

实用专家系统指南

陈 屹 于晓迪 著
王伟元 李晓光
孔壮志

科海总公司培训中心

实用专家系统指南

陈 屹 于晓迪
王伟元 李晓光 著
孔壮志

科海总公司培训中心

前 言

专家系统是近二十年发展起来的一个计算机的新分支，虽然它的历史并不长；然而，它的诞生可以看作是计算机领域的新的里程碑。以往，计算机主要是用于计算和数据处理。由于专家系统的出现，便使人们开始研究如何用计算机来模拟人的思维。目前专家系统已渗透到各个领域，并取得了令人惊叹的成果。因此专家系统是计算机应用的一个极其重要的方面，是计算机应用工作者所应掌握的一门技术。

专家系统涉及的内容很多，本书力求比较系统地论述专家系统的各个方面，使读者能对专家系统的整个体系有一个了解。本书所用的主要参考书有：Donald A. waterman 的英文版的“专家系统指南”和“建造专家系统”，傅京孙的“人工智及其应用”，吉林大学的“专家系统原理”，Elain Rich的“人工智能”，85年人工智能会议论文集，以及“计算机科学”杂志。

本书共分八章：

第一、二章 着重介绍专家系统的一些基本概念，定义和含义，以及专家系统的基本原理。

第三章 主要论述专家系统使用的几种常见的表示方法，以及相应的推理控制策略。由于我们着重在方法上，因而很容易用程序去实现。

第四章 详细地讨论了专家系统使用的几种基于不同理论的不精确推理的方法，这些方法简单而实用，可比较容易结合到所建的系统中去。

第五章 用知识工程师的方法论述了如何从专家获取知识，并重新构造和组织所获取的知识。

第六章 本书在这章比较详细地介绍了专家系统工具的特点、种类及其研究阶段的划分，并对一些实际的工具系统作了简单的描述。

第七章 介绍专家系统的评价方法。

第八章 在这一章，本书以较多的篇幅、刻画了作者实际参加、并取得成功的一个实际系统，以及系统的开发和研究过程。

本书既可作为讲授专家系统的普遍教材，又可作为从事知识工程师工作的人员的参考手册，还可作为从事计算机软件、人工智能等方面工作的人员，以及高等院校有关专业研究生和学生的参考用书。

目 录

第一章 专家系统概述	(1)
1.1 历史背景	(1)
1.2 专家系统特征	(3)
1.3 谁参与专家系统建造	(4)
1.4 专家系统类型	(6)
第二章 专家系统原理	(8)
2.1 符号、谓词演算和推理	(8)
2.2 专家系统使用的技术	(10)
2.3 专家系统的组织结构	(13)
2.4 一个专家系统示例——植物学专家系统	(15)
第三章 专家系统的知识表示	(22)
3.1 基于规则的表示方法	(22)
3.2 语义网络表示方法	(27)
3.3 框架表示方法	(41)
3.4 单元表示方法	(45)
3.5 剧本表示方法	(50)
第四章 专家系统中的不精确推理	(53)
4.1 似然推理	(53)
4.2 主观Bayes概率论方法	(55)
4.3 确定性理论方法	(61)
4.4 可能性理论方法	(64)
4.5 证据理论的方法	(68)
第五章 专家系统的建造	(74)
5.1 专家系统的适用性	(74)
5.2 知识获取过程	(77)
5.3 知识获取的主要阶段	(84)
5.4 构造专家系统的准则	(90)
第六章 专家系统工具	(94)
6.1 专家系统工具的特点及种类	(94)
6.2 专家系统工具的发展阶段	(103)
6.3 EMYCIN系统	(114)
6.4 KAS系统	(116)
6.5 EXPERT系统	(121)
6.6 HEARSAY-III系统	(124)
第七章 评价专家系统	(128)

7.1	为什么要评价专家系统.....	(128)
7.2	专家系统评价中的设计问题.....	(130)
7.3	何时评价专家系统.....	(132)
7.4	如何评价专家系统.....	(134)
7.5	一个实例的研究: R1	(137)
第八章	一个实际系统PRES的开发及其工具系统MES1的研究	(141)
8.1	PRES系统概述.....	(141)
8.2	油气资源评价体系及其概念模式.....	(142)
8.3	PRES系统结构及知识表示.....	(145)
8.4	工具系统MES1简介.....	(159)
8.5	生油条件评价专家系统软件简介.....	(175)
8.6	问题讨论.....	(178)

第一章 专家系统概述

§ 1.1 历史背景

本世纪七十年代初期的计算机革命诞生了计算机科学的一个新的分支——专家系统。由于计算机硬件的迅猛发展，使得计算机的价格下降得如此之快就连最乐观的科学家也无法估计得到。当计算机硬件工作者正在研制微片技术的时候，软件工作者，即设计和建立计算机程序的人们，正在进行一项带有根本性突破的研究。可是，这项突破并不是发明一种新的方法，诸如利用激光束或研制更快、更小的微片来编码信息；而是在计算机这个新奇领域中的一个概念上的突破：这个概念就是现在被人们称为的人工智能—AI。

AI科学家的目的是不断地探索、研究在某种程度上能够进行思维的计算机，也就是说，这种计算机解决问题所使用的方法具有人类的智能。专家系统正是适应这种要求、并经过了二十年的努力而取得的成果。图1.1表示专家系统发展前后的历史阶段。

