第四章 专家系统

在人工智能的发展过程中，研究者们逐渐认识到知识的重要性。如果计算机可以学习到人类在某个领域的专业知识，并且利用这些知识解决这个领域的专业问题，不是一种很好的人工智能的体现吗？所以，将一个专家在他本领域的专业知识总结出来，以计算机可以使用的形式表达，让计算机系统可以利用这种知识解决问题，这就是专家系统研究的动机。

世界上第一个专家系统是DENDRAL，诞生于1965年，由E.A.费根鲍姆等人在总结通用问题求解系统的成功与失败经验的基础上，结合化学领域的专门知识研制出来的，可以推断化学分子结构。后来，费根鲍姆领导的小组又研发了著名的专家系统MYCIN，它的功能是帮助医生对住院的血液感染患者进行诊断和选用抗生素HYPERLINK类药物进行治疗。这是个非常重要的专家系统，因为可以说它确定了专家系统的基本结构，为后来的专家系统研究奠定了基础。

专家系统的理论和技术不断发展，应用渗透到几乎各个领域，包括化学、数学、物理、生物、医学、农业、气象、地质勘探、军事、工程技术、法律、商业、空间技术、自动控制、计算机设计和制造等众多领域，研究者们已经开发了几千个的专家系统，其中不少在功能上已达到，甚至超过同领域中人类专家的水平，并在实际应用中产生了巨大的经济效益。

**4.1 系统概述**

我们可以将专家系统定义为：一种智能的计算机程序，其内部含有大量的某个领域专家水平的知识与经验，它运用知识和推理来解决该领域内只有专家才能解决的复杂问题。需要注意的是，这里的知识和问题均属于同一个特定领域。简而言之，专家系统是一种模拟人类专家解决领域问题的计算机程序系统。

专家系统有几个重要特征：

最高专业水平：专家系统提供最高水平的专业知识。它提供了效率，准确性和富有想象力的问题解决方案。

正确的时间反应：专家系统在非常合理的时间段内与用户进行交互。总时间必须小于专家为同一问题获得最准确解决方案所花费的时间。

良好的可靠性：专家系统需要可靠，并且不能犯任何错误。

灵活：至关重要的是它保持灵活性，因为它是专家系统所拥有的。

有效机制：专家系统必须有一个有效的机制来管理其中现有知识的汇编。

能够处理具有挑战性的决策和问题：专家系统能够处理具有挑战性的决策问题并提供解决方案。

一个专家系统的基本结构如图4.1所示。主要有六个部分构成：人机交互界面、知识库、推理机、解释器、动态数据库、知识获取。人机界面是系统与用户进行交流时的界面。通过该界面，用户输入基本信息、回答系统提出的相关问题，并输出推理结果及相关的解释等；知识库用来存放专家提供的领域知识和事实等。专家系统的问题求解过程是通过知识库中的知识来模拟专家的思维方式的；推理机针对当前问题的条件或已知信息，反复匹配知识库中的规则，获得新的结论，以得到问题求解结果。推理方式可以有正向和反向推理两种。将在下一节中进行更详细的解释；解释器能够根据用户的提问，对结论、求解过程做出说明，因而使专家系统更像一个“人”；动态数据库专门用于存储推理过程中所需的原始数据、中间结果和最终结论，往往是作为暂时的存储区；知识获取是专家系统知识库是否优越的关键，也是专家系统设计的“瓶颈”问题，通过知识获取，可以扩充和修改知识库中的内容，也可以实现自动学习功能。

