文章编号:1009-3087(2014)03-0089-06

水平划分决策表的属性约简算法

葛 浩[1,2,3],李龙澍[1,3],徐 怡[1,3],杨传健[4]

(1. 安徽大学 计算智能与信号处理教育部重点实验室,安徽 合肥 230039;2. 滁州学院 机械与电子工程学院,安徽 滁州 239000;
3. 安徽大学 计算机科学与技术学院,安徽 合肥 230601;4. 滁州学院 计算机与信息工程学院,安徽 滁州 239000)

摘 要:差别矩阵属性约简是粗糙集重要约简方法之一,但在处理不一致大数据集时存在不足。为此,首先提出决策差别集的概念,并给出基于决策差别集的属性约简定义,同时研究了由该定义获得的约简与正区域约简之间的等价性。接着,给出水平划分决策表的方法,并将子决策表分配到不同的网络节点上构建子决策差别集,并行完成核属性和属性约简求解。实例分析和UCI中数据集的实验比较表明所提出的约简算法是正确的、高效的。

关键词:粗糙集;决策差别集;核属性;属性约简

中图分类号:TP181 **文献标志码**:A

An Algorithm for Attribute Reduction Based on Horizontally Partitioning Decision Table

GE Hao[1,2,3], *LI Longshu*[1,3], *XU Yi*[1,3], *YANG Chuanjian*[4]

(1. Key Lab. of Computation Intelligence and Signal Processing of Education Ministry, Anhui Univ., Hefei 230039, China;
2. School of Mechanical and Electronic Eng., Chuzhou Univ., Chuzhou 239000, China;
3. School of Computer Sci. and Technol., Anhui Univ., Hefei 230601, China;
4. School of Computer and Info. Eng., Chuzhou Univ., Chuzhou 239000, China)

Abstract: The notion of decision discernibility set and definition of attribute reduction based on decision discernibility set were presented. It was proved that attribute reduction acquired from the definition is equivalence to attribute reduction based on positive region. And then, the method of horizontally partitioning decision table was proposed and the sub-decision table can be assigned to different network nodes and finish computing core attribute and attribute reduction based on sub-decision discernibility set. Finally, the example analysis experiment results form datasets of UCI showed that the proposed parallel algorithms are efficient and effective.

Key words: rough set; decision discernibility set; core attributes; attribute reduction

属性约简是粗糙集理论[1]重要的研究内容之一,是在保持信息系统或决策表原有分类能力的基础上,删除冗余或不相关的属性。常见的属性约简方法有差别矩阵约简方法、正区域约简方法和信息熵约简方法。差别矩阵方法理解和实现方便,得到许多学者青睐。Hu[2]设计了基于Skowron差别矩阵[3]的约简算法,但约简结果与正区域约简不一致。刘文军等[4]考虑到决策表中存在不一致信息的情况,先将决策表分成正域和负域2个部分后,创建可分辨矩阵进行属性约简,该算法的时间和空间复杂度与Hu的方法相同。作者在文献[5]中为了区别不一致对象,将信息度引入决策表,由此创建差别矩阵,并进行属性约简,解决了不一致决策表属性约简问题,但文献[5]的算法1的时间和空间复杂度并没有得到降低。高学军等[6]给出简化差别矩阵的概念,在简化差别矩阵上进行属性约简,使约简算法时间和空间开销一定程度有所降低。针对不一致决策表,Miao等[7]研究了保持不同特性情况下的

收稿日期:2013-10-14

基金项目:国家自然科学基金资助项目(5130711);安徽省自然科学基金资助项目(1308085QF114);安徽高等学校省级自然科学研究重点资助项目(KJ2013A015;KJ2012A212);滁州学院优秀青年人才基金重点项目(2013RC003);计算智能与信号处理教育部重点实验室开发课题基金资助项目

作者简介:葛 浩(1976—),男,博士生,副教授. 研究方向:数据挖掘、粒计算和粗糙集理论.

