第6卷第6期
1995年6月

软 件 学 报
JOURNAL OF SOFTWARE

Vol. 6, No. 6
Jun. 1995

归纳学习算法CAP2的研究与应用*

潘金贵 陈 彬 陈兆乾 陈世福

(南京大学计算机科学系,南京 210093)

摘要 本文提出以实例空间中状态划分概率的大小作为启发式信息,以提供的正反实例集为依据,基于二叉树分类方法的示例式归纳学习算法CAP2.它输出的分类规则是谓词演算表达式.该算法可根据用户对精度的要求控制分类深度,得到不同精度的规则,并能处理连续数据、噪音数据和利用用户提供的背景知识,既适用于同时给定概念的正、反例集的情况,也适用于只给正例集的情况.本文还介绍了CAP2算法的应用情况,并和著名的ID3算法进行了比较.CAP2已嵌入到一个自动知识获取系统.

关键词 归纳学习算法,划分概率,启发式信息.

从归纳中获取知识是人类获得新知识的极其重要的来源.归纳学习是指用实例作为问题的有限的部分描述(实例集不完备时),通过归纳得到问题的完整描述.归纳学习是机器学习领域中一个重要而被广泛研究的课题.

根据有无教师指导归纳学习又被分为示例学习和观察学习.示例学习是指教师提供基本概念的正例集与反例集,学习系统通过归纳推理产生覆盖所有正例并排斥所有反例的该概念的描述.当前在国际上有较大影响的示例学习算法有Quinlan的ID3算法[1]以及文献[2]中提到的另外几种算法.

ID3基于由实例集构造判定树的思想,是所谓TDIDT(Top Down Induction of Decision Trees)系统中的一个算法.该算法以其简明、易于实现以及学习效率高等优点得到了广泛应用,但其自身也存在一些缺陷,如生成的判定树不易理解和操作,冗余现象严重,实例集中必须含反例,不能处理连续数据等.针对其缺陷,Schlimmer、Fisher、P. E. Utgoff以及J. Cendrowska分别对其进行改进,提出了ID4、ID5、ID5R和PRISM等改进算法.我们提出的CAP1[2]算法也是这方面工作的有益实践.CAP1有其独特的优点,但同样存在很大的局限性.本文描述的CAP2算法是以实例空间中状态划分概率的大小作为启发式信息,以提供的正、反实例集为依据,基于二叉树分类方法的学习算法,它在CAP1基础上,在下面几个方面作了较大改进:

* 本文1993-09-10收到,1994-07-14定稿

本课题得到国家自然科学基金的资助.作者**潘金贵**,1952年生,副教授,主要研究领域为知识工程和机器学习.**陈彬**,1964年生,讲师,主要研究领域为专家系统,机器学习.**陈兆乾**,1940年生,副教授,主要研究领域为人工智能与机器学习.**陈世福**,1938年生,教授,主要研究领域为人工智能与机器学习.

本文通讯联系人:潘金贵,南京210093,南京大学计算机科学系

• 对搜索最佳划分状态的算法作了较大改进. CAP2 算法使用的划分概率计算公式以及其它计算公式,是经过大量实验比较后,选出的最优的一个.

• 可利用用户提供的背景知识,增强算法的归纳学习能力,同时也提高算法的效率.

• 可以处理噪音数据,引入了可信度因子来解决数据噪音问题.

• 能处理连续型数据.

从而使得 CAP2 算法具有简单、易用、计算效率高、实现代价小等性质. 该算法已成功地嵌入到一个自动知识获取系统应用于实际领域的知识获取.

1 CAP2 算法

CAP2 是以提供的实例集中状态划分概率的大小作为启发式信息、以提供的实例为依据、基于二叉树的分类方法. 它使用过程:

$induce(P_K, NP_K, Attr, List, Abstract, Left)$:推出公式 (1)

来实现. 式(1)中:

P_K:表示正例集,下文中凡是出现带足码或不带足码的 P 如无特别说明均表示正例集.

NP_K:表示反例集,下文中凡是出现带足码或不带足码的 NP 如无特别说明均表示反例集.

$Attr$:为属性集.

