

79-82

因果链

知识库

专家系统

专家系统

20

计算机科学1998 Vol. 25 No. 3

基于因果链的知识库建造^{*}

The Establishment of Knowledge Base Based on Causal Chains

张师超 唐柳英 严小卫

(广西师范大学数学与计算机系 桂林541004)

TP18

摘 要 The causal chains exist universally in knowledge base, and the propositions on a causal chain have certain causal relation. Therefore, we try to set up a method to represent and process knowledge, our method will make the structure of knowledge base more relation, and improve the reasoning efficiencies.

关键词 Causal chain, Uncertain reasoning, Problem solving, Knowledge representation, Knowledge base

一、引言

涂象初等^[1,2]建立了一种集合搜索技术,其效率比传统的逐点搜索有所提高,是解决搜索问题的一种新途径。在许多应用中,集合搜索实质上是一个具有布尔代数元的 IF-THEN 结构序列,即

命题 $(S \rightarrow) \cong (IF X_1 THEN Y_0)$

我们可以将该命题的右部看成为 $(A_1, A_2, \dots, A_n)$ 。它实质上是将论域的操作划分为 n 个独立的部分,而每一个部分 A_i 是一个链状的 IF-THEN 结构: $B_{i1} \rightarrow B_{i2} \rightarrow \dots \rightarrow B_{im}$, 其中 $B_{ij} (1 \leq i \leq n, 1 \leq j \leq m)$ 可以是一个用与、或、非逻辑符号联结起来的复合命题,即每一个 A_i 分别是一条因果链或推理链。一旦链中的某个 B_{ij} 被匹配成功时,则 B_{ij} 后的任何一个 $B_{ik} (j < k \leq m)$ 均为真并可作为结论输出(正向推理)或将它加入事实集合;若是反向推理,则 B_{ij} 前的任何一个 $B_{ik} (1 \leq k < j)$ 均可作为前提输出。有些文献建议这种因果链可以压缩成 $B_{i1} \rightarrow B_{im}$ 的简化形式。不过,有时给定的事实(或推导出的结论)可能是链中间某个 B_{ij} , 或者所要求的结论也可能是链中间的某个 B_{ij} 。因果

链就好比连接每个站的一条公共汽车路线,有的乘客从起点站坐到终点站,而有的乘客从中途上车或下车。可见,诸如此类的简化是不可取的。

我们认为,由于因果链的普遍存在,而且因果链上的命题有一定的关系。若能在知识表示或推理中将它体现出来,它不仅使知识库中的知识结构合理化,更重要的是可以提高推理速度。本文将建立一种处理因果链的知识表示方法及相应的推理模型。

二、因果链的存储

在许多应用中,可用链型方式划分知识(或论域)为 n 个部分 $A_1, A_2, \dots, A_n$, 其中每一个部分 A_i 是一条因果链 $B_{i1} \rightarrow B_{i2} \rightarrow \dots \rightarrow B_{im}$ 。对于每一条因果链,我们用一个规则连接表来存储它。这种表实质上可采用关系数据库中那种存储与管理数据的方式对规则、事实、概念进行有效的存储与管理。于是,知识库中的知识就象书库中的书本一样分门别类地存放,它能提高存取速度,特别地,可提高推理速度。规则连接表的结构如表1所示。

设有这样一条因果链:

表1 因果链上命题的存储结构

序 号	B_i	α $CF(B_{i1} \rightarrow B_{i(j+1)}) \geq \alpha$	α' $(CF(B_{i1} \rightarrow B_{i(j+1)}) \geq \alpha')$	CLOSE
1	B_{i1}	α_1	α_1'	0
2	B_{i2}	α_2	α_2'	0
3	B_{i3}	α_3	α_3'	0
4	B_{i4}	α_4	α_4'	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
m	B_{im}	α_m	α_m'	0

^{*} 本文得到国家863计划和国家自然科学基金资助。张师超 教授,主要研究领域为专家系统和时态逻辑,唐柳英 硕士,主要研究领域为计算机人工智能软件,严小卫 教授,主要研究领域为计算机软件。

