

科技大讲堂

a

**21**世纪高等学校计算机 专业实用规划教材

◎尚文倩编著

洛第人孝出版社



**a** **21**世纪高等学校计算机 专业实用规划教材



内容简介

本书系统介绍了人工智能的基本原理、基本技术、基本方法和应用领域等内容，比较全面地反映了 60 年来人工智能领域的进展，并根据人工智能的发展动向对一些传统内容做了取舍。全书共9章。第1章 介绍人工智能的基本概念、发展历史、应用领域等。其后8章的内容分为两大部分：第一部分（第2〜5章） 主要讲述传统人工智能的基本概念、原理、方法和技术，涵盖知识表示、搜索策略、确定性推理和不确定推 理的相关技术与方法；第二部分（第6〜9章）主要讲述现代人工智能的新的技术和方法，涵盖机器学习、数 据挖掘、大数据、深度学习的最新技术与方法。每章后面附有习题，以供读者练习。

本书主要作为计算机专业本科生和其他相关学科本科生相关课程教材，也可供研究生和有关科技人 员参考。

本书封面贴有清华大学出版社防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010\*2782989 13701121933

图书在版编目（CIP）数据

人工智能/尚文倩编著.一北京：清华大学出版社,2017 （21世纪高等学校计算机专业实用规划教材）

ISBN 978-7-302-46462-4

i.①人… n,①尚… ni.①人工智能iv.①tpi8

中国版本图书馆CIP数据核字（2017）第024635号

责任编辑：刘星战晓雷

封面设计：刘键

责任校对：李建庄

责任印制：李红英

出版发行：清华大学出版社

网 址：http：//www. tup. com. cn, http：//www. wqbook. com

地 址：北京清华大学学研大厦A座 邮 编：100084

社总机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup. tsinghua. edu. cn

质量反馈：010-62772015, zhiliang@tup. tsinghua. edu. cn

课件 下载：http \* / / www. tup. com. cn,010-62795954

印装者：清华大学印刷厂

经 销：全国新华书店

开 本：185mmX260mm 印 张：14.5 字 数：353千字

版 次:2017年7月第1版 印 次:2017年7月第】次印刷

印 数：1〜2000

定 价：39. 00元

产品编号：066951-01



人工智能是研究人类智能活动的规律，构造具有一定智能的人工系统，研究如何让计算 机去完成以往需要人的智力才能胜任的工作，也就是研究如何应用计算机的软硬件来模拟 人类某些智能行为的基本理论、方法和技术。人工智能是计算机科学的一个分支，被称为 20世纪世界三大尖端科技（空间技术、能源技术、人工智能）之一，也被称为*21*世纪三大尖 端技术（基因工程、纳米科学、人工智能）之一。

人工智能作为一门学科从正式提岀到现在，已经走过了一甲子的岁月，经历了风风雨 雨、起起落落。随着2016年3月AlphaG。战胜世界围棋冠军、职业九段选手李世石，人工 智能又一次受到了世人极大的关注。2014年人工智能领域全球投资总额超过19亿美元, 同比增长超50%,预计2020年全球市场规模将达到183亿美元。谷歌、微软、苹果等国外 科技巨头纷纷发力，国内企业也纷纷和顶尖技术团队合作，积极布局，如百度的大脑计划、科 大讯飞的超脑计划、京东的智能聊天机器人等。2015年7月1日，国务院印发了《关于积极 推进“互联网+ ”行动的指导意见》，将“互联网+人工智能”列为11项重点行动之一；2016 年9月国家发改委、科技部、工信部、中央网信办制定的《“互联网+”人工智能三年行动实施 方案》正式印发，计划到2018年基本建立人工智能的产业、服务和标准化体系，实现核心技 术突破。

基于此，本书在传统人工智能的基础之上增加了新的人工智能的技术与方法。本书的 第6章介绍了机器学习中的一些基本问题、基本方法和关键技术。第7章介绍了数据挖掘 的常用技术与方法。第8章介绍了大数据的最新研究进展、最新技术与方法。第9章介绍 了深度学习的常用方法与最新技术。

本书是集体智慧的结晶，全书由尚文倩主编，陈秀霞、封树超、颜梦菌、李振忠、王宇奇、 娄延伟、程宇芬、张春洁等同学为本书做出了重要贡献，在此表示衷心的感谢！本书还参考 了《2016—2020年中国人工智能行业深度调研及投资前景预测报告》，借鉴了有关教材及互 联网上的一些资料，也向这些文献的作者表达诚挚的谢意！

由于编者水平有限，书中的疏漏在所难免，敬请广大读者批评指正。

作者

2017年4月于中国传媒大学

第1章绪论 1

1. [人工智能的定义 1](#bookmark153" \o "Current Document)
2. [人工智能的发展历史 2](#bookmark156" \o "Current Document)
3. [孕育阶段 2](#bookmark159" \o "Current Document)
4. [形成阶段 3](#bookmark162" \o "Current Document)
5. [发展阶段 4](#bookmark165" \o "Current Document)

[1.3人工智能的三大学派 5](#bookmark171" \o "Current Document)

1. [1 符号主义 5](#bookmark174" \o "Current Document)
2. [连接主义 6](#bookmark177" \o "Current Document)
3. [行为主义 6](#bookmark180" \o "Current Document)
4. [人工智能研究内容与应用领域 7](#bookmark183" \o "Current Document)
5. [1 问题求解 7](#bookmark186" \o "Current Document)
6. [专家系统 7](#bookmark189" \o "Current Document)
7. [机器学习 8](#bookmark192" \o "Current Document)

[1. 4. 4 神经网络 8](#bookmark195" \o "Current Document)

[1.4.5模式识别 9](#bookmark198" \o "Current Document)

1. [数据挖掘和知识发现 9](#bookmark201" \o "Current Document)
2. [计算机视觉 9](#bookmark204" \o "Current Document)
3. [智能控制 10](#bookmark207" \o "Current Document)

[1.4.9计算智能 10](#bookmark210" \o "Current Document)

[1.4. 10 其他 10](#bookmark213" \o "Current Document)

1. [人工智能的发展趋势 10](#bookmark216" \o "Current Document)

[1.5.1 多学科交叉研究 10](#bookmark219" \o "Current Document)

[1.5.2智能应用和智能产业 11](#bookmark222" \o "Current Document)

1. [习题 11](#bookmark225" \o "Current Document)

第2章知识表示 12

1. [概述 12](#bookmark234" \o "Current Document)

[2.1.1知识及知识的分类 12](#bookmark237" \o "Current Document)

1. [1. 2 知识表示 13](#bookmark240" \o "Current Document)

[2.. 2谓词逻辑表示法 14](#bookmark243" \o "Current Document)

1. [基本概念 14](#bookmark246" \o "Current Document)

[2.2.2谓词逻辑表示法 15](#bookmark266" \o "Current Document)

[2.2.3谓词逻辑表示法的经典应用 16](#bookmark274" \o "Current Document)

[2.2.4谓词逻辑表示法的特点 18](#bookmark277" \o "Current Document)

1. [产生式表示法 18](#bookmark280" \o "Current Document)
2. [概述 18](#bookmark283" \o "Current Document)
3. [产生式系统 19](#bookmark286" \o "Current Document)
4. [产生式表示法应用举例 …… 20](#bookmark289" \o "Current Document)
5. [产生式系统的推理方式 21](#bookmark297" \o "Current Document)
6. [产生式系统的特点 22](#bookmark310" \o "Current Document)
7. [语义网络表示法 23](#bookmark320" \o "Current Document)
8. [语义网络基本概念 23](#bookmark323" \o "Current Document)
9. [语义网络中常用的语义联系 24](#bookmark326" \o "Current Document)
10. [语义网络表示知识的方法 26](#bookmark347" \o "Current Document)

[2.4.4语义网络的推理过程 30](#bookmark357" \o "Current Document)

1. [语义网络表示的特点 31](#bookmark368" \o "Current Document)
2. [框架表示法 31](#bookmark376" \o "Current Document)
3. [框架基本结构 31](#bookmark379" \o "Current Document)
4. [基于框架的推理 33](#bookmark387" \o "Current Document)
5. [框架表示法的特点 34](#bookmark394" \o "Current Document)
6. [6 习题 34](#bookmark404" \o "Current Document)

第**3**章搜索策略 36

1. [搜索的基本概念 36](#bookmark431" \o "Current Document)
2. [1.1 搜索的含义 36](#bookmark434" \o "Current Document)

[3. 1.2 状态空间法 37](#bookmark437" \o "Current Document)

[3. 1.3 问题归约法 40](#bookmark449" \o "Current Document)

[3.2状态空间搜索 43](#bookmark473" \o "Current Document)

1. [盲目搜索 43](#bookmark476" \o "Current Document)
2. [状态空间的启发式搜索 48](#bookmark495" \o "Current Document)

[3.3 博弈树的启发式搜索 54](#bookmark523" \o "Current Document)

1. [概述 54](#bookmark526" \o "Current Document)
2. [极大极小过程 55](#bookmark532" \o "Current Document)
3. 3.3 剪枝 57
4. [4 习题 58](#bookmark542" \o "Current Document)

第**4**章 确定性推理 60

[4.1推理的基本概念 60](#bookmark562" \o "Current Document)

1. [什么是推理 60](#bookmark565" \o "Current Document)

[4.1.2推理方法及其分类 61](#bookmark568" \o "Current Document)

[4.1.3推理的控制策略及其分类 63](#bookmark578" \o "Current Document)

1. [正向推理 63](#bookmark581" \o "Current Document)

[4. 1.5 逆向推理 65](#bookmark589" \o "Current Document)

[4.1,6 混合推理 67](#bookmark599" \o "Current Document)

* 1. [推理的逻辑基础 67](#bookmark602" \o "Current Document)

[4.2.1谓词公式的解释 67](#bookmark605" \o "Current Document)

* + 1. [谓词公式的永真性与可满足性 69](#bookmark611" \o "Current Document)

[4.2.3谓词公式的等价性与永真蕴含性 69](#bookmark614" \o "Current Document)

1. [谓词公式的范式 71](#bookmark627" \o "Current Document)
2. [置换与合一 71](#bookmark635" \o "Current Document)

[4.3自然演绎推理 73](#bookmark643" \o "Current Document)

[4.4归结演绎推理 74](#bookmark649" \o "Current Document)

[4.4.1子句集及其简化 74](#bookmark652" \o "Current Document)

1. [鲁滨逊归结原理 78](#bookmark667" \o "Current Document)
2. [归结演绎推理的归结策略 85](#bookmark705" \o "Current Document)

[4. 4.4用归结反演求取问题的解 89](#bookmark721" \o "Current Document)

[4.5 基于规则的演绎推理 90](#bookmark729" \o "Current Document)

1. [规则正向演绎推理 90](#bookmark733" \o "Current Document)
2. [规则逆向演绎推理 94](#bookmark752" \o "Current Document)
3. [6 习题 97](#bookmark758" \o "Current Document)

第**5**章不确定性推理 99

1. [1 概述 99](#bookmark791" \o "Current Document)

[5.1.1为什么要釆用不确定性推理 99](#bookmark794" \o "Current Document)

[5.1.2不确定性推理要解决的问题 100](#bookmark801" \o "Current Document)

1. [不确定性推理类型 101](#bookmark812" \o "Current Document)
2. [概率基础 102](#bookmark815" \o "Current Document)
3. [主观贝叶斯方法 103](#bookmark825" \o "Current Document)

[5.3.1 不确定性的表示 103](#bookmark828" \o "Current Document)

5.3.2组合证据不确定性的计算 :••…104

1. [不确定性的传递算法 104](#bookmark836" \o "Current Document)
2. 结论不确定性的合成 105
3. [可信度方法 107](#bookmark847" \o "Current Document)
4. [组合证据不确定性的计算 108](#bookmark855" \o "Current Document)
5. [不确定性的传递算法 108](#bookmark860" \o "Current Document)
6. 结论不确定性的合成 109
7. [5 证据理论 110](#bookmark865" \o "Current Document)

5. 5. 1 理论基础 110

[5. 5.2 不确定性表示 111](#bookmark875" \o "Current Document)

1. [5.3 组合证据不确定性的计算 112](#bookmark833" \o "Current Document)
2. 不确定性的更新 112
3. [模糊推理 115](#bookmark884" \o "Current Document)
4. [模糊知识的表示 115](#bookmark887" \o "Current Document)
5. [模糊概念的匹配 116](#bookmark899" \o "Current Document)
6. [模糊推理 117](#bookmark906" \o "Current Document)
7. [习题 120](#bookmark916" \o "Current Document)

第**6**章机器学习 123

1. [1 概述 123](#bookmark927" \o "Current Document)
2. [机器学习的基本概念 123](#bookmark930" \o "Current Document)

[6.1.2机器学习的发展历史 124](#bookmark936" \o "Current Document)

1. 1.3学习系统的基本模型 125

6. 1.4 学习策略 125

[6.2 记忆学习 126](#bookmark945" \o "Current Document)

6. 3 归纳学习 126

6. 3. 1 示例学习 126

1. [观察与发现学习 130](#bookmark958" \o "Current Document)
2. [决策树学习 130](#bookmark963" \o "Current Document)
3. 类比学习 133
4. [类比学习的基本过程 134](#bookmark966" \o "Current Document)
5. [属性类比学习 134](#bookmark969" \o "Current Document)
6. [转换类比学习 135](#bookmark972" \o "Current Document)
7. [派生类比学习 136](#bookmark975" \o "Current Document)
8. [联想类比学习 136](#bookmark978" \o "Current Document)

[6. 6 解释学习 137](#bookmark994" \o "Current Document)

1. [神经学习 139](#bookmark999" \o "Current Document)

[6. 7.1 感知器学习 139](#bookmark1002" \o "Current Document)

1. 反向传播网络学习 141
2. [Hopfield 网络学习 145](#bookmark1033" \o "Current Document)
3. [贝叶斯学习 147](#bookmark1038" \o "Current Document)

[6.8.2朴素贝叶斯分类算法 147](#bookmark1044" \o "Current Document)

[6.9在线机器学习 150](#bookmark1057" \o "Current Document)

1. [截断梯度法 151](#bookmark1060" \o "Current Document)
2. 前向后向切分算法 152
3. 正则对偶平均算法 154
4. FTRL 154

[6. 10 习题… 156](#bookmark1099" \o "Current Document)

第**7**章数据挖掘 158

[7.1数据挖掘概述 158](#bookmark1115" \o "Current Document)

[7.1.1数据挖掘概念与发展 158](#bookmark1118" \o "Current Document)

1. [数据挖掘的任务 159](#bookmark1121" \o "Current Document)
2. [数据挖掘的应用 159](#bookmark1127" \o "Current Document)
3. [数据挖掘过程与方法 160](#bookmark1130" \o "Current Document)
4. [分类 160](#bookmark1137" \o "Current Document)
5. [决策树分类法 160](#bookmark1140" \o "Current Document)

[7.2.2基于规则的分类器 161](#bookmark1145" \o "Current Document)

[7.2.3 朴素贝叶斯分类器 161](#bookmark1152" \o "Current Document)

[7. 2.4 基于距离的分类算法 162](#bookmark1163" \o "Current Document)

1. [聚类 162](#bookmark1172" \o "Current Document)
2. 1 概念 162
3. [3.2聚类分析的基本方法 162](#bookmark1175" \o "Current Document)
   1. [关联规则 168](#bookmark1239" \o "Current Document)
      1. 基本概念 168
      2. [关联规则挖掘算法 169](#bookmark1244" \o "Current Document)
      3. [关联规则生成 172](#bookmark1277" \o "Current Document)
   2. [习题 173](#bookmark1282" \o "Current Document)

第**8**章大数据 175

[8.1大数据概述 175](#bookmark1307" \o "Current Document)

[8.1.1大数据概念 175](#bookmark1310" \o "Current Document)

1. [1.2 特征 176](#bookmark1313" \o "Current Document)
2. 发展历程 177
3. 应用 177

[8.2数据获取 179](#bookmark1329" \o "Current Document)

8. 2. 1 网络爬虫 179

8. 2. 2 RSS 184

[8.3 数据挖掘 185](#bookmark1382" \o "Current Document)

1. [1 概述 185](#bookmark1385" \o "Current Document)

[8.3.2数据挖掘工具 186](#bookmark1388" \o "Current Document)

1. 现状与未来 188

[8.4数据分析 188](#bookmark1399" \o "Current Document)

1. 1 概述 188

[8. 4.2数据分析流程 189](#bookmark1413" \o "Current Document)

1. [数据分析方法 191](#bookmark1422" \o "Current Document)

[8. 4.4 数据分析工具 192](#bookmark1435" \o "Current Document)

[8. 5 Hadoop 194](#bookmark1454" \o "Current Document)

8. 5. 1 简介 194

8. 5. 2 分布式离线计算框架MapReduce 196

1. 5. 3 Hadoop分布式文件系统 198
2. HBase 大数据库 201

[8.6数据可视化 204](#bookmark1493" \o "Current Document)

1. 习题 205

第**9**章深度学习 206

[9.1深度学习应用背景与概述 207](#bookmark1524" \o "Current Document)

1. 1.1 应用背景 207
2. 概述 207
3. [人脑视觉机理 208](#bookmark1527" \o "Current Document)
4. [特征的概念 209](#bookmark1530" \o "Current Document)
5. [特征表示的粒度 209](#bookmark1533" \o "Current Document)
6. [初级（浅层）特征表示 210](#bookmark1536" \o "Current Document)
7. [结构性特征表示 211](#bookmark1541" \o "Current Document)
8. 特征数量 212
9. [深度学习基本思想 212](#bookmark1544" \o "Current Document)
10. [浅层学习和深度学习 212](#bookmark1547" \o "Current Document)
11. 浅层学习 212
12. [深度学习 213](#bookmark1552" \o "Current Document)

[9.5深度学习常用模型和方法 213](#bookmark1557" \o "Current Document)

1. [自动编码器 214](#bookmark1560" \o "Current Document)
2. [稀疏编码 215](#bookmark1566" \o "Current Document)
3. [深度信念网络 215](#bookmark1569" \o "Current Document)
4. [卷积神经网络 216](#bookmark1572" \o "Current Document)

[9.6深度学习展望 218](#bookmark1578" \o "Current Document)

1. [习题 219](#bookmark1584" \o "Current Document)

[参考文献 220](#bookmark1597" \o "Current Document)

自1956年人工智能的概念被第一次提及，人工智能发展至今的近60年时间里所取得 的极大发展，不容忽视，它引起了众多学科和不同专业背景的学者们日益重视，逐步成为一 门广泛的交叉和前沿科学。近些年来，随着现代计算机的不断发展及其在软硬件实现方面 取得的长足进步，人工智能正在被应用到越来越广泛的领域中。目前来看，虽然人工智能在 发展的过程中存在许多困难和挑战，但随着研究的不断深入，这些困难和挑战终将被战胜, 并将推动人工智能继续向前发展。本书主要阐述传统人工智能的基本理论、原理、方法和应 用，以及现代人工智能的新的技术与方法。

**1.1** 人工智能的定义

人工智能的定义最早可以追溯到1956年夏天，由人工智能早期研究者John McCarthy 等人提出：人工智能就是要让机器的行为看起来像是人所表现出的智能行为一样。但迄今 尚难以给出人工智能的确切定义。以下为不同学者从不同的角度、不同的层面给出的人工 智能的定义。

1978年,Bellman：人工智能是那些与人的思维相关的活动，诸如决策、问题求解和学习 等的自动化。

1985年,Haugeland：人工智能是一种计算机能够思维，使机器具有智力的激动人心的 新尝试。

同年,Charniak和McDermott：人工智能是用计算模型研究智力行为。

1990年,Kurzwell：人工智能是一种能够执行需要人的智能的创造性机器的技术。 同年,Schalokff：人工智能是一门通过计算过程力图理解和模仿智能行为的学科。

1991年,Rich Knight：人工智能研究如何让计算机做现阶段只有人才能做得好的事情。

1992年,Winston：人工智能研究那些使理解、推理和行为成为可能的计算。

1993年,Luger和Stubblefield：人工智能是计算机科学中与智能行为的自动化有关的

一个分支。

1998年,Nilsson：广义地讲，人工智能是关于人造物的智能行为，而智能行为包括知 觉、推理、学习、交流和复杂环境中的行为。

2003年,Stuart Russell和Peter Norvig：人工智能的定义可以分为4类：像人一样思 考的系统、像人一样行动的系统、理性地思考的系统和理性地行动的系统。这里“行动”应广 义地理解为采取行动或制定行动的决策，而不是肢体动作。

从不同学者对人工智能的定义中，可以归纳出人工智能需要具备判断、推理、证明、识 别、理解、感知、学习和问题求解等诸多能力。随着人工智能的不断发展和对人工智能理解 的深入，将来还会出现对人工智能的新的定义和理解。

另外，还有的专家和学者提出强人工智能和弱人工智能的概念。所谓强人工智能是指 有可能制造出真正能推理和解决问题的智能机器，并且这样的机器是有知觉的，有自我意 识。强人工智能主要分为两类：类人的人工智能，即机器的思考和推理就像人的思维一样； 非类人的人工智能，即机器产生了和人完全不一样的知觉和意识，使用和人完全不一样的推 理方式。所谓弱人工智能是指不可能制造出能真正地推理和解决问题的智能机器，这些机 器只不过看起来像是智能的，但并不真正拥有智能，也不会有自主意识。

**1.2**人工智能的发展历史

1956年夏天，由美国学者麦卡锡（McCarthy）等人发起，在美国的达特茅斯（Dartmouth）学 院举办了一次长达两个月的研讨会，重点讨论如何用机器模拟人类智能的问题。此次会议 云集了人工智能领域相关的研究者：John McCarthy（达特茅斯学院）,Marvin Minsky（哈佛 大学），Nathanniel Rochester （IBM公司）,Claude Shannon （贝尔电话实验室）以及其他数 学、神经生理学、精神病学、心理学、信息论、计算机、自然语言处理及神经网络等方面的研究 者。会上，人工智能的术语第一次被提及，这是历史上第一次人工智能研讨会，具有十分重 要的历史意义，同时也标志着人工智能学科的诞生。

到目前为止，人工智能的发展大致经历了 3个阶段：孕育阶段（1956年之前）、形成阶段 （1956—1969年）、发展阶段（1970年至今）。

**L2.1**孕育阶段

人工智能的孕育阶段大致可以认为是1956年以前的时期。这段漫长的时期中，数理逻 辑、自动机理论、控制论、信息论、仿生学、电子计算机、心理学等科学技术的发展为后续人工 智能的诞生奠定了思想、理论和物质基础。该时期的主要贡献举要如下。

公元前4世纪，希腊哲学家亚里士多德在《工具论》中提出了形式逻辑的一些主要定律， 为形式逻辑奠定了基础，特别是他的三段论，至今仍是演绎推理的基本依据。

1642年，法国数学家帕斯卡（Pascal）发明了第一台机械计算器——加法器，开创了计算 机械时代。此后，德国数学家莱布尼茨（Leibniz）在其基础上发展并制成了可进行四则运算 的计算器。他提出了“通用符号”和“推理计算”的概念，使形式逻辑符号化。这一思想为数 理逻辑以及现代机器思维设计奠定了基础。