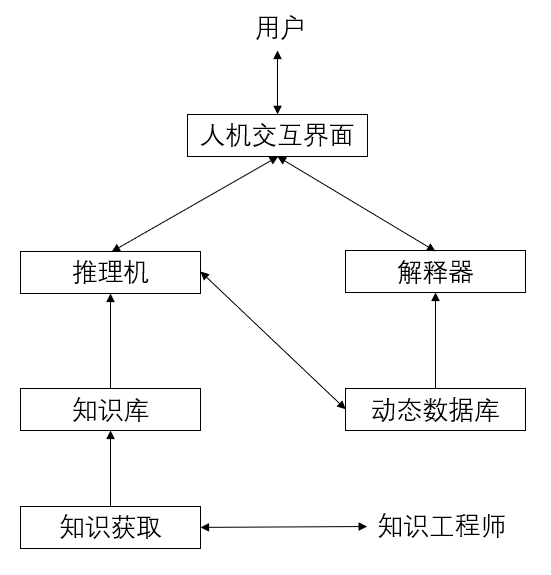


图4.1

专家系统中最重要的部分是知识库和推理机。这也是专家系统不同于一般的计算机程序系统的所在。以这两个部分为核心，专家系统可以处理非确定性的问题，它的目标不是追求问题的最佳解，而是利用知识和推理得到一个满意解。一个好的专家系统应该强调知识库与其他子系统的分离，因为知识库是存储专业知识的，这些知识是与领域强相关的，不同领域的知识库肯定不同，然而推理机等其他的子系统在不同的领域也可能具有一定的通用性。

我们一般将存放于知识库中的知识称为规则或者知识规则，一般以如下形式表示：

IF <前提> THEN <结论>

表示：当 <前提> 被满足时，可以得到 <结论>

例如：IF 阴天and湿度大 THEN 下雨

表示：如果阴天且湿度大，则会下雨。

推理机是一个执行结构，它负责对知识库中的知识进行解释，利用知识库进行推理。假设知识以规则的形式表示，推理机会根据某种策略对知识库中的规则进行预测，选择一个<前提>可以满足的规则，得到该规则的<结论>，并根据<结论>的不同类型执行不同操作。我们将在下一节中对其执行的功能进行讲解。

**4.2 推理**

专家系统中的推理机是如何利用知识库进行推理的？这个答案会根据知识表示方法的不同而有所不同。在专家系统中，规则是最常用的知识表示方法，下面以规则为例进行说明。

按照推理的方向，推理方法可以分为正向推理和逆向推理。正向推理由称为数据驱动推理，它从一组事实或者已知的前提出发，正向地使用规则，从已知条件向目标的方向进行推理。其基本思想是：检验是否有规则的前提被动态数据库中的已知事实满足，如果被满足，则将该规则的结论放入动态数据库中，再检查其他的规则是否有前提被满足；反复该过程，直到目标被某个规则推出结束，或者再也没有新结论被推出为止。由于这种推理方法是从规则的前提向结论进行推理，所以称为正向推理。由于正向推理是通过动态数据库中的数据来“触发“规则进行推理的，所以又称为数据驱动的推理。

例：设有三条规则分别为：

R1：IF A and B THEN C

R2：IF C and D THEN E

R3：IF E THEN F

并且已知A、B、D成立，求证F成立

初始时已知A、B、D在动态数据库中，根据规则R1，推出C成立，所以将C加入到动态数据库中；此时动态数据库中有C和D，根据规则R2，推出E成立，将E加入到动态数据库中；此时动态数据库中有，根据R3，推出F成立，由于得到我们需要证明的F，所以结果成立，推理结束。

逆向推理又被称为反向推理，是逆向地使用规则，先将目标作为假设，反推是否有某条规则支持该假设，即规则的结论与假设是否一致，然后看结论与假设相关的规则其前提是否成立。如前提成立（在动态数据库中进行匹配），则假设得到求证，结论被放入动态数据库中；否则将该规则的前提加入到假设集中，一个一个的求证这些假设，直到目标假设被验证为止。由于逆向推理是从假设求解目标成立，是逆向使用规则进行推理的，所以又被称为目标驱动的推理。

那么在上述例子中，如何使用逆向推理推导出F成立？

首先将F作为假设，我们知识库中发现规则R3的结论可以推导出F，然后在动态数据库中寻找R3的前提即E是否成立。因为目前动态数据库中还没有记录E是否成立，所以我们继续寻找哪条规则可以得到E，发现R2可以得到结论E，于是在数据库中依次检验R2的前提C和D是否成立。首先检验C，由于C也没有在动态数据库中，再次找结论含有C的规则，找到规则R1，发现其前提A、B在动态数据库中均成立，从而推导出C成立，将C放入动态数据库中。再检验规则R2的另一个前提条件D，由于D在动态数据库中成立，从而R2的前提全部被满足，推出E成立，并将E放入动态数据库中。此时规则R3的前提成立了，从而最终推出目标F成立。