3种属性约简,分别给出3种属性约简的差别矩阵实现。Yao等[8]提出了删除单个属性和属性集算子,通过这两个算子将差别矩阵简化为最小差别矩阵,从而获得约简。

上述方法多是对整个决策表进行研究,进而完成属性约简。当决策表规模增加,差别矩阵也不断增大。在处理大数据集时,差别矩阵常会消耗大量的存储空间,导致内存溢出。于是一些学者提出了对决策表分割后处理的方法。杨明等[9]对决策表进行垂直划分后,研究降低节点间通信代价的方法,提出分布式多决策表的近似约简。杨传健等[10]针对一致决策表,将二进制差别矩阵垂直划分后存储于外部介质中,以减少内存开销,再利用内外存交互,实现属性约简。叶明全等[11]提出将决策表水平划分,研究基于相对粒度的隐私保护属性约简算法,该方法也只针对一致决策表进行研究。钱进等[12]将决策表按照决策值划分为若干子决策表,构建二进制差别矩阵,进而获得规则集的方法,但按照决策值划分常常会导致子决策表划分不均,不利于并行处理。

在上述差别矩阵约简方法中,存储整个差别矩阵将花费很多内存空间,不利于大数据集的处理;另外,为了确保约简结果与正区域方法一致,通常需要先区分不一致对象,再进行约简,不一致对象的识别无疑增加了算法的开销。为此,作者提出基于水平划分决策表的属性约简方法,首先给出决策差别集的概念和基于决策差别集的约简定义,并证明由该定义获得的约简与正区域约简是一致的;同时提出将决策表水平划分成若干个子表,分配到分布式网络节点上,并行完成求核和属性约简。最后通过实例验证算法的可行性和高效性。

1 决策差别集相关概念与性质

设 $S=(U,A=C\cup D,V,f)$ 为一个决策表,其中,U为对象的有限集;A为属性集,包括条件属性集 C 和决策属性集 D,$C\cap D=\varnothing$;$V=\bigcup_{a\in A}V_a$,V_a 代表属性 a 的值域;$f:U\times A\to V$ 为信息函数,对 $\forall a\in A$,$x\in U$,有 $f(x,a)\in V_a$。

决策表 $S=(U,C\cup D,V,f)$,令 $P\subseteq C$,$\mathrm{IND}(P)=\{(x,y)\in U^2\mid f(x,a)=f(y,a),\forall a\in P\}$ 称为 U 的不可区分关系。不可区分关系实为等价关系,含 x 的等价类记为 $[x]_P=\{y\mid y\in U,(x,y)\in \mathrm{IND}(P)\}$。用 U/P 表示 P 在 U 上导出的划分。$P\_X=\{x\in U\mid [x]_P\subseteq X\}$ 和 $\overline{P}X=\{x\in U\mid [x]_P\cap X\neq\varnothing\}$ 分别称为 X 的下近似集和上近似集。称 $\mathrm{POS}_P(D)=\bigcup_{X\in U/D}P\_X$ 为 P 关于 D 的正区域。

在决策表 S 中,若 $\exists x,y\in U(x\neq y)$,有 $f(x,C)=f(y,C)$ 但 $f(x,D)\neq f(y,D)$,则称 S 为不一致决策表,x 和 y 为不一致对象。否则称 S 为一致决策表。

设决策表正区域的约简集为 $PRed$,对于 $\forall R\subseteq C$,$R\in PRed$ 满足以下2个条件:

1) $\mathrm{POS}_C(D)=\mathrm{POS}_R(D)$;

2) $\forall a\in R$,有 $\mathrm{POS}_C(D)\neq\mathrm{POS}_{R-\{a\}}(D)$。

1.1 决策差别集的核与约简

定义1 在决策表 S 中,令 $U/C=\{U_1,U_2,\cdots,U_m\}$,决策表可水平划分为 m 个子决策表,则子决策表表示为 $S_k=(U_k,C\cup D,V,f)(1\le k\le m)$。

定义2 子决策表 $S_k=(U_k,C\cup D,V,f)$ 的子差别矩阵定义为 $\boldsymbol{DM}_k=[m_{ij}^k]$,其中,$m_{ij}^k$ 满足:

$$m_{ij}^k=\begin{cases}\{a\in C\cup D\},f(x_i,a)\neq f(x_j,a),x_i\in U_k,x_j\in U;\\ \varnothing,\text{ otherwise}。\end{cases}$$