在60年代，AI科学家试图通过寻找解决各种各样问题的一般性方法来模拟思维的复杂过程。他们把这些一般性的方法用通用程序加以实现。然而，尽管得到了一些有意义的成就；但是，这种策略不能产生出突破性的思想。研制通用程序太难并且最终收效不大。因为，要使单一的程序能够处理问题的类型越广泛；那么，它对每种类型问题的处理能力就越浅薄。

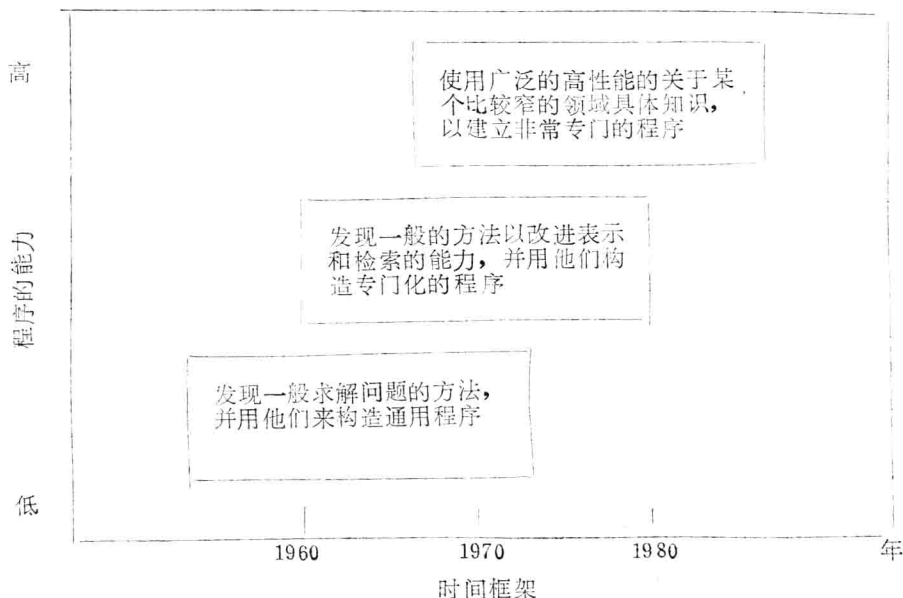


图 1.1 人工智能研究重点的转移

因此，人工智能科学家决定：必须用其他的方法使计算机程序具有智能。如果使一个完整的程序具有通用性是非常困难的话，那么是否可以把研究方向集中在专门化程序的方法和技术而将其取而代之呢？所以，在七十年代期间，科学家们着重于表示以下一些问

题：如何将问题形式化，以便问题容易被解决和被检索？如何控制寻找求解方案，以至不用过多地占用计算机存贮空间和时间？这种策略虽获得了一些成功，但仍没有获得突破性的进展。

直到70年代末期，人工智能科学家才开始注意到了某些事情的重要性：即一个程序的问题求解能力在于它所具有的处理知识的能力，而不是在于它把问题形式化工作及使用的推理方法做得有多么的漂亮。这是一个思想上的突破，这种突破可简单地用一句来说明：

要使一个程序具有智能，必须提供给这个程序大量的高性能的具体的关于某个问题领域的知识。

专家系统程序的开发，是这种向基于知识的方法转变的结果之一。已经证明：仿效某种专家求解问题的方法，比起编写模仿一个孩子领悟、理解语言，或作“常识性”推理的能力的程序来，要容易得多。这似乎使人难以理解。事实上，许多人类专家，正是由于他们对某个小的问题拥有广泛的知识而著名的。也正是由于这种限制，才有可能为计算机程序提供足以使它能有效地完成任务的知识。

建立专家系统的过程通常被称为知识工程。这种过程常常要涉及到专家系统的建立者同一个或一个以上的某一问题领域专家的一种特殊的交互形式。此时，我们把专家系统的建立者称为知识工程师。知识工程师的任务是把人类专家用来解决问题的过程、策略以及经验抽取出来，并用这些知识建成专家系统。见图1.2所示

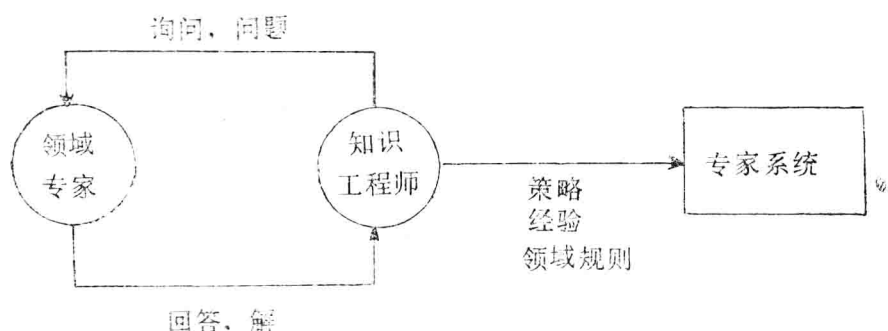


图 1.2 知识工程：把从专家得到的知识转换成计算机程序

因此，专家系统这个领域就是研究用解决某专门问题的专家知识来建立人机系统的方法和技能。专家能力包括某个具体专业领域的知识，对领域中问题的理解以及解决这些问题的技能。任何专业的知识一般都可分为两类：公开知识和个人知识。公开知识包括定义、事实和理论，这些往往已收录在教科书和参考书中。但专家的能力往往不仅限于公开发表的知识。专家们一般都使用的是在已发表的文字材料中找不到的个人知识。我们把这种凭经验求解问题的个人知识称之为启发式知识。启发式知识能使专家在适当时候作出有效的猜测，辨别出有希望的解决途径，并有效地处理错误的或不完全的数据。所以，解释并重新组织这种知识是建立专家系统的核心任务之一。