英国逻辑学家布尔(Boole)在《思维法则》一书中首次用符号语言描述了思维活动的基 本推理原则，这种新的逻辑代数系统被后世称为布尔代数。

1936年，英国数学家图灵(Turing)提出了一种理想计算机的数学模型，即图灵机模型。 这为电子计算机的问世奠定了理论基础。

1943年，心理学家麦克洛奇(McCulloch)和数理逻辑学家皮兹(Pitts)提出了第一个神 经网络模型——M-P神经网络模型。模型总结了神经元的一些基本生理特性，提出了神经 元形式化的数学描述和网络的结构方法，为开创神经计算时代奠定了坚实的基础。

1945年，冯・诺依曼(Neumann)提出了存储程序的概念。1946年，美国数学家马士利 (Mauchly)和埃克特(Eckert)研制成功第一台电子计算机ENIAC,为人工智能的诞生奠定 了物质基础。

1948年，香农(Shannon)发表了《通信的数学理论》，标志着信息论的诞生。

1948年，维纳(Wiener)创立了控制论。这是一门研究和模拟自动控制的人工和生物系 统的学科，标志着根据动物心理和行为学科进行计算机模拟研究的基础已经形成。

1950年，图灵发表论文*Computing Machinery and Intelligence* ,提出了著名的图灵测 试，该测试大致如下：询问者与两个匿名的交流对象(一个是计算机，另一个是人)进行一系 列问答，如果在相当长时间内，他无法根据这些问题判断这两个交流对象哪个是人，哪个是 计算机，那么就可以认为该计算机具有与人相当的智力，即这台计算机具有智能。

在20世纪50年代，计算机应用仅局限于数值计算，例如弹道计算。但1950年，香农完 成了人类历史上第一个下棋程序，开创了非数值计算的先河。此外,MaCarthy. Newell. Simon.Minsky等人提出以符号为基础的计算。这些成就使得人工智能作为一门独立的学 科成为一种不可阻挡的历史趋势。

**1.2.2**形成阶段

人工智能的形成阶段大约为1956-1969年。除了 1956年在美国的达特茅斯学院召开 的研讨会提出了“人工智能”的术语外，这一时期的成就还包括定理机器证明、问题求解、 LISP语言以及模式识别等。该阶段的主要研究成果如下。

1956年,Samuel研制了具有自学能力的西洋跳棋程序。该程序具备从棋谱中学习、在 实践中总结经验提高棋艺的能力。值得称道的是它在1959年战胜了设计者本人，并且在 1962年打败了美国的一个州跳棋冠军。这是模拟人类学习的一次成功的探索，同时也是人 工智能领域的一个重大突破。

1957年,Newell和Simon等人编制出了一个数学定理证明程序，该程序证明了 Russell 和Whitehead编写的《数学原理》一书中的第2章的38条定理。1963年修订的程序证明了 52个定理。1958年，美籍华人王浩在IBM 740机器上用了 3〜5min证明了《数学原理》中 命题演算的全部定理，总共220个。这些都为定理的机器证明做出了突破性贡献。

1958年,McCarthy研制出的表处理语言LISP,不仅可以处理数据，而且可以方便地处 理符号，成为人工智能程序设计语言的重要里程碑。

1960年，Newell、Shaw和Simon等人编制了能解10种不同类型课题的通用问题求解 程序(General Problem Solving,GPS) o其中所揭示的人在解题时的思维过程大致为3个阶 段：先想出大致的解题计划；然后根据记忆中的公理、定理和推理规则组织解题过程；最后 进行方法和目的分析，修正解题计划。

1965年，斯坦福大学的Feigenbaum开展了专家系统DENDRAL的研究，该专家系统 于1968年投入使用。它能根据质谱仪的试验，通过分析推理决定化合物的分子结构，其分 析能力接近甚至超过有关化学专家的水平，不仅为人们提供了一个实用的智能系统，而且对 知识表示、存储、获取、推理及利用等技术是一次非常有益的探索，对人工智能的发展产生了 深远的影响。

以上这些早期的成果表明了人工智能作为一门新兴学科有了蓬勃发展。

**1.2.3**发展阶段

从20世纪70年代开始人工智能的研究进入高速发展时期。在这期间，世界各国都纷 纷开展对人工智能领域的研究工作。至20世纪70年代末，人工智能研究遭遇了一些重大 挫折，在问题求解、人工神经网络等方面遇到许多问题，这些问题使人们对人工智能研究产 生了质疑。这些质疑声使得人工智能研究者们开始反思。终于，在1977年召开的第五届国 际人工智能联合会议上提出了知识工程的概念。此后，知识工程的兴起产生了大量的专家 系统，这些专家系统在各种领域中获得了成功应用。随着时间的推移，专家系统的问题逐渐 暴露出来，知识工程发展遭遇困境，这动摇了传统人工智能物理符号系统对于智能行为是必 要充分的基本假设，从而促进了连接主义和行为主义智能观的兴起。随后，大量神经网络和 智能主体方面的研究取得极大进展。

以下从知识工程、神经网络以及智能主体3个方面讲述人工智能在其发展阶段走过的 历程。

1. 知识工程

20世纪70年代开始，人工智能领域的研究在世界各地风生水起。这期间涌现了一批 重要的研究成果，包括1972年法国马赛大学的Comerauer提岀并实现了逻辑程序设计语 言PROLOGo斯坦福大学的Shortliffe等人开始研制用于诊断和治疗感染性疾病的专家系 统MYCIN等。但同时，人工智能研究也遭遇了重大挫折，在机器翻译方面就遇到不少问 题。例如因为多义词问题导致程序将Fruit flies like a banana翻译成“水果像香蕉一样飞 行”。在问题求解方面，即使对于具备良好结构的问题，程序也无法解决巨大的搜索空间问 题。这些问题使得人们对人工智能研究产生质疑。1973年，英国剑桥大学应用数学家 Lighthill认为“人工智能研究即使不是骗局，至少也是庸人自扰气 当时英国政府接受了他 的观点，取消了对人工智能研究的资助。由此可见，人工智能研究在这个时期遭遇了低潮。

面对困境，人工智能研究者们积极反思。1977年，在第五届国际人工智能联合会上， Feigenbaum做了题为《人工智能的艺术：知识工程课题及实例研究》的报告，首次提出了知 识工程的概念。他提出，知识工程是研究知识信息处理的学科，应用人工智能的原理和方 法，为那些需要专家知识才能解决的应用难题提供解决方法。釆用恰当的方法实现专家知 识的获取、表示、推理和解释，是设计基于知识的系统的重要技术问题。知识工程的提出以 及发展，使人工智能的研究从基于推理的模型转向基于知识的模型，使人工智能的研究从理 论走向了应用。这时期大量的专家系统被应用到各种领域并获得了成功。例如，1980年,

美国DEC公司开发了 XCON专家系统用于根据用户需求配置VAX计算机系统。通常人 类专家完成这项工作需要3小时，而该系统只用半分钟，大大提高了效率。

随着专家系统大量应用到实际中，其存在的问题也逐渐显现出来：一是交互问题，系统 只能模拟人类深思熟虑的行为,却无法处理人与环境的交互行为；二是扩展问题，传统的人 工智能方法只适合建造狭窄领域的专家系统，无法将方法推广到规模更大、领域更宽的复杂 系统中。

1. 人工神经网络

上面讲到基于知识工程的专家系统发展遇到了困境。这一困境使得人工智能研究者开 始把目光由传统的基于物理符号的系统转向区别于符号主义的连接主义和行为主义，由此 开始了人工神经网络和智能主体的研究与发展。

人工神经网络的研究起源于20世纪60年代，由于当时该领域的研究存在的局限性使 得人们几乎放弃了对人工神经网络的研究。直到1982年，美国加州理工学院物理学家霍普 菲尔特(Hopfield)使用统计力学的方法来分析网络的存储和优化特性，提出了离散的神经 网络模型，从而推动了神经计算的研究，标志着神经计算研究高潮的到来，1984年，他又提 出了连续的神经网络模型，被称为Hopfield网络模型。1985年，该模型较为成功地求解了 旅行商问题(Traveling Salesman Problem, TSP) o 1986 年,Rumelhart 和 McClelland 等人 提岀了并行分布处理理论，致力于认知的微观结构的探索，提出了反向传播(Back Propagation, BP)算法，成功解决了多层网络学习问题，成为广泛应用的神经元网络学习算法。1987年， 在美国召开了第一届神经网络国际会议，并发起成立国际神经网络学会(INNS)。

1. 智能主体

随着计算机网络、计算机通信等技术的发展，基于行为主义的智能主体的研究成为人工 智能研究的一个热点。智能主体成为人工智能的一个核心问题。1995年，斯坦福大学的 Barbara Hayes Roth在国际人工智能联合会议的特约报告中谈道：“智能的计算机主体既 是人工智能最初的目标，也是人工智能的最终目标。”近年来，智能计算机系统、智能机器人 和智能信息处理等方面的研究发展迅速，同时也取得了重大进展。

总之，从上述人工智能的发展历史可以看出，人工智能的发展经历了曲折的过程，但也 取得了许多成就。随着计算机网络技术和信息技术的不断发展，人工智能领域的研究也将 拥有更大的发展空间。相信在未来，分布式人工智能、智能系统间的通信、交互、协作等方面 的研究将给人工智能带来新的飞跃。

**1.3**人工智能的三大学派

**1.3.1**符号主义

符号主义又称逻辑主义、心理学派或计算机学派，其基本原理基于两点：物理符号系统 假设和有限合理性原理。

物理符号系统假设是Newell和Simon在1976年提出.，该假设认为：物理符合系统具 有必要且足够的方法来实现普通的智能行为。Newell和Simon把人类的一切精神活动、智 能问题归结为计算问题。一个物理符号系统主要有6种功能：输入、输出、存储、复制符号、 建立符号结构和条件性迁移。其中，建立符号结构即确定符号间的关系，条件性迁移需要依 赖已经掌握的符号继续完成后续行为。根据Newell和Simon提出的这个假设可知，一个 物理符号系统若能够完成上述6种符号操作就是具备智能的。计算机和人脑都是能操作符 号的物理符号系统，因此，计算机和人脑可以进行功能类比。据此，我们可以用计算机的符 号操作来模拟人的认知过程，相当于以此来模拟人的智能行为。

有限合理性原理是Simon提出的观点。他的观点强调：人类之所以能在大量不确定、 不完全信息的复杂环境下解决难题，原因在于人类采用了启发式搜索的试探性方法来求得 问题的有限合理解。

符号主义人工智能研究在自动推理、定理证明、机器博弈、自然语言处理、知识工程、专 家系统等方面取得了显著成果。符号主义主张从功能上对人脑进行模拟，将问题和知识以 逻辑形式表示，釆用符号推演的方式来实现推理、学习、搜索等功能。然而，由于符号主义的 核心是知识表示、知识推理和知识应用，对于“常识”问题以及不确定事物的表示和处理问题 成为符号主义需要解决的巨大难题。

**1.3.2**连接主义

连接主义又被称为仿生学派或生理学派，是基于神经元及神经元之间的网络连接机制 来模拟和实现人工智能。人类智能的物质基础是神经系统，其基本单位是神经元。这也就 是说，连接主义用人工神经网络来研究人工智能。

1943年，McCulloch和Pitts提岀第一个神经元数学模型——MP模型。该模型从结构 上对人脑进行模拟。人工神经网络方法一般先通过神经网络的学习获得知识，再利用知识 解决问题。神经网络以分布式方式存储信息，以并行方式处理信息，具有实现自组织、自学 习的能力。人工神经网络为人工智能研究提供了一种新思路，在模式识别、机器学习、图像 处理等方面很有优势。

局限于研究者对人脑的生理结构和工作机理的认识，目前的人工神经网络仅能近似模 拟人脑的局部功能，且不适合模拟人类的逻辑思维过程。因此，单靠连接机制无法解决人工 智能的所有问题。

**1.3.3**行为主义

行为主义又被称为进化主义或控制论学派，是基于控制论和“感知-动作”控制系统的人 工智能学派。其观点是：智能取决于感知和行为，取决于对外界复杂环境的适应。人类智 能是经历漫长的演化形成的，真正的智能机器也应该沿着进化的步骤走。

行为主义的基本观点可以概括为：

•知识的形式化表达和模型化方法是人工智能的重要障碍之一。

•智能取决于感知和行为，在直接利用机器对环境作用后，以环境对作用的响应为 原型。

•智能行为体现在现实世界中，通过与周围环境的交互表现出来。

•人工智能可以像人类智能一样逐步进化，分阶段发展和增强。

1991年，麻省理工学院的Brooks教授提出了无须知识表示的智能和无须推理的智能。他 认为智能只能在与环境交互过程中表现出来，不应该采用集中式的模式，应该需要具有不同的 行为模式与环境交互，并以此产生复杂行为。他成功研制出了一种由相互独立的功能单元组 成的6足机器虫，该机器虫由避让、前进和平衡等基本功能模块组成分层异步分布式网络。

行为主义实际上是从行为上进行智能模拟，使得机器具有自寻优、自适应、自学习和自 组织能力。目前，行为主义在智能控制、机器人领域取得了巨大成就，在未来，控制论、系统 工程的思想将进一步影响人工智能的发展。

符号主义、连接主义和行为主义三大思想反映了人工智能的复杂性。这三个学派从不 同的角度出发阐释了智能的特性，都取得了显著成就，但同时也存在发展的局限性。目前， 对于人工智能的研究仍没有一个统一的理论体系，这种不统一的存在促进了新思潮、新方法 的不断涌现，极大地丰富了人工智能的研究。三个学派各有优点也各有局限，可以融合这三 个学派，取长补短，为未来的人工智能研究提供新的思路。

**1.4**人工智能研究内容与应用领域

**1.4.1**问题求解

人工智能的最早应用实践就是问题求解。该领域最有名的例子就是下棋程序。问题求 解是指通过搜索的方法寻找目标解的一个合适的操作序列，并同时满足问题的各种约束。 在解决良结构问题时，通常有巨大的搜索空间，理论上可以用穷举法找到最优解，但是实践 时却由于时空约束而无法得到最优解。因此，问题求解的核心就是搜索技术。

一般搜索系统包括全局数据库、算子集和控制策略3部分。

全局数据库中含有与具体任务相关的信息，这些信息可以用来反映问题的当前状态、约 束条件和预期目标。可以根据具体问题釆用逻辑公式、数组、矩阵等不同的数据结构。状态 分量的选择应该满足必要性、独立性和充分性。各分量不同的取值组合对应不同的状态，只 有一些状态是求解问题所需要的。问题本身所具有的约束条件可以排除非法的状态和不可 能出现的状态，数据库中只保留问题的初始状态、目标状态和中间状态。

算子集也称操作规则集，用来对数据库进行操作运算。算子一般包括条件和动作两部 分。条件给出了适用于算子的先决条件，动作表述了适用算子之后的结果，即引起状态中某 些分量的变化。

控制策略可以决定下一步选用哪一个算子以及在何处应用。控制策略通常选择算子集 中最有可能导致目标状态或者最优解的算子，运用到当前状态上，不然可能会引起组合爆炸 等问题。

搜索技术的最大难点在于寻找合理有效的启发式规则。

**1.4.2**专家系统

专家系统是一个具有大量专门知识与经验的程序系统，它采用人工智能技术，根据特定

领域中一个或者多个人类专家提供的知识和经验进行推理和判断，模拟人类专家的决策过 程,用来解决一些需要专家决定的复杂问题。目前专家系统在医疗诊断、故障诊断、资源勘 探、贷款损失评估和教学等领域中得到了广泛应用。

专家系统是一种基于知识的系统，这种系统的设计方法以知识库和推理机为中心展开。 知识库包含大量的领域专家提供的知识，推理机能够使用知识库里的知识进行推理并解决 实际问题。

专家系统通常由知识库、推理机、综合数据库、人机交互界面、解释器和知识获取等部分 构成。其中，知识库用来存放专家提供的知识，知识库中知识的质量和数量决定了专家系统 的质量水平。推理机根据当前的已知信息，反复匹配知识库中的规则产生新结论，最后得到 问题的求解结果。综合数据库用来存储推理过程中的原始数据、中间结果和最终结论。解 释器对结论和求解过程作出解释。人机交互界面用于与用户进行交流。知识获取可以完成 对知识的采集和输入到知识库的过程。

专家系统的工作流程如下：用户通过人机界面提交问题和已知条件。推理机将用户输 入的信息与知识库中的规则进行匹配，并将中间结论存放在综合数据库中。若得到最终结 论则推理结束并输出结果，否则输出推理失败。最后，解释器对结论作出解释。

由于专家系统知识获取需要依赖领域专家，需要大量的人工处理。当这些信息呈现爆 炸式增长时，如何自动获取知识成为专家系统的瓶颈问题。这一问题制约了专家系统的 发展。

**1.4.3**机器学习

机器学习研究如何使计算机能够模拟人类的学习功能，从大量的数据中发现规律提取 知识,并在实践中不断完善和增强自我。机器学习是机器获取知识的重要途径，它是人工智 能研究的核心问题之一，同时也是人工智能理论研究和实际应用中的一个主要瓶颈。 Simon认为，机器学习是系统在不断重复的工作中对本身能力的一种增强和改进，这样系统 就能在下一次执行同样任务时拥有更高的执行效率。

传统的机器学习倾向于使用符号而不是数值表示，使用启发式方法而不是算法,传统的 机器学习依赖使用归纳法，而不是演绎。按照系统对导师的依赖程度可以将学习方法分类 为机械式学习、讲授式学习、类比学习、归纳学习、观察式学习等。近年来又发展了增强学习 以及数据挖掘、知识发现等学习方法。

**1.4.4**神经网络

人工神经网络简称神经网络，其研究始于1943年McCulloch和Pitts提出的神经元的 数学模型（MP模型）。人工神经网络是以对人脑和自然神经网络的生理研究成果为基础， 抽象和模拟人脑的某些机理、机制实现某方面的功能。Nielsen给出的人工神经网络定义 为：“人工神经网络是由人工建立的以有向图为拓扑结构的动态系统，它通过对连续或断续 的输入作状态响应而进行信息处理。”

目前，一般的人工神经网络适用于难以应用严格解析方法的场合，包括特征提取、联想

第1章绪论 记忆、低层次感知、模式分类以及自适应控制等。主要的研究集中在以下几个方面：

•对人工神经网络的软件模拟和硬件实现的研究。

•人工神经网络在模式识别、信号处理、知识工程、专家系统、优化组合以及机器人控 制等领域的应用。

•利用神经生理与认知科学研究人类思维和智能推理。

•深入研究神经网络算法和性能。

•开发新的神经网络数理理论，如神经网络动力学和非线性神经场等。

**1.4.5**模式识别

模式识别一般指应用电子计算机及外部设备对给定事物进行鉴别和分类，将其归入与 之相同或者相似的模式中。模式识别的一般过程包括对待识别事物采集样本、数字化样本 信息、提取数字特征、学习和识别。其核心是特征提取和学习过程。针对不同的识别对象， 模式识别可以分为以下几种类型：

•语音识别。主要研究各种语音信号的识别、翻译以及语音人机交互界面等。语音识 别技术发展得比较成熟，计算机可以识别人类语音并将其转化为文本。

•图形、图像识别。主要研究各种图形、图像的处理和识别技术，包括指纹识别、人脸 识别、车牌识别等系统早已经进入实际应用领域。

•信号识别。主要研究各种传感器信号，例如对雷达、声呐、地震波等信号的识别在军 事、地震预测和医学上早有重要应用。

•染色体识别。主要研究识别染色体以用于遗传因子研究等。

**1.4.6**数据挖掘和知识发现

数据挖掘和知识发现起源于20世纪90年代初期，在早期，它侧重于构建数据仓库，然 后运用切片、下钻、上卷等操作从海量数据中发现有用信息。如今的数据挖掘对象包括无结 构和半结构文本数据，音频、视频、图像等多媒体数据均可以成为数据挖掘的源数据来源。 在数据库的基础上，实现知识发现系统，通过综合运用统计学、粗糙集、模糊数学、机器学习 和专家系统等多种学习手段和方法，从大量的数据中提炼出抽象的知识，从而揭示出蕴含在 这些数据背后的客观世界的内在联系和本质规律，实现知识的自动获取。

**1.4.7**计算机视觉

计算机视觉是一门用计算机实现或模拟人类视觉功能的新兴学科，其主要研究目标是 使计算机具有通过二维图像认知三维环境信息的能力，这种能力不仅包括对三维环境中物 体形状、位置、姿态、运动等几何信息的感知，而且还包括对这些信息的描述、存储、识别与理 解。目前，计算机视觉已在人类社会的许多领域得到成功应用。例如，在图像、图形识别方 面有指纹识别、染色体识别等；在航天与军事方面有卫星图像处理、飞行器跟踪、成像精确 制导、景物识别、目标检测等；在医学方面有脏器的图像重建、医学图像分析等；在工业方 面有各种监测系统和生产过程监控系统等。

**1.4.8**智能控制

智能控制是一类无须或者尽可能少的人的干预就能够独立完成任务的自动控制。许多 复杂系统无法建立有效的数学模型和用常规控制理论进行定量计算和分析，而必须采用定 量的数学解析法与基于知识的定性方法的混合控制方式。随着人工智能和计算机技术的快 速发展，已有可能把控制和人工智能以及系统科学的某些分支结合起来，建立一种适用于复 杂系统的控制论和技术。

智能控制涉及的领域很多，目前主要有以下6个方面：智能机器人规划与控制、智能过 程规划、智能过程控制、专家控制系统、语音控制以及智能仪器。

1998年10月24日，宇宙飞船深空一号发射升空，其目的是为了测试12项高风险技 术。飞行的成功使其使命延长，深空一号最终在2001年12月18日退役。该软件就是称为 远程代理的一个人工智能系统，它能够规划和控制宇宙飞船的活动。

**1.4.9**计算智能

计算智能主要借鉴仿生学的思想，应属于智能学习部分，目前还没有一个统一的定义， 美国科学家贝兹德克(Bezdek)认为不具有计算适应性、计算容错力、接近人的计算速度和近 似人的误差率这几个特性，就不能称之为计算智能，其涉及的主要领域有模糊计算、神经计 算、进化计算、免疫计算等。就目前而言，在前3个领域发展相对比较成熟。

模糊计算是一种对人类智能的逻辑模拟方法，他是用模糊逻辑去模拟人类的智能行为。 其理论研究是基于1965年扎德(Zadeh)教授发表的《模糊集合》开始的。神经计算又叫做神 经网络，是计算智能的基础和核心，其主要包括人工神经元的结构和模型，人工神经网络的 互联网络结构和系统模型，基于神经网络的联结学习机制等。进化计算是一种模拟自然生 物进化过程与机制进行问题求解的自组织、自适应的随机搜索技术。

**1.4.10**其他

除了以上研究内容外，人工智能领域的研究内容还包括自然语言处理、分布式人工智 能、人工生命、机器人学、智能检索以及最近风生水起的大数据、深度学习等。

**1.5**人工智能的发展趋势

**1.5.1**多学科交叉研究

从人工智能的起源便知，它是一门多学科交叉科学，在今后同样也是，在近期的研究中 应注意其与信息科学、生物学、数学、心理学等学科的交叉研究。信息科学可以很好地为人 工智能研究提供智能模拟的物质基础和技术支持；生物学可以为人工智能研究提供自然界