$List$:为一数组变量,例如,$List[i](i \in Attr)$表示属性 i 可能取的值集.

$Abstract$:表示划分过程中使用过的抽象属性值集.

$Left$:是一布尔量,取值 *true* 表示左子树,取值 *false* 表示右子树或根,引入它是为了便于对不同的情况进行统一处理.

在给出算法描述之前,下面先对使用的一些概念进行定义.

1.1 几个定义

定义 1. 状态和状态空间

状态是指一个属性—属性值对,可表示为$\langle i, j\rangle$,所有状态组成的集合称为状态空间.

定义 2. 推出公式

设 Pr 为用户提供的精度,属性表达式 S 为推出公式,当且仅当:

$$(\forall\, x \in P(x \Rightarrow S)) \wedge (\forall\, y \in NP(\rightarrow(y \Rightarrow S))) \quad 或 \quad Pr(S) \geqslant Pr$$

定义 3. 推出公式的精度

设 $|P'|$ 是属性表达式 S 覆盖的正例个数,$|NP'|$ 表示 S 排斥的反例个数,N 为 $|P|$ 和 $|NP|$ 之和,则推出公式的精度 $Pr(S)$ 定义为:

$$Pr(S) = (|P'| + |NP'|)/N$$

定义 4. 属性表达式的可信度和分类树 T_K 的精度

设生成属性表达式 S_1 的分类树所对应的叶结点中,含有 P_1、NP_1,则 S_1 的可信度,记为 CF,定义为:

$$CF = |P_1|/(|P_1| + |NP_1|)$$

由前面定义的推出公式的精度可得出分类树 T_K 的精度:

$$Pr_K=(|P_K'|+|NP_K'|)/N \tag{2}$$

其中，$|P_K'|$和$|NP_K'|$的含义同定义 3. 式(2)也可用式(3)的一个等同公式来计算：

$$Pr_k=1-(P_C+C+NP_C+C')/N \tag{3}$$

其中 P_C 是一个统计数，表示迄今为止的划分过程中，由于右子树被剪枝而抛弃的正例个数；NP_C 也是统计数，表示已停止划分的结点中所含反例个数；C 是对 T_K 的划分中不覆盖的正例个数，即 $C=T_{2K+1}$中正例个数；C'为对 T_K 的划分中覆盖的反例个数，即 $C'=T_{2K}$中反例个数.

分类树 T_K 的精度决定了是否要对 T_K 继续划分下去，若 Pr_K满足用户给定的精度要求，则无需再划分，反之不然.

定义 5. 分类树 T_K 的划分因子

设 D_k 为分类树 T_K 的划分因子，则有定义：

$$D_K=1-(P_C+NP_C+C)/N \tag{4}$$

事实上，T_K 经过一次划分形成两棵子树 T_{2K}，T_{2K+1}，是否需要继续对右子树 T_{2K+1}进行划分，则是由划分因子 D_K决定的. 如果 D_K满足用户给定的精度，则可以剪枝，否则不可剪枝.

1.2　算法描述

CAP2 算法在状态空间中搜索一个最佳状态$\langle i,j\rangle$(最佳状态即划分概率最大的状态，划分概率的定义见定义 7)，以该状态对 P_k 和 NP_K进行分类，得到两棵子树 $T_{2K}(P_{2K},NP_{2K})$和 $T_{2K+1}(P_{2K+1},NP_{2K+1})$，即将 P_K 划分成 P_{2K}、P_{2K+1}，NP_K 划分成 NP_{2K}和 NP_{2K+1}，使$\langle i,j\rangle$对应的表达式$[i,j]$满足：

$$\forall\ x\in P_{2K}(X\Rightarrow[i,j]),\qquad \forall\ x\in NP_{2K}(X\Rightarrow[i,j]);$$
$$\forall\ x\in P_{2K+1}(\neg(X\Rightarrow[i,j])),\quad \forall\ x\in NP_{2K+1}(\neg(X\Rightarrow[i,j])).$$

接着再对 T_{2K}和 T_{2K+1}进行同样的划分，得到另一状态，递归进行上述的分类过程，直至无法再划分(或已达到用户的精度). 上述分类的输出是一个满足完备性和一致性逻辑条件的推出公式，该推出公式可很容易地转换为相应的规则集.

