

一种具有知识评估和知识求精的知识获取(KER)算法*

杨 莉 胡守仁

(计算机系)

摘 要 本文给出了一种获取多类知识的决策树算法,该算法根据所给定的属性的优先级和取值类型进行分类知识的获取。为了保证获得知识的有效性,根据科恩(Cohen)的归纳概率提出了一种证据支持程度来对所获得的知识进行评价,并相应地给出了一种知识求精的方法。

关键词 人工智能,知识获取,算法/决策树,知识评估,知识求精,归纳概率

分类号 TP311.13, TP18

智能系统要求拥有大量专门技术领域的知识,即需要知识获取。如何使智能系统具有知识获取能力呢?我们从人类获取知识的方法中可获得一些启发。人类在生产和科学实践中,往往对大量的事实进行观察记录,通过分析这些分散的甚至粗看起来十分杂乱的观察现象或数据来得出一般的规律,即人类获取知识的过程是一个归纳推理过程。为了使智能系统具有归纳推理以获取知识的能力,就是要设计出种种归纳推理算法,以使智能系统能使用这些算法,从大量的分散数据中总结出规律。由于归纳推理是由已观察到的事件过渡到尚未观察到事件的推理,即:

已观察到 e_1, e_2, \dots, e_n 是 c 和 p

没有观察到任何的 c 不是 p

\therefore 所有的 c 都是 p .

* 此项研究由国家自然科学基金资助
1989年11月5日收稿

in tightly-coupled multiprocessor system is constructed. The analytic expression of memory utility bandwidth is given by analysing the busy period of a bank. The relationship among the rate λ of the accessing memory of processors, the serving rate μ of a bank and the number m of banks is studied under the condition of keeping the memory utility bandwidth maximum.

Key Words memory, queueing theory, model/utility bandwidth, queue model

当已知 e_1, e_2, \dots, e_n 是柠檬, 并且是酸的, 且未曾尝过不酸的柠檬时, 按上述推理, 可知所有的柠檬都是酸的。由此可见, 归纳推理的结论并不逻辑地被前提所蕴涵, 结论的范围超出了前提, 因而归纳的结论不可能是完全正确的。因此, 我们应该把归纳推理过程看成是一个不断检验所获得知识的过程, 归纳过程应能够根据检验结果对知识进行评价。归纳评价不同于演绎证明。在演绎法中, 前提和结论之间存在着逻辑蕴涵或推导关系, 前提真, 结论便不可能是假的。但在归纳推理中, 假设(即获得的知识)只能从证据中获得某一程度的支持, 前提(证据或事实)和假设之间的证据支持关系表示了假设的可信程度。当归纳获得的知识具有很小的可信度时(如小于某一临界值), 应能够对此知识进行修正, 以保证所获得知识的当前合理性(所谓当前是指在目前知识集的背景下来验证所获取的知识)。

根据上述观点, 本文给出了一种获取多类知识的决策树算法与相应的知识评估和求精方法, 旨在为分类型智能系统的知识获取问题提供一条可行的途径。

1 KER算法中决策树的生成

KER算法直接从一组数据获取知识并将获取的知识表达成全称量词化的一阶公式形式。预先给定的数据是一个对象的集合, 其中每个对象具有多个属性, 每个属性表示对象某一方面的特性。属性有值, 并属于下列四种类型之一: (1) 名词类型; (2) 枚举类型; (3) 数值类型; (4) 按层次组织的结构类型。

在KER算法中, 每个对象被赋予一个类值, 因此, 我们把类值看作为对象的一个特殊属性, 设为第一属性。由已知的对象所形成的集合称为训练集。在用KER算法对训练集进行分类以获得知识时, 若不存在属于不同类但其每个属性值均相同的两个对象时, KER算法是行之有效的。在通过训练集形成决策树时, 必须选择一个属性作为决策树的树根, 然后根据该属性的值把训练集分割成一些子集合; 接着再对每个子集合分别选择另外的某个属性作为子树的根, 对每个子集合以同样的方法分别进行分割; 这个过程持续进行直至每个子集合中的对象都具有相同的类值。由于选用不同的属性做树根或子树根时所构造出来的决策树的复杂情况不同, 故将一训练集划分的代价不同。为了正确地给对象分类而又使决策树尽可能地简单, 我们采用信息论中熵的概念来选择决策树的根及子树的根。