生物生存和进化的机制；心理学科可以为人工智能研究提供认知、情感、意识等心理过程及 联系。

**1.5.2**智能应用和智能产业

智能技术将进一步与主流信息技术融合，并将应用于人类社会的各个领域和人类生活 的各个方面。有人预言，智能产业将成为社会第四产业，智能将逐步从软件中分离出来，成 为智能计算机系统的单独部分。

**1.6** 习题

1 .什么是人工智能？人工智能目前的主要成就有哪些？

1. 简述人工智能的发展历史。
2. 简述人工智能的三大学派。
3. 人工智能有哪些研究和应用领域？试举例说明人工智能研究取得了什么成果。

知识表示

**2.1**概述

**2.1.1**知识及知识的分类

知识是人们在改造客观世界的实践中积累起来的认识和经验。这些经验的描述又需要 涉及数据和信息的概念。数据是记录信息的符号，是信息的载体和表示。信息是对数据的 解释，是数据在特定场合下的具体含义。信息仅是对客观事物的一种简单描述，只有经过加 工、整理和改造等工序，并形成对客观世界的规律性认识后才能成为知识。从不同角度，可 以将知识分成不同的类型，如下所示：

按知识性质：概念、命题、公理、定理、规则、方法等

（常识性知识，即通用知识

按知识适应范围

丨领域性知识，即专业性知识

事实性知识（又称叙述性知识）

按知识的作用效果｛过程性知识

知识］ I控制性知识（又称元知识或超知识）

（确定性知识

按知识的确定性

［不确定性知识

按知识的等级：零级知识、一级知识、二级知识等

（逻辑性知识

'按知识的结构

〔形象性知识

常识性知识是指通用通识的知识。即人们普遍知道的、适用于所有领域的知识。

领域性知识是指面向某个具体专业的专业性知识，这些知识只有相应专业领域的人员 才能掌握并用来求解领域内的有关问题，如领域专家的经验等。

事实性知识也称叙述性知识，是用来描述问题或事物的概念、属性、状态、环境及条件等 情况的知识，常以“……是……”的形式出现。事实性知识主要反映事物的静态特征，是知识 库中低层的知识。例如，“我们是中国公民”“张三是一名人民教师”“小李和小张是好朋友” 等都是事实性知识。

过程性知识是用来描述问题求解过程所需要的操作、演算或行为等的规律性知识，一般 由规则、定律、定理及经验构成。

控制性知识是指有关如何选择相应的操作、演算和行动的比较、判断、管理和决策的知 识。控制性知识常与元知识有所重叠，所谓元知识是指有关知识的知识，是知识库中的高层 知识。对一个大的程序来说，以元知识或元规则形式体现控制知识更为方便，因为元知识存 于知识库中，而控制知识常与程序结合在一起出现，从而不容易修改。

确定性知识是指可以给出其真值为“真”或“假”的知识，是可以精确表示的知识。

不确定性知识是指具有不确定特性（不精确、模糊、不完备）的知识。不精确是指知识本 身有真假，但由于认识水平等限制却不能肯定知识的真假•可以用可信度、概率等描述。模 糊是指知识本身的边界就是不清楚的，例如大、小等，可以用可能性、隶属度来描述。不完备 是指解决问题时不具备解决该问题的全部知识，例如医生看病。

零级知识即陈述性知识或事实性知识,用于描述事物的概念、定义、属性，或状态、环境、 条件等，回答“是什么”“为什么"。

一级知识即过程性知识或程序性知识,用于问题求解过程的操作、演算和行为的知识， 即如何使用事实性知识的知识，回答“怎么做”。

二级知识即控制性知识或策略性知识,是关于如何使用过程性知识的知识，例如推理策 略、搜索策略、不确定性的传播策略等。通常把零级知识和一级知识称为领域知识，把二级 知识称为元知识（也称超知识）。

逻辑性知识是指反映人类逻辑思维过程的知识，一般具有因果关系或难以精确描述的 特点，是人类的经验性知识和直观感觉，例如人的为人处事的经验与风格。

形象性知识是指通过事物的形象建立起来的知识，例如一个人的相貌。

**2. 1. 2**知识表示

人工智能问题的求解是以知识表示为基础的，如何将已获得的有关知识表示成计算机 能够描述、存储、有效地利用的知识是必须解决的问题。知识表示实际上就是对知识的描 述，即用一些约定的符号把知识编码成一组能被计算机接受并便于系统使用的数据结构。 常用的知识表示方法有一阶谓词表示法、产生式表示法、语义网络表示法、框架表示法、过程 表25法、脚本表示法、本体表示法等。

不同的知识表示方法有各自的优缺点，在考虑使用哪一种知识表示方法时应遵循以下 相关原则。其一，是否能够充分表示领域性知识，即对症下药，不过有的时候也要根据具体 情况，为了克服某种方法的不足，而釆用多种知识表示方法结合，如为了弥补框架表示法不 能表示过程性知识的不足，通常将其与产生式表示法结合使用。其二，是否具备可利用性。 可利用性是指通过使用知识进行推理，以求解现实问题。如果不可利用，则会影响系统的推 理效率。其三，是否可以对知识进行组织管理。在一个智能系统初步使用时，可能会发现知

识表示方面存在某些问题，此时需要对知识进行增添或删减。长期使用后也需要进行维护。 最后，在知识表示方法选择过程中，最重要的还是要便于理解和实现。如果不能被我们自己 理解和实现，那同样还是徒劳的。

**2.2**谓词逻辑表示法

**2.2.1**基本概念

掌握每一种语言都需要先学习一些单词、语法等，在理解与掌握各种知识表示方法前, 也要掌握一些基本的概念、运算等。首先介绍论域的概念，所有讨论对象的全体构成的非空 集合称为论域。论域中的元素称为个体。在谓词逻辑中，命题是用谓词来表示的，故在此给 出谓词的定义。

定义2.1设D是论域是一个映射，其中

*D”* = {（X）,x2 <•, ,xn） I Xi *9x2 »,•,* G *D}*

则称*P*是一个〃元谓词3=1,2,3,…），记为F3,跖）。其中，珈，…，弓 为个体 变元。

定义*2.2* 设D是论域*D—D*是一个映射，则称/是D上的一个〃元函数，记为

**/（X]**,工**2**，…，石）

其中*g，…F* 是个体变元。

每一个谓词由谓词名和个体组成。其中，个体是命题的主语，用来表示某个独立存在的 事物或者某个抽象的概念；谓词名是命题的谓语，用来表示个体的性质、状态或个体之间的 关系等。

命题可以分为原子命题和复合命题，后者由前者通过联结词复合而成。下面介绍常用 联结词。

设P、Q是命题，常用的联结词如下：

（1） 「（否定或非）。对任一命题则表示对命题P的否定。

（2） V （析取）。复合命题PVQ表示P或Q的析取，艮卩P或Q。

（3） A（合取）。复合命题PAQ表示P和Q的合取，即P与Q。

（4） -（条件或蕴含）。它表示“若……则……'‘的语义。

（5） -（双条件）。它表示“当且仅当”的语义。

有了以上的联结词，可以得到表2. 1所示的真值表。

表2.1谓词真值表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| *P* | Q |  | PAQ | PVQ | P—Q |  |
| T | T | F | T | T | T | T |
| T | F | F | F | T | F | F |
| F | T | T | F | T | T | F |
| F | F | T | F | T | F | T |

命题具有如下性质：

1. 蓦等律：PVP0P,PAP0P。
2. 交换律：PVQoQVP,PAQ冃QAP。
3. 结合律：(PVQ) VR0PV(QVR),(PAQ) ARuPA(Q/\R)。
4. 分配律：PV(QAR)0(PVQ) A(PVR),PA(QVR)0(PAQ) V(PAR)。
5. 吸收律：PV(QAR)0P,PA(QVR)0F。
6. De-Morgan 法则：「(P V Q)0「PA -■ Q,「(P A Q)0*「p\j*「Q。

在一阶谓词逻辑中除了以上的连接词还有两个量词，即全称量词和存在量词，分别用来 对个体做出量的刻画。它们的符号分别为V和不。全称量词的定义为：命题(Vz)P(z)为 真，当且仅当对论域中的所有了都有P(z)为真；反之，命题为假时，当且仅当至少存在一个 使得P(xo)为假。存在量词的定义为：命题(mi)P(Q为真，当且仅当至少存在一 个工。£。使得*P(瓦)为*真；反之，命题为假，当且仅当对论域中的所有z都有P&)为假。

联结词的优先级从高到低依次为*r,A,N,f,f*

*2.2.2*谓词逻辑表示法

谓词逻辑表示法是一种基于数理逻辑的知识表示方法，人工智能所用的逻辑包括一阶 经典逻辑和除此以外的非经典逻辑。这里主要讨论一阶经典逻辑表示法，这里所提到的谓 词逻辑也指一阶谓词逻辑。谓词逻辑表示法不仅可以用来表示事物的状态、属性、概念等事 实性知识,还能表示事物的因果关系。

用谓词逻辑表示法表示知识的步骤如下：

1. 根据要表示的知识定义谓词及个体，确定每个谓词及个体的确切含义。
2. 根据所要表达的知识的语义，用适当的连词、量词把这些谓词连接起来。

例2.1用谓词逻辑表示下列知识：

1. 所有整数不是偶数就是奇数。
2. 所有父母都有自己的孩子。
3. 偶数除以2是整数。

解：首先定义谓词。

Z&)：表示工是整数。

D(t)：表示z是偶数。

S(JT)：表示1是奇数。

PARENT(z)：表示z是父母。

CHILDREN(^)：表示、是孩子。

PARENTS,y)：表示］是)的父母。

另外用。(了)表示除以2。

这样就可以表示以上知识如下：

(Vx)(Z(x) -> *D(x)* V S(x))

(Vjc)( %)(PARENT(z)PARENT(p) A CHILDREN(y))

(V^)(D(x) — Z(e(z)))

第二个谓词公式可读为：对于所有的是父母，则存在一个火夕是一个孩子，且*工* 是丁的父母。另外需要说明的是，在表示谓词时一般用大写字母,个体域用小写字母。

**2.2.3**谓词逻辑表示法的经典应用

为了更好地理解一阶谓词知识表示法，先讨论一个经典的应用一一机器人移盒子问题。 问题1：设在一个房间里，C处有一个机器人\*和。处 嬴

各有一张桌子，分别记为a桌和。桌g桌上有一个盒子，如 禮

图2.1所示。现在要求机器人走到Q处将。桌上的盒子拿 7

起放到厶桌上，再回到c处。

对于该问题不仅要考虑事物的状态、位置，还要考虑机 V

器人的动作过程。首先定义谓词如下。

TABLE(z):表示*工*是桌子。 图2・1机器人移盒子

EMPTY3)：表示了手上是空的。

表示；y 在 z 处。

HOLD(j^,w):表示夕拿着初。

ON(w,^r):表示实在飞处。

其中口的个体域是*｛a,b｝9y*的个体域是｛机器人｝次的个体域是修,们c｝，s的个体域 是｛box｝0

下面写出问题的初始状态和终止状态。

问题的初始状态：

TABLE(a)

TABLE(A)

EMPTYC robot)

AT( robot ,c)

ON(boxg)

问题的终止状态：

TABLE(a)

TABLE(b)

EMPTY(robot)

ATCrobot ,c)

ON(boxM)

机器人行动的目标就是从初始问题状态到终止问题状态，然而要很好地求解问题，还需 要完成一系列的操作，即机器人需要从*C*处移动到Q处，拿起。桌子上的盒子，然后移动*b* 处，再将盒子放下，最后再移动到［处。总的来说，可以将操作分为条件和动作。条件部分 很容易用谓词公式表示，而动作则需要通过在执行该操作前的问题状态中删去和增加相应 的谓词来实现。要实现终止状态，需要执行以下3个操作：

GOTOCz, 丁)，表示从z移动到了处。

PICKUP(z),表示拿起z。

SETDOWN(z),表示放下工。

这3个操作分别可以用条件和动作来表示：

GOTO(«z,少

条件：AT(robot,x)

动作：删除表AT(robot,x)

增加表 AT(robot,J；)

PICKUP(x)

条件：ONCbox, X),TABLE(x) , AT(robot,x), EMPTY(robot)

动作：删除表 ON(box, z),EMPTY(robot)

增加表 HOLD(robot,box)

SETDOWN(x)

条件：TABLE(z) ,AT(robot,z), HOLD(robot, box)

动作：删除表 HOLD(robot, box)

增加表()N(box, z),EMPTY(robot)

机器人在每执行一个操作前，总要先检查当前状态条件是否可以使所要求的条件得到 满足，只有满足条件的情况下才会执行操作，否则，计算机检查下一个操作所要满足的条件。 有了这些条件和动作，就可以给岀机器人移盒子的问题求解过程，如下所示：

'TABLE(a)

TABLES)

TABLE。) GOTO(c. *a)*

EMPTY(robot) " A 状态2

TABLE⑴

begin

==>初始状态1

\ EMPTY(robot)

AT(robot，*c)*

AT(robot. *a)*

、ON(box. *a)*

ON(box. *a)*

TABLE(a)

TABLE(a)

TABLE0)

HOLD( robot,box) k AT(robot. *a)*

TABLED) HOLD( robot,box)

GOTO(a. *b}*

【 A状态4

P1CKUP0)

**~~L~~** >状态3

、AT(robot. *b)*

'TABLE(a)

(TABLE(a)

GOTO。，c)

TABLE(方)

TABLE0)

SETDOWN(Z>)

**~~I~~** >状态5 <

状态6

< ON(box. *b)*

ON(box，*b)*

EMPTY(robot)

EMPTY(robot)

、AT(robot, *b)*

仔细观察上述过程，可以看出，当机器人处于状态3时，既满足GOTOa以)条件，又满 足SETDOWN&)条件，此时，机器人该如何进行操作呢？实际上，每当机器人执行下一个 操作前，都要检验下一状态是否为终止状态。若是，则问题已经得到解决；若不是，则检査 下一状态是否与之前状态相同，若相同则回溯到上一状态，执行另一操作。

从操作过程可以看出，机器人选择了 GOTO(a,6)进入状态4：

TABLE(a)

TABLE。)

HOLD（ robot, box）

AT（ robot ,。）

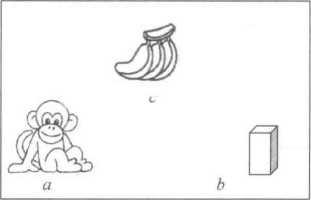
而如果执行SETDOWN（z），将变回状态2结果。

图2. 2猴子摘香蕉

在谓词逻辑中，还有一个经典应用就是猴子摘香 蕉问题，在此不作详细解释，下面仅给出问题，供读者 思考。

问题**2（**猴子摘香蕉问题）：如图2. 2所示，设房间 里有一只猴子位于。处。在c处上方的天花板上有一 串香蕉，猴子想吃，但摘不到。房间的。处还有一个箱 子，如果猴子爬到箱子上，就可以拿到香蕉。

**2.2.4**谓词逻辑表示法的特点

我们已经知道谓词逻辑表示法，主要依赖形式逻辑，利用条件和结论之间的蕴含关系。 谓词逻辑表示法在使用过程中显得接近于自然语言系统且比较灵活，接近人们对问题的理 解，易于被人们接受。该表示方法还具有模块化特点，每个知识都是相对独立的，它们之间 不直接发生某种联系，而是通过添加、删除、修改知识进行。但是，谓词逻辑表示法也有自己 的不足之处，例如该方法只能表示确定性的知识，对于模糊概念，即不确定性知识，是无法表 示的，然而，人们生活中大多数实际问题都具有某种不确定性。另外，该方法效率低，例如机 器人在移动盒子过程中，就要考虑各种条件和动作，机器人需要先移动到。处的桌子，然后 再拿起盒子。

**2.3**产生式表示法

**2.3.1**概述

产生式表示法在人工智能中的应用非常广泛，因为它的求解过程和人类求解问题的思 维过程很相像，可以用来模拟人类求解问题的思维过程。产生式系统是由美国数学家E. Post于1943年作为组合问题的形式化变换理论首先提出来的。产生式表示法也常称为产 生式规则表示法,许多成功的专家系统都釆用了这种知识表示方法。例如，1965年斯坦福 大学设计的第一个专家系统DENDRAL就采用了这种知识表示方式。1972年,Newell和 Simon在研究人类的认知模型中开发了基于规则的产生式系统。

产生式系统的知识表示方法主要包括事实和规则两种表示。

**1.**事实的表示

事实可以看做一个语言变量的值或断言或者多个语言变量间的关系的陈述句。语言变 量的值或语言变量间的关系可以是一个词。对于确定性知识，事实通常用一个三元组来表 示，即（对象，属性，值）或（关系，对象1,对象2）,其中，对象就是语言变量。对于不确定性知 识，事实通常用一个四元组来表示，即（对象，属性，值，可信度因子）或（关系，对象1，对象2, 可信度因子），其中，可信度因子是指该事实为真的可信程度，类似于模糊数学中的隶属程度，可用一个。〜1的数来表示。

例**2.2** “雪是白的”可表示为(snow, color, white)或(雪，颜色，白)。

“老王和老张是朋友”可表示为(friendship, wang, zhang)或(朋友，老王，老张)。

"李二比王三年龄要大很多”可以用四元组来表示(large than,lier, wangsan,0. 9) o 该表示可以理解为李二比王三大很多的可信度为0.9,即当可信度因子越高，事实就越 “真气

**2.**规则的表示

规则表示的是事物间的因果关系。其表现形式为

*P — Q*

或

IF *P* THEN *Q*

其中，P表示前提条件,Q表示所得到的结论或一组操作，这里需要注意的是，要得到结论， 需要前提条件必须为真。该规则又称产生式，类似于谓词逻辑中的蕴含式，但有所区别，区 别在于蕴含式只能表示确定性知识，而产生式不仅可以表示确定性知识，还能表示不确定性 知识。例如，在MYCIN专家系统中，有这样的产生式：

P：细菌革式染色阴性

形态杆状

生长需氧

*Q：*该细菌是肠杆菌属，可信度为0.8

在该规则中，所包含的事实就是不确定性知识，当前提条件满足时，就有结论“该细菌是 肠杆菌属”，可以相信的程度是0. 8O这个0. 8是规则强度，这是蕴含式所不能做到的。

**2.3.2**产生式系统

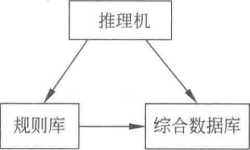
所谓产生式系统是指一组产生式相互配合，协同作用，以求得问题的解。产生式系统一 般由3个基本部分组成，分别为规则库、综合数据库以及推理 机，它们之间的关系如图2. 3所示。

图2. 3产生式系统的基本结构

规则库又称知识库，是某领域知识用规则形式表示的集 合。该集合包含了问题初始状态以及转换到目标状态所需要 的所有变化规则。规则库是产生式系统的基础。其相关特性 都将影响系统的运行效率。

综合数据库又称事实库，是用来存放当前与求解问题有 关的各种信息的数据集合。包括问题的初始状态信息、目标状态信息以及在问题求解过程 中产生的临时信息。当从规则库中取出的某规则的前提与综合数据库中的已知事实相匹配 时,该规则被激活，由该规则库得到的结论就是中间信息，将被添加到综合数据库中。

推理机又称控制系统，由一组程序组成，用来控制和协调规则库与综合数据库的运行， 决定了问题的推理方式和控制策略。即推理机按照一定的策略从规则库中选择与综合数据 库中的已知事实相匹配的规则进行匹配，当匹配有多条时，推理机应能按照某种策略从中找 出一条规则去执行，对要执行的规则，如果满足后件，则结束问题求解，如果该规则的后件不 是问题的目标，则当其为一个或多个结论时，把这些结论加入到综合数据库中，循环操作，直 至满足结束条件。

**2.3.3**产生式表示法应用挙例

例2. 3（动物识别系统）设该系统可以识别金钱豹、老虎、斑马、长颈鹿、企鹅、信天翁 6种动物。规则库包括15条规则，如下所示：

门:IF该动物有毛发THEN该动物是哺乳动物

a : IF该动物有奶THEN该动物是哺乳动物

厂3: IF该动物有羽毛THEN该动物是鸟

r4: IF该动物会飞AND会下蛋THEN该动物是鸟

r5： IF该动物吃肉THEN该动物是肉食动物

r6： IF该动物有犬齿AND有爪AND眼盯前方THEN该动物是肉食动物

r7: IF该动物是哺乳动物AND有蹄子THEN该动物是有蹄类动物

厂8: IF该动物是哺乳动物AND是反刍动物THEN该动物是有蹄类动物

r9: IF该动物是哺乳动物AND是食肉动物AND是黄褐色AND身上有暗斑点

THEN该动物是金钱豹

no: IF该动物是哺乳动物AND是食肉动物AND是黄褐色AND身上有黑色条纹

THEN该动物是老虎

rn : IF该动物是有蹄类动物AND有长脖子AND有长腿AND身上有暗斑点

THEN该动物是长颈鹿

r12： IF该动物是有蹄类动物AND身上有黑色条纹THEN该动物是斑马

r13: IF该动物是鸟AND有长脖子AND有长腿AND不会飞

THEN该动物是鸵鸟

f : IF该动物是鸟AND会游泳AND不会飞AND有黑白二色

THEN该动物是企鹅

r15: IF该动物是鸟AND善飞THEN该动物是信天翁

在此，我们讨论机器识别长颈鹿的推理过程，在推理前，明确综合数据库中已有的事实：

动物有暗斑，有长脖子，有长腿，有奶，有蹄

推理开始后，首先从规则库中取出第1条规则r,，检查其前提条件与综合数据库中已有 的事实是否匹配。因为事实库中没有毛发，与门不匹配，匹配失败。进而再从规则库中选 取第2条规则a，该前提条件是“有奶”，综合数据库中有这一条件，故匹配成功，则得到该条 件的结论，该动物是哺乳动物，但是由于该结论不是所要的终止条件（判断是某种具体动 物），于是推理机将该结论放到综合数据库中，此时综合数据库变成如下状态：

动物有暗斑，有长脖子，有长腿，有奶，有蹄，该动物是哺乳动物

接着继续从规则库中选取第3条到第6条规则进行匹配，结果匹配都失败，再取第7条 规则，该条件与综合数据库中的“该动物是哺乳动物”匹配，与之前取出的第2条规则一样, 再次更新综合数据库如下：

动物有暗斑，有长脖子，有长腿，有奶，有蹄，该动物是哺乳动物，是有蹄类动物

以此类推，直到第11条规则被执行，得到结论“该动物是长颈鹿二

根据以上推理过程，可以总结出产生式系统的问题求解一般步骤如下：

（1） 初始化综合数据库（事实库）。

（2） 检测规则库中是否有与事实库相匹配的规则，若有，则执行（3）,否则执行（4）。

（3） 更新综合数据库，即添加步骤（2）所检测到与综合数据库匹配的规则，并将所有规 则做标记。

（4） 验证综合数据库是否包含解，若有，则终止求解过程，否则转（2）。

（5） 若规则库中不再提供更多的所需信息，则问题求解失败,否则更新综合数据库，转（2）。

当仔细观察本例后可以发现，推理机是按顺序逐一进行规则验证的，那么，机器为什么 不会返回已经执行过的条件再次执行呢？这就要归结于系统控制策略问题了。在产生式系 统求解问题中，主要有两种方式，其一是不可撤回方式,其二是试探性方式。