到目前为止，专家系统已渗透到了各个领域当中，它们解决的问题一般都属于下列范畴：解释、预测、诊断、故障排除、设计，规划、监视、修正、教学和控制。专家系统不仅在AI的年轻的领域中有应用成功的记录，而且在工业和政府中也直接起着重大的作用。下面列出部分成果：

- PROSPECTOR发现了实际价值可能超过1亿美元的钼蕴藏量 (John Gaschnig, SRI)

- R1在数字设备公司 (DEC) 可根据顾客要求为VAX计算机系统配套, 尽管一些专家认为这是不大可能做到的 (McDermott 1981)

- DENDRAL在几年前就显示了超人的性能, 现在它每天为数百名各国的用户在 进行化学结构的分析 (Lindsay等人1980)

- CADUCEUS装备有比任何人都要多得多的内科知识, 并能正确地诊断使 专家都难以解决的实际病例 (Pople, Myers和Miller 1975, Pople 1981; Miller, Pople 和 Miller 1982)。

尽管专家系统在各种领域产生了一些确有成效的成果, 但是现在它的理论基础还很薄弱, 仅获得一些有限的进展。虽然一些系统使用了一些原理并进行了系统化工作, 但是对每个新的应用都需要进行创造性和挑战性的工作。迄今为止, 专家系统仍是一个经验性很强的领域, 它很少采用一般的理论。尽管如此, 核心问题已经露出, 不少工具和技术已经存在并已从某个应用系统移植到了另一个应用系统。

§ 1.2 专家系统的特征

什么是专家系统? 尽理目前专家系统已应用得非常广泛, 但是, 给专家系统下一个精确的定义却不是一件容易的事情。这里我们引用Richard O. Duda给专家系统下的一个粗略的定义。

所谓专家系统就是一个计算机程序, 它对一些重要问题提供具有专家水平的解答。专家系统具有以下三个特点:

- a. 启发性: 它能使用判断性知识及已确立理论的形式知识进行推理;
- b. 透明性: 它能够解释其推理过程并对有关他的知识的询问作出回答。
- c. 灵活性: 它能够把新的知识不断地加入到已有的知识库中。

专家系统的核心是建立系统过程中获取到的有效知识的实体。为了使决策简明, 这些知识应具有明晰性并有效地加以组织。因此, 知识的获取和整理是专家系统最重要的方面之一。这意味着它胜过为完成某种任务而进行的单纯的程序设计。

一个专家系统的最为有用的特征是用来提供帮助进行问题求解的高水平的知识。这种专门知识可以代表这一领域最有影响的专家的思想, 从而使问题求解做到准确、有效并令人信服。通常认为, 一个人之所以能成为专家; 首先, 当然是他的行为的质量。不论你执行任务的速度有多么快, 但如果执行结果不好或判断不正确, 那就不能使人感到满意。另一方面, 做出判断的速度也是一个重要的因素; 即使是最准确的诊断。倘若作得很慢, 也可能没用。较高层次的模型是对应具体现象的高度集中的, 抽象的或浓缩的描述。在一个推理循环中, 可以匹配或识别的结构层次越高, 则可以复盖的概念范围就越广。可编入专家系统的这类规则的一个有趣的特点是它的不完全性, 它们被认为仅仅是经验性的或启发式的规则。但是, 启发式的因素可能不象这种高层次规则对搜索范围减少的影响显得那么重要。

专家系统的另一个有用的特征是它的预言模型化的能力。系统可以作为已知领域的信息处理或问题求解模型。它能对已知问题的情形给出希望的解答并且表示出怎样在新

的情况下进行改变。专家系统可以对在新的情况下怎样发生变化作详细的解释。这样可以让用户去评价一个新的事实或数据的潜在的作用，并理解他们同结论的关系。类似地，用户可以评价新的策略或过程对求解问题的作用，也可以通过加入一条新的规则或修改现存的规则来看对求解问题的作用。

确定专家系统精髓的知识体，还提供了一种附加特征——一个惯例存贮。如果通过和某办公室或部门的重要人物交互来发展知识库，则知识库可以反映出这组人员的当前的策略或操作过程。这种对知识的编译使得高层次的意见变成一致，并作为参谋人员使用的最佳策略和方法的永久记录。当关键人物不在现场时，他们的专业知识仍旧存在那里。这对人员频繁调动或不固定的单位如商业特别是军队及政府部门等都是非常重要的。

专家系统的另一个特征是可以作为主要人员或重要的高级人员的训练工具。专家系统可以设计用来提供这种服务，这是因为他们已经具有必要的知识以及解释他们推理过程的能力。当然，这需要增加相应的软件以提供训练者同专家系统的友好的界面，并且这还包括示教的知识及用户模型化的知识。作为一个训练工具，专家系统向新的参谋人员提供大量的经验和策略，从这些经验和策略中可以学习所推荐的政策和方法。这种系统同样也可以训练具体某些任务的新手。