令子决策表 S_k 的子决策差别集为 SDM_k,其满足:

$$SDM_k=\begin{cases}\{\omega\},D\notin\boldsymbol{DM}_k\text{ 且 }\varnothing\neq\omega\in\boldsymbol{DM}_k;\\ \varnothing,D\in\boldsymbol{DM}_k。\end{cases}$$

其中,$\omega=\delta D$ 且 $\delta\subseteq C$。

则决策表 S 的决策表差别集定义为 $DMS=\bigcup_{k=1}^{m}SDM_k$。

性质1 子决策表 $S_k=(U_k,C\cup D,V,f)$,若 $x\in\mathrm{POS}_C(D)$,则不存在 $D\in\boldsymbol{DM}_k$;若 $\exists D\in\boldsymbol{DM}_k$,则 U_k 为不一致对象集,即 $\exists x_i,x_j\in U_k$ 有 $f(x_i,C)=f(x_j,C)\wedge f(x_i,D)\neq f(x_j,D)$。

性质2 子决策表 S_k,若 $SDM_k\neq\varnothing$,$\forall x_i\in U_k$,$\varnothing\neq m_{ij}^k\in SDM_k$,有 $x_i\in\mathrm{POS}_C(D)$;若 $x_i\in\mathrm{POS}_C(D)$ 且 $x_i\in U_k$,则 $SDM_k\neq\varnothing$。

定义3 子决策表 S_k 的核属性定义为 $\mathrm{DCORE}_k(C)=\{\delta\subseteq C\mid\delta D\in SDM_k\text{ 且 }|\delta|=1\}$。

定义4 决策表 S 的决策差别集核属性定义为 $\mathrm{DCORE}(C)=\bigcup_{k=1}^{m}\mathrm{DCORE}_k(C)$。

性质3 子决策表 S_k 核属性为 $\mathrm{DCORE}_k(C)$,则有 $\mathrm{DCORE}_k(C)\subseteq\mathrm{CORE}(C)$;若 U_k 为不一致对象集,则 $\mathrm{DCORE}_k(C)=\varnothing$。

定义5 设 S 决策差别集的约简集为 $RedM$,对于 $\forall R\subseteq C$,$R\in RedM$ 满足以下2个条件:

1) $\forall\varnothing\neq m\in DMS$,有 $m-\{D\}\cap R\neq\varnothing$;

2) $\forall a\in R$,$\exists\varnothing\neq m\in DMS$,有 $m-\{D\}\cap R-\{a\}=\varnothing$。

1.2　核与约简的等价性

定理1　设决策表 S 的核属性为 CORE(C),基于决策差别集的核属性为 DCORE(C),则有 DCORE(C) = CORE(C)。

证明:1) 先证明 DCORE(C) ⊆ CORE(C)。由于 $a \in$ DCORE(C),$\exists k \in [1..m]$,有 $aD \in SDM_k$,即 $\exists x_i \in U_k, x_j \in U$,有 $f(x_i,a) \neq f(x_j,a) \wedge f(x_i,D) \neq f(x_j,D)$,且不 $\exists x_t, x_s \in U_k$,有 $f(x_t,C) = f(x_s,C) \wedge f(x_t,D) \neq f(x_s,D)$,则说明 U_k 为一致对象集,即 $x_i \in \mathrm{POS}_C(D)$。再由 $aD \in SDM_k$,可知 $f(x_i, C-\{a\}) = f(x_j, C-\{a\}) \wedge f(x_i,D) \neq f(x_j,D)$,则 $x_i \notin \mathrm{POS}_{C-\{a\}}(D)$,故 $a \in$ CORE(C)。DCORE(C) ⊆ CORE(C) 得证。

2) 再证明 CORE(C) ⊆ DCORE(C)。即 $\forall a \in$ CORE(C),有 $a \in$ DCORE(C)。同理 1) 的证明方法,CORE(C) ⊆ DCORE(C) 可得证。

综合 1)、2),有 DCORE(C) = CORE(C)。证毕。

性质4　设 *PRed* 为决策表正区域的约简集,*DMRed* 是基于决策差别集的约简集,对于 $R \subseteq C$,$R \in PRed \Rightarrow R \in DMRed$。