不可撤回方式就是一种“一直往前走”不回头的方式，该方式是利用问题给定的局部知 识来决定选用的规则，就像动物识别系统中一样，选取一条与综合数据库进行匹配，然后作 用到综合数据库中，再选取一条新的规则进行匹配，此处在选择时不会再考虑已经用过的规 则了。某种程度上会影响系统找到问题的解，且当问题有多解时，不一定能找到最优解。

试探性方式又可以分为回溯方式和图搜索方式。回溯方式是一种碰壁回头方式，回溯 策略是一种完备而有效的策略，容易实现，且所需内存小。但使用回溯策略需要解决两方面 问题,一方面是怎样确定回溯条件，另一方面是怎样减少回溯次数。图搜索方式用图或树记 录全部求解过程，这样便于求解最优路径。

**2.3.4**产生式系统的推理方式

**1.**正向推理

正向推理也称为数据驱动式推理，从已知事实出发，通过规则库求得结论。其基本推理 过程如下：

第1步，用数据库中的事实与可用规则集中所有规则的前件进行匹配，得到匹配的规则 集合。

第2步，使用冲突解决算法，从匹配规则集合中选择一条规则作为启用规则。

第3步，执行启用规则的后件,将该启用规则的后件送入综合数据库或对综合数据库进 行必要的修改。

第4步，重复这个过程，直到达到目标或者无可匹配规则为止。

在推理过程中，当前事实可能与规则库中的多条规则匹配，而每次推理时却只能执行一 条规则，这时就产生了冲突。解决这个冲突的过程称为冲突消解。冲突消解就是从多条可 用的规则中选取一条规则作为当前执行规则。冲突消解一般的思路就是给所有可用规则排 序，然后依次从队列中取出候选规则。在不考虑利用启发知识的情况下，常用的排序依据 如下：

（1）专用与通用性排序。如果某一规则的条件部分比另一规则的条件部分所规定的情 况更为专门化，则优先使用更为专门化的规则。所谓专门化就是子条件更多。一般而言，如 果某一规则的前件集包含另一规则的所有前件，则前一规则较后一规则更为专门化。如果 某一规则中的变量在第二规则中是常量，而其余相同，则后一规则比前一规则更专门化。

1. 规则排序。通过对问题领域的了解，规则集本身就可划分优先次序。那些最适用 的或使用频率最高的规则被优先使用。
2. 数据排序。将规则中的条件部分按某个优先次序排序。
3. 规模排序。按条件部分的多少排序，条件多者优先排序。
4. 就近排序。最近使用的规则排在优先位置，这样可以让使用多的规则排在较前面 的位置而被优先获取。
5. 按上下文限制将规则分组。

对于包含启发式信息的推理，除了可以采用以上冲突消解策略外，还可以考虑以下 策略：

1. 成功率高的规则优先被执行。
2. 按规则先前执行的性价比排序。

正向推理方式的主要优点是简单明了，且能求出所有解；主要缺点是执行效率低，因为 在推理过程中可能会得出一些与目标无直接关系的事实，从而造成计算空间和时间的浪费。

1. 逆向推理

逆向推理也称为目标驱动方式推理，它是从目标出发，反向使用规则，求得已知事实。 其基本推理过程如下：

第1步，用规则库中的规则后件与目标事实进行匹配，得到匹配的规则集合。

第2步，使用冲突解决算法，从匹配规则集合中选择一条规则作为启用规则。

第3步，将启用规则的前件作为子目标。

第4步，重复这个过程，直至各子目标均为已知事实为止。

如果目标明确，使用反向推理方式的效率是比较高的，所以较为常用。

1. 双向推理

双向推理是一种既自顶向下又自底向上的推理。推理从两个方向同时进行，直至某个 中间界面上双方向结果相符便成功结束。不难想象，这种双向推理较正向或反向推理形成 的推理网络小，从而推理效率更高。

**2.3.5**产生式系统的特点

**1.**主要优点

1. 自然性。产生式表示法用“如果……则……”的形式表示知识，符合人类的思维习 惯，是人们常用的一种表达因果关系的知识表示形式，既直观自然，又便于推理。
2. 模块性。产生式规则之间没有相互的直接作用，它们之间只能通过综合数据库发 生间接联系，而不能相互调用，这种模块化结构使在规则库中的每条规则可自由增删和 修改。
3. 清晰性。规则库中的每条规则都具有统一的IF-THEN结构，这种统一结构便于 对产生式规则的检索和推理，易于设计调试，可以高效存储信息。

(4)有效性。产生式知识表示法既可以表示确定性知识，又可以表示不确定性知识； 既有利于表示启发性知识，又有利于表示过程性知识。

2.主要缺点

1. 效率较低。各规则之间的联系必须以综合数据库为媒介，并且其求解过程是一种 反复进行的“匹配-冲突消解-执行”过程，这样的执行方式将导致执行的低效率。执行产生 式系统最费时的是模式匹配，匹配的时间与产生式规则数目及数据库中元素数目的乘积成 正比，当产生式规则数目很大时，匹配时间可能超过人们的忍耐程度，同样导致效率低下。
2. 不便于表示结构性知识。由于产生式表示中的知识具有一致格式，且规则之间不 能相互调用，因此那种具有结构关系或层次关系的知识很难以自然的方式来表示。
3. 难以扩展。尽管规则形式上相互独立，但实际问题中往往彼此是相关的。这样当 规则库不断扩大时，要保证新的规则和已有规则没有矛盾就会越来越困难，规则库的一致性 越来越难以实现。
4. 控制的饱和问题。在产生式系统中存在竞争问题，实际上很难设计一个能适合各 种情况下竞争消除的策略。

因此，对于大型规则库，如要求较高的推理效率，就不宜采用单纯的产生式系统知识 模式。

**2.4** 语义网络表示法

1968年，奎廉(J. R. Quilian)在研究人类联想记忆时提出了一种心理学模型——语义网 络，认为记忆是由概念间的联系实现的。随后奎廉又把它用作知识表示。1972年，西蒙 (Simon)在他的自然语言理解系统中也采用了语义网络表示法。1975年，亨德里克(G.G. Hendrix)又对全称量词的表示提出了语义网络分区技术。

**2.4.1**语义网络基本概念

语义网络是一种用语义和语义关系来表示知识且带有方向的网络图。图中包含节点和 弧，其中，节点代表语义，即各种概念、事物、属性、状态、动作等；弧代表语义关系，表示两个 语义之间的某种联系，弧是有方向的，用来体现节点间的主次关系。为了区分不同语义以及 语义关系，每个节点和弧都必须带有标识。

例2.4用语义网络表示事实“小燕子是一种鸟气

分析：事实描述中“小燕子''和“鸟”被定义为不同语义，可用两个节点进行表示，它们之 间的语义关系应为“是一种”，则可将事实表示成如图2. 4所示。

形如图2. 4所示的结构称为语义基元，所谓语义基元是指语义网络中最基本的语义单 元，可用三元组来表示，即

(节点A,弧，节点B)

如图2. 5所示，A和B分别代表节点，R表示A和B之间的某种语义联系，该有向图称 为基本网元。所谓基本网元是指一个语义基元对应的有向图。

图2. 4 “小燕子是一种鸟”的语义网络

图2. 5基本网兀

当把多个基本网元用相应的语义联系关联在一起时,

就可得到一个语义网络，如图2. 6所示。

在语义网络中，每个节点还可以是一个语义子网络, 所以，语义网络实质上可以是一种多层次的嵌套结构。

语义网络结构

谓词逻辑表示法也可以用语义网络来表示，因为三元 组（节点1,弧，节点2）可写成P（个体1,个体2）,其中个 体1、个体2分别对应节点1、节点2,而弧及其上标注的 节点之间的关系由谓词P来体现。

例如，对“小明与小旺是同学”可以表示为三元组（小

明，同学，小旺），对应的语义网络如图2. 7所示。

如果用一阶谓词表示法表示，则可写成P（小明，小旺），谓词*P*表示小明和小旺为同学 关系。

产生式表示法也可以用语义网络来表示。Rab表示A与*B*之间的语义关系，即“如 果……那么……"，如图2.8所示。

图2.7谓词逻辑表示法的语义网络示例 图2.8产生式表示法的语义网络

**2. 4.2**语义网络中常用的语义联系

在语义网络知识表示中，通过语义关系可以将更复杂的事物关联在一起，这也就要 求我们熟悉一些基本的语义关系，如实例关系、分类关系、属性关系、聚类关系等，利用这 些语义关系可以更好地描述事物以及事物之间的关系。下面介绍一些常用的基本语义 关系。

1. 类属关系

类属关系是指具有共同属性的不同事物间的分类关系、成员关系或实例关系。它体现 的是具体与抽象、个体与集体的层次分类。其直观含义是“是一个，，“是一种”“是一只，，“是一 名”……具体层节点位于抽象层节点的下层。类属关系最主要的特征是属性的继承性，处在 具体层的节点可以继承抽象层节点的所有属性。常用的类属关系如下：

（1） ISA,艮卩IS-A,意思是“是一个”，常用来描述一个事物是另一个事物的一个实例。

如“王一是一个人"就描绘了这样的关系，如图2.9所示。

（2） AKO,即A-Kind-Of,表示一个事物是另一个事物的一种类型。如“小燕子是一种 鸟”就描绘了这样的关系，如图2. 10所示。

王一

**ISA**

小燕子

**AKO**

图2. 9 ISA的语义网络示例

图2. 10 AKO的语义网络示例

(3) AMO,即A-Member-Of,表示一个事物是另一个事物的一个成员。如“李二是中共 党员”就描绘了这样的关系，如图2. 11所示。

1. 包含关系

包含关系也称为聚类关系，是指具有组织或结构特征的“部分与整体”之间的关系。它 和类属关系最主要的区别是包含关系一般不具备属性的继承性。常用的包含关系是Part- of,含义为“是一部分”，表示一个事物是另一个事物的一部分。它连接的下层节点的属性可 能和上层节点的属性是很不相同的，即不具有继承性。

如“脑是人体的一部分”这一事实即可说明脑不一定具备身体的其他属性，如图2. 12 所示。

李二**~~［~~**~~am°t~~中共党员

图2.11 AMO的语义网络示例

Part-of

脑 ― 人体

图2. 12 Part-of的语义网络示例

**3** .位置关系

位置关系是指不同事物在位置方面的关系，节点间的属性不具有继承性。常用的位置 关系有以下几种：

1. Located-on,含义为“在……上”，表示一个物体在另一个物体之上。
2. Located-at,含义为“在”，表示某一物体处在某一位置。
3. Located-under,含义为“在……下”，表示某一物体在另一物体之下。
4. Located-inside,含义为“在 内”，表示某一物体在另一物体之内。

(5) Located-outside,含义为“在 外”，表示某一物体在另一物体之外。

例如“桌子上有本书",其语义网络表示如图2. 13所示。

**4.**时间关系

时间关系是指不同事件在其发生时间方面的先后次序关系，节点间的属性不具有继承 性。常用的时间关系有以下几种：

1. Before,含义为“在……前”，表示一个事件在另一个事件之前发生。
2. After,含义为“在……后”，表示一个事件在另一个事件之后发生。
3. During,含义为“在……期间”，表示某一事件或动作在某个时间段内发生。

例如“明朝在清朝前面”，其语义网络表示如图2. 14所示。

Located-on

桌子

明朝

Before

清朝

图2. 13 Located-on的语义网络示例

图2. 14 Before的语义网络示例

1. 属性关系

属性关系表示事物与其行为、能力、状态等属性之间的关系。常用的属性关系有以下 几种：

（1） Have,含义为“有”，表示一个节点拥有另一个节点所表示的事物。

（2） Can,含义为“能”，表示一个节点能做另一个节点的事情。

（3） Age,含义为“年龄”，表示一个节点是另一个节点在年龄方面的属性。

例如“人有手",其语义网络表示如图2. 15所示。

1. 相近关系

相近关系是指不同事物在形状、内容等方面相似或接近。常用的相近关系有以下两种：

（1） Similar-to,含义为“相似”，表示某一事物与另一事物相似。

（2） Near-to,含义为“接近”，表示某一事物与另一事物接近。

如“猫似虎",其语义网络表示如图2. 16所示。

: Have 二 Similar-to -

人 手 猫 虎

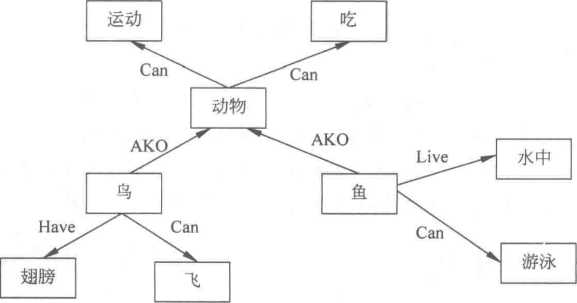
图2. 15 Have的语义网络示例 图2. 16 Similar-to的语义网络示例

上面只列出了一些常用的语义联系，其实，客观世界中，事物间的联系是各种各样、千变 万化的，在使用语义网络进行知识表示时，可根据需要随时对事物间的各种联系进行人为 定义。

**2. 4.3**语义网络表示知识的方法

1. 事实性知识的表示

事实性知识是指有关领域内的概念、事实、事物的属性、状态及其关系的描述。例如： ①动物能运动，会吃；②鸟是一种动物，有翅膀，会飞；③鱼是一种动物，生活在水中，会游 泳。以上事物和概念的表示如图2. 17所示。



1. 情况和动作的表示

1）情况的表示

在用语义网络表示那些不及物动词表示的语句或没有间接宾语的及物动词表示的语句 时，如果该语句的动词表示了一些其他情况，如动作作用的时间等，则需设立一个情况节点， 并从该节点向外引岀一组弧，用于指出各种不同的情况。例如，“一只名叫'飞飞'的小燕子 从春天到秋天占有一个巢”，其语义网络如图2. 18所示。



在图2. 18所示的语义网络中，设立了一个“占有”节点，通过由该节点向外引出的弧表 示了小燕子“飞飞”的占有物和占有时间。之所以要设立这个节点，是由于所要表示的知识不 仅指出了表示小燕子“飞飞”占有了一个巢，而且指出了它占有这个巢的时间是从春天到秋天。

另外，含有情况节点的语义网络也适于表示那些具有因果关系的知识。

2）动作和事件的表示

有些表示知识的语句涉及的动词既有主语又有直接宾语和间接宾语。也就是说，既有 发出动作的主体，又有接受动作的客体和动作所作用的客体。在用语义网表示这样的知识 时，既可以把动作设立成一个节点，也可以将所发生的动作当成一个事件，设立一个事件节 点。动作或事件节点也有一些向外引出的弧，用于指出动作的主体与客体，或指出事件的发 生动作以及该事件的主体与客体。例如，“李二给王山一块巧克力”，通过两种方式给出知识 表示的语义网络如下。

其一，增加事件节点，把“李二给王山一块巧克力”看成一个事件，如图2. 19所示。



其二，增加动作节点来表示知识的语义网络，如图2. 20所示。



图2. 20带有动作节点的语义网络

1. 逻辑关系的表示

1）合取与析取的表示

合取与析取通常通过增加合取节点和析取节点来实现，例如，用语义网表示如下事实: “听课的有硕士、博士，有男有女气

首先需要分析听课者的不同情况，可得到以下4种情况：

A.硕士、女生 B.硕士、男生 C.博士、女生 D.博士、男生

然后按照它们的逻辑关系用语义网络表示，如图2.21所示。

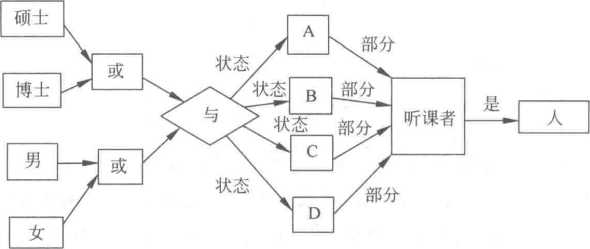


图2.21具有合取、析取关系的语义网络



是一个 是一种 是一种!

主体r~H 客体 I

图2. 22具有一个全称变量的语义网络

2）存在量词和全称量词的表示

存在量词可直接用ISA.AK0等语义关系来表示。全称量词可采用亨德里克提出的网 络分区技术，其基本思想是：把一个复杂命题划分为若干个子命题，每个子命题用一个较简 单的语义网络表示，称为一个子空间，多个子空间构成一个大空间。每个子空间看做是大空 间中的一个节点，称作超节点。空间可逐层嵌套，子空间之间用弧互相连接。例如“每个学 生都学习了一门程序设计语言课，，，用语义网络表示如图2. 22所示。

学生 学习 I

是一个

其中，G,是一个概念节点，它表示具有全称量化的一般事件。g是一个实例节点，代表 G中的一个具体例子，如上面所提到的事实。5是一个全称变量，表示任意一个学生。/是 一个存在变量，表示某一次学习。”是一个存在变量，表示某一门程序设计语言。这样，s、 之间的语义联系就构成一个子空间，它表示对每一个学生s，都存在一个学习事件，和 一门程序设计语言

在从节点g引出的3条弧中，弧“是一个”说明节点g是G,中一个实例；弧F说明它所 代表的子空间及其具体形式；弧V说明它所代表的全称量词，每一个全称量词都需要一条 这样的孤，子空间有多少个全称量词，就需要有多少条这样的弧。

另外，在网络分区技术中，要求F指向的子空间中的所有非全称变量节点都应该是全 称变量节点的函数，否则应放在子空间的外面。例如“每个学生都学习了 C语言”，其语义 网络如图2.23所示。

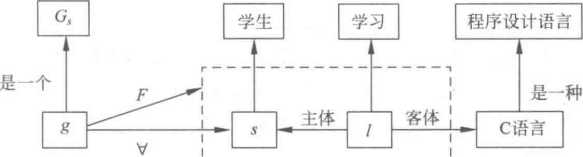


图2.23非全称变量节点不为全称变量函数的语义网络

在该图中，节点“C语言”代表一门具体的程序设计语言，是节点“程序设计语言”的一个 实例，而不是全称变量的函数，故被放到F所指的子空间的外边。

3）否定的表示

否定可分为基本语义关系的否定和一般语义关系的否定。基本语义关系的否定通过 在有向弧上直接标注该否定来解决，一般语义关系的否定通常通过引进“非”节点来 表不O

例如，“书不在桌子上”的表示如图2. 24所示。

“李二没给王山一块巧克力”的表示如图2. 25所示。

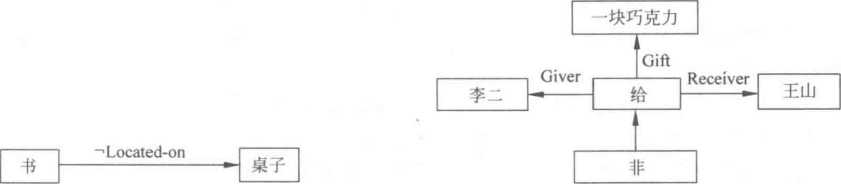


图2.24基本语义关系的否定表示 图2.25 一般语义关系的否定表示

4）蕴含的表示

在蕴含关系中，通过增加蕴含关系节点来实现知识表示，且有两条指向节点的弧，分别 表示前提条件和结论，将其分别标记为ANTE和CONSEo

例如，“如果学校组织研究生学术辩论赛，那么李二就参加”，其语义网络如图2. 26 所示。



图2.26带有蕴含关系的语义网络

**2.4.4**语义网络的推理过程

语义网络表示的问题求解系统主要由两部分组成，一是由语义网络构成的知识库，其中 存放了许多已知事实的语义网络；二是求解问题的解释程序，即推理机。语义网络采用的 推理方法一般有两种：一种是匹配，另一种是继承。

1. 匹配推理方法

匹配是指在知识库的语义网络中寻找与待求解问题相符的语义网络模式，待求解问题 是通过设立空的节点或弧来实现的。其推理过程如下：

（1） 根据待求问题的要求构造一个网络片段或者局部语义网络，这里包含着一些空的 节点或弧，即待求解的问题。

（2） 根据该局部网络到知识库中寻找所需要的信息。

（3） 当局部网络与知识库中的某个语义网络匹配时，则与未知处相匹配的事实就是问 题的解。

1. 继承推理方法

继承指将抽象事物的属性传递给具体事物，通常具有类属关系的事物之间具有继承性。 继承一般包括值继承和方法继承两种。值继承又称为属性继承，它通常是沿着ISA.AKO 等语义关系链继承。方法继承又称为过程继承，强调的是属性值不是直接继承的，而是通过 计算才能得到的，但它的计算方法是从上一层节点继承下来的，故而称之为方法继承或过程 继承。继承的一般过程如下：

（1） 建立一个节点表，用来存放待求解节点和所有以ISA.AK。等继承弧与此节点相 连的那些节点。初始情况下，表中只有待求解节点。

（2） 检查表中的第一个节点是否是有继承弧。如果有，就把该弧所指的所有节点放入 节点表的末尾，记录这些节点的所有属性，并从节点表中删除第一个节点。如果没有继承 弧，仅从节点表中删除第一个节点。

（3） 重复（2）,直到节点表为空。此时，记录下来的所有属性都是待求解节点继承来的 属性。

**2.4.5**语义网络表示的特点

语义网络表示具有以下5个特点：

1. 结构性。语义网络是一种结构化的知识表示方法，易于被询问和学习，它能将事物 属性以及事物间的各种语义联系显式地表示出来，下层节点可以继承、新增和变异上层节点 的属性，从而实现信息的共享。
2. 自然性。语义网络是一个带有标识的有向图，提供了自然的架构，可直观地把事物 的属性及事物间的语义联系表示出来，便于理解，易于转换，符合人们表达事物间关系的习 惯，自然语言与语义网络之间的转换也比较容易实现。
3. 联想性。语义网络本来是作为人类联想记忆模型提出来的，它着重强调事物间的 语义联系，体现了人类的联想思维过程。
4. 非严格性。语义网络没有公认的形式表示体系，它没有给其节点和弧赋予确切的 含义。推理过程中有时不能区分物体的“类”和“个体”的特点，因此通过推理网络而实现的 推理不能保证其正确性。
5. 复杂性。语义网络表示知识的手段是多种多样的，这虽然给其表示带来了灵活性, 但同时也由于表示形式的不一致使处理变得复杂。

**2.5**框架表示法

框架表示法是以框架理论为基础的一种结构化知识表示方法。这种表示方法可以表达 结构性的知识，能够把知识的内部结构关系以及知识间的联系表示出来，能够体现知识间的 继承属性，符合人们观察事物时的思维方式。

框架理论是明斯基于1975年作为理解视觉、自然语言对话及其他复杂行为的一种基础 提出来的。他认为，人们对现实世界中各种事物的认识都是以一种类似于框架的结构存储 在记忆中的，当遇到一个新事物时，就从记忆中找出一个合适的框架，并根据新的情况对其 细节加以修改、补充，从而形成对这个新事物的认识。

