

# 基于神经网络的交互式 自动获取知识的 N-R 方法\*

李 奕

施鸿宝

(西安交通大学计算机系 西安 710049)

(上海铁道大学计算机系 上海 200333)

**摘要** 本文为解决知识系统构造过程中的瓶颈问题——知识获取,提出了一种基于神经网络 NN(neural network)的自动获取多级推理产生式规则的 N-R 方法,该方法采用了特有的 NN 结构模型和相应的学习算法,使得 NN 在学习过程中动态确定隐层节点数的同时,也产生了样例集中没有定义的新概念,学习后的 NN 能用本文提出的转换算法转换成推理网络,最终方便地得到产生式规则集。

**关键词** 知识获取,机器学习,神经网络,产生式系统,推理。

开发知识系统的一个难点是获取有关领域的知识,构造知识库,而 NN 技术的一个主要特点是能从有关领域的样例集中自动获取(学习)该领域的知识,较之其它机器学习方法如 ID 方法来优越。<sup>[1]</sup>因此,利用 NN 技术来获取知识系统的知识是一个有效途径。<sup>[2]</sup>但 NN 技术的一个弱点是它所获取的知识是分布于网络的隐式知识,难于与推理网络相一致,难于具有可解释性,因此,借助于 NN 获取知识,构造知识库的难题成为如何确定 NN 的初始网络结构,学习过程中如何确定 NN 的隐层节点表示的中间概念,构成可解释的推理网络,最终自动转换为规则集的问题。

本文提出一种逐步获得多层网络,在每一过程中自动调整 NN 结构,学习后的网络具有可推理性和可解释性的 N-R 方法。

在介绍该方法之前,首先阐明多级推理和中间概念这 2 个术语。

本文中的知识是指多级推理的产生式规则,所谓多级推理是指经过 2 次或 2 次以上相关的产生式规则推理才得到结论的规则链,如对前提  $a_1, a_2, a_3, a_4$ , 结论  $P$ , 有下列对应关系:

if  $a_1 \wedge a_2 \wedge a_3 \wedge a_4$  then  $P$

则下列规则链

if  $a_1 \wedge a_2 \wedge a_3$  then  $X$  (I)

if  $X \wedge a_4$  then  $P$  (II)

\* 本文研究得到国家自然科学基金资助。作者李奕,1968年生,助教,主要研究领域为人工智能,神经网络,专家系统。施鸿宝,1937年生,教授,主要研究领域为计算机软件,人工智能,机器学习,神经网络。

本文通讯联系人:施鸿宝,上海 200333,上海铁道大学计算机系

本文 1995-04-24 收到修改稿

就构成  $P$  的一种二级推理,规则(I)(II)之间通过尚未定义的中间概念  $X$  发生联系.而下列规则链

- if  $a1 \wedge a2$  then  $\underline{X'}$
- if  $\underline{X'} \wedge a3$  then  $\underline{X}$
- if  $\underline{X} \wedge a4$  then  $P$

则构成  $P$  的一种三级推理,这里未定义的  $X'$  称为三级推理产生式规则的二级中间概念, $X$  称为一级中间概念.

赋予中间概念后的多级推理的产生式规则,具有推理的层次性,较符合人类思维逻辑,因而便于对获得的结论做出解释.具有多级推理产生式规则获取功能的 NN,在学习过程中获得了中间概念,即新知识,实现一种较高级的智能学习.

下面,我们将用常见的动物识别的例子来说明 N-R 方法的知识获取过程.一组样例如下所示:

- (下蛋,会飞,善飞, 信天翁) (1)
- (下蛋,会飞,能记忆方向, 鸽子) (2)
- (吃肉,有奶,有黄条纹, 虎) (3)
- (吃肉,有奶,褐黄色,黑斑点, 豹) (4)
- (有蹄,有奶,有黑条纹, 斑马) (5)
- (有蹄,有奶,有白毛, 羊) (6)

如果用户没有特别要求,用 NN 学习这组样例后,能够将样例(1)、(2)聚成一样例子集,形成一个新概念“鸟”(下蛋  $\wedge$  会飞),样例(3)、(4)聚成一类,形成新概念“肉食动物”(吃肉  $\wedge$  有奶),样例(5)、(6)聚成一类,形成新概念“草食动物”(有蹄  $\wedge$  有奶),从而得到二级推理的产生式规则,推理网络见图 6(细实线和虚线部分).