§ 1.3 谁参与专家系统建造

假设我们把专家系统的研制比做一场戏，那么唱这场戏的主角是专家系统、领域专家、知识工程师、专家系统建造工具以及用户。图1.3概括了他们的主要作用以及他们之间的相互关系。

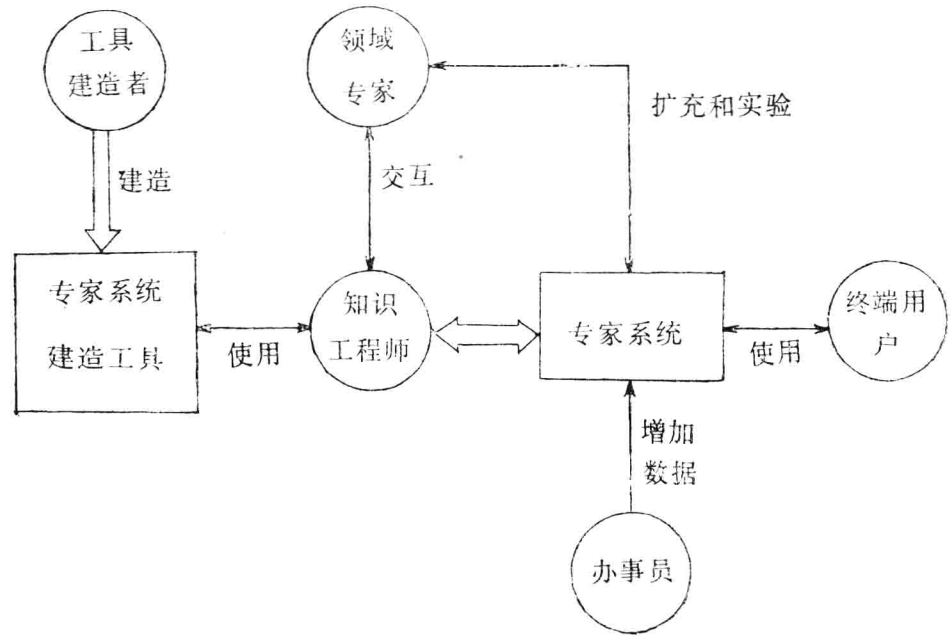


图 1.3 专家系统这场戏中的各个角色

专家系统是问题领域求解程序或计算机软件的集合。它之所以被称为系统而不是程序是因为它包含了问题求解和相应的支撑环境两个部分。其中支撑环境是用来帮助用户问题

求解程序进行交互，这部分还可以包含高级修改程序。利用它可以帮助专家系统的建立人员测试和评价程序的编码，作为一个友好的编辑工具还可以帮助专家修改专家系统中的知识和数据，并且先进的图象设备在专家系统运行时能帮助用户进行输入、输出信息。

领域专家是指具有丰富的领域知识和表达能力，在其特定的领域内对一些问题的求解具有一定影响的人。专家利用其技能和经验使对问题的求解过程更为行之有效。专家系统通常把这些问题求解策略模型化。虽然专家系统通常把一个或一个以上的专家思想模型化，但它还可以包含从其他知识来源得到的专门知识如书本上的和杂志上的。

知识工程师是具有一定计算机科学和人工智能背景的人。他知道如何建立专家系统。知识工程师同专家们进行交互，抽取、组织和整理从专家中获取到的知识，确定如何用专家系统所能接受的形式表示知识，并且还可以作为程序设计者进行编制程序。)

专家系统建造工具是知识工程师或程序员建立专家系统时所使用的程序设计语言。这些工具不同于通常的程序设计语言。这些工具是作为表示复杂的高水平概念的一种便利的手段。用AI的行话来说，工具这一术语通常涉及到建立专家系统所使用的程序设计语言和支撑环境。

用户是当专家系统建成后使用专家系统的人。用户可以是利用系统帮助发现新矿藏的科学家，可以是利用系统帮助分析案例的律师或用来学习更多有机化学知识的学生。用户一词多少有点含糊，一般是指终端用户，即专家系统的服务对象。然而在本书中，它将是任何使用专家系统的人。如图1.4，用户可以是修改专家系统建立语言的工具建造者，可以是精炼系统现存知识的知识工程师，可以是依赖系统提供建议的终端用户，甚至还可以是向系统录入数据的录入员。

专家系统的建造工具和专家系统本身是两个完全不同的概念，然而这两个概念常常被人们所混淆，即便是计算机科学家有时也很难说出他们之间的区别。我们知道，专家系统建造工具是由表示知识和存取知识所使用的语言以及支撑环境两个部分组成的，而所谓的支撑环境是指帮助用户同专家系统的问题求解部分进行交互的程序。按照这种定义，支