**2.5.1**框架基本结构

在框架理论中，框架是知识的基本单位，把一组有关的框架连接起来便可形成一个框架 体系。在框架系统中，每一个框架都有自己的名字，称为框架名。框架(frame)通常由若干 个槽(slot)组成，每个槽用来表示事物的各个方面，其根据实际需要拥有若干个侧面 (aspect),每一个侧面也可以拥有若干个值(value) o框架的基本结构如下：

Frame V框架名〉

调名1>

<侧面11 >

〈值 111>,…〈值 *11K1 >*

V侧面In, >

〈值 1 口口, …值 1B1 K1 >

〈槽名2>

〈侧面21 >

〈值 211,值 *21LX >*

<侧面2 口 2 >

〈值 2n2l>z •••<值 *2n2LD2>*

其中，某些槽值可省略，一般来说，槽值有如下几种类型：

1. 具体值value。该值按实际情况给定。
2. 默认值default.该值按一般情况给定，对于某个实际事物，具体值可以不同于默 认值。
3. 过程值procedureo该值是一个计算过程，它利用该框架的其他槽值，按给定计算 过程(或公式)进行计算得出具体值。
4. 另一框架名。当槽值是另一框架名时，就构成了框架调用，这样就形成了一个框架 链。有关框架聚集起来就组成框架系统。
5. 空值。该值等待填入。

框架的槽还可以是附加过程，称为过程附件(procedural attachment),包括子程序和某 种推理过程，这种过程也可以侧面的形式表示。附加过程根据其启动方式可分为两类。一 类是自动触发的过程，称为精灵(demon) o这类过程一直监视着系统的状态，一旦满足条件 就自动开始执行。另一类是受到调用时触发的过程，称为服务者(servant),用于完成特定 的动作和计算。

例如，给岀一个直接描述大学教师的框架。

Frame < COLLEGE TEACHER >

Name： Unit (Last name, First name)

Sex： Area(Man, Woman)

Default: Man

Age: Unit(Years)

Degree: Area(Bachelor, Master, Doctor)

Major: Unit(Major)

Paper: Area(SCI, EIZ Core)

Level: Area (A, B, C, D)

Address : <T-Address>

Telephone: HomeUnit(Number)

MobileUnit(Number)

这个框架的名字是COLLEGE TEACHER,共含有9个槽，槽名分别为Name. Sex. Age、Degree、Major、Paper、Level、Address、Telephone0 每个槽名后面的就是槽值，如 Man、Woman‘Bachelor、Master、Doctor。Area(范围)用来说明槽值仅能从后面所给内容进 行选择。Area与Default (默认)是侧面名，其后面是侧面值。尖括号“<〉”用来表示框架 名,T-Address表示教师住址框架的框架名。

下面考虑将教师和大学教师联系起来，框架表示如下。

先建立教师框架：

Frame < TEACHER >

Name: Unit (Last namez First name)

Sex： Area(Man, Woman)

Default: Man

Age: Unit(Years)

Level: Area (A, B, C, D)

Address : <T-Address >

Telephone: HomeUnit(Number)

MobileUnit(Number)

Paper: Area(SCI, EIZ Core)

再建立大学教师框架：

Frame < COLLEGE TEACHER >

AKO: TEACHER

Degree: Area(Bachelor, Master, Doctor)

Major： Unit(Major)

在该框架系统中，用了一个槽名AKO将大学教师与教师联系在一起。AKO是公用的 标准槽值名之一，称为框架中的预定义槽名。这样就建立了两个框架之间的一种层次关系， 通常称前者为父框架，后者为子框架。子框架可以继承父框架的属性。这样就可以减少框 架结构大小，而不会丢失信息。

常用的框架继承技术有defaultjf-neededjf-addedo default为相应的槽提供默认值， 如上面的“性别”，当系统没有给出值时，默认值为“男”；if-needed当槽为空且需要信息时执 行；if-added当新的信息加入槽时执行。

**2.5.2**基于框架的推理

框架表示下的知识推理方法与语义网络表示下的知识推理方法类似，即遵循匹配和继 承的原则。与语义网类似，框架表示的问题求解系统由两部分构成，一是由框架及其相互关 联构成的知识库，二是用于求解问题的解释程序，即推理机。前者的作用是提供求解问题所 需要的知识；后者则是针对用户提出的具体问题，运用知识库中的相关知识，通过推理对问 题进行求解。

求解问题的匹配推理步骤如下：

1. 把待求解问题用框架表示出来，其中有的槽是空的，表示待求解的问题，称为未 知处。
2. 与知识库中已有的框架进行匹配。这种匹配通过对相应槽的槽名及槽值逐个进行 比较来实现。
3. 使用一种评价方法对预选框架进行评价，以便决定是否接受它。
4. 若可接受，则与问题框架的未知处相匹配的事实就是问题的解。

由于框架间存在继承关系，一个框架所描述的某些属性及值可能是从它的上层框架继 承而来的，因此两个框架的比较往往牵涉到它们的上层、上上层框架，从而增加了匹配的复 杂性。

**2.5.3**框架表示法的特点

框架表示法有以下优点：

1. 结构性。框架表示法最突出的特点是善于表示结构性知识，它能够把知识的内部 结构关系以及知识间的特殊联系表示出来。
2. 深层性。框架表示法不仅可以从多个方面、多重属性表示知识，而且还可以通过 ISA.AK0等槽以嵌套结构分层地对知识进行表示，因此能用来表达事物间复杂的深层 联系。
3. 继承性。在框架系统中，下层框架可以继承上层框架的槽值，也可以进行补充和修 改，这样既减少知识冗余，又较好地保证了知识的一致性。
4. 自然性。框架能把与某个实体或实体集的相关特性都集中在一起，从而高度模拟 人脑对实体多方面、多层次的存储结构，直观自然，易于理解。

框架表示法有以下缺点：

1. 缺乏框架的形式理论。至今还没有建立框架的形式理论，其推理和一致性检查机 制并非基于良好定义的语义。
2. 缺乏过程性知识表示。框架系统不便于表示过程性知识，缺乏如何使用框架中的 知识的描述能力。框架推理过程需要用到一些与领域无关的推理规则，而这些规则在框架 系统中又很难表达。
3. 清晰性难以保证。由于各框架本身的数据结构不一定相同，从而使框架系统的清 晰性很难保证。

知识表示的方法另外还有过程表示法、脚本表示法、面向对象表示法等，不再一一赘述。

**2.6** 习题

1. 什么是知识？它有哪些特性？有哪几种分类方法？
2. 什么是知识表示？有哪几种常用的知识表示方法？
3. 一阶谓词逻辑表示法适合表示哪种类型的知识？它有什么特点？
4. 请写出一阶谓词逻辑表示法表示知识的步骤。
5. 设有下列语句，请用相应的谓词公式把它们表示出来：
6. 有的人喜欢打羽毛球，有的人喜欢打网球，有的人既喜欢打羽毛球又喜欢打网球。
7. 他每天下午都去打网球。
8. 并不是每个人都喜欢看电视。
9. 要想岀国留学，必须通过外语考试。
10. 产生式的基本形式是什么？它与谓词逻辑中的蕴含式有哪些相同和不同之处？
11. 什么是产生式系统？它由哪几部分组成？其求解问题的一般步骤是什么？
12. 什么是语义网络？语义网络表示法的特点是什么？
13. 请把下列命题用一个语义网络表示出来：
14. 树和草都是植物。

（2） 树和草都是有根有叶的。

（3） 海藻是草，且长在水中。

（4） 苹果树是果树，且结苹果。

（5） 果树是树，且会结果。

1. 什么是框架？框架的表示形式是什么？
2. 框架表示法有什么特点？请叙述框架表示法表示知识的步骤。
3. 试写岀“学生"框架的描述。

搜索策略

**3.1** 搜索的基本概念

**3.1.1**搜索的含义

如何在大量的知识甚至结构不良或非结构化的问题中获取对自己有用的信息，是人工 智能中非常重要的一部分。对于这些问题，一般很难获得其全部信息，更没有现成的算法可 供使用。因此，根据问题的实际情况，不断寻找可利用知识，从而构造一条代价最小的推理 路线，就显得尤为重要。搜索就是要寻找一个操作序列，使问题从初始状态转换到目标状 态。这个操作序列就是目标的解。因此，所谓搜索，就是根据问题的实际情况，按照一定的 策略或规则，从知识库中寻找可利用的知识，从而构造一条使问题获得解决的推理路线的过 程。搜索包含两层含义：一是要找到从初始事实到问题最终答案的一条推理路线，二是找 到的这条路线是时间和空间复杂度最小的求解路线。

通常搜索策略的主要任务是确定选取规则的方式。可根据是否使用启发式信息分为盲 目搜索和启发式搜索，也可以根据问题的表示方法分为状态空间搜索和与/或树搜索。

盲目搜索是不考虑给定问题所具有的特定知识，系统根据事先确定好的某种固定排序, 依次调用规则或随机调用规则，一般统称为无信息引导的搜索策略。由于搜索总是按照预 定的控制策略进行搜索，因此这种搜索策略具有盲目性，效率不高，不便于复杂问题的求解。 启发式搜索考虑问题领域可应用的知识，动态地确定规则的排序，优先调用较合适的规则， 加速问题的求解过程，使搜索朝着最有希望的方向前进，找到最优解。

状态空间搜索是指用状态空间法来求解问题所进行的搜索。与/或树搜索是指用问题 归约法来求解问题时进行的搜索。下面分别介绍状态空间法与问题归约法。

**3.1.2**状态空间法

在分析了人工智能研究的求解方法之后，就会发现许多问题求解方法采用了试探搜索 方法。也就是说，这些方法是通过在某个可能的解空间内寻找一个解来求解问题。这种基 于解空间的问题表示和求解方法就是状态空间法。状态空间搜索的研究焦点在于设计高效 的搜索算法，以降低搜索代价并解决组合爆炸问题。

1. 状态空间及其搜索的表示

在状态空间表示法中，问题是用“状态”和“操作”来表示的，问题求解的过程使用状态空 间来表示。

1） 状态

状态（state）是表示问题求解过程中每一步问题状况的数据结构，可以用如下形式 表示：

Sk = {Sko , Ski，…}

在这种表示方式中，当对每一个分量都给予确定的值时，就得到了一个具体的状态。其 中，每一个分量称为状态变量。实际上，任何一种类型的数据结构都可以用来描述状态，只 要它有利于问题求解，就可以选用。

2） 操作

操作（operation）也称为算符，它是把问题从一种状态变换为另一种状态的手段。当对 一个问题状态使用某种操作时，它将引起该状态中某些分量值的变化,从而使问题从一个具 体状态变为另一个具体状态。操作符可以是过程、规划、数学算子、运算符号或逻辑符号等。 操作可理解为状态集合上的一个函数，它描述状态之间的关系。

3） 状态空间

状态空间（state space）用来描述一个问题的全部状态及这些状态之间的相互关系。状 态空间常用一个三元组（S,F,G）来表示。其中：

S为问题的所有初始状态的集合，其中的每个元素表示一种状态。

F为操作的集合，用于把一个状态转换为另一个状态。

G是S的一个非空子集，为目标状态的集合。它可以是若干具体的状态，也可以是对某 些状态性质的描述。

状态空间也可以用一个赋值的有向图来表示，该有向图称为状态空间图。在状态空间 图中，节点表示问题的状态，有向边（弧）表示操作。

1. 状态空间问题的例子

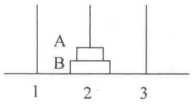
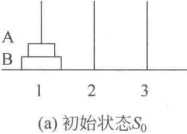
下面通过具体例子来说明状态空间法。

例**3.1**二阶梵塔问题。设有3根柱子，它们的编号分别为1号、2号、3号。在初始情 况下，1号柱子上穿有A和B两个圆盘，A比B小，A位于B的上面。要求把这两个圆盘全 部移到另一根柱子上，而且规定每次只能移动一个圆盘，任何时刻都不能使大圆盘位于小圆 盘的上面。

解：设用Sk = {Sk（）,Sq }表示问题的状态，其中Sm表示圆盘A所在的柱子号，Sy表示圆盘B所在的柱子号。全部可能的问题状态共有以下9种：

So = {1,1} S] = {1,2} S2= {1»3} S3 = {2,1} ，4 = {2,2}

S5 = (2,3} S6 = {3,1} S7 = {3,2} Sg = {3,3) 其中，初始状态S()和目标状态S4、S8如图3. 1所示。



(b)目标状态&

图3.1二阶梵塔问题的部分状态

1 2 3

(C)目标状态S8

问题的初始状态集合为Sk = {S。}，目标状态集合为G=(S4,S8}O操作分别用*AEj)* 和BQ")表示。其中表示把圆盘A从第*i*号柱子移到第j号柱子上，Bh,,)表示把 圆盘B从第，号柱子移到第，号柱子上。共有12种操作，分别是

A(l,2) A(l,3) A(2,l) A(2,3) A(3,l) A(3,2)

B(l,2) B(l,3) B(2,l) B(2,3) B(3,l) B(3,2)

根据上述9种可能的状态和12种操作，可构成二阶梵塔问题的状态空间图，如图3. 2 所示。

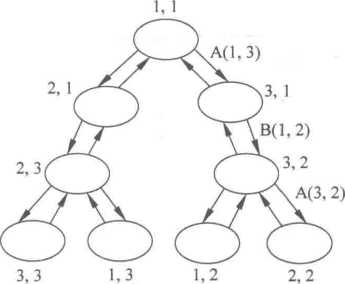


图3.2二阶梵塔的状态空间图

在图3. 2中，从初始节点(1,1)到目标节点(2,2)及(3,3)的任何一条路径都是问题的一 个解。其中，最短的路径长度是3,它由3个操作组成。例如，从初始状态(1,1)开始，通过 使用操作A(1,3)、B(1,3)及A(3,2),可到达目标状态(2,2)。

例3.2作为状态空间的经典例子，我们来观察“传教士与野人问题气 设N个传教士 带领N个野人划船渡河,且为安全起见，渡河需要遵循两个约束：①船上人数不得超过载 重限量，设为K个人；②为预防野人攻击，任何时刻(包括两岸、船上)野人数目不得超过传 教士数目。

为便于理解状态空间表示方法，可以将该问题简化为一个特例：N=3,K = 2；并以变 量m和c分别表示传教士和野人在左岸或者在船上的实际人数，变量*b*表示船是否在左岸 (值1表示船在左岸，否则为0)。从而上述约束条件转变成为

考虑到在这个渡河问题中，左岸的状态描述(m,cM)可以决定右岸的状态，所以整个问 题的状态就可以用左岸的状态来描述，以简化问题的表示。设初始状态下传教士、野人和船 都在左岸，目标状态下这三者均在右岸，问题状态以三元组表示，则问题求解任务 可以描述为

1. (0,0,0)

在此问题中，状态空间可能的状态总数为4X4X2 = 32,但由于要遵守安全约束，只有 20种状态是合法的。下面是几个不合法状态的例子：

(1,0,1),(1,2,1),(2,3,1)

鉴于存在不合法的状态，还会导致某些合法的状态不可达，例如状态(0,0,1)、(0,3,0)。 因此，此问题总共只有16种可达的合法状态：

So = (3,3,1)

S4 = (1,1,1)

*S8* = (3,2,0)

S12 = (1,1,0)

S】=(3,2,1)

S5 = (0,3,1)

*S9* = (3,1,0)

S】3 = (0,2,0)

S2 = (3,1,1)

S6 = (0,2,1)

SI0 = (3,0,0)

Sm = (0,1,0)

S3 = (2,2,1)

*S7* = (0,1,1)

*Sn* = (2,2,0)

& = (0,0,0)

有了这些状态，还需要考虑可进行的操作。在此问题中，操作是指用船把修道士或野人 从河的左岸运到右岸，或者从河的右岸运到左岸，并且每个操作都应该满足条件①、②。因 此，操作应该由两部分组成，即条件部分和动作部分。操作只有当其条件具备时才能进行, 动作则刻画了应用此操作所产生的结果。

此处用符号表示从左岸到右岸的运人操作，用符号Q表示从右岸到左岸的运人操 作.其中3表示船上的修道士数，/表示船上的野人数。通过分析可以知道有10种操作可 供选择，其操作集为

F = (F。］. Ro, Fj 1, P()2, P20，Qoi，Q10，Qi 1，Q02»Q20}

下面以Ri和Qm为例说明这些操作的条件和动作：

操作符号 条件

动作

*b =* 0»c = *c* — 1 厶=l,c = c+ l

Poi *b =* 1 = 0 或 3,c21

Q01 *b = 09m =* 0 或 3,c<2

从而可画出类似图3. 2所示的状态空间图。

例3.3猴子摘香蕉问题。在前面讨论谓词逻辑知识表示时曾经提到这一问题，现在 用状态空间法求解。

解：问题状态可用四元组来表示。其中，s表示猴子的水平位置；7表示箱 子的水平位置； > 表示猴子是否在箱子上，当猴子在箱子上时y取1,否则了取0; z表示猴 子是否拿到香蕉，当拿到香蕉时z取1,否则z取0。

所有可能的状态为

So：(如们0,0) 初始状态

:(们们0,0)

S?: (c,c,O,O)

S3 ： (c, c, 1,0)

S4: ( c, c, 1,1) 允许的操作为

目标状态

Goto(u):猴子走到位置 即(w,x,0,0)->(u,x,0,0)o

Pushbox(v):猴子推着箱子到水平位置即(z,;r,0,0)f (如如0,0)。

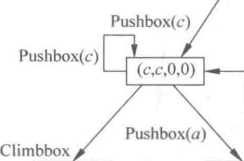
Climbbox：猴子爬上箱子，即(z,z,0,0)f(:r,z,l,0)。

Grasp：猴子拿到香蕉，即(c,c,l,O)f(c,c,l,l)。

这个问题的状态空间图如图3. 3所示。由初始状态变为目标状态的操作序列为

{Goto(6),Pushbox(c),Climbbox,Grasp}

I (姑0,0) I初始状态



*Goto(b)*

n\_\_, T ("0,0) I

I (ccLO) I

Goto(b)

(01,0) I

Pushbox(c)

Pushbox(cr)

I S，l) I

图3.3猴子摘香蕉问题的状态空间图

**3.1.3**问题归约法

问题归约法是另一种对问题进行描述及求解的办法。其基本思想是：对问题进行分解 和变换，将此问题最终变为一个子问题的集合，通过求解子问题达到求解原问题的目的。问 题归约法适用于当初始问题比较复杂时，分解后的子问题的解可以直接得到，从而解决了初 始问题的情况。

所谓“分解”是指：如果一个问题P可以归约为一组子问题…，R,并且只有当 所有子问题*P.*都有解时原问题P才有解，任何一个子问题*P.*无解都会导致原问题P无 解•则称此种归约为问题的分解，即分解所得到的子问题的“与”与原问题P等价。

所谓“变换”是指：如果一个问题P可以归约为一组子问题P「R,・・・，P”，并且子问题 P,中只要有一个有解，则原问题*P*就有解，只有当所有子问题*P,*都无解时原问题*P*才无解, 称此种归约为问题的等价变换，简称变换，即变换所得到的子问题的“或”与原问题P等价。

**1.**问题归约描述

问题归约表示由下面3个部分组成：

1. 一个初始问题的描述。
2. 一套把问题变换成为子问题的操作符。
3. 一套本原问题的描述。

从目标（要解决的问题）出发逆向推理，建立子问题以及子问题的子问题，直至最后把初 始问题归纳成为一个平凡的本原问题集合，这就是问题归约的实质。

例**3.4** 三阶梵塔问题。有3个柱子（1,2,3）和3个不同尺寸的圆盘（A,B,C）O在每 个圆盘的中心有一个孔，所以圆盘可以堆叠在柱子上。最初，全部3个圆盘都堆在柱子1 上，最大的圆盘C在底部，最小的圆盘A在顶部。要求把所有圆盘都移到柱子3上，每次只 许移动一个，而且只能先搬动柱子顶部的圆盘，还不许把尺寸较大的圆盘堆放在尺寸较小的 圆盘上。

若釆用状态空间法来求解这个问题，其状态空间有27个节点，每个节点代表柱子上的 圆盘的一种正确位置。当然，也可以用简单的问题归约法来求解此问题。

归约过程如下：

（1） 移动圆盘A和B至柱子2的双圆盘移动问题。

（2） 移动圆盘C至柱子3的单圆盘移动问题。

（3） 移动圆盘A和B至柱子3的双圆盘移动问题。

可以看出简化后的问题每一个都比原问题容易，所以原问题都可以变成易解决的本原 问题。而将一个复杂的原问题归约成一系列本原问题的过程可以很方便地用与/或树来表 示。下面介绍有关与/或树的相关内容。

1. 与/或树的相关概念

1） 节点与弧线

父节点：是一个初始问题或是可分解为子问题的问题节点。

子节点：是一个初始问题或是子问题分解的子问题节点。

或节点：只要解决某个问题就可解决其父问题的节点集合。

与节点：只有解决所有子问题，才能解决其父问题的节点集合。

端节点：没有子节点的节点。

终止节点：本原问题所对应的节点。由此可见，终止节点一定是端节点，而端节点却不 一定是终止节点。

弧线：是父辈节点指向子节点的圆弧连线。

2） 或树、与/或树和解树

把一个原问题变换成若干个子问题可用一个或树来表示，如图3.4所示。

把一个问题分解为若干个子问题可用一个与树来表示，如图3.5所示。



如果一个问题既需要通过分解，又需要通过变换才能得到其本原问题，其归约过程可以 用一个与/或树来表示，如图3.6所示。

由可解节点构成，并且由这些可解节点可以推出初始节点(它对应着原问题)为可解 节点的子树为解树。在解树中一定包含初始节点。例如，在图3.7所给的与/或树中，用 粗线表示的子树是一个解树。在该图中，节点P为原始问题节点，用，标出的节点是终止 节点。

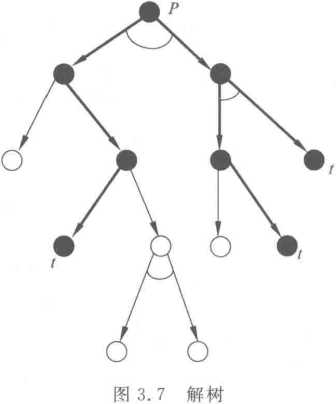
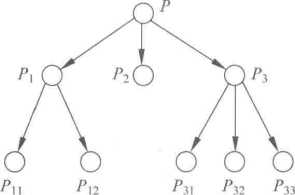


图3. 6与/或树

1. 问题归约的例子

如例3.4的三阶梵塔问题，此问题也可以用状态空间法来解，不过本例主要用它来说明 如何用问题归约法来解决问题。

首先，定义该问题的形式化表示方法。设用三元组(/"J)表示问题在任意时刻的状 态，用f表示状态的转换。在此三元组中3代表圆盘C所在的柱子号，丿代表圆盘B所在的 柱子号怯代表圆盘A所在的柱子号。

前面分解的3个子问题可分别表示如下。

1. 移动圆盘A和B至柱子2的双圆盘问题：
2. — (1,2,2)
3. 移动圆盘C至柱子3的单圆盘问题：
4. f (3,2,2)
5. 移动圆盘A和B至柱子3的双圆盘问题：
6. -> (3,3,3)