如果用户觉得二级推理的层次不够清楚,要求三级推理,则 NN 对这组样例进行一系列的学习后,另产生“哺乳动物”这个新概念,得到三级推理的产生式规则,推理网络见图 6.

作为例子,我们只演示到三级推理.而本文提出的方法能够获得  $n$  级推理( $n \geq 2$ )的产生式规则.

为了让 NN 具有上述功能,本文以一种适合于获取二级推理产生式规则的 NN 结构模型和相应的 NN 学习算法、从 NN 到可解释的推理网 RN(reasoning network)的转换算法以及由二级推理逐层学习扩展到多级推理的算法来说明 N-R 方法.

## 1 神经网络结构模型

### 1.1 有关定义

适于获取二级推理产生式规则的 NN 由输入层、隐层、输出层构成,隐层节点分为共性节点与个性节点 2 类.

定义 1(共性节点). 共性节点代表聚成一类的某样例子集所蕴含的新概念,指向共性节点的各连线的起点(输入层节点)为这个样例子集中各样例所共有的属性.

如上面例子中样例(1)、(2)聚成一类,这一类共有的属性集是{下蛋,会飞}.代表这个样例子集的共性节点表示了“鸟”这个中间概念.

**定义 2(个性节点).** 个性节点代表聚成一类的样例子集中的某一样例所持有的结论蕴含的虚概念,指向个性节点各连线的起点为该样例子集中这个样例区别于其它样例所特有的属性集.

如上面例子中样例(1)、(2)聚成一类,有一个个性节点表示该类中样例(2)的结论—“鸽子”这个虚概念,指向这个个性节点各连线的起点(输入层节点)为该样例子集中“鸽子”区别于“信天翁”的特有的属性集{能记忆方向}.

如果某个样例没有聚到任何类,则指向表示它的结论的个性节点各连线的起点代表这个样例含有的所有属性.

为简单起见,以下简称隐层的个性节点代表聚成一类的样例子集中的某一样例,共性节点代表聚成一类的某样例子集.

有了上述 2 个定义,用本文提出的方法学习后,一个被聚到某一类的样例,可被输出层节点,隐层的共性节点和个性节点,输入层节点以及它们之间的连线所表示,而未被聚到任何一类的样例,就只被输出层节点,隐层的个性节点和输入层节点以及它们之间的连线所表示.如上面例子中样例(1)、(2)聚成一类,则这 2 个样例在训练后的 NN 中表示如图 1 所示.

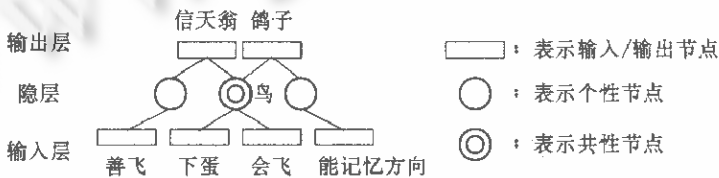


图1 样例(1)、(2)在训练后的NN中的表示

### 1.2 初始 NN 的构造

给定样例集  $S$  (含  $N$  个样例,样例的输入向量维数为  $I$ ,期望输出向量维数为  $Q$ ),如下构造 3 层的初始网络:

输入层节点数取值  $I$ ,输出层节点数取值  $Q$ ,隐层含  $2N$  个节点,其中的  $N$  个共性节点表示  $S$  中 1 个样例构成 1 个子集的最初  $N$  个子集, $N$  个个性节点表示  $N$  个不同的样例.初始网络为输入层到隐层,隐层到输出层以权值初值  $\omega_0$  连接的全连接网络.