证明:1) 首先,证明 $R \in PRed$,则 $\forall \varnothing \neq m \in DMS$,有 $m-\{D\} \cap R \neq \varnothing$。反证法。设 $\exists k \in [1..m]$ 和 $\varnothing \neq m_{ij}^k \in SDM_k$,有 $m_{ij}^k - D \cap R = \varnothing$,即 $\exists x_i \in U_k, x_j \in U$,有 $f(x_i,R) = f(x_j,R) \wedge f(x_i,D) \neq f(x_j,D)$,故 $x_i, x_j \notin \mathrm{POS}_R(D)$。由于 $\varnothing \neq m_{ij}^k \in SDM_k$,由性质2知 $x_i \in \mathrm{POS}_C(D)$,而 $x_i \notin \mathrm{POS}_R(D)$,则 $\mathrm{POS}_R(D) \neq \mathrm{POS}_C(D)$,此与正域约简定义条件(1) 矛盾。故假设不成立。

2) 然后,证明 $R \in PRed$,$\forall a \in R$,$\exists \varnothing \neq m \in DMS$,有 $m-\{D\} \cap R-\{a\} = \varnothing$。采用反证法。假设 $\exists k \in [1..m]$ 和 $a \in R$,$\forall \varnothing \neq m_{ij}^k \in SDM_k$,有 $m_{ij}^k - \{D\} \cap R - \{a\} \neq \varnothing$。则 $x_i \in U_k, x_j \in U$,有 $f(x_i, R-\{a\}) = f(x_j, R-\{a\}) \wedge f(x_i,D) \neq f(x_j,D)$,即 $x_i, x_j \in \mathrm{POS}_{R-\{a\}}(D)$。由于 $SDM_k \neq \varnothing$,有 $x_i \in \mathrm{POS}_C(D)$,而 $x_i \in \mathrm{POS}_{R-\{a\}}(D)$,则 $\mathrm{POS}_{R-\{a\}}(D) = \mathrm{POS}_C(D)$,此与正域约简定义条件 2) 矛盾。故假设不成立。

由 1)、2),性质4 得证。

定义3　设 *DMRed* 是决策表基于决策差别集的约简集,*PRed* 为决策表基于正区域的约简集,对于 $R \subseteq C$,$R \in DMRed \Rightarrow R \in PRed$。

同性质4 的方法,性质5 可以得证。

定理2　基于决策差别集的属性约简与正区域属性约简结果是等价的。

由性质4 和5 得证。

2　水平划分决策表的属性约简

2.1　属性约简串行实现

根据前面的研究,下面给出水平划分决策表的属性约简方法。由 $U/C = \{U_1, U_2, \cdots, U_m\}$ 将决策表划分为 m 个子表;然后,创建子决策表的决策差别集 $\{SDM_1, \cdots, SDM_m\}$,并求得子表核属性,合并为决策表的核;同时统计各个子表条件属性的频率,汇总条件属性频率。接着,将核属性加入约简集中,删除各个子决策差别集中含有核属性的元素;若存在子决策差别集不为空,则选择剩余条件属性中初始频率最大的属性,加入约简集中,同时删除各个子决策差别集中含有该属性的元素,直到所有子决策差别集均为空。下面先给出核属性求解算法。

算法1　核属性求解算法

输入:决策表 $S = (U, C \cup D, V, f)$,U 为对象集,C 为条件属性集,D 为决策属性集;

输出:核属性 CORE(C)。

Step1:CORE(C) = ∅;

Step2:for k = 1 to m DO

Step2.1:{对每个 S_k 创建 $\boldsymbol{DM}_k$ 并获得 SDM_k,根据定义4 求解 $\mathrm{DCORE}_k(C)$;

Step2.2:CORE(C) = CORE(C) ∪ $\mathrm{DCORE}_k(C)$;}

Step3:输出 CORE(C)。

显然,算法1 的时间复杂度为 $O(|C||U|^2)$,空间复杂度为 $O(|C||U|^2)$。

算法2　决策表的属性约简算法

输入:决策表 $S = (U, C \cup D, V, f)$,U 为对象集,C 为条件属性集,D 为决策属性集;