一般的结论及推理都是确定性的，也就是说前提成立，结论一定成立。比如在几何定理证明中，如果两个同位角相等，则两条直线一定是平行的。但是在很多实际问题中，推理往往具有模糊性、不确定性。比如“如果阴天则可能下雨”，但我们都知道阴天了不一定就会下雨，这就属于非确定性推理问题。确定和非确定性推理问题需要不同的方法来解决。

4.2.1确定性推理

我们用一个例子来展示专家系统中的确定性推理，假如你是一位动物学家，可以识别各种动物。你的朋友周末带小孩去动物园游玩并见到了一个动物，朋友不知道该动物是什么，于是给你打电话咨询，你们之间有了以下的对话：

你：你看到的动物有羽毛吗？

朋友：这个动物有羽毛

你：你看到的这个动物会飞吗？

朋友：这个动物不会飞。

你：你看到的这个动物有长腿吗？

朋友：没有长腿。

你：它会游泳吗？

朋友：会游泳。

你：颜色是黑白吗？

朋友：是。

你：这个动物是企鹅。

在以上对话中，当得知动物有羽毛后，你就知道了该动物属于鸟类，于是你提问是否会飞；当得知不会飞后，你开始假定这可能是鸵鸟，于是提问是否有长腿；在得到否定回答后，你马上想到了可能是企鹅，于是询问是否会游泳；然后为了进一步确认是否是企鹅，又问颜色是否是黑白的；得知是黑白颜色后，就可以确认该动物是企鹅。在这个过程中，你的提问是非常明确的，肯定能得到某种信息的。同时朋友的回答也是确定的能提供确切信息的。

我们也希望一个动物识别专家系统能像你一样完成以上过程，通过与用户的交互回答用户有关动物的问题。

为了实现这样的专家系统，首先要把你有关识别动物的知识总结出来，并以计算机可以使用的方式存放在计算机中。可以用规则表示这些知识。

一个规则，具有如下的格式：

（<规则名>

（if <前提>）

（then <结论>））

如“有羽毛则是鸟类”可以表示为：

（R3

（if（有羽毛））

（then（鸟类））

）

其中R3是规则名，（有羽毛）是规则的前提，（鸟类）是规则的结论。

如果前提有多个条件需要同时满足，则将多个条件并列即可。如“如果是鸟类且不会飞且会游泳且是黑白色则是企鹅“可以表示为：

（R12

（if（鸟类）

（会飞）

（会游泳）

（黑白色））

（then （企鹅）））

如果前提条件只要满足其中之一就可以，则可以用“or”将条件连接起来，表示“或”的关系，”如果是哺乳类且（有蹄或者反刍）则属于偶蹄类“可以表示为：

（R6

（if（哺乳类）

（（有蹄）or（反刍）））

（then （偶蹄类）））

根据我们对规则格式的定义，再增多几条规则，我们可以使用如下规则组成一个知识库，利用该知识库可以识别虎、豹、斑马、长颈鹿、鸵鸟、企鹅、信天翁等七种动物，知识库里面的规则可以表示为下：