## 2 相应的学习算法

构造初始网络后,要用某种学习算法调整它的权值矩阵,使得网络收敛后,NN 在功能上满足样例集给定的输入—输出对应关系,并能根据网络的部分连接分析出共性节点代表的某样例子集蕴含的新概念与哪些属性(即输入层节点)相关,个性节点代表的样例与哪些特有属性(即输入层节点)相关,以及输出层节点与哪些共性节点、个性节点相关.

为此提出一种改进的 LMS 算法,描述如下:

设有一 3 层 NN,各节点之特性为 Sigmoid 型.有  $N$  个训练样例  $(X_k, Y_k)$  ( $k=1, \dots, N$ ),其中  $X_k$  表示输入向量, $Y_k$  表示期望输出向量.任一节点  $i$  的输出表示为  $O_{i,k}$ ,其中  $i$  表示节点序号, $K$  表示样例序号.

### 2.1 前向计算

$$\text{节点 } j \text{ 的输入为 } net_{j,k} = \sum_i \omega_{ij} * O_{i,k} + \theta_j$$

节点  $j$  的输出为  $O_{jK} = f(\text{net}_{jK})$ , 其中  $f(x) = 1/(1 + e^{-x})$

对隐层、输出层的每个节点  $j$ , 计算  $O_{jK}$ .

### 2.2 后向计算

在用某个样例  $(X_K, Y_K)$  进行一趟前向计算后, 输出层、隐层的各节点都有了输出值  $O_{jK}$ . 在后向计算中, 要计算各节点的误差, 然后沿着误差下降最快的方向调整权值.

对节点  $j$ , 设其输出为  $O_{jK}$ , 期望输出为  $\text{exp}_{jK}$ , 平方型误差为  $E_{jK}$ .

$$E_{jK} = (\text{exp}_{jK} - O_{jK})^2$$

由于  $O_{jK} = f(\text{net}_{jK})$ ,  $\text{net}_{jK} = \sum_i \omega_{ij} * O_{iK} + \theta_j$

故 
$$\frac{\partial E_{jK}}{\partial \omega_{ij}} = \frac{\partial E_{jK}}{\partial O_{jK}} * \frac{\partial O_{jK}}{\partial \text{net}_{jK}} * \frac{\partial \text{net}_{jK}}{\partial \omega_{ij}} = -2 * (\text{exp}_{jK} - O_{jK}) * f'(\text{net}_{jK}) * O_{iK} \tag{I}$$

如下调整节点  $j$  的权值  $\omega_{ij}$ :  $\omega_{ij}(t+1) = \omega_{ij}(t) - \eta * \sum_{K=1}^N \partial E_{jK} / \partial \omega_{ij} \tag{II}$

其中  $\eta > 0$ , 为学习效率.

在(I)中,  $O_{jK}$ ,  $f'(\text{net}_{jK})$ ,  $O_{iK}$  都可根据前向计算中的结果而得到, 下面讨论  $\text{exp}_{jK}$  的计算.

(1)  $j$  是输出层节点  $\text{exp}_{jK}$  即为训练样例  $(X_K, Y_K)$  的  $Y_K$  的某个元素.

(2)  $j$  是隐层的个性节点 因为个性节点编号与样例编号一一对应, 所以当  $j = k$  时,  $\text{exp}_{jK} = 1$ ; 否则  $\text{exp}_{jK} = 0$ .

(3)  $j$  是隐层的共性节点 共性节点  $j$  的  $\text{exp}_{jK}$  表示样例  $K$  是否与该共性节点代表的样例子集属于同一类, 如属于同一类,  $\text{exp}_{jK} = 1$ ; 否则  $\text{exp}_{jK} = 0$ .