输出:属性约简 *Reduct*。

Step1:*Reduct* = ∅;$U/C = \{U_1, U_2, \cdots, U_m\}$;

Step2:调用算法1,求核 CORE(C),以及 m 个 SDM_k,并汇总属性初始频率;

Step3:*Reduct* = *Reduct* ∪ CORE(C),删除所有 SDM_k 中含有核属性的元素;

Step4:while($\bigcup_{k=1}^{m} SDM_k \neq \varnothing$)

Step4.1:{在 C - *Reduct* 中选择具有最大初始频率的属性,设为 a;

Step4.2:将属性 a 添加到 *Reduct* 中;

Step4.3:对所有 SDM_k,删除含有属性 a 的元素;}

Step5:输出 *Reduct*。

算法中等价类划分采用基数排序思想实现,则

Step1 时间复杂度为$O(|C||U|)$;Step2 时间复杂度为$O(|C||U|^2)$;Step3 时间复杂度为$O(|C||U|^2)$;Step4 时间复杂度为$O(|C||U|^2)$;因此,算法2的时间复杂度为$O(|C||U|^2)$,空间复杂度为$O(|C||U|^2)$。

2.2 属性约简并行实现

算法2的串行操作方式,时空开销较大。可通过并行处理技术提高求解效率:即,将划分后的子决策表分配到分布式网络的不同节点上,各个子决策表求核和子决策差别集更新操作在子节点上并行完成,提高求解效率的同时也减少空间开销。然而,实际数据集$|U/C|$的值比较大(例如,UCI的Car数据集,对象个数为1728,而$|U/C|=972$),如果为数据集分配$|U/C|$个节点常常是不现实的。因此,在具体实现中,可以将子决策表均匀分配,一个节点存放多个子决策表。

定义6 在决策表$S=(U,C\cup D,V,f)$中,令$U/C=\{U_1,U_2,\cdots,U_m\}$,子决策表为$S_k=(U_k,C\cup D,V,f)(1\leq k\leq m)$。将$m$个子决策分配到$n(n\leq m)$个子节点中,形成$n$个数据块$\{DT_1,\cdots,DT_n\}$,其中,$S=\cup_{i=1}^{n}DT_i$,$DT_i\cap DT_j=\varnothing(1\leq i\neq j\leq n)$,而$\forall k\in[1..m],\exists i\in[1..n]$,有$S_k\in DT_i$。

算法3 属性约简并行求解算法

输入:决策表$S=(U,C\cup D,V,f)$,U为对象集,C为条件属性集,D为决策属性集;

输出:属性约简$Reduct$。

Step1:$Reduct=CORE(C)=\varnothing$,$U/C=\{U_1,U_2,\cdots,U_m\}$;

Step2:将m个子决策表分为n个数据块$\{DT_1,\cdots,DT_n\}$,分配到n个节点上;

Step3:n节点并行创建子决策表的$SDM_k(1\leq k\leq m)$,求解$DCORE_k(C)$和条件属性频率;

Step4:统计属性初始频率,执行$\mathrm{CORE}(C)=\mathrm{CORE}(C)\cup\mathrm{DCORE}_k(C)$;

Step5:$Reduct=Reduct\cup\mathrm{CORE}(C)$,在子节点并行删除所有$SDM_k$中含有核属性的元素;

Step6:while($\forall SDM_k\neq\varnothing$)

Step6.1:{在$C-R$中选择具有最大初始频率的属性,设为a;

Step6.2:将属性a添加到$Reduct$中;

Step6.3:在子节点并行删除SDM_k中含有属性a的元素;}

Step7:输出$Reduct$。

采用了并行计算,算法3时间复杂度应为时间开销最多的节点(假设不考虑节点之间通信的代价),故算法时间复杂度为$\max\{O(|C||U_i||U|)\}(1\leq i\leq n)$。取节点中空间开销最大的作为算法空间复杂度,则空间复杂度为$\max\{O(|C||U_i||U|)\}$。

3 实例分析和实验比较

3.1 实例分析

为了分析本文方法的有效性,给出表1决策表,该表有6个对象,条件属性集$C=\{a,b,c,d\}$,D为决策属性。

表1 决策表 *S*

Tab.1 Decision table *S*

U	a	b	c	d	D
$x1$	0	1	0	0	1
$x2$	1	1	0	1	1
$x3$	1	1	1	0	0
$x4$	1	0	1	0	1
$x5$	1	0	1	0	0
$x6$	1	1	1	0	0