（R1

（if（有毛发））

（then（哺乳类））

）

（R2

（if（有奶））

（then（哺乳类）

）

（R3

（if（有羽毛））

（the（鸟类））

）

（R4

（if（会飞）

（下蛋））

（the（鸟类））

）

（R5

（if（哺乳类）

（（吃肉）or（有犬齿））

（眼睛前视）

（有爪））

（then（食肉类））

）

（R6

（if（哺乳类）

（（有蹄）or（反刍）））

（then（偶蹄类））

）

（R7

（if（食肉类）

（黄褐色）

（有暗斑点））

（then（豹））

）

（R8

（if（食肉类）

（黄褐色）

（有黑条纹））

（then（虎））

）

（R9

（if（偶蹄类）

（有长腿）

（有长颈）

（黄褐色）

（有暗斑点））

（then（长颈鹿））

）

（R10

（if（偶蹄类）

（有白色）

（有黑条纹））

（then（斑马））

）

（R11

（if（鸟类）

（不会飞）

（有长腿）

（有长颈）

（黑白色））

（then（鸵鸟））

）

（R12

（if（鸟类）

（不会飞）

（会游泳）

（黑白色））

（then（企鹅））

）

（R13

（if（鸟类）

（会飞））

（then（信天翁））

）

推理机是如何利用这些知识进行推理的呢？我们假设采用逆向推理进行求解。

首先，系统提出一个假设。由于一开始没有任何信息，系统只能把规则的结论部分含有动物的全部内容作为假设，并按照一定顺序进行验证。在验证的过程中，如果一个事实是已知的，比如已经在动态数据库中有记录，则直接使用该事实。动态数据库中的事实是在推理过程中由用户输入的或者是某个规则的结论。如过动态数据库中对该事实没有记录，则查看是否是某个规则的结论，如果是某个规则的结论，则检验该规则的前提是否成立，实际上就是用该规则的前提当作子假设进行验证，是一个递归调用的过程；如果不是某个规则的结论，则向用户询问，由用户通过人机交互接口获得。在以上过程中，一旦某个结论得到了验证——由用户输入的或者是规则的前提成立推出的——就将该结果加入动态数据库中，直至在动态数据库中得到最终的结果（动物是什么）结束，或者推导不出任何结果结束。

假定系统首先提出的假设是鸵鸟，则推理过程如图4.2所示。根据规则R11，需要验证其前提条件“是鸟类 且 不会飞 且 有长腿 且 有长颈 且 黑白色”。首先验证“是鸟类”，动态数据库中还没有相关信息，所以查找结论含有“（类 鸟类）”的规则R3，其前提是“有羽毛”。该结果在动态数据库中也没有相关信息，也没有哪个规则的结论含有该结果，所以向用户提出询问是否有羽毛，用户回答“Yes”，得到该动物有羽毛的结论，将有羽毛放入动态数据库。由于R3的前提条件只有这一个，所以由规则R3得出该动物属于鸟类，并将“是鸟类”这个结果加入到动态数据库中。R11的第一个条件得到满足，接下来验证第二个条件“不会飞”。同样，动态数据库中没有记载，也没有哪个规则可以得到该结论，继续询问用户，得到回答“Yes”后，将“不会飞”加入到动态数据库中。再验证“有长腿”，这时由于用户回答是“No”，表示该动物没有长腿，“没有长腿”也被放入到动态数据库中。由于“有长腿”得到了否定回答，所以R11的前提不被满足，假设“鸵鸟”不能成立。系统再次提出新的假设动物是“企鹅”，得到如图4.3所示的推理过程。结论是企鹅的规则是R12，要验证规则的前提条件“是鸟类 且 不会飞 且 会游泳 且 黑白色”，由于动态数据库中已经记录了当前动物“是鸟类”“不会飞”，所以规则R12的前两个条件均被满足。直接验证第三个条件“会游泳”和第四个条件“黑白色”，这两个条件都需要用户回答，在两个问题都得到肯定的答案后，系统得出结论——这个动物是企鹅。