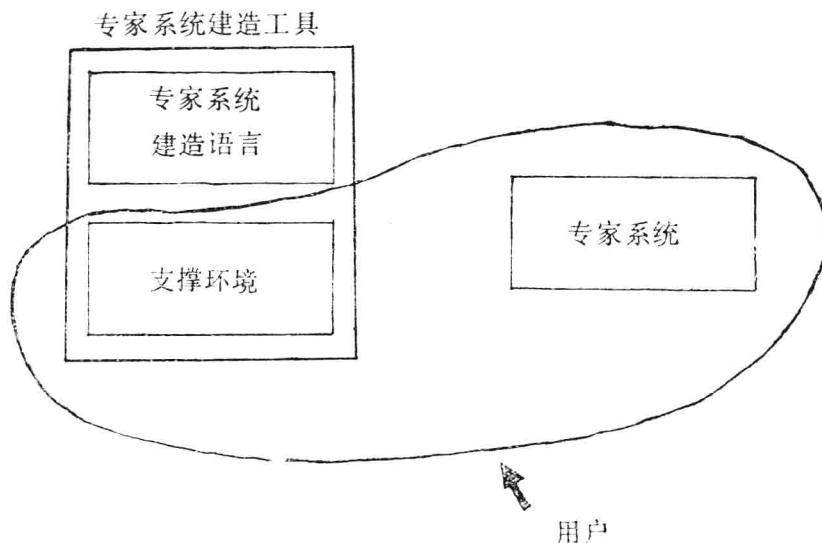


图 1.4 专家系统建造工具不是专家系统

撑环境也可以是完整的专家系统中的一部分，因为用户同专家系统的交互作用是通过支撑环境进行的，所以产生出这种混淆也是十分自然的。因此，我们有必要来区分这两种概念。图1.4详细地说明了这种区别。

§ 1.4 专家系统的类型

大多数知识工程的应用可分为几种不同的类型，表1.1汇总了这些类型。

解释系统根据观察结果来推出情况描述。这个范畴包括监视、语言理解、图像分析，化学结构说明，信号解释和许多种智能的分析。解释系统的解释是先赋予符号意义的观察数据，再用数据描述这些情况或者系统状态。

表1.1 知识工程应用的一般分类

分 类	处 理 的 问 题
解 释	由传感器数据推断情况概述
予 测	推断出已知情况的可能结果
诊 断	由观察情况推断出系统故障
设 计	根据约束条件配置出目标
规 划	设计行动
监 视	将观察情况与规划中的弱点对照
调 试	执行一个指定补救办法的计划
修 正	执行为实施给出的纠正方法而制定的计划
教 学	诊断、调试和修正学生的行为
控 制	解释、预测、修正和监视系统行为

预测系统根据给定的情况推出可能的后果。这一范畴包括天气预报、人口预测、交通预报、农业产量估计和军事预测。预测系统通过拟合给定情形的参数值应用动态参数模型。可以从模型中推导出来的结果形成了量的基础。如果不考虑概率估计，则预测系统能产生大量的可能情况。

诊断系统根据观察的情况推断系统的故障。这个范畴包括医学的，电子的，机械的和软件诊断。诊断系统使用两种技术将观察到的不正常表现与潜在的原因相比较。一种方法主要是用一个表现状态与诊断的对照表。另一种方法是将系统设计的知识同有关设计、实现或组成部分中潜在问题的知识相结合，产生同观察情况一致的故障。

设计系统开发满足设计问题要求的项目结构。这类问题包括线路设计、建筑物设计及财政问题。设计系统根据相互之间的各种关系来建立对项目的描述，并检验这些构件是否满足提出的要求。此外，许多设计系统还力求尽可能地减少那些测量费用以及测量其它的潜在设计中不希望有的特性的目标功能。这一设计问题的目的也可以通过把目的成就和目标功能合并的方式以有目的地探寻性态来归类。

规划系统设计行动方案。这些系统专门处理有关执行某些任务的设计问题，包括自动程序设计，机器人、线路、通讯、实验和军事计划问题。规划系统用行为模型推断计划执行的效果。

监视系统把对系统行为的观察同对计划成败起关键作用的特点进行比较。这些特点(或弱点)对应于计划中潜在的缺陷。一般地说,监视系统都是用两种方式来确定弱点。一类弱点是指某些一旦违犯就会使整个计划的基础理论失败的条件,另一类弱点产生于计划的某些潜在的影响会违犯计划的条件。这些要害对应于预测状态中的操作故障,现已有许多计算机辅助监视系统用于核电站、机场调动,疾病,规章制度和财政管理任务中,但这些系统还都处在实验室阶段,还不是正式的专家系统。

调试系统规定排除故障的方案。这些系统依靠计划、设计和预报的能力,对诊断出的问题产生修正的规定或建议。已有的计算机辅助排错系统以智能知识库和文本编辑的形式用计算机程序设计,但还都称不上专家系统。

修正系统开发并执行已诊断出问题的修改计划。这种系统包括排错,计划和执行的能力。计算机辅助系统已出现在自动化,网络、航天控制系统,计算机维护等领域,而专家系统还是刚刚进入这些领域。

教学系统诊断和修正学生的行为。这些系统主要包括学生感兴趣的诊断和排错子系统。它们首先建立解释学生行为的学生知识的假定描述,然后诊断学生知识的弱点,并确定一个纠正方案,最后规划一个交互式的教材,把修改的知识传授给学生。

最后一类系统称为控制。专家系统适用于管理一个系统的整体行为。为此,控制系统必须反复地解释当前的形势,预测未来,诊断预防问题的原因,形成纠正计划,并监视其执行以获得成功。控制系统处理问题的实例有机场调度,商业经营管理、战场管理和任务控制等。知识工程的技术还可以处理许多用传统数学方法不易解决的控制问题。