设每个属性 X_j 的优先级为 C_j , 训练集 $T = \{PE_1, PE_2, \dots, PE_n\}$, 其中 PE_i 表示用以获取第 i 条知识的对象集。由信息论可知, 如果一条消息的内容为真的概率为 q , 则这条信息所包含的信息量为 $\log_2\left(\frac{1}{q}\right)$ 。所以, 如果有 p_1 个对象属于 PE_1 类, \dots , p_n 个对象属于 PE_n 类, 那么“一个对象属于 PE_1 类或, \dots , 或属于 PE_n 类”的消息包含的平均信息量为:

$$I(p_1, \dots, p_n) = -\frac{p_1}{p_1 + \dots + p_n} \log_2 \frac{p_1}{p_1 + \dots + p_n} - \dots - \frac{p_n}{p_1 + \dots + p_n} \log_2 \frac{p_n}{p_1 + \dots + p_n}$$

在选择某个属性作为决策树的根或子树的根时, 单有上述的熵 $I(p_1, \dots, p_n)$ 是不够的, 必须知道当选择了某个属性作为决策树或其子树的根的条件下熵的值。设属性 X_i 具有属性值 X_{i1}, \dots, X_{iw} (当 X_i 为连续型属性时, 我们设定属性 X_i 的取值区间为 (A_{ij}, B_{ij}) , $j=1, \dots, w$, 而 X_i 可取这些区间中的任一值), 且对于任一 X_{ij} (或 (A_{ij}, B_{ij})), $j=1, \dots, w$, 训练集 T 中的 PE_1, \dots, PE_n 满足 X_{ij} (属于 (A_{ij}, B_{ij})) 的对象数为 N_{ij1}, \dots, N_{ijn} , 那么以 X_i 为树根时有关“一个对象属于 PE_1 类或, \dots , 或属于 PE_n 类”的消息包含的平均信息量即熵为:

$$E(X_i) = \sum_{j=1}^w \left[\frac{(N_{ij1} + N_{ij2} + \dots + N_{ijn})}{p_1 + p_2 + \dots + p_n} - I(N_{ij1}, \dots, N_{ijn}) \right]$$

由于每个属性对分类的贡献程度不同, 故当选择 X_i 为树 (或子树) 的根时, 它所需要的信息量 $E'(X_i) = E(X_i)/C_i$ 应为最小。例如: 表 1 是军事专家系统中关于战斗编队方法的记录, 其类值为 queue1, queue2, queue3, 分别表示战斗编队方法 1, 方法 2, 方法 3。每种战斗编队方法通过下列三个属性来表征: (1) 敌情(enemysituation), (2) 进攻正面(breadth), (3) 进攻纵深(depth)。敌情的取值为 1 和 2; 其优先级为 3, 进攻正面的优先级为 2, 其值域为: $(-\infty, 3]$; $(3, 6]$; $(6, 10]$; $(10, 15]$; $(15, \infty)$, 进攻纵深的优先级为 1, 其值域为 $(-\infty, 3]$; $(3, 6]$; $(6, 10]$; $(10, 15]$; $(15, \infty)$ 。根据计算可知: $E'(\text{enemysituation}) < E'(\text{breadth}) < E'(\text{depth})$, 故对于表 1 可得图 1 的决策树。

表 1

echelon	enemysituation	breadth	depth
queue2	1	9.77	12.58
queue3	2	6.3	10.13
queue2	1	7.75	7.67
queue1	1	11	7.31
queue2	1	9.87	5.94
queue2	1	7.83	14.05
queue2	1	7.35	12.09
queue1	1	14.45	11.47
queue2	1	8.7	3.05
queue2	1	9.13	12.56
queue1	1	12.21	7.1
queue2	1	8.4	4.39
queue1	2	13.71	4.1
queue3	2	6.67	12.21
queue3	2	5.19	12.97
queue1	1	14.16	6.97
queue2	1	7.94	4.2
queue1	1	10.56	8.33

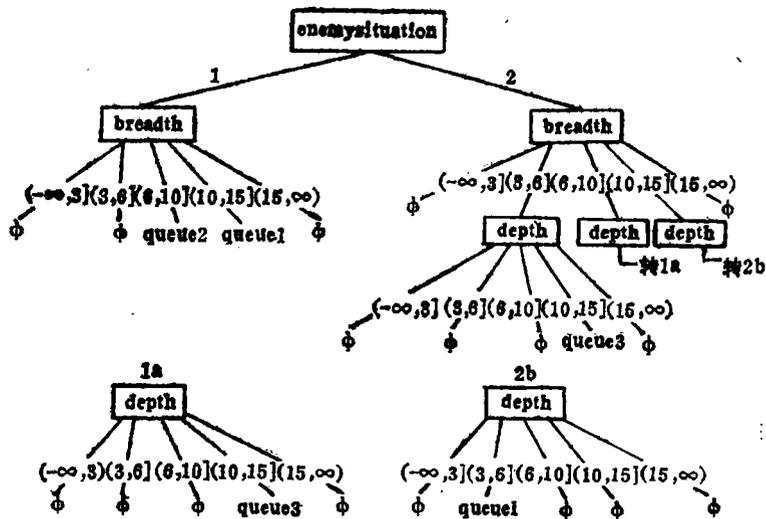


图 1

根据图 1 的决策树，可形成下列全称量词化的一阶公式：

$$(enemysituation = 1) * (10 < breadth \leq 15) \tag{1}$$

$$V(enemysituation = 2) * (10 < breadth \leq 15) * (3 < depth \leq 6) \rightarrow queue1 \tag{2}$$