实现时，对式(1)初始化：$Attr$＝知识元 A 的属性集，$Abstract$＝空，对任一的 $i\in Attr$，$List[i]$定义为属性 i 的最外层的抽象值及那些无相应抽象值的观察属性值，即如果属性 i 是结构化属性，则 $List[i]$＝{第一层抽象值及无相应抽象值的观察属性值}，若属性 i 为非结构化的，则 $List[i]$＝{属性 i 的观察属性值集}. $Induce$ 过程根据“划分概率最大原则”，选择最佳状态$\langle i,j\rangle$，其中 $i\in Attr$，$j\in List[i]$，对实例进行划分，并调整相应的 $Attr$、$List$、$Abstract$ 值，方法如下(假设从 T_K 划分成 T_{2K}、T_{2K+1})：

$$\left.\begin{aligned}
&Attr_{2k}=Attr_k-\{i\},\qquad Attr_{2K+1}=Attr_{2K}\\
&List_{2K}[i]=List_k[i]-\{j\},\qquad List_{2K+1}[i]=List_K[i]-\{j\},\\
&Abstract_{2K}=\begin{cases}Abstract_k+\{j\}; & \text{若 } j \text{ 为抽象属性值}\\ Abstract_k; & \text{其它}\end{cases}\\
&Abstract_{2k+1}=Abstract_K
\end{aligned}\right\} \tag{5}$$

其余的值继承其父结点的值. 在划分过程中可能出现下面两种情况：

P_K、NP_K 均不为空，但 $Attr$＝空，这时我们看 $Abstract$ 是否为空，若为空，则看作是数据噪音的情况，引入定义 4 中定义的 CF 解决；

若不为空,则表示划分过程作过普化,可从 *Abstract* 中选出最后一个值 j,设其相应的属性为 i,令 $A=\{i\}$,重新做 T_K 的划分.

现用类 Pascal 语言对 CAP2 算法描述如下:

```
Induce{(P_K,NP_K,Attr_K,List_K[Attr],Abstract_K,Left):推出公式};
begin
  if (Left=true) and (NP_K=Φ)
  then induce:=NULL;
  if (Left=false) and (P_K=Φ)
  then induce:=NULL;
  if (Attr_K=Φ) and (Abstract=Φ) and (|P_K|≠0) and (|NP_K|≠0)
  then CF=|P_K|/(|P_K|+|NP_K|);      /*引入CF处理噪音*/
  induce:=NULL;
  if (Atrr_K=Φ) and (Abstract_K≠NULL) and (|P_K|≠Φ) and (|NP_K≠Φ|)     /*处理过普化情况*/
  then begin
         计算 Attr_K=\{i\};Abstract_K=Abstract_K-\{j\};
         /*j为Abstract_K中最后一个元素,i为属性值j相应的属性*/
       end;
  搜索最佳划分状态⟨i,j⟩:    /*执行算法BS*/
  把 T_K 划分为 T_2K及 T_2K+1;
      P_2K={x:x∈P_K 且⟨i,j⟩∈x};
      NP_2K={x:x∈NP_k 且⟨i,j⟩∈x};
      P_2K+1=P_K-P_2K;
      NP_2K+1=NP_K-NP_2K;
  计算 C=T_2K+1中正例个数;
       C'=T_2K+1中反例个数;
       Pr_k=1-(P_C+NP_C+C+C')/N;
  if (Pr_k≥P_r)     /*检查Pr_k是否满足用户精度要求Pr,若满足,则停止划分;否则继续划分*/
  then begin
       NP_C=NP_C+C';
       induce:=δ;      /*δ为状态⟨i,j⟩对应的属性表达式,下文同*/
       end;
  根据式(5)计算:
  Attr_2K,Attr_2K+1,Abstract_2K,Abstract_2K+1,List_2K,List_2K+1;
  计算 D_K=1-(P_C+NP_C+C)/N   /*划分因子D_K决定是否要对右子树剪枝*/
  if (D_K≥Pr)
  then induce:=induce(P_2K,NP_2K,Attr_2K,Abstract_2K,List_2K,true)∧δ
  else induce:=induce(P_2K,NP_2K,Attr_2K,Abstract_2K,List_2K,true)∧δ∨
               induce(P_2K+1,NP_2K+1,Atrr_2K+1,Abstract_2K+1,List_2K+1,false)∧not δ;
end;
```

为了描述搜索最佳划分状态的算法 *BS*,现引入下面两个定义.