当  $j = K$  时,  $\text{exp}_{jK} = 1$ , 否则按下列过程计算:

设隐层有  $N$  个共性节点. 样例  $K$  与  $N$  个共性节点的关系或者是属于同一类, 或者是不属于同一类. 设  $\max_K$  为  $N$  个共性节点的输出  $O_{1K}, \dots, O_{NK}$  中的最大值. 事先给定一分类密度常数  $dis$ , 对于共性节点  $j$ , 如果  $O_{jK} > \max_K - dis$ , 则  $\text{exp}_{jK} = 1$ ; 否则  $\text{exp}_{jK} = 0$ .

根据(1)~(3)可计算出 3 种节点的  $\text{exp}_{jK}$ , 由(I)式计算  $\partial E_{jK} / \partial \omega_{ij}$ , 由(II)式就可调整权值  $\omega_{ij}$ .

### 3 神经网络到推理网络的转换算法

给定样例集  $S$ , 构造一 NN, 用改进的 LMS 算法学习, 网络收敛后, 得到一连接权值矩阵. 下面将叙述如何根据权值矩阵, 将 NN 转换成 RN.

#### 3.1 将全连接的 NN 转换成部分连接的 NN

设  $\omega_{ij}$  为节点  $i$  到节点  $j$  的连接权值,  $L_{ij}$  为 2 节点间的连接状态指示 ( $L_{ij} = 1$  表示有正连接,  $L_{ij} = -1$  表示有负连接,  $L_{ij} = 0$  表示无连接). 根据  $\omega_{ij}$  得到  $L_{ij}$  的过程就是全连接的 NN 转换成部分连接的 NN 的过程.

如果  $\omega_{ij} \geq \text{threshold1}$  ( $\text{threshold1}$  为一正常数), 则令  $L_{ij} = 1$ .

如果  $\omega_{ij} \leq \text{threshold2}$  ( $\text{threshold2}$  为一负常数), 则令  $L_{ij} = -1$ , 否则令  $L_{ij} = 0$ .

#### 3.2 去掉隐层中的某些个性节点

去掉隐层中的个性节点, 以便让输出层节点直接与输入层节点相连, 具体步骤为:

设  $L_{cj}$  为输出层节点  $C$  与隐层个性节点  $j$  连接的状态指示.

(I)如果  $L_{c_j}=0$ ,则无任何动作.

(II)如果  $L_{c_j}=1$ ,则对所有的  $|L_{j_i}|=1$  的输入层节点  $i$ ,置  $L_{c_i}=L_{j_i}$ ,并令  $L_{j_i}=0, L_{c_j}=0$ .即去掉个性节点  $j$ .

(III)如果  $L_{c_j}=-1$ ,则保留这个个性节点.

对所有的输出层节点,重复上述步骤(I),(II).图2是网络完成上述步骤的示意图.

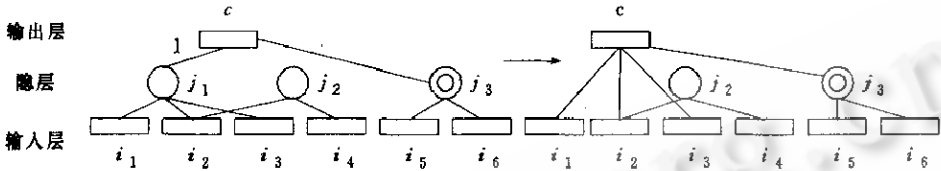


图2 去掉隐层个性节点  $j_1$

### 3.3 去掉隐层共性节点中的冗余节点

设有 2 个共性节点  $j_m, j_n (n > m)$ , 输入层共有  $I$  个节点,  $j_m, j_n$  到输入层的连接状态向量分别为  $L_{j_m} = (L_{j_m1}, \dots, L_{j_mI}), L_{j_n} = (L_{j_n1}, \dots, L_{j_nI})$ . 如  $L_{j_m} = L_{j_n}$  (意味着这 2 个节点代表同一个例子集), 则称节点  $j_n$  是节点  $j_m$  的冗余节点.