第二章 专家系统原理

§ 2.1 符号、谓词演算和推理

2.2.1 符号

Newell和Simon (1976) 在他们对物理符号系统的讨论中, 定义了作为物理模式的符号, 它可以作为符号结构的组成部分出现, 而符号结构是一些以某种物理方式相关的符号彼此连接起来而组成的。因此, 完全可以把符号看作为字符串或符号结构串, 如包含符号的表结构的数据结构。下面是符号的例子:

Apple
Runing
3.14159

下面是符号结构的例子:

(on Block1 Block2)
(Plus 5 X)
(Same-as (Father-of Pete)
(Father-of (Brother-of Pete)))

早期的AI研究对计算机科学的一个贡献是发明了符号计算用的表处理语言。这些语言提供了表处理的基本功能和存贮管理。

2.1.2 谓词

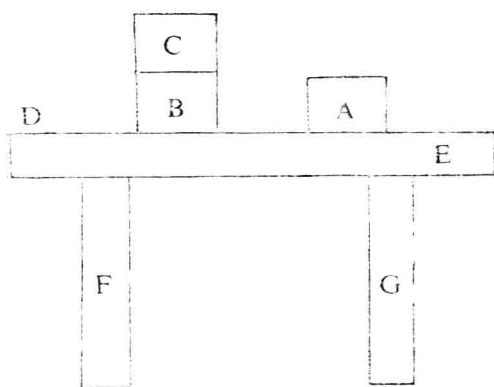
谓词演算是一种广泛研究的符号结构的形式语言, 它可在计算机内表示。让我们用一些灵活的符号结构表示事实的简单例子来考察为什么推理方法必须要知识密集才能有效。显然, 专家系统中的推理已超出了经典逻辑学的推理, 谓词演算的知识仍是理解表示和推理问题的根本基础。

在图2.1中, 桌子上放了一些小块。一些符号结构表示图中的信息, 这些符号结构是以谓词演算的一种变异的句法(前缀表示法)书写的, 由项和谓词符号组成。项用于表示事物名称, 而谓词表示事物之间的关系。本例中, A、B、C、D、E、F、G、BLOCK、TABLE、TABLE-TOP、TABLE-LEG是项, 而IS-A、PART-OF、ON是谓词名。在有关逻辑学的书中, 象图2.1中的这些简单谓词称为命题或原子公式。

除了常量和谓词使用符号, 谓词演算还可以使用函数和逻辑连接。函数表示实体之间的对应关系。例如, 若TOP-OF是一个函数符号, 则

(TOP-OF D)

表示桌子D的顶部(使用的语法与通常谓词演算的区别仅在于谓词符号的位置不同。选择前缀语法是为了与符号处理语言(如LISP)中的表结构类似)。逻辑连接用于公式的组



(IS-A A BLOCK)
 (IS-A B BLOCK)
 (IS-A C BLOCK)
 (IS-A D TABLE)
 (IS-A F TABLELEG)
 (IS-A G TABLELEG)
 (PART-OF E D)
 (PART-OF F D)
 (PART-OF G D)
 (ON A D) (ON C D)
 (ON B C)

图 2.1 放着小块桌子表示信息的谓词演算

合，包括“ $\wedge$ ”（与），“ $\vee$ ”（或）和“ $\rightarrow$ ”（蕴涵）。例如，公式“小块A或者在桌子D上或者在小块B上”可以用函数形式表示为

$\text{ON} (A D) \vee \text{ON} (A B)$

用现在的表示形式可表示为：

$(\text{OR} (\text{ON} A D) (\text{ON} A B))$

符号“ $\neg$ ”（非）尽管被用于否定一个公式的值，而不是连接两个公式，有时也称为连接词。

谓词演算表示语句时，有时会用量词，如“All Block are small”，以传统表示形式应表示为：

$(\forall X) [\text{BLOCK} (X) \rightarrow \text{SMALL}(X)]$

它的表结构表示为：

$(\text{ALL} (X) (\text{IF} (\text{IS-A } X \text{ BLOCK}) (\text{SMALL } X)))$

或 $(\text{ALL} (X) ((\text{IS-A } X \text{ BLOCK}) \rightarrow (\text{SMALL } X)))$ 。

这些公式用全称量词“ $\forall$ ”（all）表示对于定义域内变量X赋与实体的所有值公式都是成立的，这时，我们说该公式是对X量化了。在谓词演算中的另一个重要的量词是存在量词“ $\exists$ ”（exist），这个量词表示对于定义域中变量赋与实体的一些值公式也是成立的，如果允许对一些项量化，但不允许对谓词和函数符号量化的话，那么这种演算称为是一阶的。

由项和原子公式用连接符和量词组成的公式称为合适公式类（wffs）。谓词演算提供了许多组合公式的合适规则。

在谓词演算中，一个合适的公式可以通过语言的各元素与定义域中实体和关系之间的一个对应给出一个解释。对每个常量符号必须规定定义域中的一个相应的实体，且给每个谓词必须规定定义域中的一个相应关系。若计算机数据库是由符号表示组成时，则数据库的设计者可选择使用的谓词和项的词汇表并确定其含义。使用这个数据库时，就产生一个公式的解释作为定义域中的论断。例如，图2.1中的公式可解释为有关图中桌子与小块的事实。公式 $(\text{PART-OF } E D)$ 表示这样一个事实E（桌子顶部）是D（桌子）的一部分。