其中，子问题(1).(3)都是一个二阶梵塔问题，它们都可以再继续进行分解；子问题(2) 是本原问题，它已经不需要再分解。

三阶梵塔问题的分解过程可用图3. 8所示的与/或树来表示。在该与/或树中，有7个 终止节点，它们分别对应着7个本原问题。得到问题的解为

1. — (1,1,3) (1,1,3) (1,2,3) (1,2,3) — (1,2,2)
2. -> (3,2,2) (3,2,2) — (3,2,1) (3,2,1) (3,3,1)

(3,3,1) — (3,3,3)

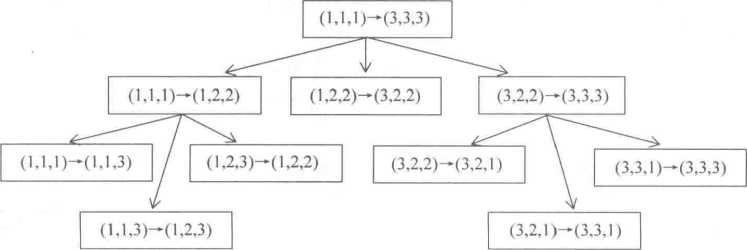


图3. 8三阶梵塔问题的与或树表示

**3.2**状态空间搜索

状态空间的搜索策略分为盲目搜索和启发式搜索两大类。下面讨论的广度优先搜索、 深度优先搜索和有界深度优先搜索都属于盲目搜索策略。

盲目搜索策略的一个共同特点是它们的搜索路线是已经决定好的，没有利用被求解问 题的任何特征信息，在决定要被扩展的节点时，并没有考虑该节点到底是否可能出现在解的 路径上，也没有考虑它是否有利于问题的求解以及所求的解是否为最优解。

**3.2.1**盲目搜索

**1.** 一般图搜索

一般图搜索是在状态空间中搜索从初始状态到目标状态解答路径的过程。由于问题的 状态空间可以用一个有向图来表示，因此状态空间搜索实际上就是对有向图的搜索。从图 搜索的角度来看，状态空间搜索的基本思想可以概括为：将问题的初始状态作为当前扩展 节点对其进行扩展，生成一组子节点，然后检查目标状态是否出现在这些节点中。如果出 现,表明搜索成功，即找到了该问题的解；如果没有出现，则再按照某一种搜索策略从已生 成的子节点中选择一个节点作为当前的扩展节点。重复上述过程，直到目标状态出现在子 节点中或者没有可供扩展的节点为止。所谓对一个节点进行“扩展”是指对该节点用某个可 用操作施加作用，生成该节点的一组子节点。

在开始搜索过程之前，先定义两个数据结构OPEN表与CLOSED表。OPEN表用于 存放刚生成的节点，对于不同的搜索策略,节点在OPEN表中的排列顺序是不同的。例如， 对广度优先搜索，节点按生成的顺序排列，先生成的节点排在前面，后生成的节点排在后面。 CLOSED表用于存放将要扩展或已扩展的节点。

搜索步骤如下：

第1步，把初始节点S放入OPEN表，并建立目前只包含S的图，记为G。

第2步，检查OPEN表是否为空，若为空则问题无解，失败退出。

第3步，把OPEN表的第一个节点取岀放入CLOSED表，并记该节点为节点

第4步，判断节点〃是否为目标节点。若是，则求得问题的解，成功退出。

第5步，考查节点宀生成一组子节点。把其中不是节点〃先辈的那些子节点记作集合 M,并把这些子节点作为节点*n*的子节点加入G中。

第6步，针对*M*中子节点的不同情况，分别进行如下处理：

•对那些未曾在G中岀现过的M成员设置一个指向父节点（即节点〃）的指针，并将它 们放入OPEN表。

•对那些先前已在G中出现过的M成员，确定是否需要修改它指向父节点的指针。

•对那些先前已经在G中出现并且已经扩展了的M成员，确定是否需要修改其后继 节点指向父节点的指针。

第7步，按某种搜索策略对OPEN表中的节点进行排序。

第8步，转第2步。

**2.**广度优先搜索

广度优先搜索又称为宽度优先搜索，是一种先生成的节点先扩展的简单策略。从初始 节点So开始逐层向下扩展，只有当同一层的节点全部被搜索完以后，才能进入下一层继续 搜索。

在搜索的过程中，要建立两个数据结构：OPEN表和CLOSED表，其形式分别如表3.1 和表3. *2*所示。

表 3.1 OPEN 表 表 3. 2 CLOSED 表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 节 | 点 | 父节点编号 | 编 | 号 | 节 | 点 | 父节点编号 |
|  | |  | | |  | |  |

OPEN表用于存放刚生成的节点，对于不同的策略，节点在此表中的排列顺序是不同 的。CLOSED表用于存放将要扩展或者已扩展的节点（节点*n*的子节点）。

所谓对一个节点进行扩展，是指用合适的算符对该节点进行操作，生成一组子节点。一 个节点经一个算符操作后一般只生成一个子节点，但对一个可使用的节点可能有多个，故此 时会生成一组子节点。需要注意的是，在这些子节点中，可能有些是当前扩展节点（节点〃） 的父节点或者祖父节点等，此时不能把这些先辈节点作为当前扩展节点的子节点。

在广度优先捜索策略中，OPEN表中的节点是按进入的先后排序，先进入OPEN表的 节点排在前，后进入的节点排在后。

因此，广度优先搜索的基本思想是：从初始节点So开始，逐层地对节点进行扩展并考 察它是否为目标节点，在第〃层的节点没有全部扩展并考察之前，不对第〃 + 1层的节点进 行扩展。其搜索过程如下：

第1步，把初始节点S。放在OPEN表中。

第2步，若OPEN表为空，则问题无解，退出。

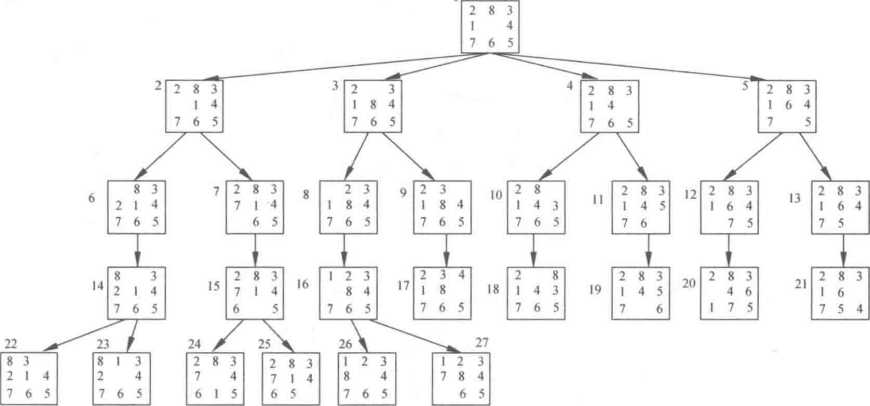
第3步，把OPEN表中的第一个节点（记为节点〃）取出放入CLOSED表中。

第4步，考察节点〃是否为目标节点。若是，则得到问题的解，成功退出。

第5步，若节点*71*不可扩展，则转第2步。

第6步，扩展节点宀将其子节点放入OPEN表的尾部，并且为每一个子节点设置指向 父节点的指针，然后转第2步。

例3.5重排九宫问题。在3X3的方格棋盘上， 分别放置了标有数字1〜8的8张牌，初始状态为So, 目标状态为Sg,如图3.9所示。可用的操作有空格左 移、空格上移、空格右移、空格下移。即只允许把位于 空格左、上、右、下的牌移入空格。要求用广度优先搜 索策略寻找初始状态到目标状态的路径。

解：应用广度优先策略，可以在第四级得到解，搜 索树如图3. 10所示。可以看出，解的路径是

S。

(a)初始状态

Sg

(b)目标状态

图3. 9重排九宫问题

So ~3 -\*• 8 —► 16 —► 26

图3. 10重排九宫的广度优先搜索

由于广度优先搜索总是在生成扩展完〃层的节点后才转到n + 1层，所以总能找到最优 解。但是实用意义不大，广度优先算法的主要缺点是盲目性较大，尤其是当目标节点距初始 节点较远时，将产生许多无用节点，最后导致组合爆炸，尽管耗尽资源，在可利用的空间中也 找不到解。

**3.**深度优先搜索

深度优先搜索总是先扩展后生成的节点。其基本思想是：从初始节点So开始，在其子 节点中选择一个最新生成的节点进行考察，如果该子节点不是目标节点并且可以扩展，则扩 展此子节点，再在此节点的子节点中选择一个最新生成的节点进行扩展，一直如此向下搜 索。当到达某一子节点，此子节点既不是目标节点又不能继续扩展时，才选择其兄弟节点进 行考察。其搜索过程如下：

第1步,初始节点放入OPEN表中。

第2步，若OPEN表为空，则问题无解，退出。

第3步，把OPEN表中的第一个节点(记为〃)取出放入CLOSED表中。

第4步，考察节点*n*是否为目标节点，若是，则问题解求出，退出。

第5步，若节点不可扩展，则转第2步。

第6步，扩展节点〃，将其子节点放入OPEN表的首部，并为其配置指向父节点的指针, 然后转第2步。

对比广度优先搜索与深度优先搜索，可以看岀二者唯一的区别是，广度优先搜索时将节 点*n*的子节点放到OPEN表的尾部，而深度优先搜索时把节点72的子节点放到OPEN表的 首部。这一不同点使得搜索的路线完全不同。

例3.6对例3. 5的重排九宫问题进行深度优先搜索。

解：用深度优先搜索可得到图3. 11所示的搜索树。但这只是搜索树的一部分，尚未到 达目标节点，仍可继续往下搜索。

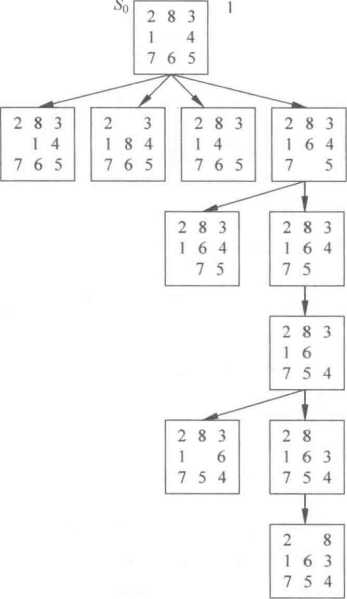


图3.11重排九宫的深度优先捜索

从深度优先搜索的算法可以看出，搜索一旦进入某个分支，就将沿着这个分支一直向下 进行下去，如果目标恰好在这一个分支上，则可以很快找到解。但是，如果目标不在此分支 上，且该分支是一个无穷分支，则搜索过程就不可能找到解。因此，深度优先搜索是一种不 完备策略，即使问题有解也不一定能够找到。此外，即使能够找到解•此解也不一定是最短 路径的解。

广度优先搜索和深度优先搜索各有不足，为了弥补各自的不足，可以釆用有界深度优先 算法。顾名思义，这是对深度优先算法给出深度限制刁m，当搜索深度达到了廿宀即使没有 找到目标，也要停止该分支的搜索，换到另一个分支进行搜索。折中的办法是广度优先和深 度优先策略的一种结合。

1. 代价树搜索

当路径的花费与弧有关时，我们常常想到的是花费最小的路径。例如，对于一个投递机器人，花费可能是两个位置之间的距离，需要解出距离最短的那条解路径。花费也可能是机 器人按照弧实施动作所需要的各种资源。而在前面讨论的各种搜索策略中，并没有将注意 力放在各边的代价上，因为默认各边的代价是相同的，且都为一个单位量。但是，对于许多 实际问题，状态空间的各个边的代价不可能完全相同。为此，需要在搜索树中给每一条边都 标上代价。这种标有代价的树称为代价树。

1）代价树的代价表示

g（〃2）*= g（7h）+ C（—，”2）*

其中，冷与如分别表示某一父节点与其子节点，g（〃）表示从初始节点S。到节点〃的代价， 用C（小，电）表示从父节点m到其子节点如的代价。

在代价树中，最小代价的路径和最短路径（即路径长度最短）是有可能不同的。代价树 搜索的目的是为了找到最优解，即找到一条代价最小的解路径。

2）代价树的广度优先搜索

代价树的广度优先搜索每次从OPEN表中选择节点以及往CLOSED表中存放节点 时，总是选择代价最小的节点。即OPEN表中节点的顺序是按照其代价由小到大排列的， 代价小的节点排在前面，代价大的节点排在后面，与节点在树中的位置无关。

代价树的广度优先搜索算法如下：

（1） 将初始节点S。放入OPEN表中，置&的代价*g（S0） = 0o*

（2） 如果OPEN表为空•则问题无解，失败退出。

（3） 把OPEN表的第一个节点取出放入CLOSED表，并记该节点为丄

（4） 考察节点〃是否为目标节点，若是，则找到了问题的解，成功退出；否则继续。

（5） 若节点〃不可扩展，则转步骤（2）；否则转步骤（6）。

（6） 扩展节点〃，生成其子节点儿（£=1,2,3,…），将这些节点放入OPEN表中，并为每 一个子节点设置指向父节点的指针。按公式g（q） = g（〃）+c（",〃,）3=l,2,3,・・・）计算各 节点的代价，并根据各节点的代价对OPEN表中的全部节点按照从小到大的顺序重新进行 排序。

（7） 转步骤（2）。

代价树的广度优先搜索策略是完备的。如果问题有解，上述算法一定能找到它，并且找 到的一定是最优解。

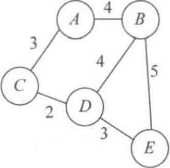
例3.7城市交通问题。设有5个城市，城市之间的交通线路 如图3. 12所示，图中的数字表示两个城市之间的交通费用，即代 价。用代价树的广度优先搜索，求出从A市出发到E市费用最小 的交通路线。

图3. 12城市交通线路图

解：图3. 12是一个网络图，不能直接用于搜索算法，需要将其 先转换为代价树。

把一个网络图转换为代价树的方法是：从起始节点A开始，把 与它直接邻接的节点作为其子节点。对其他的节点也做同样的处理。但当一个节点已作为 某个节点的直系先辈节点时，就不能再作为这个节点的子节点。例如，图中与节点*B*直接 相邻的节点有节点A、D、E,但由于A已经作为B的父节点在代价树中出现过了，因此A不 能再作为*B*的子节点。此外，图中的节点除初始节点A夕卜，其他节点都可能在代价树中出现多次，为区别它们的多次出现，分别用下标1,2,…标出，但实际上是同一个节点。

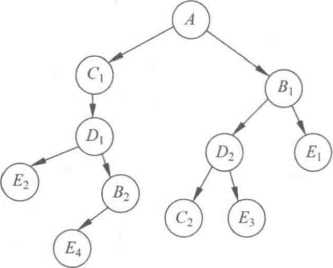
转换后的代价树如图3.13所示。

图3. 13城市交通线路图的代价树

对如图3. 13所示的代价树，按广度优先搜索可 得到最优解：

A —► *C\ -\*• D] -E[*

其代价为8。可见，从A市到E市的最小费用路线为:

A —C—E

3）代价树的深度优先搜索

代价树的广度优先搜索每次都是从OPEN表的 全体节点中选择一个代价最小的节点，而深度优先搜 索是从刚扩展的子节点中选择一个代价最小的节点。 即两种搜索策略的区别在于每次选择最小代价节点 的方法不同。

代价树的深度优先搜索算法如下：

（1） 将初始节点So放入OPEN表中，置S。的代价g（So）=O。

（2） 如果OPEN表为空，则问题无解，失败退出。

（3） 把OPEN表的第一个节点取出放入CLOSED表，并记该节点为

（4） 考察节点〃是否为目标节点，若是，则找到了问题的解，成功退出。

（5） 若节点不可扩展，则转第（2）步，否则转第（6）步。

（6） 扩展节点〃，生成其子节点孔（1=1,2,3,…），将这些子节点按边代价由小到大放入 OPEN表中的首部，并为每一个子节点设置指向父节点的指针。

（7） 转第（2）步。

对例3. 7所给出的问题，用代价树的深度优先搜索策略找到的路径为

*A* —G -\*■ *D\ —► Ei*

和广度优先搜索相比，深度优先搜索找到的路径是相同的。这只是一种巧合。一般来 说,它找到的解不一定是最优解,即代价树的深度优先搜索策略是不完备的，甚至当搜索进 入无穷分支时，算法将找不到解。

**3.2.2**状态空间的启发式搜索

3.2.1节介绍了状态空间的盲目搜索策略。例如状态空间的深度优先搜索和宽度优先 搜索。这类方法进行的是一种蛮力搜索，因而效率低。本节介绍状态空间的启发式搜索策 略，由于此种方法有较强的针对性，因此可以缩小搜索范围，提高搜索效率。

**1** .估价函数与启发式信息

3.2,1节所介绍的所有搜索算法都是无指导信息的，并没有考虑目标节点在哪里。它 们没有使用任何指引它们该去哪里的信息，除非它们无意中发现了目标。

在搜索过程中，用于决定要扩展的下一个节点的信息，即用于指导搜索过程且与具体问 题求解有关的控制信息称为启发信息；决定下一步要控制的节点称作“最有希望”的节点, 其“希望”的程度通常通过构造一个函数来表示，这种函数被称为估价函数。

1） 估价函数

估价函数的任务是估计待搜索节点的重要程度，给它们排定顺序。在这里，把估价函数 定义为从初始节点So经过节点〃到达目标节点的最小代价路径的代价估计值，它的 一般形式为

/（n） = g（〃）+/i（〃）

其中，g（〃）为初始节点So到节点*n*已实际付出的代价；九3）是从节点*n*到目标节点Sg最 优路径的估计代价，搜索的启发式信息主要由力（〃）来体现，故把九（〃）称为启发函数。对 g3）的值，可以按指向父节点的指针，从节点〃反向跟踪到初始节点S。，得到一条从初始节 点So到节点*71*的最小代价路径，然后把这条路径上的所有有向边的代价相加，就得到*g（n）* 的值。

2） 启发信息

启发信息是指与具体问题求解过程有关的，并可指导搜索过程朝着最有希望的方向前 进的控制信息。一般有以下3种：

（1） 有效地帮助确定扩展节点的信息。

（2） 有效地帮助决定哪些后继节点应被生成的信息。

（3） 能决定在扩展节点时哪些节点应从搜索树上删除的信息。

一般来说，搜索过程所使用的启发性信息的启发能力越强，扩展的无用节点就越少。

例3.8八数码难题。设问题的初始状态S。和目标状态Sg如图3. 9所示，且估价函 数为= 式中以3）表示节点〃在搜索树中的深度；W（〃）表示节点〃中

“不在位”的数码个数，请计算初始状态So的估价函数值/（So）0

解：在本例的估价函数中，取g（〃）=Q（〃），/i（〃）=W（〃）。此处用So到*n*的路径上的 单位代价表示实际代价，用〃中“不在位”的数码个数作为启发信息。一般来说，某节点中的 “不在位”的数码个数越多，说明它离目标节点越远。

对初始节点So,由于a（So）=o,w（s（））=3,因此有

/（So）=O + 3 = 3

这个例子仅是为了说明估价函数的含义及估价函数值的计算。在问题搜索过程中，除 了需要计算初始节点的估价函数之外，更多的是要计算新生成节点的估价函数值。

**2. A**算法

在搜索的每一步都利用估价函数，3）=g3） + /i3）对OPEN表中的节点进行排序， 则该搜索算法称为A算法。由于估价函数中带有问题自身的启发性信息，因此，A算法也 称为启发式搜索算法。

根据搜索过程中选择扩展节点的范围，启发式搜索算法可分为全局择优搜索算法和局 部择优搜索算法。其中，全局择优搜索算法每当需要扩展节点时，总是从OPEN表的所有 节点中选择一个估价函数值最小的节点进行扩展。局部择优搜索算法每当需要扩展节点 时，总是从刚生成的子节点中选择一个估价函数最小的节点进行扩展。下面主要讨论全局 择优搜索算法。

'全局择优搜索算法的搜索过程可描述如下：

（1） 把初始节点 So 放入 OPEN 表中 J（So）=g（So）+/i（S。）。

（2） 如果OPEN表为空，则问题无解，失败退出。

1. 把OPEN表的第一个节点取出放入CLOSED表，并标记该节点为
2. 考察节点〃是否为目标节点，若是，则找到了问题的解，成功退出。
3. 若节点不可扩展，则转第(2)步。
4. 扩展节点〃，生成其子节点仏(7= 1,2,3,…)，计算每一个子节点的估价值/(«,)(£=!, 2,3,…)，并为每个子节点设置指向父节点的指针，然后将这些子节点放入OPEN表中。
5. 根据各节点的估价函数值，对OPEN表中的全部节点按从小到大的顺序重新进行 排序。
6. 转第(2)步。

由于上述算法的第(7)步要对OPEN表中的全部节点按估价函数值从小到大重新进行 排序，这样在算法第(3)步取出的节点就一定是OPEN表的所有节点中估价函数值最小的 一个节点。因此，它是一种全局择优的搜索方式。

对上述算法进一步分析还可以发现：如果取估价函数f(〃)=g(〃)，则它将退化为代价 树的广度优先搜索；如果取估价函数则它将退化为广度优先搜索。可见，广 度优先搜索和代价树的广度优先搜索是全局择优搜索的两个特例。

例**3.9** 八数码难题。设问题的初始状态So和目标状态Sg如图3. 9所示，估价函数 与例3. 8相同，请用全局择优搜索解决该问题。

解：这个问题的全局择优搜索树如图3. 14所示，在图3. 14中，每个节点旁边的数字是 该节点的估价函数值。例如，对节点烏，其估价函数值的计算为

/(Sz) = d(S?) + W (S2) = 2 + 2 = 4

从图3. 14还可以看出，该问题的解为

*Sq* —\*■ S] —► S? -S3 —\*■ Si



So

| 2 8 3 | 4 | *2* 3 | *S\* | 2 8 3 | 5 | 2 8 3 |
| --- | --- | --- | --- | --- | --- | --- |
| 】4 |  | 1 8 4 |  | 1 4 |  | 1 6 4 |
| 7 6 5 |  | 7 6 5 |  | 7 6 5 |  | 7 5 |

4

2 8 3

1 4

7 6 5

| 8 3 | 6 | 2 8 3 | 4 | 2 3 | 6 | 2 3 |
| --- | --- | --- | --- | --- | --- | --- |
| 2 1 4 |  | 7 1 4 |  | 1 8 4 |  | 1 8 4 |
| 7 6 5 |  | 6 5 |  | 7 6 5 |  | 7 6 5 |

4

1 2 3

8 4

7 6 5

I 2 3 8 4

7 6 5

图3. 14全局择优搜索树

**3.** *A'*算法

1） A\*算法概述

A\*算法也是一种启发式搜索方法，它是对扩展节点的选择方法做了一些限制，选用了 一个比较特殊的估价函数，这时的估价函数/3）=义（〃）+力（〃）是对函数*r* 3） = g\* 3） + /T 3）的一种估计或近似，即，/（〃）是对广（〃）的估计，g（〃）是对g\* 3）的估计，人（7?）是对 人\* （〃）的估计。

