

事务处理专家系统及开发工具Guru

北京工业大学

刘椿年 钟 宁 刘树钧 译

27

科 海 培 训 中 心

一九八七、十一

事物处理专家系统及开发工具Guru

北京工业大学

刘椿年 钟 宁 刘树钧 译

科 海 培 训 中 心

一九八七·十一

译者附言

人工智能，特别是专家系统技术目前已在国内引起了广泛的兴趣。将专家系统技术汇入事务处理这一计算机应用的主流将极大地提高管理决策过程的效率和质量，具有深远的理论和实际意义。美国MDBS公司1986年推出的协同集成化人工智能环境（或称为事务处理专家系统开发工具）Guru是基于这种思想的最新软件，它在IBM个人计算机上运行，国内已得到该系统的演示软盘。

美国Purdue大学的C.W.Holsapple博士和A.B.Whinston博士合著的《事务处理专家系统及其开发工具——Guru》（英文原名为Manager's Guide to Expert Systems Using Guru）一书使用Guru从事务处理的角度讲解八十年代的专家系统技术，立论严谨，表述通俗，对于广大的管理人员，计算机系统开发人员，计算机科学、管理科学和经济学的师生都有极大的参考价值。两位作者是美国管理科学、经济学和计算机科学界的知名学者，著述颇多，且具有丰富的实践经验和教学经验。

为了使国内读者了解专家系统技术与常规的事务处理技术（数据库、电子表格、图形生成、文字处理和过程性模型等）相结合这一新的潮流并应用到自己的工作上去，我们根据美国DOW JONES-IRWIN出版社出版的1986年英文原著并参考日文译本将此书翻译成中文。配合Guru的演示软盘，本书将对国内人工智能技术的推广普及起到积极的作用。本书既可作为一般的专家系统技术的教材和参考书，又可作为使用Guru开发事务处理专家系统的指南。

最后，为了使读者能够顺利学会Guru演示盘的用法，我们把“Guru演示软件使用指南”翻译出来作为附录，放在本书的最后。

北京工业大学

刘椿年、钟宁、刘树钧

1987.8于北京

1987.8/25

序 言

人工智能正在急速地从象牙之塔走向现实世界。在人工智能的各个分支中,应用专家系统被期望获得最大的经济效益。工业界估计,专家系统的市场1984年几乎为零,而到1990年将增长到十亿美元以上。我们认为,专家系统应用的最大潜在领域是事务处理——特别是管理决策过程的支持。本书探讨事务处理专家系统的潜力并解释如何将这种潜力变成现实。

不久前,人工智能的奠基人之一,诺贝尔奖金获得者Herbert Simon指出:

“从现在起用不了几年,一本书就难以概括写出人工智能技术在管理中的应用了。”*

几年很快地过去了,Simon的预言确已成为事实。1981年提出的将人工智能推理技术与通常的事务处理计算设施相结合的理论基础在今天已产生了开发人工智能决策支持系统(如事务处理专家系统)的各种可能性。本书是首批严肃认真地考虑人工智能在管理方面应用的著作之一。本书不打算全面概述所有人工智能技术在管理方面的应用,而仅就事务处理专家系统的可能性进行深入的探讨。

这本指导管理人员学习专家系统的著作将使读者熟悉事务处理专家系统,它们是什么,如何开发它们,它们是如何工作的以及它们的意义。对事务处理专家系统的全部问题我们都是从事务处理计算的角度加以考察的。不是计算机科学家或程序员的读者也能从本书得到收获。若熟悉某些常见的事务处理软件,读者将能更全面地理解本书内容的意义,但它们并不是必需的预备知识。**

本书的一大特色是提供了一个事务处理专家系统的应用实例——销售定额的设定,此例子随着本书行文的展开而不断完善。每引进一个新的概念,都用这个例子对之加以解释,以加深理解。销售定额问题是常见的,很容易被管理人员读者们所理解。同时,它又是足够丰富的,可为引入的各种概念提供进行解释的实例。这一不断丰富和深化的事务处理应用实例使本书的叙述具有连续性的重要特点,它还提示了那些读者可能希望用到自己的事务处理专家系统中去的功能。