$$(enemysituation = 1) * (6 < breadth \leq 10) \rightarrow queue2 \tag{3}$$

$$(enemysituation = 2) * (3 < breadth \leq 6) * (10 < depth \leq 15) \tag{4}$$

$$V(enemysituation = 2) * (6 < breadth \leq 10) * (10 < depth \leq 15) \rightarrow queue3 \tag{5}$$

由此可见，我们所提出的决策树算法不仅可对具有离散值（可枚举）属性的知识进行自动获取，也可对具有连续性（不可枚举）属性的知识进行自动获取，从而扩大了知识获取的范围。

2 KER算法中知识的有效性检验及评估

在KER算法中进行知识获取的算法是根据给定的训练集进行分类知识的获取，其归纳推理过程是由给定的一组对象（单称命题）得到一组一般的知识（全称命题）。因此，归纳结论的正确与否与系统中当前知识集的大小有关。但无论进行归纳推理的训练集有多大，也无论对所得到的归纳结论进行多少次反复检验，都不能证明其为真或为假。因而由KER进行知识获取所得到的知识只能从证据（实例）中获得某一程度的支持和指明它在当前知识集中的可信程度。

在KER中，我们根据科恩的归纳概率提出了一种证据支持程度来对所获得的知识进行评价。下面我们对此方法进行简单的介绍。

科恩的归纳概率建立在可检验的全称检验句 $\forall x_1, \dots, x_n (A(x_1, \dots, x_n) \rightarrow B(x_1, \dots, x_n))$ 的归纳支持的基础上，但什么是全称检验句 $H = \forall x_1, \dots, x_n (A(x_1, \dots, x_n) \rightarrow B(x_1, \dots, x_n))$ 的归纳支持呢？科恩采用相关变量法，简单地说，如果 H 有 n 个相关变量，在检验中通过了 i 个相关变量，则说 H 有 i/n 级归纳支持（这里 i/n 不是分数，

只是这种等级的一个标志)。我们把科恩的这一思想引入KER算法,即把影响全称检验句成立与否的因素称为属性,如敌情,进攻正面等。一个属性由一些属性值所组成,如敌情 $=\{1,2\}$ 。

现在可以对全称检验句 H 的相关属性(即属性)进行检验。如果 V_1 是 H 的属性,则对 V_1 进行检验是指对 V_1 的每个属性值分别进行检验。如果 V_1 和 V_2 都是 H 的相关属性,则对 $\{V_1, V_2\}$ 进行检验是指对 V_1 和 V_2 中各取一个属性值的所有可能的组合进行检验。对于 H 的任何一组有限的相关属性,对其检验的意义是类似的。

设 A 是 H 的一组相关属性,如果在对 A 的检验中没有找到 H 的反例,则说 H 通过了 A 。这样可以得到全称检验句 H 的证据支持如下:设 B 是 H 的所有相关属性的集合, A 是 H 所通过的相关属性集,则 H 的证据支持就可用 $([A, B], i/n)$ 来进行表示。这里 i, n 分别表示属性集 A, B 中元素的个数。

因为用KER进行知识获取所获得的知识是全称量词化的一阶公式,而一阶公式可由全称简单句和联结词 \wedge (合取), \vee (析取)经过复合而构成。为了在KER中能对所获得的知识进行检验和评估,仅需要给出全称检验句和 \wedge, \vee 所组成的复合句的证据支持。下面对 \wedge, \vee 分别进行讨论。

(1) $H \wedge G$, 其中 H 和 G 为全称检验句,设 H, G 的证据支持分别为 $([A_1, B_1], i_1/n_1), ([A_2, B_2], i_2/n_2)$ 。令 $A = (A_1 \wedge A_2) \vee (A_1 \setminus B_2) \vee (A_2 \setminus B_1)$ (这里“ \setminus ”表示集合的差运算), $B = B_1 \vee B_2$, i 和 n 为 A, B 中元素的个数。显然 $H \wedge G$ 的所有相关属性的集合为 B 。因为 $A_2 \setminus B_1$ 中的任一相关属性都不是 H 的相关属性,所以,相对于 H 来说,关于 A 的检验就是关于 $(A_1 \wedge A_2) \vee (A_1 \setminus B_2)$ 的检验,因此 H 通过了每个这样的检验,即 H 通过了 A 。同理, G 也通过了 A 。又任给 $V \in B \setminus A$,当 $V \in B_1 \setminus A_1$ 时, H 没有通过 \vee ;当 $V \in B_2 \setminus A_2$ 时, G 没有通过 \vee 。因此, $H \wedge G$ 通过的相关属性集合正好是 A ,这样 $H \wedge G$ 的证据支持即为: $([A, B], i/n)$ 。