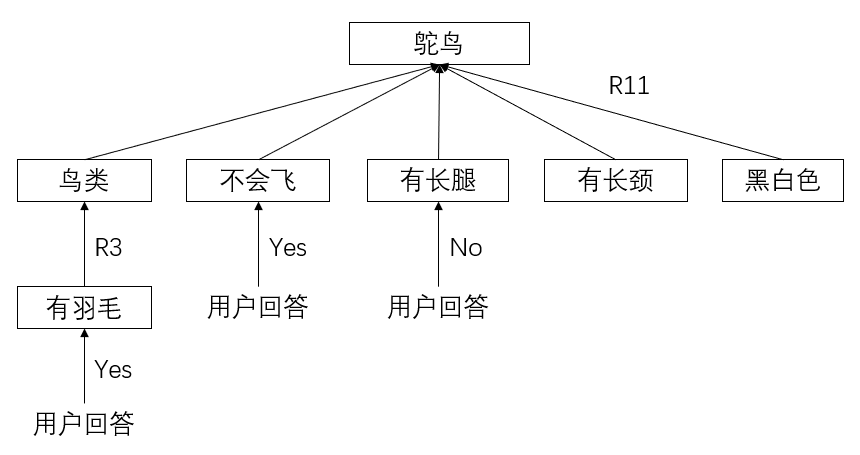


图4.2

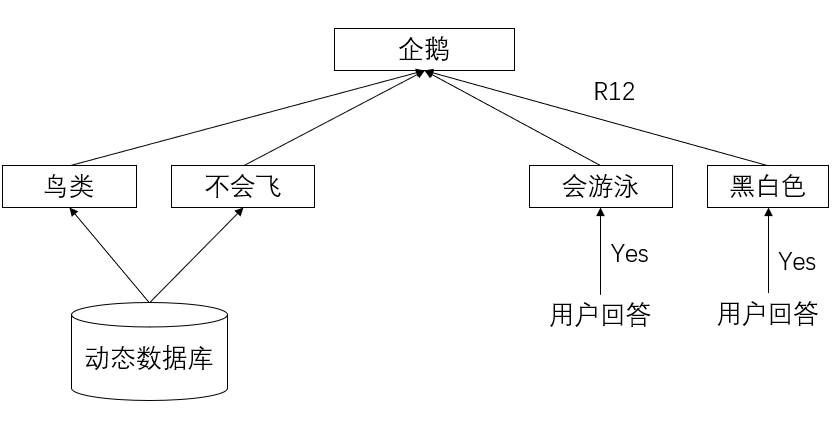


图4.2

以上我们给出了一个简单的专家系统实例以及它是如何工作的。实际的系统中，为了提高效率，可能要比这复杂得多，例如动物识别系统，肯定要识别的不只是七种动物，所以肯定会更加复杂，规则更多。那么需要解决的问题就是如何提高匹配速度以提高系统的工作效率和如何提出假设以便系统尽快地得出答案等等。

我们也可以看到，再以上例子中，从用户那里得到的回答都是yes或者no这样很明确的回答，规则也很明确，如果满足前提条件那么就必然能得出结论，比如是鸟类或者是企鹅等，而不会得出可能是鸟类，可能是企鹅这样模糊的结论。这种推理就是确定性推理。

4.2.2 非确定性推理

在前面给出的一个识别七种动物的专家系统的简单例子中，每个规则都是确定性的，也就是说满足了什么条件，结果就一定是什么。但现实生活中的很多实际问题是非确定性问题。比如如果阴天则下雨。阴天就是一个非确定性的东西，参杂了人的主观意志，有的人可能认为有些云彩就算阴天，有的人认为乌云密布才算是阴天。而且满足了阴天的条件也不一定下雨，只是在人们的常识中，天阴得越厉害，下雨的可能性就越大，但不能说阴天就一定下雨。这就是非确定性的问题，没有办法给条件和结论进行百分百肯定，解决非确定性问题需要使用非确定性推理方法。

目前有不少非确定性推理方法，各有优缺点，下面我们以著名的专家系统MYCIN中使用的可信度方法（certainty factor，CF方法）为例进行说明。

事实A为真的可信度用CF（A）表示，取值范围为[-1，1]，当CF（A）=1时，表示A肯定为真；当CF（A）=-1时，表示A为真的可信度为-1，也就是A肯定为假。CF（A）>0表示A以一定的可信度为真；CF（A）<0表示A以一定的可信度且值为(-CF（A）)为假；或者说A为真的可信度为CF（A），由于此时CF（A）为负，实际上A为假；CF（A）=0表示对A一无所知。在实际使用时，一般会给出一个绝对值比较小的区间，只要在这个区间就表示对A一无所知。

例如：

CF（阴天）=0.7，表示天气为阴天的可信度为0.7。

CF（阴天）=-0.7，表示天气为阴天的可信度为-0.7，我们也可以理解为阴天的相反面即晴天的可信度为0.7。

具有可信度的规则表示为如下形式：

IF A THEN B CF（B，A）

其中A是规则的前提；B是规则的结论；CF（B，A）是规则的可信度，又称规则的强度，表示当前A为真时，结论B为真的可信度。同样，规则的可信度CF（B，A）取值范围也是[-1，1]，取值大于0表示规则的前提和结论是正相关的，取值小于0表示规则的前提和结论是负相关的，即前提越是成立则结论越不成立。