表2 子差别矩阵 DM_1

Tab.2 Decision discernibility matrix DM_1

U	$x1$	$x2$	$x3$	$x6$	$x4$	$x5$
$x1$	$\varnothing$	ad	acD	acD	abc	$abcD$

表3 子差别矩阵 DM_2

Tab.3 Decision discernibility matrix DM_2

U	$x1$	$x2$	$x3$	$x6$	$x4$	$x5$
$x2$	ad	$\varnothing$	cdD	cdD	bcd	$bcdD$

表4 子差别矩阵 DM_3

Tab.4 Decision discernibility matrix DM_3

U	$x1$	$x2$	$x3$	$x6$	$x4$	$x5$
$x3$	acD	cdD	$\varnothing$	$\varnothing$	bD	b
$x6$	acD	cdD	$\varnothing$	$\varnothing$	bD	b

表5 子差别矩阵 DM_4

Tab.5 Decision discernibility matrix DM_4

U	$x1$	$x2$	$x3$	$x6$	$x4$	$x5$
$x4$	abc	bcd	bD	bD	$\varnothing$	D
$x5$	$abcD$	$bcdD$	b	b	D	$\varnothing$

分析表1,有$U/C=\{\{x1\},\{x2\},\{x3,x6\},\{x4,x5\}\}$,划分为4个子决策表,分别创建子差别矩阵。根据定义2获得子决策差别集$SDM_1=\{acD,acD,abcD\}$,$SDM_1=\{cdD,cdD,bcdD\}$,$SDM_1=\{acD,acD,cdD,cdD,bD,bD\}$,$SDM_4=\varnothing$;统计$SDM_1,SDM_2,SDM_3,SDM_4$中条件属性频率,有$f(a)$

$=5, f(b)=4, f(d)=5, f(c)=10$;由于 $bD \in SDM_3$,则 $b \in \mathrm{CORE}(C)$。此时,$Reduct = \mathrm{CORE}(C) = \{b\}$,删除 SDM_1, SDM_2, SDM_3 中含有核属性的元素;然后,从 $C\text{-}Reduct = \{a,c,d\}$ 中选择初始频率最大的属性 c,加入约简 $Reduct = \{b,c\}$,删除 SDM_1, SDM_2, SDM_3 中含有属性 c 的元素后,均为空集;约简结束。最终 $Reduct = \{bc\}$。

3.2　实验比较

为了验证本文约简算法的时空性能,选取 UCI 中的数据集作为测试数据;主从机均采用 Intel I5-3470 3.2 GHz,RAM 3.4 G;实验环境为 Windows XP,VS. NET 2005 平台的 VC + + 环境。

1)以 UCI 数据库中 8 个数据集为实验数据,采用基于简化差别矩阵的约简算法(设为算法 A:对简化差别矩阵仅存储其下三角矩阵,在下三角差别矩阵上进行约简),本文算法 2 和 3(采用 1 个主节点和 3 个子节点)进行比较,结果如表 6 所示。其中,|COR|表示核属性个数,|R|表示约简中属性个数,Space 表示算法空间开销(单位:MB),Time 表示算法时间开销(单位:s)。

