一种基于改进差别矩阵的属性约简增量式更新算法

杨 明
(南京师范大学计算机科学系 南京 210097)

摘 要 粗糙集方法提供了一种新的处理不精确、不完全与不相容知识的数学工具.属性约简是粗糙集理论的重要研究内容之一,已有的大多数属性约简算法主要考虑信息系统(或决策表)不变的情况,有关属性约简的增量式更新算法却报道不多.为此,文中提出一种基于改进差别矩阵的属性约简增量式更新算法,主要考虑对象动态增加情况下属性约简的更新问题.该算法可通过快速更新差别矩阵,在动态求解核的基础上,利用原有的属性约简有效地进行属性约简的增量式更新,因而可提高属性约简的更新效率.理论分析表明,该文提出的算法是有效可行的.

关键词 粗糙集;差别矩阵;核;属性约简;增量式更新
中图法分类号 TP311

An Incremental Updating Algorithm for Attribute Reduction Based on Improved Discernibility Matrix

YANG Ming
(*Department of Computer Science, Nanjing Normal University, Nanjing* 210097)

Abstract Rough set theory is a new mathematical tool to deal with imprecise, incomplete and inconsistent data. Attribute reduction is one of important parts researched in rough set theory. Many existing algorithms mainly aim at the case of stationary information system or decision table, very little work has been done in updating of an attribute reduction. Therefore, in this paper, the authors introduce an incremental updating algorithm for attribute reduction based on discernibility matrix in the case of inserting, which only inserts a new row and column, or deletes one row and updates corresponding column when updating the decernibility matrix. After dynamically computing a core, attribute reduction can be effectively updated by utilizing the old attribute reduction. Theoretical analysis shows that the algorithm of this paper is efficient and feasible.

Keywords rough set; discernibility matrix; core; attribute reduction; incremental updating

1 引 言

波兰数学家 Pawlak 在 20 世纪 80 年代初提出的粗糙集(Rough Set,RS)是一种新的处理不精确、不完全与不相容知识的数学理论[1],近年来该理论在机器学习、数据挖掘及模式识别等多个领域得到了广泛的应用[2-3].在粗糙集理论中,属性约简是重要研究内容之一,也是知识获取的关键步骤,因此属性约简研究备受粗集研究者的关注,也取得了很大的进展[4-10].现有的属性约简大体上可分为基于差别矩阵或在此上的改进的属性约简算法[5-7]、基于正区域的属性约简算法[8-9]及基于启发式的属性约简算法[10-11] 3 种,但这些算法都是针对静态的信息系

收稿日期:2006-01-20;修改稿收到日期:2006-09-19.本课题得到国家自然科学基金(70371015)、江苏省自然科学基金(BK2005135)和江苏省高校自然科学研究项目基金(05KJB520066)资助.**杨 明**,男,1964 年生,博士,教授,主要研究领域为数据挖掘和知识发现、机器学习、粗集理论与应用.E-mail: yangm_163@163.com.

统或决策表,不适合信息系统或决策表动态变化的情况.

现实世界是发展变化的,信息系统(或决策表)中的对象在不断动态变化,已得到的属性约简将可能不再有效,这就需要对属性约简进行动态修改.为此,刘宗田教授提出属性最小约简的增量式算法,但该算法主要考虑不带决策属性的信息系统情况下的属性约简[12].文献[13]虽然可对决策表的属性约简进行动态更新,但计算量大.于是,王国胤教授等提出基于正区域的属性约简增量式更新算法[14],提高了属性约简的更新效率,但该算法未能有效利用更新核,对于对象增加而引起新的不一致情况还需改进.

现有的很多高效属性约简算法都是从核开始的,为了对核进行高效更新,文献[15]提出一种基于改进差别矩阵的核增量式更新算法,该算法考虑对象动态增加情况下核的高效更新,而求解核的主要目标之一是求解属性约简,因而如何以核的高效更新算法为基础进行高效属性约简算法研究就成了本文的主要目标.为此,本文提出一种基于改进差别矩阵的属性约简增量式更新算法,主要考虑对象动态增加情况下属性约简的更新问题.该算法可通过快速更新差别矩阵,在动态求解核的基础上,有效地利用原有的属性约简进行属性约简的增量式更新,因而可有效改进属性约简的更新效率.理论分析表明,本文提出的算法是有效可行的.

2 粗糙集概念

粗糙集理论的要点是将分类与知识联系在一起,并用等价类关系形式化表示分类,可理解为:知识是使用等价类 R 对离散空间 U 的划分,记为 $U/R=\{X_1,X_2,\cdots,X_n\}$,称为 X_i 为 U/R 的等价类.为节省篇幅,在此仅介绍和属性约简及核有关的一些概念,关于粗糙集的其它一些概念可参见文献[2-3].

