文章编号: 1001-0920 (2015) 01-0065-06 DOI: 10.13195/j.kzyjc.2013.1527

# 基于近似决策熵的属性约简

江 峰[1], 王莎莎[1], 杜军威[1], 眭跃飞[2]

(1. 青岛科技大学 信息科学技术学院，山东 青岛 266061；2. 中国科学院 计算技术研究所，北京 100080)

**摘 要:** 粗糙集理论已被证明是一种有效的属性约简方法. 目前有许多启发式属性约简算法已被提出, 其中基于信息熵的属性约简算法受到了广泛的关注. 为此, 针对现有的基于信息熵的属性约简算法问题, 定义一种新的信息熵模型——近似决策熵, 并提出一种基于近似决策熵的属性约简 (ADEAR) 算法. 通过在多个 UCI 数据集上的实验表明, 与现有算法相比, ADEAR 算法能够获得较小的约简和较高的分类精度, 具有相对较低的计算开销.

**关键词:** 粗糙集；属性约简；信息熵；近似决策熵

**中图分类号:** TP311 **文献标志码:** A

# Attribute reduction based on approximation decision entropy

*JIANG Feng*[1], *WANG Sha-sha*[1], *DU Jun-wei*[1], *SUI Yue-fei*[2]

(1. College of Information Science and Technology，Qingdao University of Science and Technology，Qingdao 266061，China；2. Institute of Computing Technology，Chinese Academy of Sciences，Beijing 100080，China．Correspondent：JIANG Feng，E-mail：jiangkong@163.net)

**Abstract:** The rough set theory is proved to be an effective method for attribute reduction. By now, many heuristic attribute reduction algorithms have been proposed, where the information entropy-based attribute reduction algorithms have received much attention. To solve the problems of the current information entropy-based attribute reduction algorithms, a new model of information entropy, approximate decision entropy, is defined, and an approximate decision entropy-based attribute reduction algorithm, called ADEAR, is also proposed. Some experiments are carried out on several UCI data sets. The experimental results show that ADEAR algorithm can obtain smaller reducts and higher classification accuracies than the current algorithms, and the computational cost of ADEAR algorithm is relatively low.

**Keywords:** rough sets；attribute reduction；information entropy；approximation decision entropy

## 0 引 言

属性约简是粗糙集理论的核心研究内容[1-2]. 近年来, 粗糙集的研究者提出了许多启发式属性约简方法[3-12], 这些方法通常采用属性重要性作为启发式信息. 目前, 在粗糙集中关于属性重要性的定义主要有 2 种观点: 代数观点和信息论观点. 其中, 前者以代数学中的不可分辨关系和集合运算为基础, 而后者则以信息论中的信息熵为基础[4].

信息熵由香农在 1948 年提出, 主要用以解决信息的量化度量问题[13]. 近年来, 信息熵的概念被逐渐引入到粗糙集中, 并出现了知识熵、粗糙熵和条件熵等新概念[3-5,12,14-17]. 相应地, 基于熵的属性重要性和基于熵的属性约简也应运而生. 苗夺谦等[3, 9]提出了一种基于互信息的启发式约简算法; 王国胤等[4]利用条件信息熵进行决策表的约简; Liang 等[15]提出了知识的粗糙熵及粗糙集的粗糙熵等概念, 并提出一种基于信息量的约简算法[5,10].

现有的基于熵的属性约简方法通常只是单纯地从信息论的观点出发来定义属性重要性[3-5,12], 很少有人考虑将传统的代数观点与信息论观点相结合来定义属性重要性. 实际上, 正如文献 [4]中所指出的, 属性重要性的代数定义与信息论定义具有较强的互补性: 前者考虑的是属性对论域中确定分类子集的影响, 而后者考虑的是属性对论域中不确定分类子集的影

**收稿日期:** 2013-11-04；**修回日期:** 2014-01-03.

**基金项目:** 国家自然科学基金项目(60802042, 61273180)；山东省自然科学基金项目(ZR2011FQ005, ZR2011FQ026)；山东省高等学校科技计划项目(J11LG05).

**作者简介:** 江峰(1978−), 男, 副教授, 博士, 从事人工智能、粗糙集等研究；王莎莎(1990−), 女, 硕士生, 从事数据挖掘、粗糙集的研究.

响. 因此, 有必要将这两者结合起来, 综合考虑多种因素, 从而得到一种更加全面的属性重要性度量机制. 另外, 现有的基于熵的属性约简算法的时间复杂度还比较高, 这方面也有待改进[3-5,12].