本书以非形式化的、面向实用的风格把专家系统的基础知识与构造事务处理专家系统的实践活动揉合在一起。本书避开了对于理解事务处理专家系统不是必要的那些抽象概念和技术。我们尽力使本书能为非技术人员所理解,同时使它对多种类型的读者均有实用价值。首先,读者可以是希望对人工智能及其在事务处理方面的应用有所了解的管理人员。特别地,本书将帮助这类读者明确他们可以从事务处理专家系统中合理地期望得到什么。

*取自《决策支持系统基础》(R.H.Bonczek, C.W.Holsapple和A.B.Winston合著,纽约学术出版社1981年)的前言(H.Simon)。

**这类软件的例子是1-2-3电子表格软件, dBASE文件管理程序, CHART-MASTER图形生成程序或KnowledgeMan集成化系统。

这类读者可以更清楚地明瞭这一技术对他们个人以及对他们所管理的机构所提供的机会和提出的挑战。

第二类读者是应用系统的专业开发人员，如公司的信息系统部门、软件开发公司和咨询公司的专业人员。如果你是属于这一类读者的话，你可能已经觉察到了，潜在的用户们已越来越希望专业开发人员提供事务处理专家系统。但精通事务处理专家系统的机会和技术的专业开发人员目前还很少。本书从事务处理的角度介绍专家系统，就是试图在这一鸿沟上架起一座桥梁。这一独特的观点使读者可简便地识别出人工智能中可用于构造先进的事务处理专家系统的概念和技术。

第三大类读者群出现在正式的教育领域内，也包括公司的培训班。如果你是负责专家系统原理和应用的教學工作的，你会发现本书是一本合适的参考书。本书虽然集中讨论了事务处理专家系统，它与管理科学以外的教学内容也是密切相关的。在诸如科学、工程、农业、医学等领域的应用训练中，本书提供的材料也是很有用的。

即使你不是经理、软件开发人员或教师，本书对你仍然是有用的。例如，你可能是一个希望追踪软件最新进展的计算机爱好者，或者是一个希望了解专家系统如何有效地应用于事务处理问题解决的人工智能迷。

本书分为四篇：

第一篇 事务处理专家系统的背景

第二篇 事务处理专家系统的基本概念

第三篇 事务处理专家系统的建造

第四篇 我们需要什么——今天与明天

第一篇将你引入人工智能世界。重点是关于推理系统，从Simon和Newell的开创性工作开始，一直讲到经典的专家系统。通过对事务处理计算的发展历史的回顾，我们清楚地看到，在事务处理的计算机应用方面，专家系统技术是下一个革命性的然而又是很自然的步骤。直到1985年，用于真正事务处理方面的问题解决的成功的专家系统尚不多见。第一篇最后说明这种现象出现的原因以及为什么这种状况正在改变之中。

第二篇解释事务处理专家系统的基本概念。我们将专家系统和构造专家系统的工具严格地区分开来。我们描述了专家系统的三大成分（用户接口、推理机和规则集）。我们考察了规则的概念，它是推理活动的一个片断的体现。随后我们描述推理机是怎样使用一组规则进行推理来解决问题的。我们讨论了带有各种不精确性的推理。最后，我们探讨了通过激活推理机进行专家系统咨询的各种可能性。所有这些基本概念都是通过一个不断扩展的事务处理专家系统的例子来说明的。这个例子是提供关于设立销售定额的提议的专家系统。

第三篇提供大量对专家系统开发人员来说是最感兴趣的高级专家系统概念。这些实践性的论题中包括规则说明策略，对推理机的推理进行实施的控制，各种置信度代数的比较，以及模糊变量管理的技术。第四篇在开始时总结了选择专家系统开发工具时应考虑的那些问题。然后展望了明天的基于知识的组织。在这种组织中，用于决策支持的人工智能专家系统将起着决定性的作用。为了使读者能够快速查到常用术语的含义，书后提供了一份名词解释。

所有的例子都是在IBM微机上用Guru实际实现了的。Guru是一个受到高度评价的

用于构造和使用事务处理专家系统的人工智能环境。*我们认为,灵活性、强大的功能和使用方便这三者结合,使Guru在探索事务处理专家系统的可能性方面成为一个有价值的指南。跟着Guru,你可以简便地抓住理解专家系统在事务处理方面的应用所需要的根本的思想、原理、技术和术语。无论你在构造和使用你自己的事务处理专家系统时实际使用什么软件,这些根本性的东西对你都是有用的。