定义 6. 在 $T_K(P_K,NP_K,Attr_K,List_K,Abstract_K)$ 的划分过程中,对任意 $\langle i,j\rangle$ 且 $i\in Attr_K, j\in List[i]$,

有定义:

$$n(\langle i,j\rangle,P_K)=|\{x:x\in P_K \text{ 且 } \langle i,j\rangle\in x\}|;$$
$$\bar{n}(\langle i,j\rangle,P_K)=|P_K|-n(\langle i,j\rangle,P_K);$$
$$n(\langle i,j\rangle,NP_K)=|\{x:x\in NP_K \text{ 且 } \langle i,j\rangle\in x\}|;$$
$$\bar{n}(\langle i,j\rangle,NP_K)=|NP_K|-n(\langle i,j\rangle,NP_K);$$
$$OB\_Set=\{\langle i,j\rangle:n(\langle i,j\rangle,P_K)\neq 0 \text{ 且 } n(\langle i,j\rangle,NP_K)=0\}$$

事实上,$n(\langle i,j\rangle,P_K)$表示状态$\langle i,j\rangle$在 P_K 中出现的次数,$n(\langle i,j\rangle,NP_K)$表示$\langle i,j\rangle$在 NP_K 中出现的次数,因而 OB_Set 定义的是只在 P_K 中出现而不在 NP_K 中出现的状态的集合.

定义 7. 划分概率

设 $LP(\langle i,j\rangle,T_K)$为状态$\langle i,j\rangle$在 T_K 上的划分概率,则有定义:

$$LP(\langle i,j\rangle,T_K)$$
$$=\begin{cases}0; & \text{当 } n(\langle i,j\rangle,P_K)=0 \text{ 且 } n(\langle i,j\rangle,NP_K)=0 \text{ 时}\\ \dfrac{n(\langle i,j\rangle,P_K)^2\times(\bar{n}(\langle i,j\rangle,P_K)+\bar{n}(\langle i,j\rangle,NP_K)}{(|P_K|+|NP_K|)^2\times(n(\langle i,j\rangle,P_K)+n(\langle i,j\rangle,NP_K))\times(\bar{n}(\langle i,j\rangle,P_K)+1)}; & \text{其他}\end{cases}$$

$LP(\langle i,j\rangle,T_K)$是一个经验公式,以此作为评价函数可搜索到最佳划分状态.

有了上述定义,搜索最佳划分状态的算法 BS 可描述如下.

算法初始化:对任意$\langle i,j\rangle$,$i\in Attr_K$,$j\in List_K[i]$,计算 OB_Set,$LP(\langle i,j\rangle,T_K)$.

case1. $OB\_Set\neq\Phi$,则所求$\langle i,j\rangle$满足$\langle i,j\rangle\in OB_Set$,且对任意的$\langle s,t\rangle\in OB_Set$ 有:

$$n(\langle s,t\rangle,P_K)\leqslant n(\langle i,j\rangle,P_K)$$

case2. 若 $OB\_Set=\Phi$,则所求$\langle i,j\rangle$满足:$i\in Attr_K$,$j\in List_K[i]$,且对任意状态$\langle s,t\rangle$,$s\in Attr_K$,$t\in List_K[s]$,有:

$$LP(\langle s,t\rangle,T_K) < LP(\langle i,j\rangle,T_K)$$

或

$$LP(\langle s,t\rangle,T_K) = LP(\langle i,j\rangle,T_K)$$

且

$$n(\langle s,t\rangle,T_K) < n(\langle i,j\rangle,T_K).$$

1.3 CAP2 算法的性质

归纳学习系统是从外界提供的信息中归纳出知识,这就要求归纳出的知识相对所提供的信息具有完备性和一致性[3].