针对上述问题, 本文将提出一种新的信息熵模型——近似决策熵, 并基于该模型重新定义属性重要性. 在定义近似决策熵和属性重要性时, 将传统的代数观点与信息论观点结合起来. 首先, 从代数观点出发, 利用粗糙集中的近似精度来定义近似决策熵; 其次, 从信息论观点出发, 利用条件熵来定义近似决策熵. 通过近似精度与条件熵的结合, 可以使本文所定义的属性重要性比现有的属性重要性度量方法更加全面, 也更能体现出各方面因素对属性重要性的影响.

本文基于近似决策熵提出了一种新的属性约简(ADEAR)算法. 在约简过程中, 近似决策熵呈现出非严格单调性, 从而保证了其用于属性约简的合理性. 另外, 在ADEAR算法中引入基数排序和增量式的思想来计算划分$U/\mathrm{IND}(B)$, 从而使得ADEAR算法的计算性能优于现有的同类算法.

## 1 粗糙集理论的相关概念

在粗糙集中, 决策表是一个五元组$\mathrm{DT}=(U,C,D,V,f)$. 其中: $U$为对象集合; $C$和$D$分别为条件属性集和决策属性集; $V=\bigcup\limits_{a\in C\bigcup D}V_a$为所有属性论域的并集; $f:U\times(C\bigcup D)\to V$为一个函数, 使得对于任意$a\in C\bigcup D$和$x\in U$, $f(x,a)\in V_a$.

给定决策表$\mathrm{DT}=(U,C,D,V,f)$, 对于任意$B\subseteq C\bigcup D$, 都可以定义一个$U$上的不可分辨关系$\mathrm{IND}(B)$[2], 使得

$$\mathrm{IND}(B)=\{(x,y)\in U\times U:\forall a\in B(f(x,a)=f(y,a))\}.$$

关系$\mathrm{IND}(B)$将$U$划分为多个等价类, 所有这些等价类的集合构成了$U$的一个划分, 记为$U/\mathrm{IND}(B)$.

**定义1**(上近似、下近似) 给定决策表$\mathrm{DT}=(U,C,D,V,f)$, 对于任意$B\subseteq C\bigcup D$和$X\subseteq U$, 集合$X$的$B$-上近似和$B$-下近似分别定义为[2]

$$\underline{X}_B=\bigcup\{[x]_B\in U/\mathrm{IND}(B):[x]_B\subseteq X\},$$

$$\overline{X}_B=\bigcup\{[x]_B\in U/\mathrm{IND}(B):[x]_B\bigcap X\neq\varnothing\}.$$

**定义2**(近似精度) 给定决策表$\mathrm{DT}=(U,C,D,V,f)$, 对于任意$B\subseteq C\bigcup D$和$X\subseteq U$, 集合$X$在关系$\mathrm{IND}(B)$下的近似精度定义为[2]

$$\alpha_B(X)=\frac{|\underline{X}_B|}{|\overline{X}_B|}.$$

## 2 近似决策熵

**定义3**(近似决策熵) 给定决策表$\mathrm{DT}=(U,C,D,V,f)$, 令$U/\mathrm{IND}(D)=\{Y_1,Y_2,\cdots,Y_m\}$. 对于任意$B\subseteq C$, 令$U/\mathrm{IND}(B)=\{X_1,X_2,\cdots,X_n\}$. 将决策属性集$D$相对于$B$的近似决策熵定义为

$$\mathrm{ADE}(D|B)=-\sum_{i=1}^{m}\log_2(2-\alpha_B(Y_i))\times\sum_{j=1}^{n}(p(X_j)\times p(Y_i|X_j)\times\log_2 p(Y_i|X_j)).$$

其中: $\alpha_B(Y_i)$为$Y_i$在$\mathrm{IND}(B)$下的近似精度, $p(X_j)=\dfrac{|X_j|}{|U|}$, $p(Y_i|X_j)=\dfrac{|X_j\bigcap Y_i|}{|X_j|}$, $1\leqslant i\leqslant m$, $1\leqslant j\leqslant n$.

由定义3可知, 在定义近似决策熵时, 将粗糙集中的近似精度引入到传统的条件熵模型中: 条件熵可以有效度量知识的粒度大小, 但不能有效度量知识的完备性(即知识在刻画某个粗糙集$X$时的完备性); 相反, 粗糙集中的近似精度可以有效度量知识关于粗糙集$X$的完备性, 但不能度量知识的粒度大小. 仅仅只考虑知识的粒度大小或者知识的完备性显然都存在片面性. 因此, 有必要将这两者结合起来研究粗糙集中的不确定性度量问题. 相对于传统的信息熵模型而言, 近似决策熵不仅可以度量知识的粒度大小, 而且可以度量知识的完备性, 从而提供了一种更加全面的不确定性度量机制.