云很浓→天下雨→地会湿→今天的球赛可能推迟

其中 $CF(\text{云很浓} \rightarrow \text{天下雨}) \geq \alpha = 0.98$, $CF(\text{天下雨} \rightarrow \text{云很浓}) \geq \alpha' = 0.42$, CLOSE 这个字段初始值全部赋为 0, 也就是说, 对于任一 B_i , 在初始状态时, 没有一条规则被激活。在推理过程中, 若某个 B_i 作为某个规则的前提时, 且该规则被匹配成功, 则令其相应的 CLOSE 值置为 1, 表明规则 $B_i \rightarrow B_{i+1}$ 已被激活, 将该规则关闭, 往后的推理可越过此规则。特别地, B_i 后的每个 $B_t (j \leq t \leq m)$ 的 CLOSE 值也同时被置为 1 并关闭。

若 B_i 作为某个规则的前提时, 且 B_i 是由 $C_{i1}, C_{i2}, \dots, C_{in}$ 用与、或、非逻辑符号联结起来的复合命题, 那么其可能性和必要性计算与文[3]一致。

这种知识库的知识更新操作是在因果链上进行的。当一条新规则增加到知识库中去时, 先将该规则按类分配到某一个规则连接表中去, 即将新规则加入到相应的因果链上。然后找到表中与该规则相关的规则。相关的规则之间可能存在一些冲突, 它需要知识获取机制协调, 或由人工解决。若新规则不属于任何一条链, 则建立一条新链。删除、修改等更新操作也是在因果链上进行, 其操作方式与增加新知识类似。

三、推理模型

依照上述存贮知识的方式, 其推理过程的基本形式是: 设给定事实集合为 F 。对于每一条因果链, 若第一个命题的 CLOSE 值为 0, 则从链的第一个元素(或命题)即 B_1 开始进行匹配, 一旦某个 B_i 被匹配为真, 则对于任一 $B_t (m \geq t \geq j+1)$, 每一个 B_t 均作为真命题加入到事实集合 F 。若所要求的结论包含在新的事实集 F 中, 则输出结果并结束推理; 否则, 重复这一推理过程。同理, 对于反向推理, 可根据目标从表中的最末命题 B_m 开始, 对 B_i 的每个命题进行匹配, 一旦某个 B_i 被匹配为真, 则对于任一 $B_k (1 \leq k \leq j)$, 每一个命题 B_k 均作为前提输出。

可能性与必要性测度的传播由如下推理模式确定:

$$\begin{aligned} N(p \rightarrow q) &\geq \alpha \\ N(q \rightarrow p) &\geq \alpha' \\ \beta &\leq N(p) \leq \beta' \\ \gamma &\leq \Pi(p) \leq \gamma' \end{aligned}$$

$$\min(\alpha, \beta) \leq N(q) \leq \begin{cases} 1, & \text{当 } \alpha \leq \beta' \\ \beta', & \text{其他。} \end{cases}$$

$$\max(1 - \alpha', \gamma') \geq \Pi(q) \geq \begin{cases} \gamma, & \text{当 } \alpha > 1 - \gamma' \\ 0, & \text{其他。} \end{cases}$$

通常, 在可能性理论和 Dempster-Shafer 理论中, 都用两个数来表示不确定性, 即命题被指派一个

• 80 •

可能性程度 $\Pi(p)$ 和一个必要性程度 $N(p)$, 其实 $\Pi(p)$ 和 $N(p)$ 分别相当于通常我们所说的命题 P 的隶属度 $u_A(p)$ 及其可信度 $CF(p)$ (本文主要讨论模糊命题)。在以下的讨论中, 我们均用 $\Pi(p)$ 和 $N(p)$ 分别来代替 $u_A(p)$ 和 $CF(p)$ 。

这里, γ, γ' 分别为当 B_i 作为某个规则的条件(或前提)时的可能性的上界和下界, β, β' 分别为当 B_i 作为规则的条件或前提时的必要性的上界和下界, 而 α, α' 分别为该规则的充分性和必要性 ($q \rightarrow p$ 或 $p \rightarrow q$) 的下界, α 与 α' 的意义同表 1。