隐层共性节点中, 如  $j_n$  是  $j_m$  的冗余节点, 则对所有的  $L_{c_j}=1$  的输出层节点  $C$ , 置  $L_{c_j_m}=1, L_{c_j_n}=0$ , 然后去掉节点  $j_n$ . 图 3 是这一步骤的示意图.

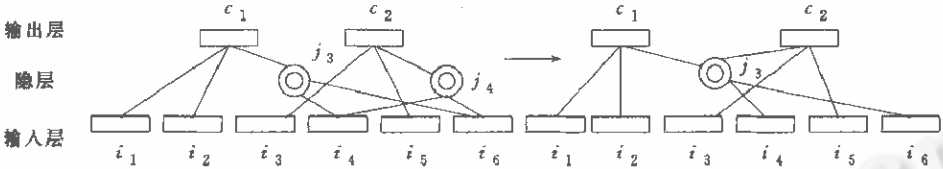


图3 去掉共性节点  $j_3$  的冗余节点  $j_4$

### 3.4 从神经网络到推理网络和产生式规则集

设输出层有  $Q$  个节点, 则对每个输出层节点  $C_i (i=1, \dots, Q)$ , 重复下列步骤:

(I)从节点  $C_i$  出发, 沿着存在的连接, 深度遍历整个 NN, 得到一连通子网.

(II)在这个连通子网中, 将其输入层、隐层、输出层节点分别替换成 RN 的前件、中间概念和结论.

全部输出层节点完成步骤(I),(II)后, 将网络中起始于不同节点, 终于同一节点的不同连接之间的关系翻译为“与导出关系”, 其中  $L_{ij}=1$  的联边的起点构成与项的一个正逻辑变量,  $L_{ij}=-1$  的联边的起点构成与项的一个负逻辑变量, 就得到推理网络. 由推理网络可方便地转换成产生式规则集. 图 4 是上述步骤的示意图.

## 4 多级推理网络的构造

给定样例集, 用第 1~3 节中的方法构造、学习、转换 NN, 得到二级推理的产生式规则. 其中的中间概念可经过交互式过程由专家赋予含义. 如专家感到 NN 得到的中间概念难于赋予含义, 或中间概念的内涵太多, 可将各中间概念作为结论与对应的前件形成新的训练样例集, 未聚成类的原训练样例也可加入到这个样例集中. 按照上面的过程再学习一次. 将这

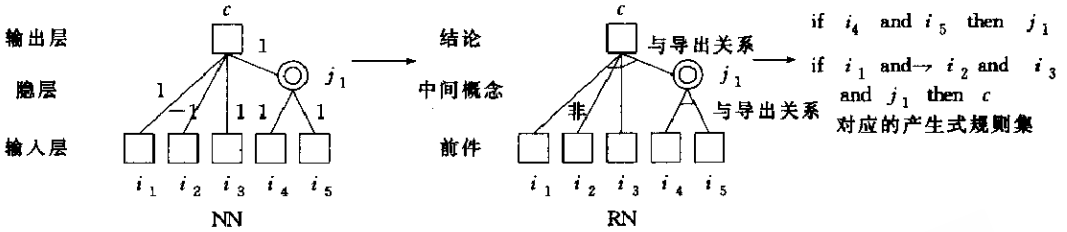


图4 从NN到RN和产生式规则集

次学习形成的推理网络  $N_2$  作为第 1 次学习得到的推理网络  $N_1$  的中间概念层嵌入到  $N_1$  中, 形成具有 2 层中间概念层的推理网, 从而获得三级推理的产生式规则. 这个过程可以一直持续下去, 直到交互过程中专家满意才结束, 用 NN 获取多级推理产生式规则的过程如图 5 所示.