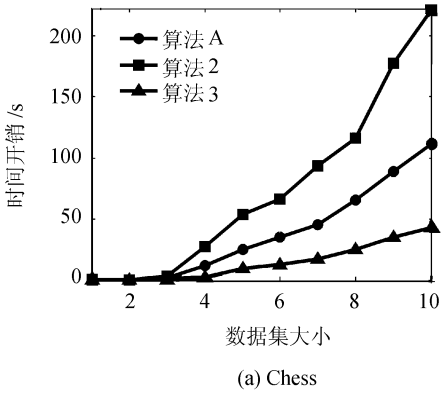
表 6　3 个算法性能比较

Tab. 6　Performance comparison of three reduction algorithms

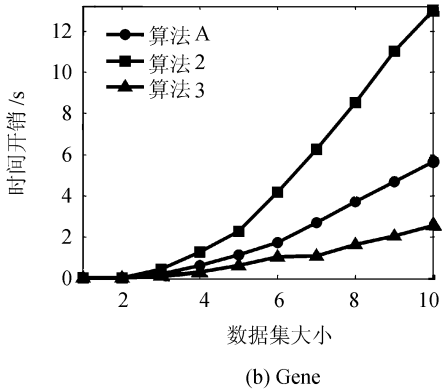
DataSet	|U|	|C|	|COR|	|U/C|	|R|	算法 A		本文算法 2		本文算法 3	
						Space/MB	Time/s	Space/MB	Time/s	Space/MB	Time/s
Zoo	101	16	2	59	5	1.544	0.003	1.732	0.005	1.452	0.002
Balance	625	4	4	625	4	10.996	0.071	20.576	0.141	6.212	0.036
Cancer	683	9	1	449	4	6.964	0.069	12.472	0.230	4.332	0.031
Tic-Tac-Toe	958	9	0	958	8	24.272	0.291	47.044	0.587	12.636	0.142
Car	1 728	6	6	972	6	25.944	0.167	50.300	0.334	10.112	0.089
Chess	3 196	36	27	3 196	29	541.216	111.063	1 079.896	221.050	271.848	42.719
Gene	3 190	60	0	3 005	10	740.676	5.661	1 478.524	13.001	365.196	2.464 8
Nursery	12 960	8	8	12 960	8	内存溢出		内存溢出		2 093.496	16.041

由表 6 可以发现,算法 A 的时间和空间开销要低于算法 2,这是由于算法 A 仅存储差别矩阵的下三角部分,而算法 2 基本上需要存储整个决策差别集且采用串行处理方式。算法 3 的时间开销要明显低于算法 A,这是因为算法 3 采用并行处理方式,大大提高了处理效率。而且算法 3 将决策表分成多个子决策表后,分布到不同的节点执行,大大降低了算法的空间开销;因此,处理 Nursery 数据集时,算法 A 和算法 2 均因空间消耗过大,而造成内存溢出,但算法 3 却可以很好地处理。

2)对 Chess 和 Gene 数据集分别按 10%,20%,…,100% 比例选择数据,构成新的数据集,采用算法 A、本文算法 2 和算法 3 分别对 10 个数据集(标号为 1,2,…,10)进行测试,结果如图 1 所示。



(a) Chess



(b) Gene

图 1　数据量变化情况的时间开销

Fig. 1　Time-consuming comparison when adding data

可以发现,随着数据量增加 3 个算法的时间开销均不断增大,但算法 2 增加的速度最快,呈现指数型增长,算法 A 时间开销大约为算法 2 的一半,而算法 3 的时间开销增加较慢,基本为一个斜率较低的线性函数。

3)采用 Chess、Gene 和 Nursery 数据集,将计算

机节点数从1增加到6,测试算法3的加速比,结果如图2所示。

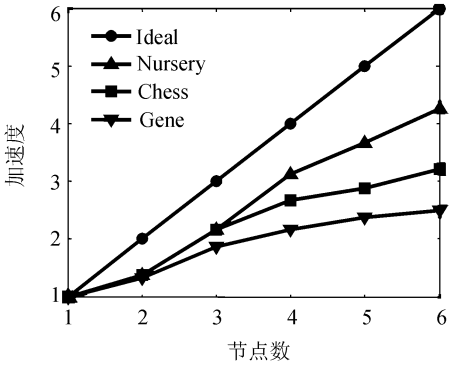


图2 加速比

Fig. 2 Speedup

理想的加速比曲线应为一条直线,由于通信代价的存在,实际的加速比要低于理想曲线。由图2可以看出,算法3在处理数据集Nursery时有较好加速比,处理数据集Gene数据集时加速较低,Chess加速比曲线处在Nursery和Gene数据集之间。这是由于Nursery的条件属性较少,仅为8个且约简即为核属性,通信代价较少;而Gene有60个条件属性且无核属性,通信代价较大;而Chess的条件属性和核属性情况处于Gene和Nursery数据集之间,故其加速比曲线也处于两者之间。