函数*厂3*的定义是这样的：它表示从节点So到节点/?的一条最佳路径的实际代价 加上从节点〃到目标节点Sg的一条最佳路径的代价之和。而g\*3）就是从节点S。到节点 〃之间最小代价路径的实际代价，疽3）则是从节点〃到目标节点S&的最小代价路径上的 代价。既然g3）是对g\*（〃）的估计，所以g（Q是比较容易求得的，它就是从初始节点So 到节点〃的路径代价，这可通过由节点〃到节点S。回溯时把所遇各段弧线的代价加起来而 得到，显然恒有g（〃）2g\* 3）。九3）是对*疽3）*的估计，它依赖于有关问题领域的启发信 息，是上述提到的启发函数，其具体形式要根据问题特性来进行构造。在A\*算法中要求启 发函数人3）是*h\* 3）*的下界，即对所有的*n*均有*h（n）V扩3）。*这一要求十分重要，它能 保证A\*算法找到最优解。理论分析表明，若问题存在最优解，则此限制就可能保证找到最 优解。虽然，这个限制可能产生无用搜索，但是不难想象，当某一节点〃的九（〃）＞疽3）,则 该节点就有可能失去优先扩展的机会，因而导致得不到最优解。

2） A\*算法性质

*A\**算法具有可釆纳性、单调性和信息性。

（1）可采纳性。

所谓可采纳性是指对于可求解的状态空间图（即从状态空间图的初始节点到目标节点 有路径存在）来说，如果一个搜索算法能在有限步内终止，并且能找到最优解，则称该算法是 可釆纳的。分3步证明如下：

1. 对于有限图，A\*算法一定会在有限步内终止。

对于有限图，其节点个数是有限的。可见A\*算法在经过若干次循环之后只可能出现 两种情况：或者由于搜索到了目标节点而终止；或者由于OPEN表中的节点被取完而终 止。不管发生哪种情况，A\*算法都在有限步内终止。

1. 对于无限图，只要初始节点到目标节点有路径存在，则A\*算法也必然会终止。

该证明分两步进行。证明在A\*算法结束之前，OPEN表中总存在节点z'。该节点是 最优路径上的一个节点，且满足

设最优路径是& ,而口2，・・・，石,书；。由于A\*算法中的人伝）满足 （z）,所以

/（s0）,/（^）,/（x2）,-,/（^）均不大于 *mcs；*）=广（S。）。

又因为A\*算法是全局择优的，所以在它结束之前，OPEN表中一定含有S。，円，戏，…， 中的一些节点。设是最前面的一个，则它满足

心）。\*.（S）

至此，第一步证明结束。

现在来进行第二步的证明。这一步用反证法，即假设*A\**算法不终止，则会得出与上一 步矛盾的结论，从而说明A\*算法一定会终止。

假设A\*算法不终止，并设。是图中各条边的最小代价（乃）是从S。到节点乃的最 短路径长度，则显然有

g\* O”）2 Q\* （了“）X *e*

又因为

g（z”）2 g\* 6）

所以有

g（z”）2 *d\** （x„） X *e*

因为

*hCx„）* 2。' *f（1n）*2 g6）

故得到

*f（Kn）* 2 *d\* （工n）* X *e*

由于A\*算法不终止，随着搜索的进行〃\*（］“）会无限增长，从而使 EC也无限增长。 这就与上一步证明得出的结论矛盾。因为对可解状态空间来说（So）一定是有限值。

所以，只要从初始节点到目标节点有路径存在，即使对于无限图，A，算法也一定会终止。

1. A\*算法一定终止在最优路径上。

假设A\*算法不是在最优路径上终止，而是在某个目标节点*t*处终止，即A,算法未能 找到一条最优路径，则

四）=g（Q>r\*（S。）

但由②的证明可知，在A\*算法结束之前，OPEN表中存在节点工‘，它在最优路径上，且 满足

心）W广（S。）

此时，A\*算法一定会选择7’来扩展而不会选择f,这就与假设矛盾。显然，A\*算法一 定终止在最优路径上。

根据可纳性的定义及以上证明可知A\*算法是可采纳的。同时由上面的证明还可知， A\*算法选择扩展的任何一个节点/都满足如下性质：

心）9\* （SQ

（2）单调性。

在A\*算法中，若对启发性函数加以适当的单调性条件限制，就可使它对所扩展的一系 列节点的估价函数单调递增（或非递减），从而减少对OPEN表或CLOSED表的检查和调 整，提高搜索效率。

所谓单调性限制是指满足如下两个条件：

1. /i（Sg）=O。
2. 设可是节点心的任一子节点，则有

*h（jCi*） — *h^Xj*） *< cCxt 9Xj*）

其中，Sg是目标节点,C（X, 是节点*JCt*到其子节点Zj的边代价。

若把上述不等式改写为如下形式：

*hXxi） W h^Xj*） 4- c（x, ）

就可看出节点右到目标节点最优费用的估价不会超过从s到其子节点丐的边代价加上 从勺到目标节点最优费用的估价。

可以证明，当A\*算法的启发式函数/i（z）满足单调限制时，有如下两个结论：•

1. 若A,算法选择节点*右*进行扩展，则

g（z”）= g\*（x„）

1. 由A•算法所扩展的节点序列的估价值是非递减的。

这两个结论都是在从”满足单调限制时才成立的。否则，它们不一定成立。例如，对 于第②个结论，当人（了）不满足单调限制时，有可能某个要扩展的节点比以前扩展的节点的 估价值小。

（3）信息性。

A，算法的搜索效率主要取决于启发函数 E,在满足*•质）*的前提下，方3）的 值越大越好。人3）的值越大，表明它携带的与求解问题相关的启发信息越多，搜索过程就 会在启发信息指导下朝着目标节点逼近，少走弯路，提高搜索效率。

设£（］）与是对同一问题的两个估价函数：

*f\* （x） = gi （x） + *h\* （x）

*工）*=g2（z） + *h2（JT）*

&.与A；分别是以方Gr）与人伝）为估价函数的A•算法，且设对多有非目标节点］均有

*hi （jc） ＜ h2* （x）

在此情况下，我们将证明Aj扩展的节点数不会比扩展的节点数少，即扩展的 节点集是扩展的节点集的子集。可用归纳法证明如下。

设K表示搜索的深度。当K = 0时，结论显然成立。因为若初始状态就是目标状态， 则与人：都无须扩展任何节点。若初始状态不是目标状态，它们都要对初始节点进行 扩展，此时与扩展的节点数是相同的。

设当搜索树的深度为K— 1时结论成立，即凡扩展了的前K —1代节点，也都 扩展了。此时，只要证明扩展的第K代的任一节点孙也被Aj扩展就可以了。

由假设可知，AU扩展的前K—1代节点A：也都扩展了。因此在A『搜索树中有一条 从初始节点S。到乃的路径，其费用不会比搜索树中从S。到乃的费用更大，即

g】3）v g2（m

假设不扩展节点孔，就表示能找到另一个具有更小估价值的节点进行扩展并 找到最优解。此时有

即

g】3）+/tS）2尸（So）

对上述不等式应用如下关系式

gl（3）＜ g2（Ht）

得到

*h\*（X\*）2 广（So）— *g2（m*

这与我们最初的假设"（工）＜方2&）矛盾。

由此可得出“Aj扩展的节点数不会比扩展的节点数少”这一结论是正确的，即启 发函数所携带的启发性信息越多，搜索时扩展的节点数越少，搜索效率越高。

3) A\*算法应用举例

例3.10修道士和野人问题(简称MC问题)。设在河的左岸有三个野人、三个修道士 和一条船，修道士想用这条船把所有的野人运到对岸，但是受以下条件的约束：

一是修道士和野人都会划船，但每次船上至多可载两个人。

二是在河的任一岸，如果野人数目超过修道士数目，修道士就会被野人吃掉。

如果野人会服从任何一次过河安排，请规划一个确保修道士和野人都能过河，且没有修 道士被野人吃掉的安全过河计划。用A•算法解决该问题。

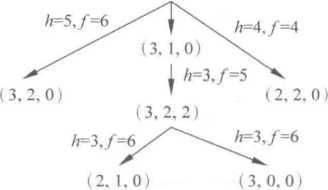
解：用m表示左岸的修道士人数“表示左岸的野人数M表示左岸的船数，用 表示问题的状态。

对于A•算法，首先需要确定估价函数。设*g(n)=d(n) Ji(n)=m+c~2b,*则有

*f(n)* =g(〃)+/i(〃)= + 〃 一 *2b*

式中混(Q为节点的深度。通过分析可知0(〃)£矿3),满足A,算法的限制条件。

MC问题的搜索图如图3.15所示。在该图中，每个节点旁边还标出了该节点的*h*值和/•值。



(3, 3, 1 )

宀=/\^*八2,户6*

(3,3,1)

*h=2J=7*

(0, 2,0)

*时 f=7 |*

(2, 2, I)

(L 1,0)

(0. 3, 1)

*h=lj=8* I

(0. 1,0)

*h=0J=8*

(0, 2, 1)

(0. 0. 0 )

图3. 15修道士和野人问题的搜索图

**3.3**博弈树的启发式搜索

**3.3.1** 概述

博弈是一类富有智能行为的竞争活动.如下棋、打牌、战争等。博弈可以分为双人完备 信息博弈和机遇性博弈。所谓双人完备信息博弈，就是两位选手对垒，轮流走步，每一方不

第3章搜索策略 仅知道对方已经走过的棋步，而且还可以估计出对方未来的走步。对弈的结果是一方赢，另 一方输.或者是双方和局。这类博弈的实例有象棋、围棋等。所谓机遇性博弈，是指存在不 可预测性的博弈，如掷币等。对机遇性博弈，由于不具备完备信息，因此本节不讨论。本节 主要讨论双人完备信息博弈问题。

在双人完备信息博弈的过程中，双方都希望自己能获胜。因此，当任何一方走步时，都 是选择对自己最为有利，而对另一方最为不利的行动方案。假设博弈的一方为MAX,另一 方为MIN。在博弈过程中的每一步，可供MAX和MIN选择的行动方案都可能有很多种。 从MAX方的观点来看，可供自己选择的那些行动方案之间是“或”的关系，原因是主动权掌 握在MAX手里，选择哪个方案完全是由自己决定的；而那些可供对方选择的行动方案之 间是“与”的关系，原因是主动权掌握在MIN的手里，任何一个方案都可能被MIN选中， MAX必须防止那种对自己最为不利的情况的发生。

若把双人完备信息博弈过程用图表示出来，就可以得到一棵与/或树，这种与/或树称为 博弈树。在博弈树中，那些下一步该MAX走步的节点称为MAX节点，而下一步该MIN 走步的节点称为MIN节点。博弈树具有如下特点：

（1） 博弈的初始状态是初始节点。

（2） 博弈树中的“或”节点和“与”节点逐层交替出现。

（3） 整个博弈过程始终站在某一方的立场上。所有能使自己一方胜利的终局都是本原 问题，相应的节点是可解节点；所有使对方获胜的终局都是不可解节点。例如，站在MAX 方，所有能使MAX方获胜的节点都是可解节点，所有能使MIN方获胜的节点都是不可解 节点。

**3.3.2**极大极小过程

对简单的博弈问题，可以生成整个博弈树，找到必胜的策略。但对于复杂的博弈，如国 际象棋,大约有IO"。个节点，可见要生成整个搜索树是不可能的。一种可行的方法是使用 当前正在考察的节点生成一棵部分博弈树，由于该博弈树的叶节点一般不是哪一方的获胜 节点，因此，需要利用估价函数八〃）对叶节点进行静态估值。一般来说，那些对MAX有利 的节点，其估价函数取正值；那些对MIN有利的节点，其估价函数取负值；那些使双方均 等的节点，其估价函数取接近于0的值。

为了计算非叶节点的值，必须从叶节点向上倒推。对于MAX节点，由于MAX方总是 选择估值最大的走步，因此,MAX节点的倒推值应该取其后继节点估值的最大值。对于 M1N节点，由于MIN方总是选择使估值最小的走步，因此MIN节点的倒推值应取其后继 节点估值的最小值。这样一步一步地计算倒推值，直至求出初始节点的倒推值为止。由于 我们是站在MAX立场上，因此应该选择具有最大倒推值的走步。这一过程称为极大极小 过程。

下面给出一个极大极小过程的例子。

例3.11 一字棋游戏。设有一个3行3列的棋盘，如图3. 16所示，两个棋手轮流走 步,每个棋手走步时往空格上摆一个自己的棋子，谁先使自己的棋子成三子一线为赢。设 MAX方的棋子用X标记，MIN方的棋子用。标记，并规定MIN方先走步。

解：为了对叶节点进行静态估计，规定估价函数e(P)如下：

•若F是MAX的必胜局，则e(P) = + oc；

•若P是MIN的必胜局，则e(F) = —8；

•若P对MIN,MAX都是胜负未定局，则

*e(P)* = e(+P) -e(-P)

式中,e( + P)表示棋局P上有可能使X成三子一线的数目,e(-P)表示棋局P上有可能使

O成三子一线的数目。例如，对于如图3. 17所示的棋局有估价函数值

e(F) = 6 — 4 = 2

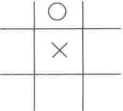
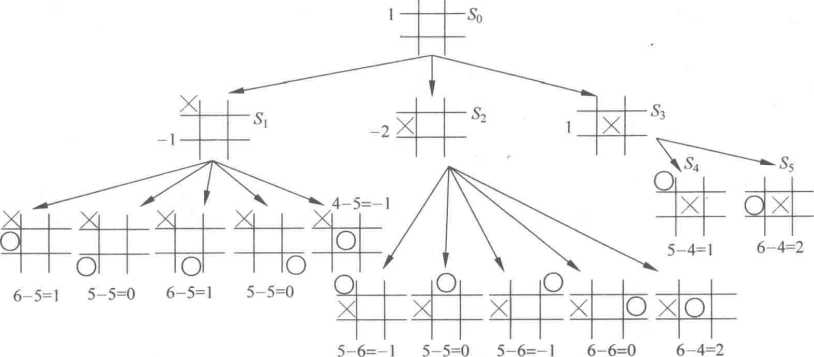


图3.16 —字棋棋盘 图3. 17棋局1

在搜索过程中，具有对称性的棋局认为是同一棋局。例如，如图3. 18所示的棋局可以 认为是同一个棋局，这样能大大减少搜索空间。图3. 19给出了第一着走棋后生成的博弈 树。图中叶节点下面的数字是该节点的静态估值,非叶节点旁边的数字是计算岀的倒推值。 从图中可以看出，对MAX来说S3是一着最好的走棋，它具有较大的倒推值。

, — -kF



% 1

图3.18对称棋局的例子

**3. 3. 3** *a-p* 剪枝

上述极大极小过程是先生成与/或树，然后再计算各节点的估值，这种生成节点和计 算估值相分离的搜索方式需要生成规定深度内的所有节点，因此搜索效率低。如果能在 生成节点的同时对节点进行估值，从而可以剪去一些没用的分枝，这种技术称为a-/?剪枝 过程。

*a~p*剪枝的方法如下：

（1） MAX节点的a值为当前子节点最大倒推值。

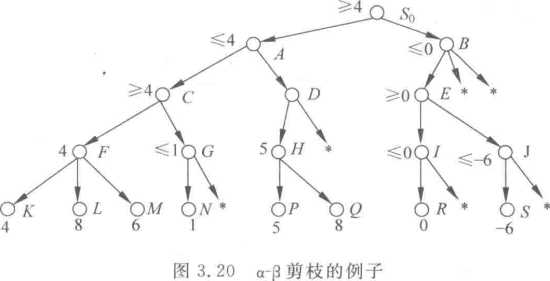
（2） MIN节点的值为当前子节点最小倒推值。

。不剪枝的规则如下：

（1） 任何MAX节点〃的。值大于或等于它先辈节点的£值，则n以下的分枝可停止搜 索，并令节点〃的倒推值为a。这种剪枝称为0剪枝。

（2） 任何MIN节点”的0值小于或等于它先辈节点的a值，则n以下的分枝可停止搜 索，并令节点〃的倒推值为傷 这种剪枝称为。剪枝。

下面来看一个。不剪枝的具体例子，如图3. 20所示。其中，最下面一层端节点下面的 数字是假设的估值。



在图3. 20中，由节点K、L、M的估值推出节点F的倒推值为4,即F的夕值为4,由此 可推出节点C的倒推值（24）。记C的倒推值的下界为4,不可能再比4小，故C的。值为 4O由节点N的估值推出节点G的倒推值（Wl）,无论G的其他子节点的估值是多少,G的 倒推值都不可能比1大。事实上，随着子节点的增多，G的倒推值只可能是越来越小。因 此，1是G的倒推值的上界，所以G的值为1。另外，已经知道C的倒推值（24） *,G*的其他 子节点又不可能使C的倒推值增大。因此，对G的其他分枝不必再进行搜索，这就相当于 把这些分枝剪去。由F、G的倒推值可推出节点C的倒推值为4,再由C可推岀节点A的倒 推值（W4）,即A的值为4O另外，由节点P、Q推岀的节点*H*的倒推值为5,此时可推岀 *D*的倒推值（25）,即D的。值为5。此时，。的其他子节点的倒推值无论是多少都不能使 *D*及A的倒推值减少或者增加，所以D的其他分枝被剪去，并可确定A的倒推值为40用 同样的方法可推出其他分枝的剪枝情况，最终推出So的倒推值为4。

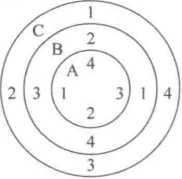
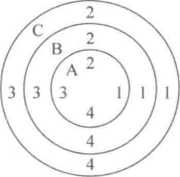
**3.4** 习题

1. 什么是搜索？有哪两大类不同的搜索方法？两者的区别是什么？
2. 什么是状态空间？用状态空间法表示问题时，什么是问题的解？什么是最优解？最 优解唯一吗？
3. 请写出状态空间图的一般搜索过程。在搜索过程中OPEN表和CLOSED表的作用 分别是什么？有何区别？
4. 什么是盲目搜索？主要有哪几种盲目搜索策略？
5. 什么是宽带优先搜索？什么是深度优先搜索？有何不同？
6. 什么是与树？什么是或树？什么是与/或树？什么是可解节点？什么是解树？
7. 有一个农夫带一只狼、一只羊和一筐青菜从河的左岸乘船到右岸，但受到下列条件 的限制：
8. 船太小，农夫每次只能带一样东西过河。
9. 如果没有农夫看管，则狼要吃羊，羊要吃菜。

请设计一个过河方案，使得羊和菜都能不受损失地过河，画出相应的状态空间图。

1. 圆盘问题.设有大小不等的3个圆盘A、B、C套在一根轴上，每个盘上都标有数字

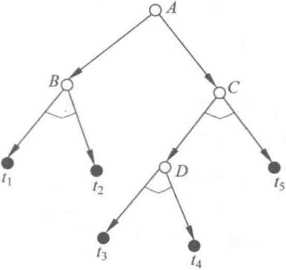
1、2、3、4,并且每个圆盘都可以独立的绕轴做逆时针转动，每次转动90°,其初始状态S。和目 标状态Sg如图3. 21所示，请用广度优先搜索和深度优先搜索求岀从So到6的路径。



(a)初始状态So (b)目标状态席

图3.21题8示意图

1. 设有如图3. 22所示的与/或树，请分别用与/或树的广度优先搜索和深度优先搜索 求出解树。



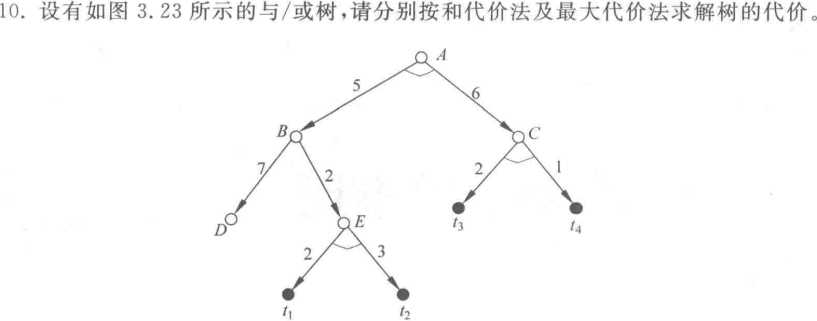


图3. 23题1。2K意图

11.设有如图3. 24所示的博弈树，其中最下面的数字是假设的估值。请对该博弈树做 如下工作：

（1） 计算各节点的倒推值。

（2） 利用剪枝技术剪去不必要的分枝。

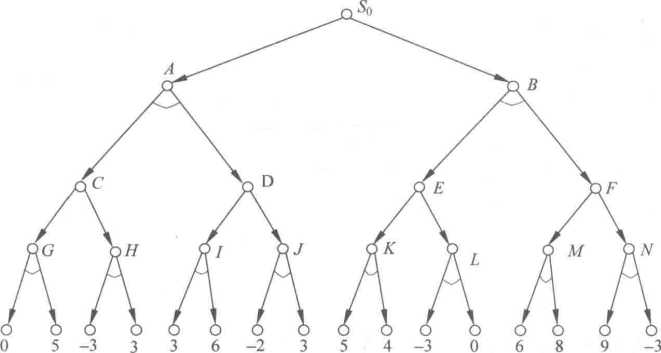


图 3.24

题11示意图

确定性推理

一个智能系统不仅应该拥有知识，而且还应该能够很好地利用这些知识，即利用知识推 理和求解问题。智能系统的推理过程实际上就是一种思维过程。按照推理过程中所用到的 知识的确定性，可分为确定性推理和不确定性推理。

若在推理中所用的知识都是精确的，即可以把知识表示成必然的因果关系，然后进 行推理，推理的结论或为真，或为假，这种推理就叫做确定性推理。反之，在人类知识中， 有相当一部分属于人们的主观判断，是不精确和含糊的，这些知识归纳总结出来的推理 规则往往也是不精确的，基于这种不精确的推理规则进行推理，形成的结论也是不确定 的，这种推理就叫做不确定性推理。本章重点讨论确定性推理，不确定性推理将在第5 章讨论。

**4.1**推理的基本概念

**4.1.1**什么是推理

在现实生活中，人们对各种事物进行分析、综合并最后做出决策时，通常是从已知的事 实出发，通过运用已掌握的知识，找出其中蕴含的事实或归纳岀新的知识，这一过程通常称 为推理。因此，从智能技术的角度来说，所谓推理就是按照某种策略由已知判断推出另一种 判断的思维过程。在人工智能系统中，推理通常是由一组程序来实现的，人们把这一组用来 控制计算机实现推理的程序称为推理机。例如，在医疗诊断专家系统中，知识库存储专家经 验及医学常识，数据库存放病人的症状、化验结果等初始事实，利用该专家系统为病人诊治 疾病实际上就是一次推理过程，即从病人的症状及化验结果等初始事实出发，利用知识库中 的知识及一定的控制策略，对病情做岀诊断，并开岀医疗处方。像这样从初始事实出发，不 断运用知识库中的已知知识逐步推出结论的过程就是推理。

**4.1.2**推理方法及其分类

推理方法主要解决在推理过程中前提与结论之间的逻辑关系，以及在不确性推理中不 确定性的传递问题。推理可以有很多种不同的分类方法，例如，可以按照推理的逻辑基础、 所用知识的确定性、推理过程的单调性以及是否使用启发性信息等角度来划分。