给定决策表$\mathrm{DT}=(U,C,D,V,f)$, 假设$\mathcal{P}(U)$为$U$上的所有划分的集合, 可以在$\mathcal{P}(U)$上定义一个偏序关系$\preceq$, 使得对于任意$P_1,P_2\in\mathcal{P}(U)$, 都满足$P_1\preceq P_2\Leftrightarrow\forall X\in P_1,\exists Y\in P_2(X\subseteq Y)$.

**定理1** 给定决策表$\mathrm{DT}=(U,C,D,V,f)$, 其中$U=\{x_1,x_2,\cdots,x_q\}$. 令$U/\mathrm{IND}(D)=\{Y_1,Y_2,\cdots,Y_m\}$, 并且对于任意$B\subseteq C$, 令$U/\mathrm{IND}(B)=\{X_1,X_2,\cdots,X_n\}$. 近似决策熵$\mathrm{ADE}(D|B)$满足如下性质:

1) $0\leqslant\mathrm{ADE}(D|B)\leqslant\log_2 q$;

2) 如果$U/\mathrm{IND}(B)\preceq U/\mathrm{IND}(D)$, 则$\mathrm{ADE}(D|B)$取最小值;

3) 如果$U/\mathrm{IND}(D)=\{\{x_1\},\{x_2\},\cdots,\{x_q\}\}$且$U/\mathrm{IND}(B)=\{U\}$, 则$\mathrm{ADE}(D|B)$取最大值.

**证明** 1) 首先, 由定义1和定义2可以得出, 对于任意$B\subseteq C$, $0\leqslant\alpha_B(Y_i)\leqslant 1$, 其中$1\leqslant i\leqslant m$. 进一步可以得出, 对于任意$1\leqslant i\leqslant m$, $0\leqslant\log_2(2-\alpha_B(Y_i))\leqslant 1$.

其次, 对于任意$Y_i\in U/\mathrm{IND}(D)$, 只需要考虑

$U/\mathrm{IND}(B)$ 中与 $Y_i$ 的交集不为空的等价类 $X_j$. 因为如果 $Y_i \bigcap X_j = \varnothing$, 则存在

$$p(X_j) \times p(Y_i|X_j) \times \log_2 p(Y_i|X_j) = 0.$$

对于任意 $Y_i \in U/\mathrm{IND}(D)$, 令集合 $\mathcal{S}_i = \{E \in U/\mathrm{IND}(B) : Y_i \bigcap E \neq \varnothing\}$ 包括 $U/\mathrm{IND}(B)$ 中所有与 $Y_i$ 的交集不为空的等价类, 则可以得出, 对于任意 $E \in \mathcal{S}_i$, 存在

$$\log_2 \frac{1}{q} \leqslant \log_2 p(Y_i|E) \leqslant 0,$$

且有

$$\sum_{i=1}^{m} \sum_{E \in \mathcal{S}_i} \frac{|Y_i \bigcap E|}{|U|} = \sum_{i=1}^{m} \frac{|Y_i|}{|U|} = 1.$$

综合上面推出的 2 个结论, 可以得出

$$\log_2 \frac{1}{q} \leqslant \sum_{i=1}^{m} \sum_{E \in \mathcal{S}_i} \log_2(2 - \alpha_B(Y_i)) \times \frac{|Y_i \bigcap E|}{|U|} \times \log_2 p(Y_i|E) \leqslant 0.$$

而

$$-\sum_{i=1}^{m} \sum_{E \in \mathcal{S}_i} \log_2(2 - \alpha_B(Y_i)) \times \frac{|Y_i \bigcap E|}{|U|} \times \log_2 p(Y_i|E) = -\sum_{i=1}^{m} \log_2(2 - \alpha_B(Y_i)) \times \sum_{j=1}^{n} (p(X_j) \times p(Y_i|X_j) \times \log_2 p(Y_i|X_j)).$$

因此可以证明 $0 \leqslant \mathrm{ADE}(D|B) \leqslant \log_2 q$.