现在, 我们用上一节的例子来说明可能性与必要性的传播。假设已知事实 $\Pi(\text{云很浓}) \in [\gamma, \gamma'] = [0.85, 0.9]$, $N(\text{云很浓}) \in [\beta, \beta'] = [0.7, 0.75]$ 。当规则连接表中“云很浓”(某个 B_i) 被匹配成功时, 则根据上面传播方式可得到:

$$\begin{aligned} N(\text{云很浓} \rightarrow \text{天下雨}) &\geq 0.9 \\ N(\text{天下雨} \rightarrow \text{云很浓}) &\geq 0.84 \\ 0.7 &\leq N(\text{云很浓}) \leq 0.75 \\ 0.85 &\leq \Pi(\text{云很浓}) \leq 0.9 \end{aligned}$$

$$0.7 \leq N(\text{天下雨}) \leq 0.75$$

$$0.9 \geq \Pi(\text{天下雨}) \geq 0.85$$

这一推理模式使得我们可以从规则的条件的可能性及必要性的上界和下界, 以及规则的充分性和必要性的下界(以上这些值均存放在表 1 形式的链中), 推得结论的可能性及必要性的上界和下界。再将所得的结果作为下一规则的条件的可能性及必要性的上界和下界, 以及从表中查出下一规则的充分性和必要性的下界; 进一步推得下一个结论的可能性及必要性的上界和下界。如此依次往下推, 直至求出链后部所有命题的可能性及必要性的上界和下界。

下面我们考虑不确定性规则的并行组合, 即已知 n 条规则的集合“若 p_i 则 q ”, 其中 $p_1, \dots, p_n$ 和 q 均为简单命题或复合命题, 每条命题都得到一对关于命题 q 的可能性分布 $\Pi_i(q)$ 和必要性分布 $N_i(q)$ 。我们的目标在于把所有的 $\{\Pi_i(q), N_i(q)\}$ 组合成一个单一的偶对 $\{\Pi(q), N(q)\}$ 。

设 $w_1, w_2, \dots, w_n$ 分别为专家给定每条规则的权重, 权重也可以用如下方式求得。假定我们有数据:

p_i	p_{i1}	p_{i2}	p_{i3}	$\dots$	p_{in}
q	q_1	q_2	q_3	$\dots$	q_n

则由概率论的相关系数的公式, 得前提与结论的相关系数:

$$r_i = \frac{n \sum (q_j^2 (p_{ij} - (p_i - q_j)) - (\sum (p_{ij} - (p_i - q_j))) (\sum q_j))}{\sqrt{n \sum (p_{ij} - (p_i - q_j))^2 - (\sum (p_{ij} - (p_i - q_j)))^2} \sqrt{n \sum q_j^2 - (\sum q_j)^2}}$$

$$(p_i = (\sum p_{ij})/2, q = (\sum q_j)/2)$$

由此, 我们可得出第 i 条规则其结论 q 的权重为:

$$w_i = r_i / (r_1 + r_2 + \dots + r_n), \sum w_i = 1$$

设 $N_1(q) = [l_1, u_1], N_2 = [l_2, u_2], \dots, N_n = [l_n, u_n]$,
则组合多个规则的方式如下:

$$N(q) = [l, u], \text{ 其中 } l = \sum_{i=1}^n w_i l_i, u = \sum_{i=1}^n w_i u_i.$$

同理可得, $l(q) = [v, w]$.

四、推理算法

为简化问题描述,本算法中用二元组 (C, c) 来表示模糊命题 p , 其中 C 为命题的精确描述, c 为命题的不确定刻画. 算法 NA 如下:

```
(1) 输入事实集 T;  
    若  $(C, c) \in T$ , Do  
        Begin  
            若  $\{B_{ij}, [Y, Y'], [B, B'], a, c', \text{CLOSE}\}$ ;  
            /* 作为一个规则的前提条件 */  
            IF  $B_{ij} = C$  THEN  $B_{ij}' := c$   
        end;  
(2)  $b := \text{false}$ ;  
(3) FOR  $i := 1$  TO  $n$  DO  
    BEGIN  
        Q 是规则连接表  $i$ ;  
(4) IF  $B_{ij}$  的 CLOSE  $\neq 1$  THEN  
(5) FOR  $j := 1$  TO  $m$  DO  
    BEGIN  
        计算  $F(B_{ij}')$ ;  
        /*  $F(B_{ij}') =$   
        { .true, 如果  $Y \leq B_{ij}' \leq Y'$ , 即  $B_{ij} \rightarrow B_{ij+1}$  匹配成功;  
        { .false, 否则. */  
        IF  $F(B_{ij}')$  THEN  
            Begin  
                 $b := \text{true}$   
                得到  $B_{ij}$  之后的所有结点  $B_{ij+1} (j < t \leq m)$ , 依次计算  
                所有的  $\{B_{ij}, [Y, Y'], [B, B'], a, c', \text{CLOSE}\}$  并将所得的结果  
                全部加入 T 中并输出, 若某个  $B_{ij}$  为所求结果, 则输出结果并转 (8);  
                并置  $B_{ij}, B_{ij+1}, \dots, B_m$  的 CLOSE 字段值为 1, 表明规则  $B_{ij} \rightarrow$   
                 $B_{ij+1}, B_{ij+1} \rightarrow B_{ij+2}, \dots, B_{ij+m-1} \rightarrow B_m$  匹配成功且被关闭.  
            end;  
    END;  
END;  
(6) IF  $b$  THEN (2);  
(7) 输入“失败”;  
(8) END ALL.
```