信息系统 IS 是一个四元组 $\langle U,Q,V,f\rangle$,其中,U 是一组对象的非空有限集合,称为论域;设有 n 个对象,则 U 可表示为 $U=\{x_1,x_2,\cdots,x_n\}$,Q 是属性集合,$V=\bigcup_{a\in Q}V_a$,V_a 为属性 a 的值域集;f 是 $U\times Q\rightarrow V$ 的映射.属性集合 Q 通常分为条件属性集 C 与决策属性集 D.对 $B\subseteq Q$,无差别关系 $IND(B)$ 定义为 $\{(x,y)\in U^2\mid\forall a\in B,f(x,a)=f(y,a)\}$,通过 $IND(B)$ 将 U 划分为若干个类 $E_i(1\leq i\leq|U/IND(B)|)$.为便于叙述,设条件属性集合 C 中有 m 个属性:$C_1,C_2,\cdots,C_m$,其值域为有限离散集合,并用 $|\cdot|$ 表示集合的基.不失一般性,假设仅有一个决策属性 D,其取值范围是 $1,2,\cdots,k$.由 D 导出的等价类构成 U 的一个划分:$\{\psi_1,\psi_2,\cdots,\psi_k\}$,其中,$\psi_i=\{x\in U:f(x,D)=i\}$,$i=1,2,\cdots,k$.

定义 1. 在信息系统(或决策表)中,若一些数据具有相同的条件属性而具有不同的分类,则称这类数据是不一致的,否则为一致的;若两个不同的对象 x 和 y 具有相同的条件属性而具有不同的分类,则称 x 和 y 为不一致的(也称为 C-不一致的),否则称 x 和 y 为一致的(也称为 C-一致的).

定义 2. 设 $X\subseteq U$ 为论域的一个子集,$P\subseteq C$,X 的关于 P 的下近似为 $\underline{P}X(U)=\{x\in U:[x(U)]_p\subseteq X\}$;其中,$[x(U)]_p$ 表示 U 中所有与 x 在关系 $IND(P)$ 下是等价的元素构成的集合.

定义 3. 设 $P\subseteq C$,对划分 $\{\psi_1,\psi_2,\cdots,\psi_k\}$ 的 P-近似精度为 $\gamma_P(U)=\sum_{i=1}^{k}|\underline{P}\psi_i(U)|/|U|$,其中记 $P(U)=\sum_{i=1}^{k}|\underline{P}\psi_i(U)|$.

定义 4. 设 $P\subseteq C$,若 $\gamma_P(U)=\gamma_C(U)$,且不存在 $R\subset P$,使得 $\gamma_R(U)=\gamma_C(U)$,则称 P 为 C 的一个(相对于决策属性 D 的)属性约简.称满足 $\gamma_P(U)=\gamma_C(U)$ 的条件属性子集 P 为候选属性约简.所有 C 的属性约简的交称为 C 的核(简称核),记为 $Core(C,U)$.

为方便计,用 $R(U)$ 表示由信息系统(或决策表)的所有属性约简集,$CR(U)$ 表示所有候选属性约简集.

定义 5. 如果属性 $a\in C$ 满足 $\gamma_{C-\{a\}}(U)<\gamma_C(U)$,则称属性 a 为不可缺少的(indispensable),否则,称属性为冗余的.

性质 1. 属性 $a\in Core(C,U)$ 当且仅当 a 是不可缺少的属性.

3 改进的差别矩阵与核增量式更新方法

求核是很多属性约简算法的关键步骤,为有效地进行核的增量式更新,文献[15]在引入下面的定义 6 和定理 1 后,提出核的增量式更新算法.

定义 6[15]. 对给定的信息系统 IS,定义差别矩阵 $\boldsymbol{M}_1=\{m_{ij}\}$ 为

$$m_{ij}=\begin{cases}\{a\in C: f(x_i,a)\neq f(x_j,a)\}, \text{当 } f(x_i,D)\neq f(x_j,D)\text{时，这里，} x_i\in U_1,\ x_j\in U_1\\ \{a\in C: f(x_i,a)\neq f(x_j,a)\}, \text{其中，} x_i\in U_1,\ x_j\in U_2'\\ \varnothing, \text{其它情况时}\end{cases} \tag{1}$$

其中，$U_1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U_2=U-U_1$，$U_2'=delrep(U_2)$，这里的函数 $delrep(U_2)$ 描述如下：

```
begin
  U2' = ∅;
  for 任意 x∈U2 do
    if 不存在 y∈U2' 使得 ∀a∈C, f(x,a)=f(y,a) 且
      f(x,D)≠f(y,D) then
        U2' = U2' ∪ {x};
  return U2';
end
```

性质 2.　对 $U_1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U_2=U-U_1$，若 $U_2\neq\varnothing$，则 $|U_2'|<|U_2|$.

定理 1[15].　对于信息系统 IS，若记 $IDM(C,\boldsymbol{M}_1)=\{m_{ij}\mid m_{ij}\in\boldsymbol{M}_1$ 且 m_{ij} 为单个属性$\}$，则有 $IDM(C,\boldsymbol{M}_1)=Core(C,U)$，即当且仅当某个 m_{ij} 为单个属性时，该属性属于核 $Core(C,U)$.

为简化问题的讨论，文献[15]约定当信息系统(或决策表)动态改变时，决策属性 D 的取值范围仍为 $1,2,\cdots,k$. 没有特别说明，本文也遵循该约定. 设 $U_1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U_2=U-U_1$，$U_2'=delrep(U_2)$，对新增对象 x，若 $\forall y\in(U_1\cup U_2')$，$x$ 和 y 是一致的，则称 x 与 $(U_1\cup U_2')$ 一致；若 $\exists y\in U_1$ 使得 x 和 y 是不一致的，则称 x 与 U_1 不一致；若 $\exists y\in U_2'$ 使得 x 和 y 是不一致的，则称 x 与 U_2' 不一致.