(2) $H \vee G$, 其中 H 和 G 为全称检验句,设 H, G 的证据支持分别为 $([A_1, B_1], i_1/n_1), ([A_2, B_2], i_2/n_2)$;由“ \vee ”的意义, $([A_1, B_1], i_1/n_1)$ 和 $([A_2, B_2], i_2/n_2)$ 都是 $H \vee G$ 的证据支持。这个结果说明,一般一个复合句可能不止一个证据支持,而有一个证据支持的集合。

(3) 设 H 和 G 分别有证据支持的集合 Γ_1 和 Γ_2 ,则 $H \wedge G$ 的证据支持集是所有 Γ_1 中的元素和 Γ_2 中的元素的复合组成的集合, $H \vee G$ 的证据支持集就是 $\Gamma_1 \vee \Gamma_2$ 。

3 KER中知识的求精

在KER中,由知识获取算法获得的分类知识经过检验,均对应着一个证据支持,以表明它在当前知识集中的可信程度。当所获得的知识的可信度小于某一临界级时,我们称归纳所得的结果不是非常有效的,还需要不断进行精化。在KER中,我们提供三种知识精化手段:

- (1) 扩充:由知识 K_1 转变为较强的 K_2 (即 K_2 是 K_1 的一般化)。
- (2) 收缩:由知识 K_1 变为较弱的 K_2 (即 K_2 是 K_1 的特殊化)。
- (3) 更换:由含有对象 h 的知识 K_1 转变为含有和 h 不相容的 h' 的无矛盾的 K_2 。

扩充和收缩的实现过程是根据一批新的正确事例，对那些没有达到精度的知识进行测试。如果应属于某一知识 A 的对象 B 不能被 A 所满足，则这时通过对所获得的概念删去一些约束（对于一阶公式来说，就是通过减少其前提）以扩大大概念的描述范围，以达到能正确描述所有事例（对象）的目的。如果不属于知识 A 的事例 B 被 A 所满足，则这时通过对所获得的概念 A 增加一些约束（对于一阶公式来说，就是通过增加其前提）以缩减概念的描述范围，达到能正确描述所有事例的目的。对于更换，其目标不是获得一般的新知识，而是获得无误的新知识。由包含 h 的 K_1 直接更换为包含和 h 不相容的 h' 的 K_2 ，便不能满足避免获得错误知识的要求。因此，我们把合理的更换分解为由收缩和扩充组成的修改程序，这样从含有 h 的 K_1 到含有 h' 的 K_2 的更换是通过由 K_1 到 K^* ($K^* \subset K_1$) 的收缩和由 K^* 到 K_2 的扩充来得到。

至此，已经简介了KER中三个主要功能的设计思想。

4 结束语

本文所提出的一种具有知识评估和知识求精的知识获取算法作为知识获取系统GKD—NKAS中进行分类型知识获取的手段，已在micro VAX II上用GKD—Prolog加以实现，此算法被用于某专家系统中进行实际知识库的构造，取得了良好的效果。实际运用表明，在归纳推理中引入证据支持对所获得的知识进行检验和评估，并根据评估结果对知识进行精化的方法，可以为解决知识获取的有效性提供一种手段。

参 考 文 献

- [1] Michalsky R S. A Theory and Methodology of Inductive Learning. Machine Learning, An AI Approach, 1983
- [2] Cohen L J and Hesse M B. Applications of Inductive Logic. Clarendon Press, 1980
- [3] Quinlan J. Induction of Decision Tree. Machine Learning, Vol.1. No. 1. Kluwer Academic Publishers, 1986
- [4] Shannon C E and Weaver W. The Mathematical Theory of Communication. Urbana, IL: University of Illinois, 1964

KER: A Knowledge Acquisition Algorithm with Knowledge Evaluation and Knowledge Refinement

Yang Li Hu Shouren

(Department of Computer Science)

Abstract

This paper gives a decision—tree algorithm acquiring many kinds of knowledge. The algorithm depends on the given priority and type of attributes to acquire the classified type of knowledge. In order to ensure the effectiveness of the acquired knowledge, a kind of evidence support degree to evaluate knowledge is proposed according to Cohen's inductive probability, and a kind of knowledge refinement method is given correspondingly.

Key Words artificial intelligence, knowledge acquisition, algorithm/decision—tree, knowledge evaluation, knowledge refinement, inductive probability