本书说明,一个事务处理计算的人工智能环境(如Guru)怎样使你创造一个专家,咨询一个专家,以及自己成为一个专家。通过创造一个专家,你把使用知识的经验编入一个计算机系统,通过咨询一个专家,你可以积极地利用别人的推理经验去处理各种知识以导出新的知识(即建议)。通过自己成为一个专家,你在处理包含于数据库、电子表格、模型、正文、图形等中间的知识时直接使用你自己的推理能力。允许一个人扮演任一种或全部这三种角色的人工智能环境可以成为明天的基于知识的组织的基础。

本书附有几幅Guru在PGS彩色监视器上实际生成的彩色屏幕的照片。书中包含的Guru生成的打印输出是送往IBM的廉价图形打印机后产生的。若使用较昂贵的打印机或绘图仪,将能输出更精致的文字和图表,并带有彩色**。

最后,我们对许多从事实际工作的经理和管理学院的学生表示感谢,他们参加了我们的学习班、讨论会,对本书的写作起了重要作用。本书所使用的关于销售定额的例子部分地是基于K.Ferguson提出的建议。在手稿的最后准备阶段,Dr.R.Adelsman, Dr.R.Johnson, S.Thomson和M.Williamson提出的建设性意见使本书获益非浅。我们还感谢K.Smith的勤奋不倦的打字工作和K.Freeman在制图和编制索引方面给予的帮助。

Purdue大学的Krannert管理科学研究生院的管理信息研究中心(MIRC)的学术气氛特别有助于产生本书所提供的那些深刻思想和观点。MIRC受到IBM、GE和其他公司的慷慨支持。IBM市场部的副主任Mr.F.G.Rodgers(已退休)和计划与项目管理主任Mr.C.Bowen以多种方式支持了MIRC的研究计划,我们向他们表示感谢。我们也感谢MOBS公司的Dr.G.Koehler和Dr.M.Gagle(以及其他技术人员),他们提供了用于演出本书例子的Guru软件。

C.W.Holsapple.
A.B.Whinston

*如参见M.Williamson,“在Guru中,事务处理界终于找到了第一个真正属于人工智能的微机软件包。”PC周刊3, nos.11-14, 1986

**译者注:中译本将彩色屏幕照片改画成示意图,放在第三章最后以备后面各章参考。示意图是黑白的,但标出各项内容的颜色,以便与书中行文对照。

目 录

第一篇 事务处理专家系统的背景

第一章 人工智能世界	(1)
1.1 什么是人工智能	(1)
1.2 自然语言系统	(1)
1.3 推理系统概述	(3)
1.4 其他的人工智能研究领域	(8)
1.5 小结	(9)
第二章 事务处理计算——从记录保存到专家系统	(12)
2.1 记录保存	(12)
2.2 决策与支持	(14)
2.3 对通用性的需求	(15)
2.4 集成化软件	(17)
2.5 事务处理专家系统——障碍之所在	(18)
2.6 事务处理专家系统——困难的解决	(19)
2.7 小结	(21)

第二篇 事务处理专家系统的基本概念

第三章 熟悉专家系统	(23)
3.1 专家系统的结构	(23)
3.2 开发工具	(24)
3.3 一个应用场景	(25)
3.4 潜在的效益	(26)
3.5 小结	(28)
附录 彩色屏幕示意图	(28)
第四章 推理规则	(33)
4.1 规则的表示	(33)
4.2 规则的管理	(35)
4.2.1 用正文处理器管理规则集	(36)
4.2.2 用菜单管理规则集	(36)
4.2.3 规则的归纳	(37)
4.3 小结	(38)
第五章 使用规则的推理	(39)
5.1 ADVISOR规则集	(39)