2) 如果 $U/\mathrm{IND}(B) \preceq U/\mathrm{IND}(D)$, 则对于任意 $1 \leqslant i \leqslant m$, $U/\mathrm{IND}(D)$ 中的等价类 $Y_i$ 为 $U/\mathrm{IND}(B)$ 中的若干个等价类的并集. 因此, 由定义 1 和定义 2 可知, 对于任意 $1 \leqslant i \leqslant m$, $\alpha_B(Y_i) = 1$. 在此基础上, 再由定义 3 可以得到 $\mathrm{ADE}(D|B) = 0$.

3) 如果 $U/\mathrm{IND}(D) = \{\{x_1\}, \{x_2\}, \cdots, \{x_q\}\}$, 并且 $U/\mathrm{IND}(B) = \{U\}$, 则由定义 1 和定义 2 可知, 对于任意 $1 \leqslant i \leqslant q$, $\alpha_B(\{x_i\}) = 0$, 并且有

$$p(U) \times p(\{x_i\}|U) \times \log_2 p(\{x_i\}|U) = \frac{-\log_2 q}{q}.$$

再由定义 3 可得 $\mathrm{ADE}(D|B) = \log_2 q$. □

**定理 2**　给定决策表 $\mathrm{DT} = (U, C, D, V, f)$, 其中 $U = \{x_1, x_2, \cdots, x_q\}$. 对于任意 $B \subseteq C$ 和 $a \in C - B$, 有 $\mathrm{ADE}(D|B) \geqslant \mathrm{ADE}(D|B \bigcup \{a\})$.

**证明**　首先, 由不可分辨关系和偏序关系 $\preceq$ 的定义可以得出 $U/\mathrm{IND}(B \bigcup \{a\}) \preceq U/\mathrm{IND}(B)$. 再由定义 1 可知: 对于任意 $Y \subseteq U$, $|\underline{Y}_{B \bigcup \{a\}}| \geqslant |\underline{Y}_B|$ 且 $|\overline{Y}_{B \bigcup \{a\}}| \leqslant |\overline{Y}_B|$. 因此, 由定义 2 可以进一步得出

$$\alpha_{B \bigcup \{a\}}(Y) \geqslant \alpha_B(Y).$$

另外, 假设 $U/\mathrm{IND}(D) = \{Y_1, Y_2, \cdots, Y_m\}$, 由文献[4]中的引理 1 可得

$$-\sum_{i=1}^{m} \sum_{X \in U/\mathrm{IND}(B \bigcup \{a\})} (p(X) \times p(Y_i|X) \times \log_2 p(Y_i|X)) \leqslant -\sum_{i=1}^{m} \sum_{X \in U/\mathrm{IND}(B)} (p(X) \times p(Y_i|X) \times \log_2 p(Y_i|X)).$$

综合上面所推出的 2 个结论和定义 3 可以得出 $\mathrm{ADE}(D|B) \geqslant \mathrm{ADE}(D|B \bigcup \{a\})$. □

定理 2 表明了在属性约简过程中, 近似决策熵的变化规律呈现非严格单调性. 因此, 根据定理 2 可以在算法 ADEAR 中使用近似决策熵来度量属性的重要性, 并以此作为寻找约简的启发式信息.

**定义 4** (基于近似决策熵的约简)　给定决策表 $\mathrm{DT} = (U, C, D, V, f)$, 对于 $\forall B \subseteq C$, 若存在 $\mathrm{ADE}(D|B) = \mathrm{ADE}(D|C)$, 并且对于任意 $b \in B$, $\mathrm{ADE}(D|B - \{b\}) > \mathrm{ADE}(D|C)$, 则称 $B$ 为 $C$ 在决策表 DT 中相对于 $D$ 的一个约简.

**定义 5** (基于近似决策熵的核)　给定决策表 $\mathrm{DT} = (U, C, D, V, f)$, 对于任意 $a \in C$, 如果 $\mathrm{ADE}(D|C - \{a\}) > \mathrm{ADE}(D|C)$, 则称属性 $a$ 为 $C$ 在决策表 DT 中相对于 $D$ 的一个核属性.

**定义 6** (基于近似决策熵的属性重要性)　给定决策表 $\mathrm{DT} = (U, C, D, V, f)$, 对于任意 $B \subset C$ 和 $a \in C - B$, 将属性 $a$ 在决策表 DT 中相对于 $B$ 和 $D$ 的重要性定义为

$$\mathrm{Sig}(a, B, D) = \mathrm{ADE}(D|B) - \mathrm{ADE}(D|B \bigcup \{a\}).$$

## 3　属性约简 (ADEAR) 算法

由于在 ADEAR 算法中需要反复地计算关系 $\mathrm{IND}(B)$ 对于 $U$ 的划分 $U/\mathrm{IND}(B)$, 为了降低 ADEAR 算法的复杂度, 将采用基数排序的思想来计算 $U/\mathrm{IND}(B)$[18].