4 结 论

基于差别矩阵的属性约简是粗糙集约简理论中重要研究内容之一,由于决策表中常存在不一致信息,造成差别矩阵约简与正区域约简不一致;另外,对整个差别矩阵进行操作将消耗大量的存储空间,在处理大数据集时,会导致内存溢出。为此,作者提出了决策差别集的概念,在该差别集上完成属性约简,而无需事先区分一致与不一致对象;为了提高算法处理效率,提出了决策表水平划分方法,将子决策表分配到分布式系统的不同节点上,在不同节点并行求解子差别矩阵和子决策差别集,完成求核和属性约简算法。实例分析与UCI数据集实验结果表明,文中的并行求解算法是有效的。下一步将在云计算环境的MapReduce编程模式下,研究如何实现大数据的处理。

参考文献:

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

[2] Hu X H, Cercone N. Learning in relational databases: a rough set approach[J]. International Journal of Computational Intelligence, 1995, 11(2): 323 – 338.

[3] Skowron A, Rauszer C. The discernibility matrices and functions in information systems[M]//Intelligent Decision Support, Handbook of Applications and Advances of the Rough Sets Theory. Dordrecht, Netherland: Kluwer Academic Publishers, 1991: 331 – 362.

[4] Liu Wenjun, Gu Yundong, Feng Yanbin, et al. An improved attribute reduction algorithm of decision table[J]. Pattern Recognition and Artificial Intelligence, 2004, 17(1): 119 – 123. [刘文军,谷云东,冯艳宾,等. 基于可辨别矩阵和逻辑运算的属性约简算法的改进[J]. 模式识别与人工智能,2004,17(1):119 – 123.]

[5] Ge Hao, Li Longshu, Yang Chuanjian. Discernibility matrix based on credibility and attribute reduction method[J]. Journal of Sichuan University: Engineering Science Edition, 2011, 43(5): 146 – 152. [葛浩,李龙澍,杨传健. 可信度差别矩阵及其属性约简[J]. 四川大学学报:工程科学版,2011,43(5):146 – 152.]

[6] Gao Xuedong, Ding Jun. An attribution reduction algorithm based on simple discernibility matrix[J]. Systems Engineering Theory and Practice, 2006, 26(6): 101 – 107. [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,2006,26(6):101 – 107.]

[7] Miao D Q, Zhao Y, Yao Y Y, et al. Relative reducts in consistent and inconsistent decision tables of the Pawlak rough set model[J]. Information Sciences, 2009, 179(24): 4140 – 4150.

[8] Yao Y Y, Zhao Y. Discernibility matrix simplication for constructing attribute reducts[J]. Information Sciences, 2009, 179(7): 867 – 882.

[9] Yang Ming, Yang Ping. Approximate reduction based on conditional information entropy over vertically partitioned multi-decision table[J]. Control and Decision, 2008, 23(10): 1104 – 1108. [杨明,杨萍. 垂直分布多决策表下基于条件信息熵的近似约简[J]. 控制与决策,2008,23(10):1104 – 1108.]

[10] Yang Chuanjian, Ge Hao, Li Longshu. Attribute reduction of vertically partitioned binary discernibility matrix[J]. Control and Decision, 2013, 28(4): 563 – 568. [杨传健,葛浩,李龙澍. 垂直划分二进制可分辨矩阵的属性约简[J]. 控制与决策,2013,28(4):563 – 568.]

[11] Ye Mingquan, Wu Changrong. Privacy-preserving attribute reduction algorithm based on relative granularity over horizontally partitioned multi-decision tables[J]. Application Research of Computers, 2010, 27(10): 3701 – 3704. [叶明全,伍长荣. 水平划分多决策表下基于相对粒度的隐私保护属性约简算法[J]. 计算机应用研究,2010,27(10):3701 – 3704.]

[12] Qian Jin, Meng Xiangping, Liu Dayou, et al. A mining algorithm for concise decision rules based on rough sets theory[J]. Control and Decision, 2007, 22(12): 1368 – 1372. [钱进,孟祥萍,刘大有,等. 一种基于粗糙集理论的最简决策规则挖掘算法[J]. 控制与决策,2007,22(12):1368 – 1372.]

(编辑 杨 蓓)