定理 1. CAP2 算法从给定实例集归纳生成的指定概念的规则集是完备的(当用户给的精度是 100%时). 即:对于给定的正例集 P_C 和归纳生成的规则集 H_C,有对任意的 $e\in P_C$,存在 $D\in H_C$,$e\Rightarrow D$.

证明:设 A 为一属性表达式,$example$ 为 P_C 的任一正例,学习算法开始时 $A=$空. 假设每次算法从状态空间搜索到的最佳划分状态为$\langle i,j\rangle$,其相应的属性表达式为$[i,j]$,且$\langle i,j\rangle$把 T_K 划分为 T_{2K}及 T_{2K+1}. 若 $e\in T_{2K}$,则 $A_{2K}=A_K\wedge[i,j]$,显然,对任意的 $e\in P_{2K}$有 $e\Rightarrow A_{2K}$. 算法终止时(该算法对实例空间经过多次划分后,必然终止),一定是下面情况:

$example$ 在树的叶结点 T_P 中出现,且 $P=2m$,$NP_p=$空,显然 $A_P\in H_C$,$example\Rightarrow A_P$.

注意:当用户给的精度<100%时,可能出现 $example\in T_{2m+1}$,但因 T_{2m+1}被截枝,无规则属于 H_C 能覆盖 $example$,这是因用户精度要求造成的,并不与完备性矛盾.

定理 1 得证.

定理 2. CAP2 算法从给定的实例集中归纳生成的指定概念的规则集是一致的. 即:对

于给定的正例集 P_C、反例集 NP_C 及归纳生成的规则集 H_C，满足：

$$\forall e \in NP_C \forall D \in H_C((\rightarrow(e \Rightarrow D)) \text{或} (e \Rightarrow D) \text{且} e \in P_C)$$

证明：用反证法证明. 设 e 是 NP_C 中任一元素，学习算法终止时，若存在一个 $D \in H_C$，且 D 满足 e，则 e 一定被划分在分类树的某个左子树上. 又因算法终止的条件是左子树中均是正例，或右子树中均是反例，或有噪音时，左子树中是相同前提但不同分类的正、反例.

case1. 此左子树中均是正例，则 $e \in P_C$，与 $e \in NP_C$ 矛盾，得结论：$\rightarrow(e \Rightarrow D)$；

case2. 此左子树中是噪音数据，则 $e \in NP_C$，结论成立. □

1.4 CAP2 算法与 ID3 算法的比较

CAP2 算法与 ID3 算法都是以实例为基础的树分类算法，但各有特点，现列表作比较如下：

表 1　CAP2 算法与 ID3 算法比较

算法	对实例的要求	属性的要求	背景知识	结果形式	分类基础	结束条件
CAP2	可无反例 可有噪音	离散、连续型 数据均可	有	一阶谓词表达式	基于划分概率	满足用户 精度要求
ID3	要有反例 无噪音	离散型数据	无	判定树	基于熵(entroy)	结点中实例 属同一类

此外，我们还以目前较流行的 Marlon Nűnez 的分类问题 MT[4]、Quinlan 关于人群分类问题 MAN[1]以及胆结石判定问题 GS 为例，分别用 CAP2 算法及 ID3 算法进行规则归纳，得到表 2 的结果(是在 PC 386sx/20 微机上运行的).

表 2　CAP2 算法与 ID3 算法在实例应用中的比较数据

实例	实例数		分类树结点数		规则覆盖的最多实例数		规则覆盖的最少实例数		总规则数		运行时间(秒)	
	正	反	CAP2	ID3	CAP2	ID3	CAP2	ID3	CAP2	ID3	CAP2	ID3
GS	18	30	25	22	17(1)	17(1)	1(4)	1(6)	11	13	16.04	15.33
MAN	3	5	7	6	3(1)	3(1)	1(1)	1(1)	4	4	1.04	0.98
MT*	7	5	9	15	4(1)	2(1)	1(1)	1(10)	5	11	3.90	2.91
MT	7	5	13	15	4(1)	2(1)	1(3)	1(10)	7	11	3.20	2.91

注：小括号中的数据表示相应的规则数目

2 CAP2 算法的应用

CAP2 算法已嵌入归纳学习系统 NDKAS 中作为核心学习算法，并成功地用于“新构造控水专家系统[5]”的知识库的自动构造.