另外, 本文还提出了一种增量式计算 $U/\mathrm{IND}(B)$ 的方法. 给定决策表 $\mathrm{DT} = (U, C, D, V, f)$, 对于任意 $B \subseteq C$ 和 $a \in B$, 如果已经计算出划分 $U/\mathrm{IND}(B - \{a\}) = \{E_1, E_2, \cdots, E_h\}$, 则可以针对每个 $E_i$, 计算划分 $E_i/\mathrm{IND}(\{a\})$, $1 \leqslant i \leqslant h$. 而 $\bigcup_{i=1}^{h} (E_i/\mathrm{IND}(\{a\})) = U/\mathrm{IND}(B)$. 通过充分利用之前已经计算出的划分 $U/\mathrm{IND}(B - \{a\})$, 可以使得计算 $U/\mathrm{IND}(B)$ 的时间复杂度降低为 $O(|U|)$.

**算法 1**　计算近似决策熵.

输入: 决策表 $\mathrm{DT}=(U,C,D,V,f)$, $B\subseteq C$, 划分 $U/\mathrm{IND}(B)=\{X_1,X_2,\cdots,X_n\}$, $U/\mathrm{IND}(D)=\{Y_1,Y_2,\cdots,Y_m\}$ 和 $U/\mathrm{IND}(B\bigcup D)$;

输出: 近似决策熵 $\mathrm{ADE}(D|B)$.

1) 初始化. 对于每个 $1\leqslant j\leqslant n$, 令 $\mathrm{Flag}[j]=F$, 并且对于每个 $X_j\in U/\mathrm{IND}(B)$, 令 $N(X_j)$ 表示 $X_j$ 的编号.

2) 对于任意 $x\in U$, 根据 $U/\mathrm{IND}(B)$ 和 $U/\mathrm{IND}(B\bigcup D)$ 分别计算等价类 $[x]_B$ 和 $[x]_{B\bigcup D}$ 的势.

3) 对于每个 $1\leqslant i\leqslant m$, 循环执行:

① 对于每个 $x\in Y_i$, 如果 $|[x]_B|=|[x]_{B\bigcup D}|$ 且 $\mathrm{Flag}[N([x]_B)]=F$, 则令

$$\mathrm{count1}=\mathrm{count1}+|[x]_B|,$$

$$\mathrm{Flag}[N([x]_B)]=T;$$

② 令 $\mathrm{LA}[i]=\mathrm{count1}$.

4) 对于每个 $1\leqslant j\leqslant n$, 令 $\mathrm{Flag}[j]=F$.

5) 对于每个 $1\leqslant i\leqslant m$, 循环执行:

① 对于每个 $x\in Y_i$, 如果 $\mathrm{Flag}[N([x]_B)]=F$, 则令

$$\mathrm{count2}=\mathrm{count2}+|[x]_B|,$$

$$\mathrm{Flag}[N([x]_B)]=T;$$

② 令 $\mathrm{UA}[i]=\mathrm{count2}$, $\alpha_B(Y_i)=\dfrac{\mathrm{LA}[i]}{\mathrm{UA}[i]}$;

③ 对于每个 $x\in Y_i$, 如果 $\mathrm{Flag}[N([x]_B)]=T$, 则令 $\mathrm{Flag}[N([x]_B)]=F$.

6) 对于每个 $1\leqslant i\leqslant m$, 循环执行:

① 对于每个 $x\in Y_i$, 若 $\mathrm{Flag}[N([x]_B)]=F$, 则令

$$\mathrm{count3}=\mathrm{count3}+\frac{|[x]_B|}{|U|}\times\frac{|[x]_{B\bigcup D}|}{|[x]_B|}\times\log_2\frac{|[x]_{B\bigcup D}|}{|[x]_B|},$$

$\mathrm{Flag}[N([x]_B)]=T$;

② 令 $\mathrm{ADE}(D|B)=\mathrm{ADE}(D|B)-\log_2(2-\alpha_B(Y_i))\times\mathrm{count3}$;

③ 对于每个 $x\in Y_i$, 如果 $\mathrm{Flag}[N([x]_B)]=T$, 则令 $\mathrm{Flag}[N([x]_B)]=F$.

7) 返回 $\mathrm{ADE}(D|B)$.