对 $U_1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U_2=U-U_1$，$U_2'=delrep(U_2)$，由定义 6 得到的差别矩阵为 $\boldsymbol{M}_1$，若新增对象为 x，则只要得到 $(U_1\cup U_2'\cup\{x\})$ 的差别矩阵(简记为 $\boldsymbol{M}_1(x)$)，便可由定理 1 求得核. 因此，文献[15]将核的增量式更新归于差别矩阵的更新问题，且将 $\boldsymbol{M}_1$ 的更新分下列三种情况进行：

(1) 若 x 与 $(U_1\cup U_2')$ 一致，则在 $\boldsymbol{M}_1$ 中增加对象 x 对应的行和列，$U_1=U_1\cup\{x\}$；

(2) 若 x 与 U_1 不一致，即 $\exists y\in U_1$ 使得 x 和 y 是不一致的，则删除对象 y 所在的行，修改对象 y 所在的列，$U_2'=U_2'\cup\{y\}$，$U_1=U_1-\{y\}$；

(3) x 与 U_2' 不一致，则 $\boldsymbol{M}_1$ 保持不变.

依据上述分析，核的增量式更新算法描述如下.

算法 1[15].　IUAC(Incremental Updating Algorithm of a Core)

```
输入：(1) U1 = ⋃_{i=1}^{k} C̲ψi(U), U2 = U − U1, U2' =
          delrep(U2)，差别矩阵为 M1；
      (2) 新增对象为 x
输出：M1(x) 及相应的核 Core(C, U∪{x})
begin
(1) if x 与 (U1∪U2') 一致 then
      {
        在 M1 中增加一行和一列，并由定义 6 得到矩阵
          元素的值；
        U1 = U1 ∪ {x};
      }
    else
      if x 与 U1 不一致 then
        {
          在 U1 找到与 x 不一致的对象 y；
          删除对象 y 所在的行；
          修改对象 y 所在的列；
          U2' = U2' ∪ {y}, U1 = U1 - {y};
        }
(2) 由定理 1 得到核 Core(C, U∪{x})；
end
```

定理 2[15].　设 $U_1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U_2=U-U_1$，$U_2'=delrep(U_2)$，由定义 6 得到的差别矩阵为 $\boldsymbol{M}_1$，若新增对象为 x，则由算法 1 可得到正确的核.

由算法 1，可高效求得核，而求解核的主要目标之一是求解属性约简，因而如何以核的高效更新算法为基础进行高效属性约简算法研究就成了本文的主要目标.

4　属性约简的增量式更新

4.1　属性约简的增量式更新算法

由定义 4 可知，决策表的某个属性约简本质上就是发现保持 C-近似精度 γ_C 不变的某个属性子集 P. 为此，本文通过对新增对象 x 引起 $C(U)$ 的改变来求解属性约简，而 $C(U)$ 的改变是由于新增对象 x 的插入而引起的. 新增对象 x 的插入对近似精度的影响可分以下三种情况进行讨论.

(1) x 与 U_2 中的某个对象不一致；

(2) x 与 U_1 中的某个对象产生不一致；

(3) x 与 $U_1\cup U_2$ 中的对象是一致的.

对情况 1，对任意的 $P\subseteq C$，对象 x 的增加并不

改变 $P(U)$ 的值，即 $P(U)=P(U\cup\{x\})$，因此从直觉上可以判断原属性约简保持不变. 对情况 2 和情况 3，属性约简可能发生改变，即原属性约简不再是属性约简，一些新的属性约简将产生，因而结合核的更新和原属性约简集高效求解新的属性约简就成了本文的主要任务. 为此，本文在引入下面的定义 7 后，提出定理 3、定理 4 和定理 5.

定义 7. 若 $P\subset C$，且两个不同的对象 x 和 y，对 $\forall a\in P$ 满足 $f(x,a)=f(y,a)$，$f(x,D)\neq f(y,D)$，则称 x 和 y 为 P-不一致的，否则称 x 和 y 为 P-一致的.

推论 1. 对 $P\subset C$，若两个不同的对象 x 和 y 是 P-一致的，则对象 x 和 y 是一致的；反之，结论不成立.

性质 3. 若 P 为候选属性约简，且 $P=Core(P,U)$，则 P 为属性约简.

证明. 反证法. 因 P 为候选属性约简，故 $\forall x,y\in P(U)$，有 x 和 y 是 P-一致的. 假设 P 不是属性约简，即有 $Q\subset P$ 为属性约简，则 $\forall x,y\in U_1$，有 x 和 y 是 Q-一致的；取 $a\in(P-Q)$，因 $P=Core(P,U)$，则有 $a\in Core(P,U)$，故由定义 6 可知，$\exists s\in U_1,t\in U$ 使得 $\forall b\in(P-\{a\})$ 有 $f(s,b)=f(t,b)$，$f(s,D)\neq f(t,D)$，$f(s,a)\neq f(t,a)$ 成立，即 s,t 是 Q-不一致的，这与 Q 为属性约简矛盾. 证毕.