根据“新构造控水理论”，一个地质带是否富含地下水资源取决于该地区断裂带的方向性，扭动性等 10 多个属性构成的地质特征，而每个属性表现的地质特征(属性值)又呈现复杂的多样性. 例如：

断裂带有一定的切割性(属性)呈现如下特征：

①北西西走向的断裂切割所有的断裂;

②北西西走向的断裂切割北北东走向的断裂;

③北东东走向的断裂切割北北西走向的断裂;

④北东东走向的断裂切割石炭岩及其中的侵入岩和矿脉;

⑤其它特征.

每个找水实例都是由这些属性及其属性值构成的一阶逻辑描述.CAP2 算法从上百条这些专家提供的找水实例中归纳出 20 多条经验规则,这些经验规则满足一致性和完备性条件,并经专家认定是完全正确的规则.

从应用的结果来看,我们发现:

(1)随着实例集增大,归纳出的经验规则集趋于正确,而此时的实例集只是实例空间中很小一部分.

(2)算法计算时间随着实例集增大呈线性增长.

为进一步了解该算法的应用,下面给出用 CAP2 算法处理 Marlon Nűnez 关于分类问题的示例[5].

首先,用概念描述语言 CL,对理论 plus_class 描述如下:

```
1  domain theory declaration
2   theory plus_class_m
3    universe
4      concept Forma is nominal (square,triangle,circle,ellipse);
5      concept Color is nominal (red,blue,yellow,pink);
6      concept Size is nominal (big,small,medium);
7      conoept Material is nominal (metal,plastio,leather,wood)
8  end;
9      isa_relation
10       Color|(red,blue,yellow:PRIMARY);
11       Size|(big,medium:NOT_SMALL big,small:NOT_MEDIUM);
12       Forma|(square,triangle:POLYGON # circle,ellipse:CONIC)
13       end
14   end
15   end;
16   example descriptions
(1) plus_class:-Forma=square and Color=red and Size=big and Material=metal;
(2) plus_class:-Forma=square and Color=blue and Size=small and Material=plastic;
(3) plus_class:-Forma=triangle and Color=yellow and Size=medium and Material=metal;
(4) plus_calss:-Forma=ellipse and Color=blue and Size=big and naterial=leather;
(5) plus_class:-Forma=ellipse and color=pink and Size=medium and Material=wood;
(6) plus_class:-Forma=circle and Color=blue and Size=big and Material=wood;
(7) plus_class:-Forma=triangle and Color=blue and Size=medium and Material=plastic;
(8) not plus_class:-Forma=triangle and Color=pink and Size=big and Material=leather;
(9) not plus_class:-Forma=medium and Color=pink and Size=medium and Material=leather;
(10) not plus_class:-Forma=circle and Color=red and Size=small and Material=plastic;
(11) not plus_class:-Forma=circle and Color=blue and Size=small and Material=metal;
```

(12) not plus_class:—Forma=ellipse and Color=yellow and Size=small and Material=plastic

end;

goal

learn concept(plus_class);

learn metarule()

end.

设用户给的精度为90%,

行1—15描述的是背景知识,即图1中所示 IS_A 关系:

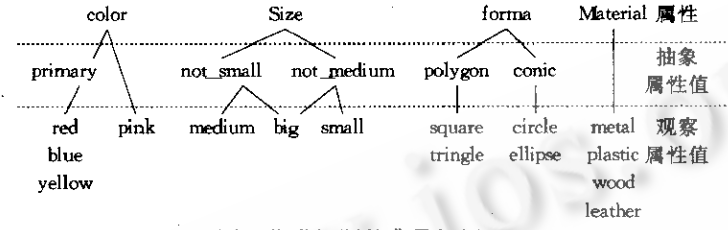

图1 分类问题的背景知识

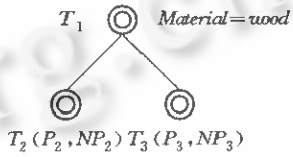

图2

行15以后描述的即为分类问题的有关实例.