定理1 在由 n 个规则连接表(即 n 条因果链)组成的知识库中, 算法 NA 或者可在 $O(s^2)$ 时间内得到所求结果; 或者失败返回, 其中 s 为知识库中的规则总数.

证明: m 是 B_{ij} 的个数. 从算法 NA 的 (6) 可知, 每次外循环至少确定一条规则被匹配成功, 否则失败返回.

设知识库中共有 s 条规则, 则至多执行算法 NA 的 (3) 的次数为 s . 由于因果链中有 $m-1$ 条规则, 所以至多需 $m-1$ 步就可以执行完一条因果链, 即算法的 (5). 又设 m 为所有链中的最长链的命题个数, 那么由此算法 NA 可以在 $O(snm)$ 时间内得到所求的结果(如果有的话). 因为执行外、内循环实质上就是对知识库的规则进行匹配, 所以这里的 nm 可用 s 代替, 从而该推理算法的时间复杂性为 $O(s^2)$.

从算法 NA 可看出, 算法的空间复杂度与传统的方式一致.

在建立专家系统的现有推理模型中, 1992 年,

Chen S. 等发表了一个基于模糊矩阵与事实向量作某种操作而进行推理的模型, 该算法的复杂度约为 $O(s^3)$. 而集合推理模型算法的复杂度虽为 $O(s^2)$, 但其空间复杂度却为 $O(s^3)$, 用大空间换取速度, 又降低了推理效率. 由此可见, 算法 NA 至少对某类问题的求解效率比较高.

设按传统的逐步匹配方式推理的算法为 A, 显然可以得到如下结果:

定理2 设 KB 为知识库, T 为给定的事实集合. 若用算法 A 在 KB 和 T 下推出结果 C, 则在 KB 和 T 下用算法 NA 也可以推出 C.

证明: 设 C 是算法 A 在 KB 和 T 下依次运用 l 条规则 $R^i (i=1, 2, \dots, l, R^i \in \text{KB})$ 推出的结果, 那么在相同的 KB 和 T 下, 对于算法 NA:

① 若 $R^i, i=1, 2, \dots, l$, 属于同一条因果链上的规则. 由算法 NA 的 (3) 和 (5), 一定存在某个 i , 使得在规则连接表 i 中的规则 R^i 匹配成功, 并且其后的规则 $R^2, \dots, R^l$ 的前提和结论均可全部加入 T 中, 从中得到结果 C.

② 若 $R^i, i=1, 2, \dots, l$, 可分成 k 类, 每一类属于同一条因果链, 即每一类属于同一个规则连接表, 这样 l 条规则就属于 k 条因果链.

由算法 NA 的 (3), 通过循环, 可从总的 n 类中, 找出这 k 类 (k 个规则连接表). 对于每一类运用与 ① 类似的方法激活因果链并作用于事实集 T, 最后可从事实集 T 中得到结果 C.

由上可知, 若用算法 A 在 KB 和 T 下推出结果 C, 则在 KB 和 T 下用算法 NA 也可推出 C.

定理3 $A \neq C$, 则 $NA \neq C$.

定理4 $NA \Rightarrow C$, 则 $A \Rightarrow C$.

定理5 $NA \neq C$, 则 $A \neq C$.

定理6 NA 与 A 在功能上是等价的.

定理3至定理6的证明与定理2的证明相类似.