引理 1. 设新增对象为 x，$P\subset C$，$\boldsymbol{M}_1(x)$ 为由算法 1 得到的差别矩阵，对 $y\in U_1$，若在 $\boldsymbol{M}_1(x)$ 中 y 所对应的行中存在某个矩阵元素 m_{ij} 使得 $m_{ij}\neq\varnothing$，$(m_{ij}\cap P)=\varnothing$，则 y 与 U 中某个元素 z 为 P-不一致的.

证明. 不妨设 m_{ij} 对应的列对象 $z\in U_1$，由 $m_{ij}\neq\varnothing$ 可得 $f(y,D)\neq f(z,D)$，而由 $(m_{ij}\cap P)=\varnothing$ 可知 $f(y,b)=f(z,b)(\forall b\in P)$，即 y 和 z 是 P-不一致的. 对 $z\in U_2'$ 类似可证. 证毕.

引理 2. 设新增对象为 x，$P\subset C$，$M_1(x)$ 为由算法 1 得到的差别矩阵，对于 U_1 中的 P-一致对象 y，若 y 所对应的行中，存在矩阵元素 m_{ij} 使得 $(m_{ij}\cap P)$ 为单个属性 $\{a\}$，则有 $a\in Core(P,U)$.

证明. 类似于定理 1 可证.

依据引理 1 和引理 2，利用差别矩阵可有效地求解核 $Core(P,U)$，详情可参见算法 2.

定理 3. 对新增对象 x，若 x 与 U_2 中的某个对象不一致，$P\in R(U)$ 是一个属性约简，则 P 为 $(U\cup\{x\})$ 的属性约简，即 $P\in R(U\cup\{x\})$.

证明. 因 x 与 U_2 中的某个对象 y 不一致，由 $P\in R(U)$ 可证 $\bigcup_{i=1}^{k}\underline{P}\psi_i(U\cup\{x\})=\bigcup_{i=1}^{k}\underline{P}\psi_i(U)=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)=\bigcup_{i=1}^{k}\underline{C}\psi_i(U\cup\{x\})$，且对任意 $S\subset P$，若 $|\bigcup_{i=1}^{k}\underline{P}\psi_i(U\cup\{x\})|\geq 1$（即 $|\bigcup_{i=1}^{k}\underline{P}\psi_i(U)|\geq 1$），则有 $|\bigcup_{i=1}^{k}\underline{S}\psi_i(U\cup\{x\})|<|\bigcup_{i=1}^{k}\underline{P}\psi_i(U\cup\{x\})|$. 故 $P\in R(U\cup\{x\})$. 证毕.

可见，由定理 3 可知，当新增对象 x 与 U_2 中的某个对象不一致时，有 $R(U)=R(U\cup\{x\})$ 成立，即新增对象不改变属性约简.

定理 4. 对新增对象 x，若 x 与 U_1 中的某个对象 y 不一致，$P\in R(U)$ 是一个属性约简，则有

(1) 如果 $|[y(U)]|_P=1$，则 $\gamma_P(U\cup\{x\})=\gamma_C(U\cup\{x\})$，即 P 为 $(U\cup\{x\})$ 的一个候选属性约简；

(2) 如果 $|[y(U)]|_P>1$，则 $\gamma_P(U\cup\{x\})<\gamma_C(U\cup\{x\})$，即 P 不是 $(U\cup\{x\})$ 的一个候选属性约简.

证明.

(1) 由 $P\in R(U)$ 是一个属性约简可得 $P(U)=C(U)$，$S(U)<P(U)(S\subset P)$. 又由 $|[y(U)]|_P=1$ 及 x 与 U_1 中的某个对象 y 不一致可知，$P(U\cup\{x\})=C(U\cup\{x\})=P(U)-1$，即 $\gamma_P(U\cup\{x\})=\gamma_C(U\cup\{x\})$.

(2) 由 $P\in R(U)$ 是一个属性约简可得 $P(U)=C(U)$，$S(U)<P(U)(S\subset P)$. 不妨设 $|[y(U)]|_P=t(t>1)$，由 x 与 U_1 中的某个对象 y 不一致可知 $P(U\cup\{x\})=P(U)-t$，$C(U\cup\{x\})=P(U)-1$，即 $\gamma_P(U\cup\{x\})<\gamma_C(U\cup\{x\})$. 证毕.

依据定理 4，当新增对象 x 与 U_1 中的某个对象 y 不一致时，可分以下两种情况进行讨论：(1) $|[y(U)]|_P=1$；(2) $|[y(U)]|_P>1$. 对情况 1，由属性约简 $P\in R(U)$ 可得候选属性约简 $P\in CR(U\cup\{x\})$，由性质 3 进一步判断 P 是否是属性约简，若是，则可得 $P\in R(U\cup\{x\})$；否则，由算法 1 求得核 $Core(C,U\cup\{x\})$，并以该核为起点，类似于文献[6]逐步扩展 $(P-Core(C,U\cup\{x\}))$ 中的属性得到属性约简 $S\in R(U\cup\{x\})(S\subset P)$. 对情况 2，属性约简 $P(P\in R(U))\notin CR(U\cup\{x\})$，可从核 $Core(C,U\cup\{x\})$ 开始，逐步扩展属性得到一个属性约简 S.