一条规则的可信度可以理解为当前提肯定为真时，结论为真的可信度。

例如：

IF 阴天 THEN 下雨 0.7

表示：如果阴天，则下雨的可信度为0.7。

IF 晴天 THEN 下雨 -0.7

表示：如果晴天，则下雨的可信度为-0.7，即如果晴天，则不下雨的可信度为0.7。

若规则的可信度CF（B，A）=0，则表示规则的前提和结论之间没有任何相关性。

例如：

IF 堵车 THEN 下雨 0

表示：堵车和下雨之间没有任何联系。即没有办法根据堵车的前提得知是否下雨。

规则的前提也可以是复合条件。

例如：

IF 阴天and湿度大 THEN 下雨 0.6

表示：如果阴天且湿度大，则下雨的可信度为0.6。

规则前提可以是复合条件，复合条件可以通过逻辑运算表示。常用的逻辑运算有“与”“或”“非”，在规则中可以分别用“and”“or”“not”表示。在可信度方法中，具有可信度的逻辑运算规则如下：

1. CF（A and B）=min {CF（A），CF（B）}
2. CF（A or B）=max {CF（A），CF（B）}
3. CF（not A）=-CF（A）

1表示“A and B”的可信度，等于CF（A）和CF（B）中最小的一个；2表示“A or B”的可信度，等于CF（A）和CF（B）中最大的一个；3表示“not A”的可信度等于A的可信度的负值。

例如，已知：

CF（阴天）=0.7

CF（湿度大）=0.5

则：

CF（阴天 and 湿度大）=0.5

CF（阴天 or 湿度大）=0.7

CF（not 阴天）=-0.7

前面提到过，规则的可信度可以理解为当规则的前提肯定为真时，结论的可信度。如果已知的事实不是肯定为真，也就是事实的可信度不是1时，如何从规则得到结论的可信度呢？在可信度方法中，规则运算的规则按照如下方式计算：

已知：

IF A THEN B CF（B，A）

CF（A）

则：CF（B）=max {0，CF(A)} \*CF（B，A）

由于只有当规则的前提为真时，才有可能推出规则的结论，而前提为真意味着CF（A）必须大于0；CF（A）<0的规则，意味着规则的前提不成立，不能从该规则推导出任何与结论B有关的信息。所以在可信度的规则运算中，通过max {0，CF（A）}筛选出前提为真的规则，并通过规则前提的可信度CF（A）与规则的可信度CF（B，A）相乘的方式得到规则的结论B的可信度CF（B）。

如果一条规则的前提不是真，即CF（A）<0，则通过max {0，CF（A）}得到CF（B）=0，表示该规则得不出任何与结论B有关的信息。注意！这里CF（B）=0，只是表示通过该规则得不出任何与结论B有关的信息，并不表示对B就一定一无所知，因为还有可能通过其他的规则推导出与B有关的信息。

例如，已知：

IF 阴天 THEN 下雨 0.7

CF（阴天）=0.5

则：CF（下雨）=0.5\*0.7=0.35，即从该规则得到下雨的可信度为0.35.已知：

IF 湿度大 THEN 下雨 0.7

CF（湿度大）=-0.5

则：CF（下雨）= max {0，0.5}\*0.7=0\*0.7=0，即通过该规则得不到下雨的信息。现实中可以理解为因为湿度大的可信度为-0.5，所以现在不能得到湿度大的前提，所以也无法通过依靠湿度大的规则来得到下雨的结论。