5.2	未知变量的挑战	(42)
5.2.1	使未知变量成为已知的三种方法	(42)
5.2.2	定制用户接口	(44)
5.3	用户对话之外集成化提供的能力	(45)
5.3.1	规则集内的数据库管理	(46)
5.3.2	规则集内的电子表格	(47)
5.3.3	规则集内的过程性模型	(48)
5.3.4	规则集内的咨询	(49)
5.3.5	规则集内的图形功能	(49)
5.3.6	集成化的知识获取	(51)
5.4	基于规则的推理	(51)
5.5	正向推理	(52)
5.5.1	正向推理之例	(52)
5.6	咨询结果的表示	(54)
5.7	反向推理	(55)
5.7.1	反向推理的一般描述	(56)
5.7.2	推理流程的控制	(57)
5.7.3	反向推理之例	(57)
5.8	小结	(59)
第六章	不精确性的处理	(60)
6.1	置信因子	(60)
6.2	前提的置信因子	(61)
6.2.1	前提的连接置信度	(61)
6.2.2	前提的确认置信度	(61)
6.3	变量值的置信度	(62)
6.3.1	报告变量值的置信度	(62)
6.3.2	获取用户的不精确信息	(63)
6.4	规则动作的置信度	(64)
6.4.1	将动作的置信度传给变量	(64)
6.4.2	结论中不同动作的不同置信度	(65)
6.4.3	可变置信度	(65)
6.5	为同一变量值触发多个规则	(65)
6.5.1	替换赋值与追加赋值	(66)
6.5.2	确认变量的值	(66)
6.6	模糊变量	(67)
6.6.1	对于模糊变量的推理	(67)
6.6.2	报告模糊变量的值和置信度	(69)
6.7	未知的界限	(70)
6.8	小结	(70)

第七章 推理机的激活	(72)
7.1 咨询环境	(72)
7.1.1 咨询的上下文	(72)
7.1.2 环境控制	(75)
7.1.3 知识的安全	(76)
7.2 请求咨询的方法	(78)
7.2.1 推理机的命令激活	(78)
7.2.2 推理机的菜单驱动激活	(79)
7.2.3 推理机的自然语言激活	(79)
7.2.4 激活推理机的定制用户接口	(79)
7.2.5 人工智能应用系统	(82)
7.3 解释能力	(83)
7.3.1 咨询中间的解释	(83)
7.3.2 咨询后的解释	(84)
7.4 小结	(86)

第三篇 事务处理专家系统的建造

第八章 规则集的开发	(87)
8.1 构造专家系统的时机	(87)
8.2 对程序员的旁白	(89)
8.3 捕捉专家系统的开发时机	(90)
8.3.1 目标的设定	(90)
8.3.2 计划	(90)
8.4 开发的过程	(92)
8.5 分析和问题的定义	(95)
8.6 规则集的规范说明	(95)
8.6.1 发现变量	(96)
8.6.2 发现规则	(99)
8.6.3 初始序列和完成序列	(101)
8.7 专家测试	(102)
8.7.1 用HOW和WHY命令进行检测	(102)
8.7.2 自动追踪	(103)
8.7.3 定制追踪	(104)
8.7.4 保存和使用存贮的测试情况	(105)
8.7.5 提高咨询的速度	(107)
8.8 人机接口的作成	(109)
8.9 用户测试	(110)
8.10 装配	(110)

8.11 小结.....	(110)
第九章 推理的控制	(112)
9.1 推理的严密性.....	(112)
9.1.1 正向推理的严密性	(113)
9.1.2 反向推理的严密性	(114)
9.2 规则选择的顺序.....	(115)
9.2.1 先后顺序 (F) 标准	(117)
9.2.2 优先级 (P) 标准	(117)
9.2.3 成本 (C) 标准	(118)
9.2.4 未知变量数 (U) 标准	(119)
9.2.5 最高置信度 (H) 标准	(119)
9.2.6 随机 (R) 标准	(119)
9.2.7 组合选择标准	(119)
9.3 前提的评价策略.....	(120)
9.3.1 未知变量的顺序	(120)
9.3.2 计时	(121)
9.4 对未知变量的处理.....	(121)
9.5 推理控制变量的设定.....	(123)
9.6 小结.....	(123)
第十章 置信因子代数	(125)
10.1 未知的阈值	(125)
10.2 表达式的置信因子	(126)
10.2.1 连接置信因子代数	(126)
10.2.2 确认置信因子代数	(131)
10.3 变量的置信因子	(134)
10.3.1 规则内置信度	(136)
10.3.2 规则间置信度	(137)
10.3.3 控制方法的组合使用	(137)
10.3.4 合理共存	(138)
10.4 对开发者的辅助功能	(139)
10.4.1 置信度的自动输入输出	(139)
10.4.2 置信因子代数的交互实验法	(140)
10.4.3 定制的置信因子代数	(144)
10.5 小结	(144)
第十一章 对模糊变量的操作	(146)
11.1 对一个变量赋以多个值	(146)
11.1.1 推理前的赋值	(146)
11.1.2 在规则结论中的赋值	(147)
11.1.3 追减赋值	(149)