定理 5. 对新增对象 x，若 x 与 $(U_1\cup U_2)$ 中的对象是一致的，$P\in R(U)$ 是一个属性约简，则有

(1) 如果 $[x(U\cup\{x\})]_P\subseteq\psi_i(1\leq i\leq k)$，即 x 与 U_1 中的对象是 P-一致的，则 P 是一个属性约简.

(2) 如果 $\exists 1\leq i\neq j\leq k$ 使得 $[x(U\cup\{x\})]_P\cap\psi_i\neq\varnothing$，$[x(U\cup\{x\})]_P\cap\psi_j\neq\varnothing$，即 x 与 U_1 中的某个对象 y 是 P-不一致的，则 $\gamma_P(U\cup\{x\})<\gamma_C(U\cup\{x\})$，即 P 不是 $(U\cup\{x\})$ 的一个候选属性约简.

证明.　由 x 与 $(U_1\cup U_2)$ 中的对象是一致的，$P\in R(U)$ 是一个属性约简可知 $P(U)=C(U)$，且 $C(U\cup\{x\})=C(U)+1$.

(1) 由 $[x(U\cup\{x\})]_P\subseteq\psi_i\ (1\leq i\leq k)$ 可知 $P(U\cup\{x\})=P(U)+1$，即 $\gamma_P(U\cup\{x\})=\gamma_C(U\cup\{x\})$. 现只要证 $\forall S\subset P$ 有 $\gamma_S(U\cup\{x\})<\gamma_P(U\cup\{x\})$ 成立即可. 因 $\forall S\subset P$ 有 $S(U)<P(U)$，而 $S(U\cup\{x\})\leq S(U)+1$，即有 $S(U\cup\{x\})<P(U)+1$（因 $S(U)<P(U)$），从而 $S(U\cup\{x\})<P(U\cup\{x\})$，即 $\gamma_S(U\cup\{x\})<\gamma_P(U\cup\{x\})$，故 P 是一个属性约简.

(2) 假设 $|[x(U\cup\{x\}]_P\cap\psi_i|=t_1$，$|[x(U\cup\{x\}]_P\cap\psi_j|=t_2$，则由已知条件知 $t_1>0$，$t_2>0$，$P(U\cup\{x\})=P(U)-t_1-t_2$，而 $C(U\cup\{x\})=P(U)+1$，故有 $\gamma_P(U\cup\{x\})<\gamma_C(U\cup\{x\})$，即 P 不是 $(U\cup\{x\})$ 的一个候选属性约简.　　证毕.

由定理 5，当 x 与 $(U_1\cup U_2)$ 中的对象是一致的，类似于定理 4，可分为以下两种情况进行讨论：(1) 新增对象 x 与 U_1 中的对象是 P-一致的；(2) 新增对象 x 与 U_1 中的某个对象 y 是 P-不一致的. 对情况 1，若 $P\in R(U)$，则 $P\in R(U\cup\{x\})$；对情况 2，可类似定理 4 的第 2 种情况进行属性约简求解.

依据上述分析，对象增加情况下的属性约简更新算法描述如下.

算法 2.　IUAARI(Incremental Updating Algorithm of Attribute Reduction for Inserting).

输入：(1) $U_1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U_2=U-U_1$，
　　$U_2'=delrep(U_2)$，差别矩阵为 $\boldsymbol{M}_1$；
　(2) $P\in R(U)$；
　(3) 新增对象为 x
输出：一个属性约简 $S\in R(U\cup\{x\})$
begin
(1) 运行算法 1 得更新差别矩阵 $\boldsymbol{M}_1(x)$ 及核 $Core(C,U\cup\{x\})$；
(2) //依据 x 与 U_1，U_2 中对象的一致和不一致，进行 //属性约简更新
　(2.1) if x 与 U_2 中的某个对象不一致 then　　//由定理 3
　　$S=P$；
　else
　(2.2) if x 与 U_1 中的某个对象 y 不一致 then　　//由定理 4
　{
　　if $|[y(U)]|_P==1$ then
　　{
　　$Core(P,U\cup\{x\})=GetCore(U_1,U,P,M_1(x))$；//求解 $Core(P,U\cup\{x\})$
　　if $P==Core(P,U\cup\{x\})$ then
　　$S=P$；
　　else
　　{
　　$S=Core(C,U\cup\{x\})$；
　　从 $(P-S)$ 选择重要的属性进行扩展直到 $S=Core(S,U\cup\{x\})$；//见文献[6]
　　}}
　　else
　　{
　　$S=Core(C,U\cup\{x\})$；
　　从 $(C-S)$ 选择重要的属性进行扩展直到 $S=Core(S,U\cup\{x\})$；
　　}
　}
　else
　{
　(2.3) if 新增对象 x 与 U_1 中的对象是 P-一致的 then //定理 5
　　$S=P$；
　　else
　　if 新增对象 x 与 U_1 中的某个对象 y 是 P-不一致的 then
　　{ $S=Core(C,U\cup\{x\})$；
　　从 $(C-S)$ 选择重要的属性进行扩展直到 $S=Core(S,U\cup\{x\})$；
　　}
　}
(3) 求得一个属性约简 $S(S\in R(U\cup\{x\}))$
end
其中，$Core(P,U\cup\{x\})$ 可通过下列函数求得
function $GetCore(U_1,U,P,\boldsymbol{M}_1(x))$
{$core=\varnothing$；
　for each $y\in U_1$ do
　　if y 与 U 中的任意对象 z 是 P-一致的 then　　//由引理 1
　　for y 所对应行中的每个矩阵元素 m_{ij} do　　//由引理 2
　　if $m_{ij}\cap P=\{a\}$ then
　　$core=core\cup\{a\}$；
　return $core$；
}

定理 6.　设 $U_1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U_2=U-U_1$，$U_2'=delrep(U_2)$，由定义 6 得到的差别矩阵为 $\boldsymbol{M}_1$，若新增对象为 x，$P\in R(U)$，则由算法 2 可求得一个

属性约简.