初始时,$Pc=0, NP_C=0, Left=false$;

$P_1=\{1,2,3,4,5,6,7\}, NP_1=\{8,9,10,11,12\}$;

$Attr=\{Shape, Color, Size, Material\}$;

$Abstract=$空;

$List[Shape]=\{POLYGON, CONIC\}$;

$List[Color]=\{PRIMARY, pink\}$;

$List[Size]=\{NOT\_SMALL, NOT\_MEDIUN\}$;

$List[Material]=\{metal, plastic, leather, wood\}$;

$Induce(P_1, NP_1, false)$;

第1次划分:$P_1\neq$空,求最佳划分状态$\langle i,j\rangle$,由 BS 算法得:

$$OB\_Set=\{\langle Material, wood\rangle\},$$

此时,则$\langle Material, wood\rangle$即为最佳划分状态,$T_1$被划分成 T_2及 T_3,如图2所示.

其中,$P_2=\{5,6\}, NP_2=$空,$P_3=\{1,2,3,4,7\}, NP_3=\{8,9,10,11,12\}$,又由于 $P_C=0, NP_C=0$,得 T_1的精度:

$$Pr_1=1-(P_C+NP_C+C+C')/N=1-5/12=58.3\%<Pr$$

T_1要继续划分下去.因为划分因子:$D_1=1-(P_C+NP_C+C)/N=58.3\%<Pr$ 不能截枝,则:

$$Induce=\{(Material=wood)\wedge Induce(P_2, NP_2, true)\},$$

$$not(Material=wood)\wedge Induce(P_3, NP_3, false)\}$$

第2次划分:$NP_2=$空,得 $Induce(P_2, NP_2, false)$,=空;求 $Induce(P_3, NP_3, false)$,$P_3\neq$空,所以找最佳划分状态$\langle i,j\rangle$,调用过程 BS 得:$OB_Set=$空,与 $case2$情况相对应,求得:$Lp((Color, PRIMARY))$最大,为0.0625,则$\langle Color, PRIMARY\rangle$即为所求状态,对 T_3进行划分得 T_6, T_7(见图3):

其中,$P_6=\{1,2,3,4,7\}, NP_6=\{10,11,12\}, P_7=$空,$NP_7=\{8,9\}, C=|P_7|=0, C'=|NP_6|$

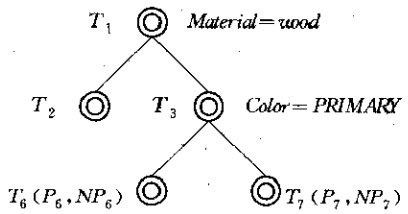

图3

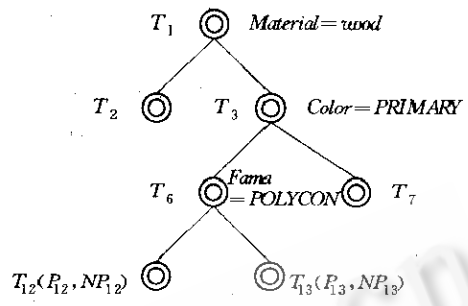

图4

$=3$，$P_C=0$，$NP_C=0$，可计算出 T_3 的精度：$Pr_3=1-(P_C+NP_C+C+C')/N=75\%<Pr$，$T_3$ 还需划分下去. 又因：

$$D_3=1-(P_C+NP_C+C)/N=100\%>Pr,$$

所以 T_7 可截枝，$P_C<-C$，则：

$Induce=\{(Material=wood),not(Material=wood)\wedge(Color=PRIMARY)\wedge Induce(T_6)\}$.