11.2	对开发者的辅助功能	(151)
11.2.1	模糊变量的自动输入输出	(151)
11.2.2	对模糊变量的交互式操作	(152)
11.3	小结	(155)

第四篇 我们需要什么——今天与明天

第十二章	选择一个专家系统开发工具的准则	(156)
12.1	规则集的特性	(156)
12.1.1	规则	(157)
12.1.2	变量	(160)
12.1.3	初始化	(162)
12.1.4	安全	(162)
12.1.5	窗口	(162)
12.1.6	完成	(163)
12.2	规则集的开发	(163)
12.2.1	规则集管理	(164)
12.2.2	规则集测试	(166)
12.3	咨询	(167)
12.3.1	推理机特性	(167)
12.3.2	推理机的激活	(171)
12.3.3	推理机的环境	(172)
12.4	事务计算	(173)
12.4.1	数据库管理	(174)
12.4.2	Ad Hoc查询	(175)
12.4.3	统计分析	(176)
12.4.4	电子表格分析	(176)
12.4.5	图形	(177)
12.4.6	正文处理	(177)
12.4.7	表格管理	(178)
12.4.8	定制报表的生成	(178)
12.4.9	远程通讯	(179)
12.4.10	过程模型	(179)
12.5	零售商的支持	(180)
12.5.1	零售商的历史	(181)
12.5.2	文稿	(181)
12.5.3	销售服务	(182)
12.6	性能	(183)
12.7	价格	(184)

12.8	小结	(185)
第十三章	基于知识的组织	(186)
13.1	知识工作者	(186)
13.1.1	技术的发展	(187)
13.1.2	训练	(188)
13.1.3	合作	(188)
13.2	基础结构	(189)
13.3	基于知识的组织的管理	(191)
13.4	基于知识的组织的实现	(193)
13.5	小结	(195)
附录 I	Guru演示学习软件使用指南.....	(197)
附录 II	名词解释.....	(219)

第一篇 事务处理专家系统的背景

第一章 人工智能世界

今天的经理们虽然很少触及人工智能(AI)，但在今后几年内，人工智能将开始对经理们的活动产生巨大而广泛的影响。再向远看一点，到本世纪末，将能到处看到过去三十年间AI研究的实际成果。什么是组织，它们是如何管理的，以及决策是如何产生的，对于这些问题的传统概念将会改变。在这一转变中，人工智能技术的应用将起着重大的作用。所有这一切对今天的经理和组织构成了重大挑战和机会。虽然这一转变会伴随着阵痛，但它将导致经理及其组织的生产率的巨大提高。那些在应用人工智能方法帮助管理方面走在前头的组织较之落后将具有显著的竞争优势。

人工智能的最大报偿大概将出现在专家系统领域。这种系统使用人工智能的推理技术为用户提出的问题提供专家级的咨询意见。专家系统使用从人类专家中搜集出来的关于如何解决特定类型的一个问题或一组相关问题的经验。这种系统的潜在效益是多样的，从经验的传播与普及，直到一个组织的推理知识的形式化和保存。通常提到的关于这种系统的障碍如开发周期长、开发费用高、开发工具缺乏以及孤立于数据处理的主流之外等将会消失，因为已经出现了新型的专家系统开发工具。

在详细考察专家系统这一领域之前，有必要回过头来看一看人工智能的更广阔的领域。本章是导论性的，对人工智能领域作一简短的回顾，从它的最初出现一直到最近的重大发展——把AI技术与管理信息处理方法的主流汇合起来的软件环境。

1.1 什么是人工智能

人工智能致力于使计算机这样的机器具有显示智能行为的能力。智能行为意为，若该行为由人完成，一般被看作是智能的。智能的两根柱石是理解自然语言的能力和推理的能力。它们反过来又代表了AI领域的两个主要研究内容。这些研究导致发现了使计算机能够理解自然语言和使用推理解决问题的实用技术。重要的一点是要懂得，这些技术并非一定要使用人们在执行这些任务时所用的同样的内部机构，但是它们产生的结果多多少少可与人做的结果相比。