通常情况下，得到同一个结论的规则不止一条，也就是说可能会有多个规则得出同一个结论，但是从不同规则得到同一个结论的可信度可能并不相同。

例如，有以下两条规则：

IF 阴天 THEN 下雨 0.8

IF 湿度大 THEN 下雨 0.5

且已知：

CF（阴天）=0.5

CF（湿度大）=0.4

从第一条规则，可以得到CF（下雨）=0.5\*0.8=0.4

从第二条规则，可以得到CF（下雨）=0.4\*0.5=0.2

那么究竟CF（下雨）应该时多少呢？这就是规则合成问题。我们可以通过一定的计算方法，将两条规则合成。

在可信度方法中，规则的合成计算如下：

设：从规则1得到CF1（B），从规则2得到CF2（B），则合成后有：

CF（B）=CF1（B）+CF2（B）-CF1（B）\*CF2（B），当CF1（B）和CF2（B）均大于零

=CF1（B）+CF2（B）+CF1（B）\*CF2（B），当CF1（B）和CF2（B）均小于零

=CF1（B）+CF2（B），其他

这样，上面的例子合成后的结果为：

CF（下雨）=0.4+0.2-0.4\*0.2=0.52

如果是三个及三个以上的规则合成，则采用两个规则先合成一个，在与第三个合成的办法，以此类推，实现多个规则的合成。

下面给出一个用可信度方法实现非确定性推理的例子。

已知：

R1：IF A1 THEN B1 CF（B1，A1）=0.8

R2：IF A2 THEN B1 CF（B1，A2）=0.5

R3：IF B1 and A3 THEN B2 CF（B2，B1 and A3）=0.8

CF（A1）=CF（A2）=CF（A3）=1

计算：CF（B1），CF（B2）

由R1：CF1（B1）=CF（A1）\*CF（B1，A1）=1\*0.8=0.8

由R2：CF2（B1）=CF（A2）\*CF（B1，A2）=1\*0.5=0.5

合成得到：CF（B1）=CF1（B1）+CF2（B1）-CF（B1）\*CF2（B1）=0.8+0.5-0.8\*0.5=0.9

CF（B1 and A3）=min{CF（B1），CF（A3）}=min{0.9，1}=0.9

由R3：CF（B2）=CF（B1 and A3）\*CF（B2，B1 and A3）=0.9\*0.8=0.72

答：CF（B1）=0.9，CF（B2）=0.72

4.4 专家系统的应用

专家系统是最早走向实用的人工智能技术。世界上第一个实现商用并带来经济效益的专家系统是DEC公司的XCON系统，该系统拥有1000多条人工整理的规则，帮助为新计算机系统配置订单，1982年开始正式在DEC公司使用，据估计它为公司每年节省了4000万美元。

在军事上的著名实例是在1991年初的海湾战争中。当时以美国为首的多国部队遇到一个非常严重的问题，就是怎样尽快地把大量的军队（约50万人）和物资装备（约130亿磅）从美国和欧洲运到沙特阿拉伯境内。政府资助的一个小组在6个星期内开发了一个规划专家系统来规划运输工作，大量的SUN工作站运到了沙特。美国军队使用这个专家系统用于自动的后勤规划和运输日程安排，这项工作同时涉及5万个车辆、货物和人，而且必须考虑起点、目的地、路径以及解决所有参数之间的冲突。AI规划技术使得一个计划可以在几小时内产生，而用旧的方法则需要花费几个星期。由于使用了专家系统，使得重大的运输任务如期完成。军事空运司令部司令汉·约翰逊上将不无得意地说：“历史上还没有哪个国家运送物资和人员能这么多、这么快和这么远。”

清华大学于1996年开发的一个市场调查报告自动生成专家系统也在某企业得到应用，该系统可以根据市场数据自动生成一份市场调查报告。该专家系统知识库由两部分组成，一部分知识是有关市场数据分析的，来自企业的专业人员，根据这些知识对市场上相关产品的市场形势进行分析，包括市场行情、竞争态势、动态、预测发展趋势等；另一部分知识是有关报告自动生成的，根据分析出的不同市场形势撰写出不同内容的图、文、表并茂的市场报告，并通过多种不同的语言表达生成丰富多彩的市场报告。

相比于专家系统在其他领域的应用，医学领域是较早应用专家系统的领域，像著名的MYCIN就是一个帮助医生对血液感染患者进行诊断和治疗的专家系统。我国也开发过一些中医诊断专家系统，比如总结著名中医专家关幼波先生的学术思想和临床经验基础上研制的“关幼波胃脘病专家系统”等。

在我国，石油、地质、医疗、农业、工业、气象等部门，“七五”期间共开发出26个实用的专家系统，其中10个达到了国际先进水平。中科院合肥智能所研制的施肥专家系统，针对性强，效果好，在全国15个省的70多个县推广应用，增产粮食5N5亿千克，棉花7N7万担，节省化肥25万吨，为农民增加收入4亿元。

专家系统创始人之一、美国斯坦福大学教授埃·费根鲍姆在20世纪80年代中期对世界上许多国家和地区的专家系统应用情况做了一番调查研究，他总结说：几乎所有的专家系统至少能将人的工作效率提高10倍，有的达到100倍甚至300倍；使用专家系统节约了大量的资金。

参考导论