谓词演算的优点是这种语言可以很好地理解对许多有关语句的表示。De Morgan定

律就是一个处理标准解释的例子。因为，在谓词中， $(\text{NOT} (\text{AND } A \ B))$ 等价于 $(\text{OR} (\text{NOT } A) (\text{NOT } B))$ 。另一个例子是使用括号表示量化的辖域。下面两个公式分别表示 “Everyone has someone that he loves” 和 “There is someone who loves everybody”。

$$\begin{aligned} & (\text{ALL } (X) (\text{SOME } (y) (\text{Loves } x \ y))) \\ & (\text{SOME } (x) (\text{ALL } (y) (\text{Loves } x \ y))) \end{aligned}$$

2.1.3 推理

为了使系统能推理，它必须能从已知的事实推导出新的事实。这需从旧的符号结构中，动态地产生出新的符号结构。在谓词演算中，对一组 wffs 应用推理规则能产生新的 wffs。例如，在图2.1的情形下，我们有下列关于谓词 ABOVE 的规则：

$$(\text{ALL } (x) (\text{ALL } (y) ((\text{ON } x \ y) \rightarrow (\text{ABOVE } x \ y))))$$

“如果x在y的上面，则x是在y的上方”。

$$\begin{aligned} & (\text{ALL } (x) (\text{ALL } (y) (\text{ALL } (z) (\text{AND } (\text{ABOVE } x \ y) \\ & (\text{ABOVE } y \ z)) \rightarrow (\text{ABOVE } x \ z)))) \end{aligned}$$

已知图3.1中的 wffs，我们可以应用第一条规则推导出小块在它所放的物体之上，即：

$$\begin{aligned} & (\text{ABOVE } A \ D) \quad (\text{ABOVE } C \ D) \\ & (\text{ABOVE } B \ C) \end{aligned}$$

第二条规则使我们能用 ABOVE 关系的传递性推出： $(\text{ABOVE } B \ D)$

从逻辑上讲，这是不完备的。因为叙述规则仅是领域上的规则，也就是图2.1定义中的 wffs。还没有提到逻辑推理规则——有关用已知公式产生导出公式的逻辑的规则。所以，上述推理隐含地使用了两条推理规则，即假言推理和全称消去推理：

假言推理： $A, A \rightarrow B \vdash B$

全称消去推理： $A(\forall x), w(x) \vdash w(A)$

这里符号 $\vdash$ 的意思是“产生”。假言推理可陈述为从 wffs A 和 $A \rightarrow B$ ，我们可以推出公式 B 。与之类似，全称消去推理只是用上了全称量词。从公式 $(\forall x) w(x)$ 产生出公式 $w(A)$ ，这里 A 是任意常量符号。常用的全称消去推理的例子是如下推理公式：“张三是人”并且“所有的人是要死的”，产生出“张三是要死的”结论。

将其直接应用到专家系统是用计算机中的存贮公式表示“现在相信什么”，并应用推理规则导出新的事实和相信：这种思路是好的，但作为专家系统中表示知识的手段还不够。其难度是在手里只掌握事实是不够的，还必须要知道如何使用它们。

§ 2.2 专家系统使用的技术

目前，大多数计算机是根据传统程序的决策逻辑来完成任务的，但这些程序不容易容纳大量有效知识。程序由两个不同的部分即算法和数据组成。算法决定如何解决某类特殊问题，数据描述手头特定问题的参数。然而人类知识并不适合这种模型，因为大量的知识是由一些专门的技能片段构成的，运用大量知识需要有新的方法将决策片段组织成有用的实体。专家系统将这些片段收集在知识库中，然后通过访问知识库来对某些特殊问题作出

推导。因此，专家系统在其组织方式，收编方式，执行方式，以及与人交往中产生的印象等方面与传统程序是截然不同的。

专家系统的开发者要收集大量的判断性和启发性知识片段，他们要自动解决这样的问题、即要求机器按照最适于手头数据的推理路径来工作，要使系统在发展中不断接收新的知识，要使系统在必要时运用自己的知识对其行为作出有意义的解释。

图2.2给出了专家系统的基本构造块。基础层是构成几乎所有应用的基础技术：符号程序设计，谓词演算、搜索和启发式方法。专家系统是通过产生候选解，然后加以评价来解决问题的。通常，求解涉及到对所给数据使用启发式的规则，推导出逻辑的或可能的结果，并证明这些结果满足目标。这些行动相应谓词演算的推理和证明机制。虽然目前大多数知识系统实际并不使用形式逻辑程序，但它们达到了相同的效果，谓词演算为其一般受限的推理和证明能力提供了形式的基础。