1.2 自然语言系统

关于自然语言处理的AI研究的中心目标是要使人们能够用自己的自然语言与计算机对话，而不必用计算机语言。自然语言系统的用户不需要学习或记住任何专用语言就能够向计算机询问所需的信息。与计算机谈话就象与一个人类助手谈话一样。唯一的区别在于问题不是用嘴讲出来的，而是从键盘上输入进去的——随着语音识别技术的发展，就连这一

区别也将消失。

在与自然语言系统对话时，你不必操心遵守严格的计算机语言的语法问题。日常使用的词、句，甚至只言片语都照样能够使用。如同在与他人谈话时一样，你可以提出对方不一定完全理解的询问，它可以包含其含意是未知的新词、多义词，无意中用错的词等等。仍如人际对话一样，自然语言系统将与你进一步对话以澄清你的询问，它能够学会你在对话中引入的新词语的含义。在随后的谈话中再提到这些词语它就懂了。这是制定系统中的词汇表以包括你自己的常用语、隐语、缩略语等的方便的手段。

某些自然语言系统能够检查出询问中的拼写错误并自动为你改正。例如你不小心把list错写为lsit，系统会猜到你其实是在输入list。出于礼貌，系统将会问你这一解释是否正确。当然这也不排除你是在向系统的词汇表中定义一个新项目lsit。

如同人际对话时一样，自然语言系统能够在前面所提询问的上下文中理解新的询问。例如，你可能会问：

谁在纽约？

系统可能会反问你“纽约”是指纽约市还是纽约州。当你回答说是指市后，系统将列出一张在纽约市工作的你的雇员的清单。然后，你可能会问：他们薪金是多少？

系统将回应一个两列的报表，第一列是纽约市雇员的姓名，第二列是他们各自的薪金。如果你认为这一信息最好显示为饼状图，你的下一个要求可能为：

把它们画成饼状图。

结果是一张显示纽约市雇员的各种薪金级别所占百分比的饼状图。如果你下面想看一下同一信息的条形图，你可以说：

把它们画成条形图。

同一信息，原先显示为饼状图，现在变成了条形图。

今天，自然语言系统最广泛的应用是数据文件或数据库中的信息存取。在理解一个询问的过程中，自然语言系统用该数据文件或数据库管理系统所能处理的语言建立相应的存取命令。该文件或数据库管理软件然后执行所得到的命令以便实际存取所需信息并把它提供给用户。这样，用户不必知道关于文件或数据库是如何组织的任何知识。某些高级的自然语言系统还能处理关于数据修改、数据计算、统计分析、图形生成、模型执行甚至专家系统咨询的请求。

第一流的自然语言系统可以提高经理和数据处理专业人员双方的生产效率。它允许经理（作为计算机偶然的用户）直接获得对临时的信息需求的回答。没有这样的系统，经理要末就得找数据处理人员（如果有的话）来帮忙，要末就得自己学习数据库管理系统的查询语言或菜单接口（如果有的话）。这两种办法哪一个也不能让人满意：第一个办法将造成数据处理人员负担过重（他们的时间宝贵，薪金很高），而第二个办法将迫使经理去学习一种新的语言，而且若不经常使用，还得一遍又一遍地重学。自然语言的解决方法使数据处理人员能够集中精力去搞大量的、重复出现的任务的应用系统，而使经理集中精力去搞决策工作而不是学习计算机。

表1—1总结了今天已商品化的某些自然语言系统。它们分别在微型机、小型机和大型机上运行。在多数情况下，自然语言系统的能力有多么强或有多么复杂，并不主要取决于机器的大小。除了文件管理系统，几乎所有重要类型的数据库管理系统均成功地配有自然

表1-1 商品化自然语言系统

自然语言系统	厂 家	主机	类 型	数 据 源	图形支持
Clout	Microsim	微机	孤立软件	R:base	无
Guru	MDBMS, Inc,	微机 小型机	集成软件	Guru 数据库	条形图 饼状图 直线等
INTEL LECT	人工智能公司	大型机	孤立软件	IDMS, ADASAS, Vsam, FOCUS	条形图
K-Chat	MDBS, Inc	微型机 小型机	集成软件	KnowledgeMan 数据库	条形图 饼状图 直线等
RAMIS II English	Mathematica products Group Inc.	大型机	孤立软件	RAMIS II	无
THEMS	Frey Associates	小型机	孤立软件	Datatrieve, DBMS-32	无