证明. 若 P 为$(U\cup\{x\})$的属性约简,则已得证;否则,无论 P 是候选属性约简或不是候选属性约简,则从核 $Core(C,U\cup\{x\})$开始,可得到一个属性约简[6]. 证毕.

可见,对新增对象 x 和已知属性约简 P,依据算法 2 可有效判别 P 是否为对象更新后的属性约简或候选属性约简或不是候选属性约简. 对 P 是候选属性约简或不是候选属性约简的情况,算法 2 可利用算法 1 快速求得核,并从核出发,结合现有基于差别矩阵的快速求属性约简算法可得一个属性约简.

4.2 示例说明

实例 1. 表 1 所示的是一张二值数据表,其中共有 5 个对象和 5 个属性,$C=\{C_1,C_2,C_3,C_4\}$为条件属性集,D 为决策属性.

表 1 数据表

元素	数据值				
	C_1	C_2	C_3	C_4	D
x_1	1	0	1	0	2
x_2	1	0	1	0	1
x_3	1	1	1	0	3
x_4	0	1	0	0	2
x_5	0	1	1	1	2

对实例 1,$U_1=\{x_3,x_4,x_5\}$,$U_2=\{x_1,x_2\}$,$U_2'=\{x_1\}$,由定义 6 可得差别矩阵 $\boldsymbol{M}_1$ 如下:

$$\begin{array}{c} \\ x_3 \\ x_4 \\ x_5 \end{array}\begin{array}{c} \begin{array}{cccc} x_3 & x_4 & x_5 & x_1 \end{array} \\ \begin{bmatrix} \varnothing & \{C_1,C_3\} & \{C_1,C_4\} & \{C_2\} \\ \{C_1,C_3\} & \varnothing & \varnothing & \{C_1,C_2,C_3\} \\ \{C_1,C_4\} & \varnothing & \varnothing & \{C_1,C_2,C_4\} \end{bmatrix} \end{array}$$

由定理 1 可得,实例 1 的核为$\{C_2\}$. 由定义 4 可知,$\{C_1,C_2\}$和$\{C_2,C_3,C_4\}$为实例 1 的两个属性约简.

为进一步说明算法 2,对实例 1,下面通过增加不同的对象来说明对象插入后的属性约简更新情况.

(1) 对实例 1, 若新增对象 x 为$\{1,0,1,0,3\}$,则 x 与 x_1 不一致,由算法 1 和算法 2 可知,差别矩阵保持不变,核和属性约简不改变.

(2) 对实例 1,若新增对象 x 为$\{1,1,1,0,1\}$,则 x 与 x_3 不一致,由算法 1 知 $\boldsymbol{M}_1(x)$为

$$\begin{array}{c} \\ x_4 \\ x_5 \end{array}\begin{array}{c} \begin{array}{cccc} x_3 & x_4 & x_5 & x_1 \end{array} \\ \begin{bmatrix} \{C_1,C_3\} & \varnothing & \varnothing & \{C_1,C_2,C_3\} \\ \{C_1,C_4\} & \varnothing & \varnothing & \{C_1,C_2,C_4\} \end{bmatrix} \end{array}$$

对 $P=\{C_1,C_2\}$,因$|[x_3]_{\{C_1,C_2\}}(U)|=1$,由算法 2(定理 4 的情况 1)可知 $P=\{C_1,C_2\}$为决策表更新后的一个候选属性约简,进一步可得 $S=\{C_1\}$为一个属性约简. 同理,由 $P=\{C_2,C_3,C_4\}$可得$\{C_3,C_4\}$为一个属性约简.

(3) 对实例 1,若新增对象 x 为$\{0,1,0,0,3\}$,则 x 与 x_4 不一致,由算法 1 知 $\boldsymbol{M}_1(x)$为

$$\begin{array}{c} \\ x_3 \\ x_5 \end{array}\begin{array}{c} \begin{array}{cccc} x_3 & x_4 & x_5 & x_1 \end{array} \\ \begin{bmatrix} \varnothing & \{C_1,C_3\} & \{C_1,C_4\} & \{C_2\} \\ \{C_1,C_3\} & \varnothing & \varnothing & \{C_1,C_2,C_4\} \end{bmatrix} \end{array}$$

对 $P=\{C_1,C_2\}$,因$|[x_4]_{\{C_1,C_2\}}(U)|=2>1$,由算法 2(定理 4 的情况 2)可知 $P=\{C_1,C_2\}$不是决策表更新后的一个候选属性约简. 从核 $Core(C,\{x_1,x_2,x_3,x_4,x_5,x\})=\{C_2\}$出发,可得到$\{C_1,C_2,C_3\}$为一个新的属性约简.