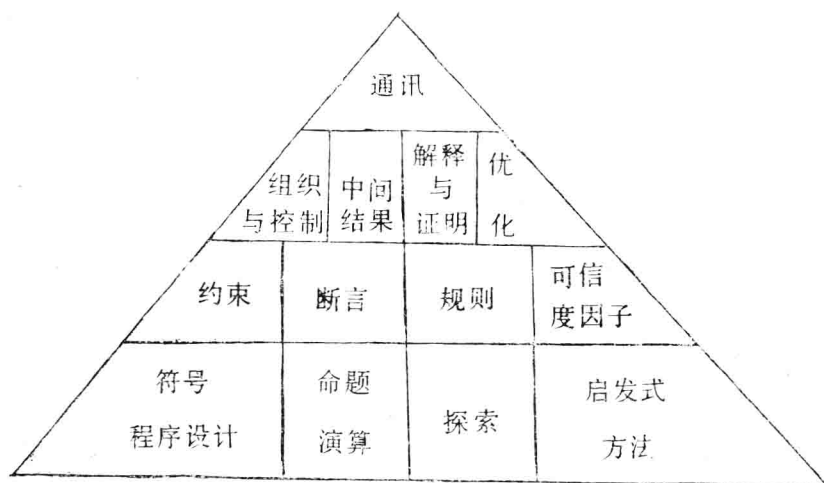


图2.2 知识系统使用的技术

在知识系统技术的第二层，给出了最常用的知识表示形式：约束、断言、规则及可信因子。约束的例子如：

“两个不同的物理实体不能占据同一个空间”。

“在人寿保险单中指定的每一个受益的人或保险一方的健康必须有经济利害关系。”知识系统包含约束是为了表示对允许状态、值或结论的限制。事实上，某些知识系统主要是通过识别和满足复杂符号约束集的能力来导出值。因此，专家系统扩充了可计算的满足约束的问题类。知识系统与主要考虑线性约束的计算机系统不同，它可考虑任意符号的约束，如空间、时间或逻辑关系的要求。

断言数据库提供了存贮和检索命题的方法。断言对应于真命题，即事实。断言的例子为：

“瑞典国王访问我公司，以探讨与西海岸高技术公司的可能关系。”

“摩根是一只狗”

“摩根是我的狗的名子”

许多简单形式的断言对实现关系数据库是很有用的，但是更复杂的模式却不能如此。一般说来，目前大多数知识系统都有自己专门断言的数据子系统。

规则表示特定形式的陈述性或命令性知识。为了说明命令性规则，试看例子：

“如果发现病人发烧或流鼻涕，则怀疑他可能是感冒”。该规则告诉知识系统是如何动作的。一条相应的陈述性规则是告诉系统可以相信什么，而未说明怎么行动。

“如果病人感冒，则他可能发烧或流鼻涕。”大多数知识系统都运用上述的一种或两种规则形式。一般地说，陈述性规则描述世界上事物的工作方式。另一方面，命令性规则指示了知识系统在运行过程中必须使用的启发式方法。

可信因子指定了知识系统对其数据、规则或结论应赋予怎样的置信程度和有效程度。这些可信因子可能反映了处理误差和不确定性的各种不同模式。某些系统运用贝叶斯条件概率来估计不确定性。另外一些系统则利用完全主观的方式，如1.0表示确定，-1.0表示否定命题的确定，0.0表示不表态或无证据。许多人曾都致力于改进可信因子技术。但这种努力在很大程度上可以证明是毫无结果的。首先，正由于不存在有效而形式化的其他可选方法，因此专家系统必须对其结论的强度准确作出估计；所以，人们不能利用任何形式化消去这一决策过程的主观因素。其次，许多不同的可信因子模式都工作得同样好。而知识系统工作良好是在于它能模拟人的特性，如果人需要计算复杂的数字公式来确定可信因子，那么他是不可能很好地解决问题的。人之所以工作得十分出色，主要是因为他的知识一般发挥得足够好。这种知识对解决重要问题是有效，足够和全面的。因此，专家系统正是利用了人类知识的这种功能。

在知识系统技术的第三层给出了组织与控制、中间结果、解释与证明和优化。专家系统按照所体现的结构设计原理来组织和控制自己的活动。例如，通过查找具有充分可信的证据，一个医疗诊断专家系统可以从它知道的所有潜在的疾病出发进行向后推理。首先它可以考虑把具有较高优先级的疾病作为最可能的疾病。然后也可以按照最可能的特定并发症来寻找证据。仅当遇到大量完全不一致的数据时，才开始考虑下一个疾病。当专家系统按此方式运行时，可采取深度优先的逆向拉链控制模式。每一个不同的控制模式有其相应的知识库组织方式及搜索该知识库和应用知识的适当修改的推理机制。因此，控制和组织是密切相联的。

中间结果出现在所有的系统中。由于专家系统可能面临高难性能的要求，因此它常常需要有效地使用中间结果。例如，在一个逆向拉链的系统中，所考虑的几个可能选择都需要少量公共证据。这种证据的收集可能需要大量的计算和推理。一旦知识系统评价了该证据，它将保存起来以备以后再使用。在每种组织和控制模式中，类似的临时存贮和重用问题都会出现。目前，大多数专家系统都是采用专用的或即席的方法来实现这些功能的。

由于知识系统一般要解释和论证它的结论，所以它必须兼顾各类可能用户的利益。许多应用领域的终端用户需要听听知识系统的建议。而系统是通过解释结论是怎样产生的而能给用户一个合理的印象。为了形成一个解释，系统将专家的启发性规则和断言变换成若干的推理路线。一条推理路线给出一个初始假设集和启发式规则集是怎样产生一个特殊结论的过程。终端用户一般认为这些解释象规则本身一样也是似乎可信的。与知识系统交互的其他人也可利用这种解释能力。知识库维护人员（包括专家和技术人员）则是通过测试实例来估价性能，并对专家系统一再论证，即论证系统是否得出了正确的决策以及是否依据正确的理由。

在知识系统中，优化技术起着重要的作用。和其他计算机的应用一样，知识系统必须