**算法 2** ADEAR 算法.

输入: 决策表 $\mathrm{DT}=(U,C,D,V,f)$;

输出: 存放约简结果的集合 $R$.

1) 初始化. 令集合 $\mathrm{Core}=\varnothing$, $R=\varnothing$.

2) 计算 $U/\mathrm{IND}(D)$, $U/\mathrm{IND}(C)$ 和 $U/\mathrm{IND}(C\bigcup D)$.

3) 利用算法 2 计算 $\mathrm{ADE}(D|C)$.

4) 对于任意 $a\in C$, 循环执行:

① 计算 $U/\mathrm{IND}(C-\{a\})$, $U/\mathrm{IND}((C-\{a\})\bigcup D)$;

② 利用算法 2 计算 $\mathrm{ADE}(D|C-\{a\})$;

③ 若 $\mathrm{ADE}(D|C-\{a\})>\mathrm{ADE}(D|C)$, 则令 $\mathrm{Core}=\mathrm{Core}\bigcup\{a\}$.

5) 令 $R=\mathrm{Core}$. 如果 $R$ 等于空集, 则令 $\mathrm{Temp}=\mathrm{ADE}(D|C)+1$; 否则, 首先计算 $U/\mathrm{IND}(R)$ 和 $U/\mathrm{IND}(R\bigcup D)$, 再计算 $\mathrm{ADE}(D|R)$, 并令 $\mathrm{Temp}=\mathrm{ADE}(D|R)$.

6) 当 $\mathrm{Temp}\neq\mathrm{ADE}(D|C)$ 时, 循环执行:

① 对于每一个 $a\in C-R$, 循环执行:

基于 $U/\mathrm{IND}(R)$ 和 $U/\mathrm{IND}(R\bigcup D)$, 增量式计算 $U/\mathrm{IND}(R\bigcup\{a\})$ 和 $U/\mathrm{IND}(R\bigcup\{a\}\bigcup D)$;

利用算法 2 计算 $\mathrm{ADE}(D|R\bigcup\{a\})$, 并由此得到属性 $a$ 的重要性 $\mathrm{Sig}(a,R,D)$.

② 从 $C-R$ 中选择重要性最大的属性 $a'$ (如果有多个, 则随机选择一个).

③ 令 $\mathrm{Temp}=\mathrm{ADE}(D|R\bigcup\{a'\})$, 并且令 $R=R\bigcup\{a'\}$.

7) 返回约简结果 $R$.

在最坏的情况下, 算法 1 的时间复杂度和空间复杂度均为 $O(|U|)$. 而算法 2 的时间复杂度为 $O(|C|^2\times|U|)$, 空间复杂度为 $O(|C|\times|U|)$.

## 4 实验结果

为了验证 ADEAR 算法的性能, 在 7 个 UCI 数据集上进行了实验: 1) Tic-tac-toe endgame (Tic); 2) Wisconsin breast cancer (Breast); 3) Congressional voting records (Vote); 4) Zoo; 5) Lymphography(Lymph); 6) Mushroom (Mush); 7) Soybean-small (Soyb)[19]. 其中 Vote 数据集所使用的是它的一个子集(包含 300 条记录).

采用 Java 语言实现 ADEAR 算法, 硬件环境如下: Intel 处理器 2.0 GHz, 2 GB 内存. 将 ADEAR 与如下 6 个具有代表性的约简算法进行了比较: 1) 基于正区域的算法 (POS)[1]; 2) 基于条件信息量的算法 (CIQ)[12]; 3) 基于条件熵的算法 (CE)[4]; 4) 基于分辨矩阵的算法 (DISM)[6]; 5) 基于遗传算法的算法 (GA)[20]; 6) 基于粒子群优化和粗糙集理论的算法 (PSORS)[21].

首先比较不同算法的约简大小. 表 1 给出了每个约简算法在不同数据集上的约简大小. 其中, POS、CE、DISM、GA 和 PSORS 的实验结果可以从文献[21]中得到. 另外, 本文基于 Java 实现了 CIQ.