语言接口。这包括古老的层次数据库管理方式,以及现在更为流行的关系方式和CODASYL方式。

大多数自然语言系统在本质上是独立的,其作用相当于在文件或数据库管理上面附加的应用程序。但在个别情况下,自然语言处理已被完全集成为一个单一程序的组成部分,该程序执行数据库管理、统计、绘图、电子表格和其他类型的处理。这一集成化是向着更有力的一代自然语言系统迈进的重要一步,新系统能够理解对于过程性模型和基于计算机的推理的请求。

1.3 推理系统概述

要求计算机系统能够对事实和断言进行推理以解决问题是人工智能研究领域的核心部分。一般说来,推理系统具有三大组成部分或子系统:

知识系统,存放某一问题领域的事实和断言。

语言系统,陈述所需解决的具体问题。

问题处理系统,利用知识系统的内容以解决语言系统所陈述的问题。

关于推理系统性质的研究集中于开发建立这三个子系统的越来越有力越来越灵活的技术。

虽然在人工智能文献中经常用数据库一词来代替知识系统,我们却不打算这样做。人

人工智能研究人员发明的在知识系统中表达知识的方法包括数组结构、语义网络、属性分级、各种表结构和谓词演算公式集合。这些方法都很有意思的也是很有用的，但与今日人们广泛使用的数据库管理系统却很少有共同之处。你也不必非要理解它们才能建立或使用事务处理专家系统。我们把数据库一词留作表示数据库管理系统所管理的数据实体。后面我们将要谈到，人工智能的知识表达方法可以有效地与今天的数据处理人员非常熟悉的数据库方法结合起来。这是把人工智能推理系统结合到事务处理计算领域中去的关键。

在三个子系统中，语言系统受到的注意最少：一般推理系统所提供的对所要解决的问题或所要达到的目标的陈述功能都相当简单。看起来它们允许对问题用自然语言语句进行规定，但一般只不过是提示用户对一组参数提供值罢了。用户提供的值有效地定义了推理系统所要处理的问题。

例如，假定有一个推理系统是用来解决个人在股票中投资多少的问题。它的知识系统应包含有关当前股票市场状况的各种事实以及在各种情况下个人应如何投资的断言。该系统的用户通过回答一系列关于该个人性质（如资产、收入水平、年龄和抚养人等）的预定的询问来给出特定的问题。这些回答刻画了一个特定的个人，因此也就构成了一个特定的个人，因此也就构成了一个特定的问题陈述。有些推理系统随后将这种问题与知识系统放在一起进行推理；另一些推理系统并不把问题陈述看作知识系统的一部分。不管是哪一种情况，问题处理系统都会使用知识系统的内容进行推理以解决特定个人的投资问题。

人工智能对推理系统的研究集中于问题处理系统，因为这是对知识系统中所存放的事实和断言进行实际推理的部分。对一特定的问题，目标是检查一系列有关的事实和断言以导出或发现对问题的解。在推理过程的各个步骤中，问题处理程序可能会要求用户提供进一步信息或对所考虑问题作进一步澄清。

如同人类推理一样，对于问题处理程序来说重要的一点是能够区别哪些是与手头的问题有关的知识，哪些是无关的。另一个重要之点是要按逻辑顺序考虑有关的知识，而不是随意浏览或跳到结论上。否则，问题的“解”可能是不正确的，或者要花费太多的时间才能得到。人工智能研究人员与逻辑学家已发明了许多在推理过程中关于下一步应检查什么知识的策略。

一般地，推理策略可分为两大类：正向推理与反向推理。正向推理开始于有关问题领域的基本知识。这些知识按特定的顺序被检查，问题处理程序在每一步中记住被检查过的知识片断的推论。这一过程一直进行到发现能够提供正在被处理的具体问题的解的那些结论时为止。反向推理则从原始问题陈述开始。这个问题被分解为子问题，子问题又被进一步分解为子问题，如此等等。其思想是较小的问题一般要比较大的问题容易解决。就是说，它有可能仅仅通过查看知识系统中有关的事实或断言来解决。通过对问题的所有（甚至是某些）子问题的解决，问题本身也能被容易地解决掉。