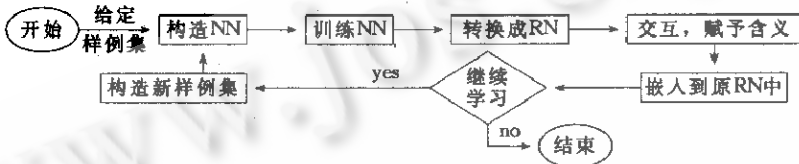


图5 获取多级推理的产生式规则

对于本文第 1 节中给出的动物识别的样例集, 以三级推理为例, 说明如何获得多级推理的产生式规则. 首先根据给定的样例集, 构造网络, 网络经改进的 LMS 算法学习, 用转换算法转换后, 得到如图 6 中细实线和虚线(虚线原应为细实线, 再次学习后将删除, 故用虚线表示)所构成的二级推理网.

其中  $j_1, j_2, j_3$  是形成的中间概念, 经过人机交互过程, 分别给  $j_1, j_2, j_3$  赋予“鸟”、“食肉动物”、“草食动物”的含义, 根据上述推理网, 就能得到二级推理的产生式规则集.

如果我们觉得第 1 次学习获得的中间概念的内涵太多, 使得二级推理产生式规则的逻辑性较弱, 可以将获得的 3 个中间概念作为结论与对应前件形成新的训练样例集. 依循同样的过程对新样例集进行学习转换, 得到图 6 中粗实线所构成的二级推理网, 其中  $j_4$  是新形成的中间概念. 经过交互过程, 给  $j_4$  赋予“哺乳动物”的含义.

第 2 次学习得到的推理网作为第 1 次学习形成的推理网的中间概念嵌入其隐层, 产生图 6 中细实线和粗实线所构成的具有 2 级中间概念的三级推理网(虚线表示已被删去的连接).

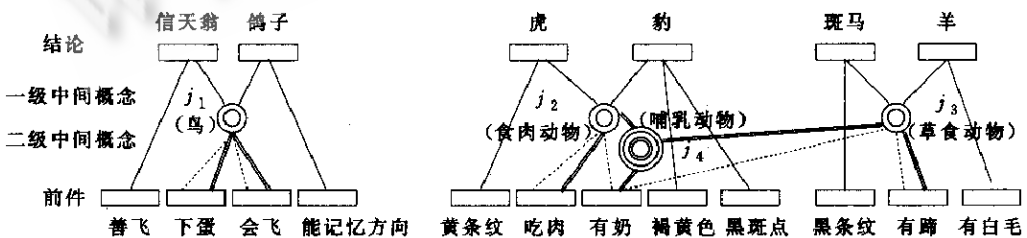


图6 二级至三级推理网络的变化过程

## 5 结 论

本方法用于儿科常见病病例集,获得的多级推理的产生式规则,在推理得出结论时有一定的层次性、逻辑性,便于人们理解,满足专家系统的要求.这种基于 NN 的获取知识的 N-R 方法较之知识工程师人工直接获取知识库中的规则集来得简便,快速.

### 参考文献

- 1 Fu L M. Rule generation from neural networks. IEEE Trans. on Systems, Man, and Cybernetics, 1994, 24(8): 1114~1124.
- 2 Galland S L. Connectionist expert system. CACM, 1988, 31(2):152~169.

## THE N-R METHOD OF ACQUIRING KNOWLEDGE AUTOMATICALLY AND INTERACTIVELY BASED ON NEURAL NETWORK

Li Yi

(Department of Computer Science Xi'an Jiaotong University Xi'an 710049)

Shi Hongbao

(Department of Computer Science Shanghai Railway University Shanghai 200333)

**Abstract** This paper presents a N-R method of acquiring multi-level reasoning production rules automatically based on NN(neural network), so as to resolve the bottle neck problem in building a knowledge-based system—knowledge acquisition. This method adopts a characteristic NN structure model and corresponding learning algorithm, so that N-R method can fix the number of hidden nodes dynamically and form new concepts which are not defined in the sample set in the course of learning. Having been trained, NN is transformed into an RN(reasoning network) with the proposed transform algorithm, from which production rules can be obtained easily.

**Key words** Knowledge acquisition, machine learning, neural network, production system, inference.