**表 1　不同算法的约简大小比较**

| 数据集 | 约简中所包括的属性个数 | | | | | | |
|---|---|---|---|---|---|---|---|
| | POS | CIQ | CE | DISM | GA | PSORS | ADEAR |
| Tic | 8 | 8 | 7 | 8 | 8 | 8 | 8 |
| Breast | 4 | 4 | 4 | 5 | 4 | 4 | 4 |
| Vote | 9 | 8 | 11 | 8 | 9 | 8 | 8 |
| Zoo | 5 | 5 | 10 | 5 | 6 | 5 | 5 |
| Lymph | 6 | 6 | 8 | 7 | 8 | 7 | 6 |
| Mush | 5 | 4 | 5 | 6 | 5 | 4 | 4 |
| Soyb | 2 | 2 | 2 | 2 | 6 | 2 | 2 |

由表 1 可知, ADEAR 算法在大部分数据集上都能够获得较小的约简. 除了 Tic 数据集之外, 本文所提出的算法都能够获得最小约简, 即使是在 Tic 上, ADEAR 算法所产生的约简也非常接近最小约简.

下面比较不同约简算法所对应的分类精度. 首先借鉴 Wang 等[21]设计的实验方法, 采用十折交叉验证的方式来评估每个约简的分类精度, 并使用 Rough Set Exploration System (RSES) 中的 LEM2 算法从约简之后的训练集中提取决策规则[16,22]; 然后利用这些规则对测试集进行测试 (冲突通过 Standard Voting 方法来消解)[22].

**表 2　不同算法的分类精度比较**　　%

| 数据集 | 分类精度 | | | | | | |
|---|---|---|---|---|---|---|---|
| | POS | CIQ | CE | DISM | GA | PSORS | ADEAR |
| Tic | 94.42 | 96.33 | 77.89 | 86.21 | 93.05 | 96.32 | 96.33 |
| Breast | 95.94 | 98.63 | 94.20 | 95.94 | 95.65 | 95.80 | 98.97 |
| Vote | 94.33 | 97.94 | 92.33 | 93.67 | 94.0 | 95.33 | 98.51 |
| Zoo | 96.0 | 97.75 | 94.0 | 94.0 | 92.0 | 96.0 | 97.75 |
| Lymph | 85.71 | 81.5 | 72.14 | 74.29 | 70.0 | 75.71 | 81.5 |
| Mush | 100 | 100 | 90.83 | 100 | 100 | 99.70 | 100 |
| Soyb | 100 | 100 | 100 | 100 | 97.50 | 100 | 100 |

由表 2 可知, ADEAR 算法的分类性能在大多数情况下都优于或等于其他算法. 除了在 Lymph 上低于 POS 算法之外, 在其他数据集上, 本文所提出的算法都具有最高的精度. 因此, 总体而言, ADEAR 算法具有更好的分类性能.

通过分析实验结果可以发现, ADEAR 在 Lymph 上的分类精度低于 POS. 这是因为 Lymph 中包含了 6 个离群点, 这些离群点的存在影响了 ADEAR 的分类性能. 相对而言, POS 受这些离群点的影响要小一些, 因此, 其分类精度高于 ADEAR. 针对上述问题可以考虑在利用近似决策熵进行属性约简之前, 预先采用某种离群点检测方法将离群点找出, 尽量避免离群点对属性约简的影响, 从而进一步提高 ADEAR 的性能.

最后比较 ADEAR 算法与文献[10]中的 2 个约简算法 SCE 和 FSPA-SCE 的运行时间.

**表 3　不同算法的运行时间比较**　　s

| 数据集 | 运行时间 | | |
|---|---|---|---|
| | SCE | FSPA-SCE | ADEAR |
| Tic | 4.500 0 | 3.109 4 | 0.093 |
| Breast | 1.343 8 | 0.843 8 | 0.047 |
| Mush | 162.640 6 | 159.593 8 | 1.656 |

由表 3 可知, ADEAR 的运行时间远小于 SCE 和 FSPA-SCE. 这是因为在 ADEAR 中采用了基数排序和增量式的思想来计算划分 $U/\mathrm{IND}(B)$[18], 从而极大地减少了运行时间.

## 5　结　　论

本文将传统的代数观点与信息论观点结合起来定义属性重要性, 并由此提出了一种新的属性约简算法. 基数排序和增量式思想的引入使所提出的约简算法具有较低的计算开销. 另外, 在真实数据集上的实验表明, 所提出的算法不仅能够获得较小的约简结果, 而且具有较好的分类性能.

可以进一步考虑将本文的方法扩展到邻域粗糙集或者模糊粗糙集模型中[7-8], 设计一种不需要离散化就能够直接处理连续型属性的约简方法.

## 参考文献(References)


[1] Hu X H. Knowledge discovery in databases: An attribute-oriented rough set approach[D]. Regina: Regina University, 1995.

[2] Pawlak Z. Rough sets[J]. Int J of Computer and Information Sciences, 1982, 11(5): 341-356.