(4) 对实例 1,若新增对象 x 为$\{1,1,0,1,3\}$,则 x 与原决策表中的对象一致,由算法 1 知 $\boldsymbol{M}_1(x)$为

$$\begin{array}{c} \\ x_3 \\ x_4 \\ x_5 \\ x \end{array}\begin{array}{c} \begin{array}{ccccc} x_3 & x_4 & x_5 & x & x_1 \end{array} \\ \begin{bmatrix} \varnothing & \{C_1,C_3\} & \{C_1,C_4\} & \varnothing & \{C_2\} \\ \{C_1,C_3\} & \varnothing & \varnothing & \{C_1,C_4\} & \{C_1,C_2,C_3\} \\ \{C_1,C_4\} & \varnothing & \varnothing & \{C_1,C_3\} & \{C_1,C_2,C_4\} \\ \varnothing & \{C_1,C_4\} & \{C_1,C_3\} & \varnothing & \{C_2,C_3,C_4\} \end{bmatrix} \end{array}$$

对 $P=\{C_1,C_2\}$,因$[x(U\cup\{x\})]_P\subseteq\psi_3$,由算法 2(定理 5 的情况 1)可知 $P=\{C_1,C_2\}$为决策表更新后的一个属性约简.

(5) 对实例 1,若新增对象 x 为$\{0,1,0,1,3\}$,则 x 与原决策表中的对象一致,由算法 1 知 $\boldsymbol{M}_1(x)$为

$$\begin{array}{c} \\ x_3 \\ x_4 \\ x_5 \\ x \end{array}\begin{array}{c} \begin{array}{ccccc} x_3 & x_4 & x_5 & x & x_1 \end{array} \\ \begin{bmatrix} \varnothing & \{C_1,C_3\} & \{C_1,C_4\} & \varnothing & \{C_2\} \\ \{C_1,C_3\} & \varnothing & \varnothing & \{C_4\} & \{C_1,C_2,C_3\} \\ \{C_1,C_4\} & \varnothing & \varnothing & \{C_3\} & \{C_1,C_2,C_4\} \\ \varnothing & \{C_4\} & \{C_3\} & \varnothing & \{C_1,C_2,C_3,C_4\} \end{bmatrix} \end{array}$$

对 $P=\{C_1,C_2\}$,因$[x(U\cup\{x\})]_P\cap\psi_3\neq\varnothing$,$[x(U\cup\{x\})]_P\cap\psi_2\neq\varnothing$,由算法 2(定理 5 的情况 2)可知 $P=\{C_1,C_2\}$不是决策表更新后的一个候选属性约简. 设 $S=Core(C,\{x_1,x_2,x_3,x_4,x_5,x\})=\{C_2,C_3,C_4\}$,因 $S=Core(S,\{x_1,x_2,x_3,x_4,x_5,x\})$,可得 S 为一个新的属性约简.

从示例说明可见,对象增加的多数情况下仅需少量计算便可求得属性约简,这与决策表渐进更新后的重要特征相对稳定是一致的. 而在决策表更新引起其重要特征有较大变化的情况下(即原属性约简不是候选属性约简),利用核的快速更新算法得到核,进而从得到的核出发,可快速求得一个属性约简.

4.3 算法的复杂度分析

对 $U_1=\bigcup_{i=1}^{k}\underline{C}\psi_i(U)$，$U_2=U-U_1$，$U_2'=delrep(U_2)$，若动态增加的对象为 x，下面对算法 2 的空间复杂度和时间复杂度进行分析.

1）空间复杂度.

算法 2 和算法 1 的空间复杂度相同，而算法 1 的空间复杂度为 $O(|U_1|\times(|U_1|+|U_2'|))$. 由 $|U_1|\times(|U_1|+|U_2'|)\leq|U_1|\times|U|\leq|U|\times|U|$ 知，当 U 中不一致对象较多时，算法 2 可有效降低空间复杂度.

2）时间复杂度.

由算法 2 知，对新增对象 x，最坏的情况为原属性约简 P 不再为候选属性约简，该情况下的运行时间包括：(1) 判别 x 与其它对象是否为 P-一致性的时间，其时间复杂度为 $O(|U_1|+|U_2'|)$；(2) 求解核的时间. 对算法 1，判别对象 x 与 U_1 和 U_2' 一致性的时间复杂度为 $O(|U_1|+|U_2'|)$，修改差别矩阵的复杂度最多为 $O(2\times|U_1|+|U_2'|)$，扫描差别矩阵求核的时间最多为 $(|U_1|+1)\times(|U_1|+|U_2'|+1)$，故算法 1 的时间复杂度为 $O(|U_1|\times(|U_1|+|U_2'|))$；(3) 从核出发，逐步扩展重要属性的时间，其时间复杂度为 $O(|U_1|\times(|U_1|+|U_2'|))$；(4) 判别 S 是否等于 $Core(S,U)$ 的时间，其时间复杂度为 $O(|U_1|\times(|U_1|+|U_2'|))$（由引理 1 和引理 2 知）. 因此，在最坏的情况下，算法 2 的时间复杂度最多为 $O(|S|\times|U_1|\times(|U_1|+|U_2'|))$.

