信息粒度 $IG(Red' \cup \{c_7\}) = 11/81$ 。

选择属性度量值 $Sig(c_i, Red', D)$ 的最大值所对应的属性存入属性集 Red' 中, 通过计算可知条件属性 $\{c_5, c_7\}$ 满足, 则随机选择条件属性 c_5 存入属性约简 Red' 中, 即 $Red' = Red' \cup \{c_5\}$; 根据算法的 Step 9 可知, 信息粒度 $IG(Red') = IG(C')$, 则输出新属性约简 $Red' = \{c_1, c_3, c_4, c_6, c_5\}$, 算法结束。因此, 通过上述实例分析可知, 本文的算法有效利用在条件属性 C 下对样本集的划分和在原属性约简 Red 下对样本集的划分, 无需重新求解它们的划分, 同时利用原属性约简结果快速求解出新决策表的属性约简, 有效降低了算法的计算复杂度。

4.2 实验比较

为了进一步验证本文算法的有效性和可行性,我们将本文的算法和文献[7,8]中的属性约简算法进行了性能比较和分析。数据集来自 UCI 机器学习数据库中的真实数据集,数据集情况如表 2 所列,其中,算法的运行环境为:Core(TM) Duo CPU P8700, 2.53GHz, RAM 为 4G,采用 C++ 语言在 VC++6.0 中编程实现。对于表 2 中的数据集,我们将数据集中的条件属性集分为两部分:从属性集中随机选取 10% 的属性集作为动态增加的属性集,剩余的数据集作为基准属性集;其中原属性约简的结果是从基准属性集求解的。当属性集动态增加到基准属性集时,对原属性约简进行动态更新;为了算法比较的公平性和准确性,算法的计算时间为算法的整体执行时间,同时上述 3 个算法分别执行 10 次,计算其平均执行时间(时间单位为秒)。算法的约简结果和整体执行时间如表 3 所列。

表 2 UCI 实验数据集情况

数据集	样本数	属性数	决策类数
Breast-cancer	286	9	2
Vehicle	846	18	4
Car	1728	6	4
Chess	3196	36	2
Mushroom	8124	22	2

表 3 算法的约简结果(R)和整体执行时间(T)

数据集	本文算法		文献[7]算法		文献[8]算法	
	R	T	R	T	R	T
Breast-cancer	5	0.63	5	2.06	5	2.72
Vehicle	11	2.17	11	5.75	12	6.10
Car	6	1.90	6	5.01	6	7.35
Chess	15	7.24	15	18.46	15	25.20
Page Blocks	6	10.59	6	31.53	7	44.38
Mushroom	5	18.46	5	75.84	5	100.61

从表 3 中的实验结果可知,3 种算法的属性约简结果是相同的,其主要原因是 3 种算法都是从属性的区分性出发对属性的重要性进行度量。在计算效率方面,与文献[7,8]中的属性约简算法比较,本文算法对小规模数据集的属性约简进行动态更新,在计算时间上的优势并不明显。如数据集 Breast-cancer,本文的计算时间为 0.63s,文献[7]的计算时间为 5.75s,文献[8]的计算时间为 6.10s。但是对规模较大的数据集,本文算法在计算效率上体现出较明显的优势,如在数据集 Page Blocks 和 Mushroom 这两个数据集上计算时间差别明显,特别是与文献[8]中算法比较时,本文算法的计算效率更显著。其主要原因是本文的动态算法能够有效利用原有的信息粒度及新增的属性集对象的区分性,使得当属性动态增加到数据集中时,算法仅需在原信息粒度的基础上计算新增的属性集对信息粒度的变化量,从而对属性约简进行快速更新;而文献[7,8]则需重新求解条件属性和决策属性下的划分,同时需计算属性的重要性。综合分析可知,本文算法与其他两个属性约简算法进行比较时,3 个算法的计算时间会随着数据集规模的增大而增加,而对于大规模的数据集,本文算法的计算效率要优于文献[7,8]中的算法,同时约简结果的传承性更好。