[3] 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.
(Miao D Q, Hu G R. An heuristic algorithm of knowledge reduction[J]. Computer Research and Development, 1999, 36(6): 681-684.)

[4] 王国胤, 于洪, 杨大春. 基于条件信息熵的决策表约简[J]. 计算机学报, 2002, 25(7): 759-766.
(Wang G Y, Yu H, Yang D C. Decision table reduction based on conditional information entropy[J]. Chinese J of Computers, 2002, 25(7): 759-766.)

[5] Liang J Y, Xu Z B. The algorithm on knowledge reduction in incomplete information systems[J]. Int J of Uncertainty, Fuzziness and Knowledge-Based Systems, 2002, 10(1): 95-103.

[6] Hu K Y, Lu Y C, Shi C Y. Feature ranking in rough sets[J]. AI Communication, 2003, 16(1): 41-50.

[7] Hu Q H, Xie Z X, Yu D R. Hybrid attribute reduction based on a novel fuzzy-rough model and information

granulation[J]. Pattern Recognition, 2007, 40(12): 3509-3521.

[8] Hu Q H, Yu D R, Liu J F, et al. Neighborhood rough set based heterogeneous feature subset selection[J]. Information Sciences, 2008, 178(18): 3577-3594.

[9] 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.

[10] Qian Y H, Liang J Y, Pedrycz W, et al. Positive approximation: An accelerator for attribute reduction in rough set theory[J]. Artificial Intelligence, 2010, 174(9/10): 597-618.

[11] 杨明, 杨萍. 基于广义差别矩阵的核和属性约简算法[J]. 控制与决策, 2008, 23(9): 1049-1054.
(Yang M, Yang P. Algorithms based on general discernibility matrix for computation of a core and attribute reduction[J]. Control and Decision, 2008, 23(9): 1049-1054.)

[12] 李鸿. 基于条件信息量的知识相对约简算法[J]. 中国矿业大学学报, 2005, 34(3): 378-382.
(Li H. Algorithm for relative reduction of knowledge in information systems based on a conditional information quantity[J]. J of China University of Mining and Technology, 2005, 34(3): 378-382.)

[13] Shannon C E. The mathematical theory of communication[J]. Bell System Technical J, 1948, 27(3/4): 373-423.

[14] Düntsch I, Gediga G. Uncertainty measures of rough set prediction[J]. Artificial Intelligence, 1998, 106(1): 109-137.

[15] Liang J Y, Shi Z Z, Li D Y, et al. Information entropy, rough entropy and knowledge granularity in incomplete information systems[J]. Int J of General Systems, 2006, 35(6): 641-654.

[16] 代建华, 潘云鹤. 一种基于分类一致性的决策规则获取算法[J]. 控制与决策, 2004, 19(10): 1086-1090.
(Dai J H, Pan Y H. Algorithm for acquisition of decision rules based on classification consistency rate[J]. Control and Decision, 2004, 19(10): 1086-1090.)

[17] Zhao H B, Jiang F, Wang C P. An approximation decision entropy based decision tree algorithm and its application in intrusion detection[C]. Proc of the 6th Int Conf on Rough Set and Knowledge Technology. Chengdu: Springer-Verlag, 2012: 101-106.

[18] 徐章艳, 刘作鹏, 杨炳儒, 等. 一个复杂度为$\max(O(|C||U|), O(|C|^2|U/C|))$的快速属性约简算法[J]. 计算机学报, 2006, 29(3): 391-399.
(Xu Z Y, Liu Z P, Yang B R, et al. A quick attribute reduction algorithm with complexity of $\max(O(|C||U|), O(|C|^2|U/C|))$[J]. Chinese J of Computers, 2006, 29(3): 391-399.)

[19] Bay S D. The UCI KDD repository[DB/OL]. University of California, Irvine, 1999. http://kdd.ics.uci.edu.

[20] Wroblewski J. Finding minimal reducts using genetic algorithms[C]. The 2nd Annual Joint Conf on Information Sciences. North Carolina: Atlantis Press, 1995: 186-189.

[21] Wang X Y, Yang J, Teng X L, et al. Feature selection based on rough sets and particle swarm optimization[J]. Pattern Recognition Letters, 2007, 28(4): 459-471.

[22] Skowron A, Bazan J, Son N H, et al. RSES 2.2 User's Guide[EB/OL]. [2005-01-19]. http://logic.mimuw.edu.pl/rses.

(责任编辑: 闫 妍)