有趣的是，回顾50年代中期，当时最早的一个研究推理系统的中心是一个管理学院，即卡内基理工学院（现为卡内基-梅隆大学）的工业管理研究生院。虽然人工智能常常被认为是计算机科学的一个领域，可是当卡内基的H. Simon和A. Newell开始他们的关于推理系统的性质的研究时，世界上还没有一个计算机科学系存在*。虽然有极少的数的

* 第一个计算机科学系是Purdue大学在60年代初建立的。

管理学院从事不断扩大推理系统的研究,但是大多数都既不做研究也不搞这方面的教学。目前,有迹象表明这一状况正在起变化。因为有效的推理是管理的一个主要方面,

与RAND公司的J.C.Show合作,Newell和Simon在1959年开发了Logic Theorist系统[16],这个系统使用反向推理解决谓词演算中的问题(即证明定理),它的知识系统最初是一小组公理。1957年他们开始了一项远为大胆的计划并持续了十年之久。这就是里程碑式的通用问题求解器。(GPS)[8,17]。与Logic Theorist不一样,GPS不是设计成专门解决谓词演算或任一具体领域的问题的。相反,它致力于体现可用于广泛的问题领域的一般的推理方法。换句话说,它试图规定一个问题解决的一般方法。

GPS的目的是提供一个问题解决系统,其软件无论对哪一个问题领域均不需改变。这是通过将所有与问题有关的知识,包括问题领域特有的推理知识存放在知识系统内,而不是混在问题处理软件中来实现的。这样,通过改变其知识系统的内容,GPS可以解决各种不同领域的问题,包括谓词演算,符号积分,消解定理证明及各种游戏(如汉诺塔问题,传教士与野蛮人问题)。

具体问题向GPS描述为现存的初始情况和所求的目标情况。知识系统包含着运算符(推理知识),每一运算符允许问题处理程序确定从一个给定情况可以达到另外的哪一个情况。为了解决一个问题,问题处理程序搜索运算符以发现一系列运算符——一个逻辑流——它们可以初始情况出发,经过各种中间情况达到目标。问题正是在建立这个推理流程的过程中得到解决的。这种搜索是诱导控制的——这是一种问题解决过程,它一般是有效的,但不能完全保证总是能得到解。

GPS本身从它所能解决的问题的广度和深度来说,实际上只具有有限的通用性,但它所体现的原则是很有意义的[18]。Simon和Newell的开创性工作揭示了推理的本质,对于这一领域在创建年代的其他研究者提供了指南,对推理系统的研究作出了巨大贡献。他们的学生继续在推理系统的领域里做出重要的贡献。

七十年代初出现了新的重大突破。这涉及到建立在很狭窄问题领域里集中解决困难问题的推理系统。这里强调解决现实生活里的问题如疾病诊断,推断未知分子的化学结构,以及应用数学分析。因为这样系统能够解决只有相应问题领域里的专家才能解决的问题,所以被称为专家系统。其中最早的一个系统DENDRAL是在六十年代中期开始研究的。到七十年代末出现了大批的专家系统。

表1-2给出了具有代表性的专家系统的样本。关于这些专家系统的一个最重要的事实是它们确实是实用的。这一点是无可辩驳的。它们解决原本需要人类专家才能解决的问题。今天它们中间的许多系统例行地被用于提供专家咨询。但直到1985年,还极少有(如果真有的话)值得注意的专家系统是被建立来解决管理的或事务处理问题的。

与通用问题解决系统GPS不同,专家系统的问题处理程序并不是设计成为通用的。它们是为非常具体的问题领域定制的。专家系统的问题处理软件中含有具体领域所特有的推理方法。这样DENDRAL的问题处理系统并不适于细菌血液感染性疾病的诊断。反之,MYCIN的问题处理程序也不足以承担分子结构分析的任务。专家系统的实现者各自从头建立自己专用的问题处理软件,通常是使用LISP这样的专用程序设计语言。LISP已有25年历史,适合处理一组相关的符号表。实现一个专家系统所需的时间通常为许多人年。因此实现的代价很高,它需要人工智能专家而不是数据处理专业人员这一主流派。