结束语 信息粒度是度量决策表中属性重要度的主要方法之一,本文针对决策表中存在属性增加的情况,给出了一种基于信息粒度的动态属性约简模型,详细讨论了当新属性增加时,给出的信息粒度的动态更新计算方法。该算法有效利

用了条件属性集和决策属性集对决策表的划分,无需重新求解它们的划分,有效提高了信息粒度的计算效率,同时有效利用原决策表的约简结果,使得约简结果保持了较好的传承性。本文的信息粒度的动态计算方法也适用于当决策表中出现属性删除的情况,是一种相对高效的动态属性约简方法。

参考文献

- [1] Pawlak Z. Rough sets[J]. International Journal of Computer and Information Science, 1982, 11(5): 341-356
- [2] Pawlak Z, Skowron A. Rudiments of rough sets[J]. Information Sciences, 2007, 117(1): 3-37
- [3] Parthaldin N M, Shen Q. Exploring the boundary region of tolerance rough sets for feature selection[J]. Pattern Recognition, 2009, 42(5): 655-667
- [4] 纪霞, 李龙澍, 陈圣兵. 改进的启发式 LEM2 快速规则获取算法[J]. 小型微型计算机系统, 2010, 31(11): 2278-2281
- [5] 钱文彬, 杨炳儒, 徐章艳, 等. 基于差别矩阵的不一致决策规则获取算法[J]. 计算机科学, 2013, 40(6): 215-218
- [6] 徐章艳, 刘作鹏, 杨炳儒, 等. 一个复杂度 $O(|C||U|)$, $O(|C|^2|U/C|)$ 的快速属性约简算法[J]. 计算机学报, 2006, 29(3): 391-399
- [7] 孙丽君, 苗夺谦. 基于粒度计算的特征选择方法[J]. 计算机科学, 2008, 35(4): 14-15
- [8] 黄国顺, 曾凡智, 陈广义, 等. 基于区分能力的 HU 差别矩阵属性约简算法[J]. 小型微型计算机系统, 2012, 33(8): 1800-1804
- [9] 叶明全, 胡学钢, 胡东辉, 等. 基于属性值分类的多层次粗糙集模型[J]. 模式识别与人工智能, 2013, 5(26): 481-491
- [10] 蒋云良, 杨章显, 刘勇. 不协调信息系统快速属性分布约简方法[J]. 自动化学报, 2012, 38(3): 382-388
- [11] 钱进, 苗夺谦, 张泽华, 等. MapReduce 框架下并行知识约简算法模型研究[J]. 计算机科学与探索, 2013, 7(1): 35-45
- [12] 杨明. 一种基于改进差别矩阵的属性约简增量式更新算法[J]. 计算机学报, 2007, 30(5): 815-822
- [13] Zhang J B, Wang J S, Li T R. A comparison of parallel large-scale knowledge acquisition using rough set theory on different MapReduce runtime systems[J]. International Journal of Approximate Reasoning, 2014, 55(3): 896-907
- [14] Liang J Y, Wang F, Dang C Y. A group incremental approach to feature selection applying rough set technique[J]. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(2): 294-308
- [15] Xu Y T, Wang L S, Zhang R Y. A dynamic attribute reduction algorithm based on 0-1 integer programming[J]. Knowledge-Based Systems, 2011, 24(8): 1341-1347
- [16] Zhang J B, Li T R. Rough sets based matrix approaches with dynamic attribute variation in set-valued information systems[J]. International Journal of Approximate Reasoning, 2012, 53(4): 620-635
- [17] 官礼和, 王国胤. 决策表属性约简集的增量式更新算法[J]. 计算机科学与探索, 2010, 4(5): 436-444
- [18] Wang F, Liang J Y, Dang C Y. Attribute reduction for dynamic data sets[J]. Applied Soft Computing, 2013, 13(1): 676-689
- [19] Cheng Y. The incremental method for fast computing the rough fuzzy approximations[J]. Data & Knowledge Engineering, 2011, 70(1): 84-100
- [20] Liu D, Li T R, Ruan D. An incremental approach for inducing knowledge from dynamic information systems[J]. Fundamenta Informaticae, 2009, 94: 245-260