五、因果链的扩展

现在, 我们提出对该方法的进一步扩展, 即由因果链^{扩充}→因果二元树^{扩充}→因果多元树. 因果树在应用中存在更为普遍, 如案件诊断系统. 有关因果树的存储与推理模型有待进一步的探讨, 并且推广也有推理算法, 从而可提高推理效率.

设有如下图形的因果关系:

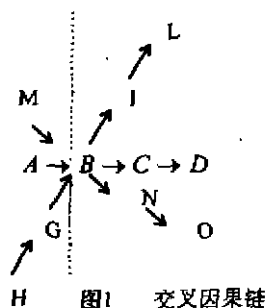


图1 交叉因果链

神经网络的解释机制^{*})

Explanation Mechanism in Neural Networks

贵忠华 刘振凯 严新民

(西北工业大学 西安710072)

摘 要 Neural Networks(NNs) have no explicit, declarative knowledge representation and therefore have considerable difficulties in generating explanation. We think that additional structures and algorithms are needed for generating explanation in NNs. This paper discusses the necessity of explanation in NNs, analyses the explanation methods which have been published, and proposed a case/rule-based explanation method for NNs.

关键词 Neural networks, Explanation, Inference, Knowledge representation, Case

解释就是人工智能系统对用户提问给出一个清晰的、完全的和易于理解的回答,即对其行为作出合理说明、使系统具有透明性。解释的类型虽然很有限,但对于用户是否接受一个推理系统却是绝对关键的。专家系统的应用表明,如果没有任何解释的话,势必影响用户对结论的可信度,降低系统的可接受性。因此,在专家系统研究中,人们作出了很多努力来允许产生解释,且这些解释至少在一定程度上是合情合理的。

神经网络可用于知识表示和推理,但是在本质上神经网络需要解释吗?一个极端的情况是完全信任的训练。例如,在神经网络的学习过程中,当能得到很大的一个训练样本集合,有一个非常有效的训练方法就足够了,可以不需要解释。这时网络能够容易地识别以前训练过的模式和不在训练样本集合里的一些新的实例(即泛化)。然而,对现实世界的应用而言,自然环境是不断变化的,推理者必须能处理不完全和不一致的信息;学习可能被中断;在一个适当的时间间隔里必须完成训练;甚至一个完美的训练过程也不影响用户对解释的需要,由于解释已成为任何推理系统中的一个关键功能。因此,解释机制在

神经网络中同样也是必备的。

人们已尝试性地提出了一些实现神经网络解释机制的方法,这些方法可分为下面三种:(1)采用基于规则的专家系统进行解释;(2)从神经网络中提取用于解释的规则;(3)基于实例的解释方法,即“举例说明”,下面将对这些方法的优缺点进行分析,并提出了一种基于实例/规则的解释方法。

1 采用基于规则的专家系统进行解释

这是 Caudill 于1991年提出的一种方法^[2],其基本思想是构造一个附加的、基于规则的专家系统用于辅助神经网络的解释,如图1所示。当用户要求解

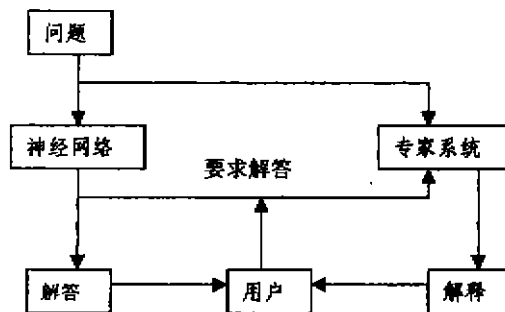


图1 用专家系统辅助神经网络的解释

^{*})国家自然科学基金资助。

该树可从 B 处分成一棵因果树(右半部分)和一棵倒因果树(左半部分)。

因果树中的传播体制与因果链中的传播体制是相似的,但它又具有树自己本身的特点:每个结点与以它为根的子树中的每个子结点有因果关系。象图1中的交叉因果链被分成两棵因果树后,应注意考虑它们的联系。

参 考 文 献

- [1]涂象初等,集合搜索的一些命题,科学通报,1988,33(12),957
- [2]涂象初等,集合搜索作为具有布尔代数元的 IF-THEN 结构,科学通报,34(9)1989
- [3]张尧庭等,专家系统中不确定性方法学,广西师范大学出版社,1994,6版