可见，在最坏的情况下，算法 2 因可快速更新核，因而其性能优于文献[6]的非增量式属性约简算法. 文献[14]对于对象增加而引起新的不一致情况还有待改进，如：对示例说明中的情况 3，若取原属性约简 $P=\{C_1,C_2\}$，则由文献[14]的算法 1 得候选属性约简为 $A=\{C_1,C_2\}$，但容易验证 A 不是候选属性约简，因而依据文献[14]的算法 1 对 A 进一步约简不能得到一个属性约简，因此，本文也是对文献[14]的补充和改进.

5 结　语

本文提出一种基于改进差别矩阵的属性约简增量式更新算法，主要考虑对象动态增加情况下属性约简的更新问题. 该算法可通过快速更新差别矩阵，在动态求解核的基础上，利用原有的属性约简有效地进行属性约简的增量式更新，为属性约简的增量式更新提供了一条新的途径.

参　考　文　献

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

[2] Pawlak Z. Rough set approach to multi-attribute decision analysis. European Journal of Operational Research, 1994, 72(3): 443-459

[3] Liu Qing. Rough Sets and Rough Reasoning. Beijing: Science Press, 2001(in Chinese)
(刘清. Rough 集及 Rough 推理. 北京：科学出版社，2001)

[4] Hu X H, Cercone N. Learning in relational databases: A rough set approach. Computational Intelligence 1995, 11(2): 323-338

[5] Jelonek J, Krawiec K, Slowinski R. Rough set reduction of attributes and their domains for neural networks. Computational Intelligence, 1995, 11(2): 339-347

[6] Ye Dong-Yi. An improvement to jelonek's attribute reduction algorithm. Acta Electronica Sinica, 2000, 28(12): 81-82. (in Chinese)
(叶东毅. Jelonek 属性约简算法的一个改进. 电子学报，2000，28(12)：81-82)

[7] Wang Jue, Wang Ju. Reduction algorithm based on discernibility matrix the ordered attributes method. Journal of Computer Science and Technology, 2001, 16(6): 489-504

[8] Liu Shao-Hui, Sheng Qiu-Jian, Wu Bin, Shi Zhong-Zhi, Hu Fei. Research on efficient algorithms for Rough set methods. Chinese Journal of Computers, 2003, 26(5): 524-529(in Chinese)
(刘少辉，盛秋戬，吴斌，史忠植，胡斐. Rough 集高效算法的研究. 计算机学报，2003，26(5)：524-529)

[9] Guan J W, Bell D A. Rough computational methods for information systems. Artificial Intelligences, 1998, 105(1-2): 77-103

[10] Miao Duo-Qian, Hu Gui-Rung. A heuristic algorithm for reduction of knowledge. Journal of Computer Research & Development, 1999, 36(6): 681-684(in Chinese)
(苗夺谦，胡桂荣. 知识约简的一中启发式算法. 计算机研究与发展，1999，36(6)：681-684)

[11] Woblewski J. Finding minimal reducts using genetic algorithm. Warsaw University of Technology: ICS Research Report, 16/95, 1995

[12] Liu Zong-Tian. An Incremental arithmetic for the smallest reduction of attributes. Acta Electronica Sinica, 1999, 27(11): 96-98(in Chinese)
(刘宗田. 属性最小约简的增量式算法. 电子学报，1999，27(11)：96-98)

[13] Orlowska M, Orlowski M. Maintenance of knowledge in dynamic information systems//Slowinski R ed. Proceedings of

the Intelligent Decision Support. Handbook of Applications and Advances of The Rough Set Theory, Dordrecht: Kluwer Academic Publishers, 1992: 315-330

[14] Hu Feng, Wang Guo-Yin, Huang Hai, Wu Yu. Incremental attribute reduction based on elementary sets//Proceedings of the 10th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing. Regina, Canada, 2005: 185-193

[15] Yang Ming. An incremental updating algorithm of the computation of a core based on the improved discernibility Matrix. Chinese Journal of Computers, 2006, 29(3): 407-413 (in Chinese)
(杨明. 一种基于改进差别矩阵的核增量式更新算法. 计算机学报, 2006, 29(3): 407-413)

YANG Ming, born in 1964, Ph. D., professor. His research interests include data mining and knowledge discovery, machine learning, rough sets theory and its applications.

Background

This work is supported by the National Natural Science Foundation of China under grant No. 70371015, the National Science of Jiangsu under grant No. BK2005135 and the National Science Research Foundation of Jiangsu Province under grant No. 05KJB520066.

The main objective of this paper is to provide a novel framework for incremental updating of attribute reduction for the case of inserting. By the algorithms, the decernibility matrix only needs to be inserted for a new row and column, or deleted for an old row and updated for corresponding column when adding a new object in a decision table or an information system, while the decernibility matrix must be generated completely using traditional algorithms. For an old attribute reduction R, we can efficiently judge that which one of the following three cases will be occurred by using our algorithms for inserting of a new object: 1) R is also a new attribute reduction; 2) R becomes a candidate attribute reduction; 3) R is not a candidate attribute reduction. Hence, a new attribute reduction can be efficiently acquired by using the incremental updating algorithm of a core. Compared to the existing algorithms for incremental updating of attribute reduction, the new algorithm can effectively save space and efficiently reduce time-consuming. How to apply the algorithm to intrusion detection is future work.