第3次划分：$P_6\neq$空，所以找最佳划分状态$\langle i,j\rangle$，调用过程 BS 得：

$OB_Set=$空，与 $case2$ 情况相对应，求得：$Lp(\langle Forma,POLYGON\rangle)$最大，为0.5，则$\langle Forma,POLYGON\rangle$即为所求状态，对 T_6 进行划分得 T_{12}，T_{13}（见图4）：

其中，$P_{12}=\{1,2,3,7\}$，$NP_{12}=\{4\}$，$P_{13}=$空，$NP_{13}=\{10,11,12\}$，$C=|P_{13}|=0$，$C'=|NP_{12}|=1$，$P_C=0$，$NP_C=0$，可计算出 T_6 的精度：

$Pr_6=1-(P_C+NP_C+C+C')/N=91.7\%>Pr$，故 T_6 停止划分.

$$Induce=\{(material=wood),not\ (Material=wood)\wedge(Color=PRIMARY)\wedge(Forma=POLYGON)\}$$

综合上述3次划分得规则集：

if $Material=wood$ then $plus\_class$

if not $(material=wood)$ and $Color=red\ or\ blue\ or\ yellow$ and $Forma=square\ or\ triangle$ then $plus\_class$

若用户所给的精度不是90%，而是100%，则 T_6 还要继续划分下去，所得推出公式为：

if $Material=wood$ then $plus\_class$

if not $(Material=wood)$ and $Color=red\ or\ blue\ or\ yellow$ and $Forma=square\ or\ triangle$ then $plus\_class$

if not $(Material=wood)$ and $Color=red\ or\ blue\ or\ yellow$ and $not(Forma=square\ or\ triangle)$ and $Size=big$ then $plus\_class$

由此例可看出，CAP2算法能从给出的实例集中直接归纳出需要的规则，且这些规则满足完备性和一致性条件.

3 结　论

本文提出的CAP2算法，是在CAP1的基础上作了较大的改进，并对它使用的搜索最佳

划分状态的算法以及多种计算公式进行了大量实验比较后优选出来的，具有简单、实用、计算效率高、实现代价小等良好特性，具体说，有如下特点：(1)该算法把正例集作为分类的依据，同时考虑反例集的影响，不仅适用于给定概念的正、反实例集的情况，也适用于只给概念的正例集的情况.(2)可根据用户对精度的不同要求控制分类深度，得到不同精度的规则集.(3)该算法输出的规则是谓词演算表达式，这种逻辑形式的规则表达能力强.(4)能处理噪音数据和连续数据.它通过引入可信度因子的方法来解决数据的噪音问题.(5)可以直接使用用户显式提供的启发式信息—背景知识，从而大大提高概括程度和时空效率.

该算法已嵌入到归纳学习系统NDKAS中作为其核心学习算法，成功地应用于"新构造控水专家系统"知识库的自动构造，取得了满意的效果.

参考文献

1 Quinlan J R. Learning efficient classification procedures and their application to chess and games. In: Michalski R S, Garbonell J, Mitchell T M eds, Machine Learning: an Artificial Intelligence Approach, Palo Alto. CA: Tioga Press, 1983.

2 陈世福，潘金贵，陈彬等.一种概念获取算法CAP及其应用.计算机学报，1991，**14**(**8**)：586－595.

3 Michalski R S. A theory and methodology of inductive learning. In: Michalski R S, Garbonell J, Mitchell T M eds, Machine Learning: an Artificial Intelligence Approach, Palo Alto. CA: Tioga Press, 1983.

4 Marlon Nunez. The use of background knowledge in decision tree induction. Machine Leaning, 1991, **6**(**3**): 1201－1210.

5 陈世福，潘金贵等.勘探地下水专家系统NCGW的设计与实现.计算机学报，1989，**12**(**6**)：452－457.

THE RESEARCH AND APPLICATIONS ON THE ALGORITHM CAP2 FOR INDUCTIVE LEARNING

Pan Jingui Chen Bin Che Zhaoqian Chen Shifu

(*Department of Computer Science, Nanjing University, Nanjing* 210093)

Abstract In this paper, an inductive learning algorithm CAP2 is described, which is a binary-tree classification method, with the division probability of state space as heuristic information, based on the provided example set. CAP2 can satisfy the requirement of precision of users to induce the rule set of the proper precision, and it can deal with continuous data, noisy data. Background knowledge provided by users can be used by CAP2. CAP2 has been successfully applied in real world. Also its comparison with ID3 is given.

Key words Inductive learning algorithm, division probability, heuristic information.