**1.**按推理的逻辑基础分类

按照推理的逻辑基础，常用的推理方法可分为演绎推理和归纳推理。

1） 演绎推理

演绎推理是从已知的一般性知识出发，推出蕴含在这些知识中的适合于某种个别情况 的结论。它是一种由一般到个别的推理方法，其核心是三段论。常用的三段论由一个大前 提、一个小前提和一个结论3个部分组成。其中，大前提是已知的一般性知识或推理过程得 到的判断；小前提是关于某种具体情况或某个具体实例的判断；结论是由大前提推出的， 并且适合于小前提的判断。

例如，有如下3个判断：

（1） 计算机系的学生都会编写程序。

（2） 程强是计算机系的一位学生。

（3） 程强会编写程序。

这是一个三段论理论。其中，（1）是大前提，（2）是小前提，（3）是经演绎推理出来的结 论。从这个例子可以看出，“程强会编写程序”这一结论是蕴含在“计算机系的所有学生都会 编写程序”这个大前提中的。因此，演绎推理就是从已知的大前提中推导出适合于小前提的 结论，即从已知的一般性知识中抽取所包含的特殊性知识。由此可见，只要大前提和小前提 是正确的，则由它们推出的结论也必然是正确的。

2） 归纳推理

归纳推理是从一类事物的大量特殊事例出发，推出该类事物的一般性结论。它是一种 由个别到一般的推理方法。归纳推理的基本思想是：先从已知事实中猜测出一个结论，然 后对这个结论的正确性加以证明确认。数学归纳法就是归纳推理的一种典型例子。对于归 纳推理，如果按照所选事例的广泛性可分为完全归纳推理和不完全归纳推理；如果按照推 理所使用的方法可分为枚举归纳推理和类比归纳推理等。

完全归纳推理是指在进行归纳时需要考察相应事物的全部对象，并根据这些对象是否 都具有某种属性来推出该类事物是否具有此种属性。例如，某公司选购一批计算机，如果对 每台计算机都进行了质量检测，并且都合格，则可以得出结论：这批计算机的质量是合 格的。

不完全归纳推理是指在进行归纳时只考察了相应事物的部分对象，就得出了关于该事 物的结论。例如，某公司选购一批计算机，如果只是随机抽查了其中的部分计算机，便可根 据这些被抽查的计算机的质量来推测出整批计算机的质量。

枚举归纳推理是指在进行归纳时，如果已知某类事物的有限可数个具体事物都具有某 种属性，则可推出该类事物都具有此种属性。设小辺2，…辺”是某类事物A的具体事物，若 已知釦，阪，…，七都具有属性B,并且没有发现反例，那么当*71*足够大时，就可以得岀“事物 A中的所有事物都具有属性B”这一结论。

例如，设有如下事例：

王强是计算机系学生，他会编写程序。

高华是计算机系学生，他会编写程序。

李明是计算机系学生，他会编写程序。

当这些具体事例足够多时，就可归纳出一个一般性的知识：

凡是计算机系的学生，就一定会编写程序。

类比归纳推理是指在两个或两类事物有许多属性都相同或相似的基础上，推出它们在 其他属性也相同或相似的一种归纳推理。

设分别是两类事物的集合：

*A = {ax ,a2 ^a3 »•,,}>* B = {。】，缶

并设％与"总是成对出现，且当《有属性*P*时0就有属性Q与之对应，即

P(a,-) —► Q(6f) £ = 1,2,3,…

则当A与B中有新的元素对出现的时候，若已知/有属性P0有属性Q,即P(/)f Q(y)o

类比归纳推理的基础是相似原理，其可靠程度取决于两个或两类事物的相似程度，以及 这两个或两类事物的相同属性与推岀的那个属性之间的相关程度。

3)演绎推理与归纳推理的区别

演绎推理与归纳推理是两种完全不同的推理。演绎推理是在已知领域内的一般性知识 的前提下，通过演绎求解一个具体问题或者证明一个给定的结论。这个结论实际上早已蕴 含在一般性知识的前提中，演绎推理只不过是将其解释出来，因此它不能增殖新知识。

在归纳推理中，所推出的结论是没有包含在前提内容中的。这种由个别事物或现象推 出一般性知识的过程是增殖新知识的过程。

1. 按所用知识的确定性分类

按所用知识的确定性，推理可以分为确定性推理和不确定性推理。所谓确定性推理，是 指推理所使用的知识和推出的结论都是可以精确表示的，其真值要么为真，要么为假，不会 有第3种情况出现。本章主要讨论的是确定性推理。

所谓不确定性推理，是指推理时所用的知识不都是确定的，推岀的结论也不完全是确定 的，其真值会位于真假之间。由于现实世界中大多数事物都具有一定程度的不确定性，并且 这些事物很难用精确的数学模型来进行表示和处理，因此不确定性推理也就成了人工智能 的一个重要研究课题。不确定性推理将在第5章进行讨论。

1. 按推理过程的单调性分类

按照推理过程的单调性，或者说按照推理过程所得岀的结论是否越来越接近目标,推理 可分为单调推理与非单调推理。

所谓单调推理是指在推理过程中，每当使用新知识后，所得到的结论会越来越接近目 标，而不会出现反复的情况，即不会因为新知识的加入否定了前面推出的结论•从而使得推 理过程又退回到先前的一步。

所谓非单调性推理是指在推理过程中，当某些新知识加入后，会否定原来推出的结论， 使推理过程退回到先前的一步。非单调性推理往往是在知识不完全的情况下发生的。在这 种情况下，为使推理能够进行下去，就需要先进行某些假设，并在此假设的基础上进行推理。 但是，当后来由于新的知识的加入，发现原来的假设不正确时，就需要撤销原来的假设及以 此假设为基础推出的一切结论，再运用新的知识重新进行推理。

**4.1.3**推理的控制策略及其分类

推理过程不仅依赖于所用的推理方法，也依赖于推理的控制策略。推理的控制策略是 指如何使用领域知识使推理过程尽快达到目标的策略。由于智能系统的推理过程一般表现 为一种搜索过程，因此，推理的控制策略又可分为推理策略和搜索策略。其中，推理策略主 要解决推理方向、冲突消解等问题，如推理方向控制策略、求解策略、限制策略、冲突消解策 略等；搜索策略主要解决推理线路、推理效果、推理效率等问题。

推理方向用来确定推理的控制方式，即推理过程是从初始证据开始到目标，还是从目标 开始到初始证据。按照对推理方向的控制，推理可分为正向推理、逆向推理和混合推理等。 无论哪一种推理方式，系统都需要有一个存放知识的知识库，一个存放初始证据和中间结果 的综合数据库和一个用于推理的推理机。求解策略是指仅求一个解还是求所有解或最优解 等。限制策略是指对推理的深度、宽度、时间、空间等进行的限制。冲突消解策略是指当 推理过程有多条知识可用时，如何从这多条可用知识中选岀一条最佳知识用于推理的策 略。常用的冲突为选择知识的依据。新鲜知识优先，是指把知识前提条件中事实的新鲜 性作为选择知识的依据。例如，综合数据中后生成的事实比先生成的事实具有更大的新 鲜性.

对于推理控制策略所包含的推理策略和搜索策略，本章主要讨论推理策略。

**4.1.4**正向推理

正向推理是一种从已知事实出发，正向使用推理规则的推理方法，也称为数据驱动推理 或者前向链推理。其基本思想是：用户需要实现提供一组初始证据，并将其放入综合数据 库。推理开始后，推理机根据综合数据库中的已有事实，到知识库中寻找当前可用知识，形 成一个当前可用知识集，然后按照冲突消解策略，从该知识集中选择一条知识进行推理，并 将新推出的事实加入综合数据库，作为后面继续推理时可用的已知事实。如此重复这一过 程，直到求出所需要的解或者知识库中再无可用知识为止。

正向推理过程可用如下算法描述：

（1） 用户提供的初始证据放入综合数据库。

（2） 检查综合数据库中是否包含了问题的解。若已包含，则求解结束，并成功退出；否 则，执行下一步。

（3） 检査知识库中是否有可用知识。若有，形成当前可用知识集,执行下一步；否则，转（5）。

（4） 按照某种冲突消解策略，从当前可用知识集中选出一条知识进行推理，并将推出的新事实加入综合数据库，然后转（2）。

（5）询问用户是否可以进一步补充新的事实，若可补充，则将补充的新事实加入综合数 据库，然后转（3）；否则表示无解，失败退出。

以上算法的流程图如图4.1所示。

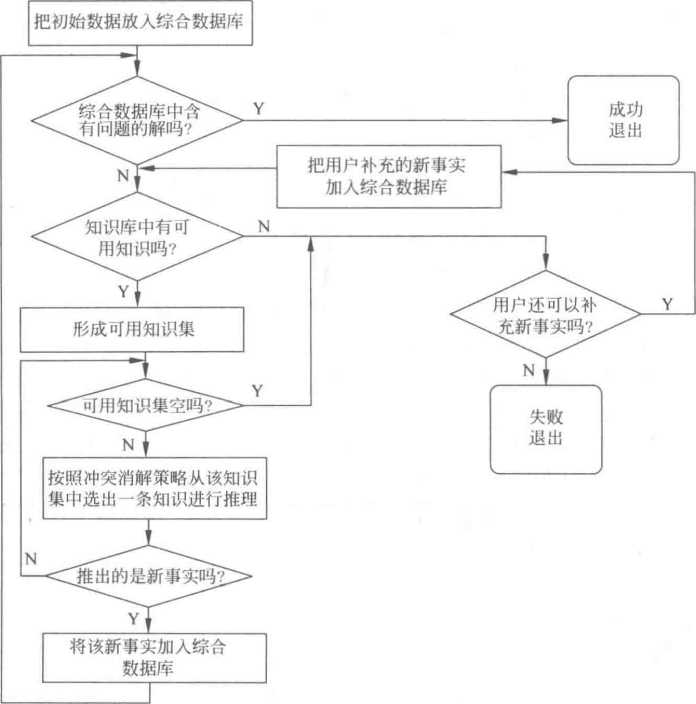


图4.1正向推理流程图

仅从正向推理的算法来看好像比较简单，但实际上，推理的每一步都还有许多要做的工 作。例如，如何根据综合数据库中的事实在知识库中选取可用知识；当知识库中有多条知 识可用时,应该先使用哪一条知识等。这些问题涉及知识的匹配方法和冲突消解策略，如何 解决，以后将进行讨论。

作为对正向推理过程的说明，下面给出一个简单的例子。

例4.1请用正向推理完成以下问题的求解。

假设知识库中包含以下两条规则：

r, ： IF B THEN C

r2 ： IF A THEN *B*

已知初始证据A,求证目标C。

解：本例的推理过程如下。

推理开始前，综合数据库为空。推理开始后，先把初始证据A放入综合数据库，然后检 查综合数据库中是否含有该问题的解，回答为“N”。接着检查知识库中是否有可用知识，显 然r2可用，形成仅含a的知识集。从该知识集中取出a，推出新的事实B,将B加入综合 数据库，检查综合数据库中是否有目标C,回答为“N”。再检查知识库中是否有可用知识， 此时由于B的加入使得门可用，形成仅含门的知识集。从该知识集中取岀门，推出新的事 实C,将C加入综合数据库，检查综合数据库中是否含有目标G回答为“Y”。说明综合数 据库中已经包含问题的解，推理过程成功结束，目标C得证。

正向推理的优点是比较直观，允许用户主动提供有用的事实信息，适合于诊断、设计、预 测、监控等领域的问题求解。其主要缺点是推理无明确的目标，求解问题时可能会执行许多 与解无关的操作，导致推理效率较低。

**4.1.5**逆向推理

逆向推理是一种以某个假设目标作为出发点的推理方法，也称为目标驱动推理或逆向 链推理。其基本思想是：首先根据问题求解的要求，将要求证的目标（称为假设）构成一个 假设集，然后从假设集中取出一个假设对其进行验证，检查该假设是否在综合数据库中，是 否为用户认可的事实。当该假设在数据库中时，该假设成立，若此时假设集为空，则成功退 出；若假设不在综合数据库中，但可被用户证实为原始证据时，将该假设放入综合数据库， 此时若假设集为空，则成功退出；若假设可由知识库中的一个或多个知识导出，则将知识库 中所有可以导出该假设的知识构成一个可用知识集，并根据冲突消解策略，从可用知识集中 取出一个知识，将其前提中的所有子条件都作为新的假设放入假设集。重复上述过程，直到 假设集为空时成功退岀，或假设集非空但可用知识集为空时失败退岀为止。

逆向推理过程可用如下算法描述：

（1） 问题的初始证据和要求证的目标（称为假设）分别放入综合数据库和假设集。’

（2） 从假设集中选取一个假设，检查该假设是否在综合数据库中。若在，则该假设成 立。此时，若假设集为空，则成功退出；否则，仍执行（2）。若假设不在数据库中，则执行下 一步。

（3） 检查该假设是否可由知识库中的某个知识导出。若不能由某个知识导出，则询问 用户该假设是否为可由用户证实的原始事实。若是，则该假设成立，并将其放入到综合数据 库，再重新寻找新的假设；若不是，则转（5）。若能由某个知识导出，则执行下一步。

（4） 知识库中可以导岀该假设的所有知识构成一个可用知识集。

（5） 检查可用知识集是否为空。若空，失败退岀；否则，执行下一步。

（6） 按照冲突消解策略从可用知识集中取出一个知识，继续执行下一步。

（7） 该知识的前提中的每个子条件都作为新的假设放入假设集，转（2）。

以上算法的流程图如图4.2所示。

对例4. 1的问题，如果采用逆向推理方法，其推理过程如下。

推理开始前，综合数据库和假设集均为空。推理开始后，先将初始证据A和目标C分 别放入综合数据库和假设集，然后从假设集中取出一个假设C,查找假设C是否为综合数据 库中的已知事实，回答为“N”。再检查C是否能被知识库中的知识所导出，发现C可由八 导出，于是小被放入可用知识集。由于知识库中只有门可用，故可用知识集中仅含门。接

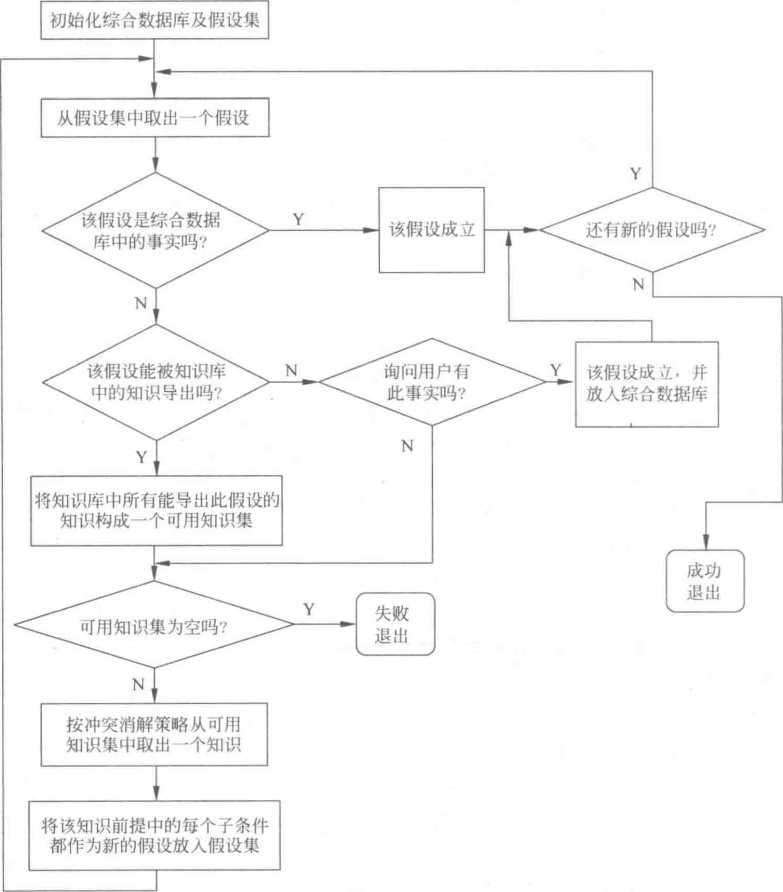


图4.2逆向推理的流程图

着从可用知识集中取出门，将其前提条件*B*作为新的假设放入假设集。从假设集中取岀 B,检查*B*是否为综合数据库中的事实，回答为“N”。再检查*B*是否能被知识库中的知识所 导出，发现B可由l导出，于是a被放入可用知识集。由于知识库中只有r2可用，故可用 知识集中仅含a。从可用知识集中取出a，将其前提条件A作为新的假设放入假设集。然 后从假设集中取出A,检查*A*是否为综合数据库中的事实，回答为“Y”，说明该假设成立。 由于无新的假设，故推理过程成功结束，于是目标C得证。

逆向推理的主要优点是不必寻找和使用那些与假设无关的信息和知识，推理过程的目 标明确，也有利于向用户提供解释，在诊断性专家系统中较为有效。其主要缺点是，当用户 对解的情况认识不清的时候，由系统自主选择假设目标的盲目性比较大，若选择不好，可能 需要多次提岀假设，会影响系统效率。

**4.1.6**混合推理

由以上讨论可知，正向推理和逆向推理都有各自的优缺点。当问题较为复杂时，单独使 用其中的一种，都会影响到推理效率。为了更好地发挥这两种算法各自的长处，避免各自的 短处，互相取长补短，可以将它们结合起来使用。这种把正向推理和逆向推理结合起来所进 行的推理称为混合推理。

混合推理可以有很多具体的实现办法。例如，可以釆用先正向推理，后逆向推理的方 法；也可以采用先逆向推理，再正向推理的方法；还可以采用随机选择正向和逆向推理的 方法。由于这些方法仅是正向推理和逆向推理的某种结合，因此对这3种情况不再进行 讨论。

**4.2**推理的逻辑基础

在前文中，讨论了知识表示的逻辑基础，已经引入了谓词逻辑的一些简单概念。本节主 要讨论推理所需要的一些逻辑基础。

**4.2.1**谓词公式的解释

在命题逻辑中，命题公式的一个解释就是对该命题公式各个命题变元的一次真值指派。 有了命题公式的解释，就可以根据这个解释求出该命题公式的真值。但谓词逻辑不同，由于 谓词公式中可能包含有个体常量、个体变元或者函数，因此不能像命题公式那样直接通过真 值指派为谓词分别指派真值。下面给出谓词公式的解释的含义。

定义4.1设D是谓词公式P的非空个体域，若对*P*中的个体常量、函数和谓词按照 如下规则赋值：

1. 为每个个体变量指派*D*中的一个元素。
2. 为每个〃元函数指派一个从D”到D的映射，其中

*Dn* = {&i 口2，…，i”)I zi 口2 ,了“ G *D}*

1. 为每个〃元谓词指派一个从D”到{F,T}的映射。

则称这些指派为P在D上的一个解释。

例4.2 设个体域D={1,2},求公式A=( V7)Cy)P(工,丁)在D上的解释，并指出在 每一种解释下公式A的真值。

解：由于公式A中没有包含个体常量和函数，因此可以直接为谓词指派真值，设有

|  |  |  |  |
| --- | --- | --- | --- |
|  | P(l,2) | P(2,l) | P(2,2) |
| T | F | T | F |

这就是公式A在D上的一个解释。从这个解释中可以看岀： •当*x=l.y=* 1时，有卩&以)的真值为T。

•当工=2点=1时，有P(X,y)的真值为T。

即对H在D上的任意取值，都存在丁=1使的真值为To因此，在此解释下公 式A的真值为To

需要注意的是，一个谓词公式在其个体域上的解释是不唯一的。例如，对公式A,若给 出另一组真值指派：

|  |  |  |  |
| --- | --- | --- | --- |
|  | P(l,2) | P(2,l) | P(2,2) |
| T | T | F | F |

这也是公式A在D上的一个解释。从这个解释可以看出：

•当*x=l,y =* 1时，有*P(x.y)*的真值为T。

•当了=2,y = 1时，有P(X,Y)的真值为F。

同样：

•当了=1,丁=2时，有F(z,))的真值为F。

•当卫=2,丁 = 2时，有P(X,Y)的真值为T。

即对h在D上的任意取值，不存在一个了使得P&以)的真值为T。因此，在此解释下 公式A的真值为F。

实际上，A在D上共有16种解释，在这里不一一列举。

例4.3 设个体域D={1,2},求公式B=( Vz)P(fa),")在D上的解释，并指出在该 解释下公式B的真值。

解：设对个体常量。和函数，&)的真值指派为

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *a* | | /(l) | | /(2) | |
| 1 | | 1 | | 2 | |
| 对谓词的真值指派为 | | | | | |
| P(l,l) | P(l,2) | | P(2,l) | | P(2,2) |
| T | X | | T | | X |

这里，由于已知指派。=1,所以P(l,2)和P(2,2)不可能出现，故没有给它们指派 真值。

上述指派是公式B在D上的一个解释。在此解释下有：

•当 1=1 时,a = l 使得 P(1,1) = TO

•当 1=2 时 s=l 使得 P(2,1) = T。

即对丁在。上的任意取值，都存在。=1使P(/(^),a)的真值为T。因此，在此解释下 公式B的真值为T。

由上面的例子可以看出，谓词公式的真值都是针对某一个解释而言的，它可能在某一个 解释下真值为T,而在另一个解释下真值为Fo

**4.2.2**谓词公式的永真性与可满足性

为了以后推理的需要，下面先定义谓词公式的永真性、永假性、可满足性与不可满足性。 定义4.2如果谓词公式*P*对非空个体域*D*上的任一解释都取得真值T,则称P在D 上是永真的；如果*P*在任何非空个体域上均是永真的，则称*P*永真。

由此定义可以看出，要判定一个谓词公式为永真，必须对每个非空个体域上的每个解释 逐一进行判断。当解释的个数有限时，尽管工作量大，公式的永真性毕竟还是可以判定的； 但当解释个数无限时，其永真性就很难判定了。

定义4. 3对于谓词公式P,如果至少存在*D*上的一个解释，使公式*P*在此解释下的真 值为T,则称公式F在D上是可满足的。

谓词公式的可满足性也称为相容性。

定义4.4如果谓词公式P对非空个体域D上的任一解释都取真值F,则称P在D上 是永假的；如果*P*在任何非空个体域上均是永假的，则称*P*永假。

谓词公式的永假性又称不可满足性或不相容性。

**4.2.3**谓词公式的等价性与永真蕴含性

谓词公式的等价性和永真蕴含性可分别用相应的等价式和永真蕴含式来表示，这些等 价式和永真蕴含式都是演绎推理的主要依据，因此也称它们为推理规则。

**1.**等价式

谓词公式的等价式可定义如下。

定义4.5设P与Q是D上的两个谓词公式，若对D上的任意解释，F与Q都有相同 的真值，则称F与Q在D上是等价的。如果*D*是任意非空个体域，则称P与Q是等价的, 记作P0Q。

常用的等价式如下。

1. 双重否定律：
2. 交换律：

*P* V Q0Q V P, *P* A Q0Q A *P*

1. 结合律：

(F V Q)V V(Q V R)

(P A Q) A R0F A (Q /\ R)

1. 分配律：

P V(Q A R)0(P V Q)A (F V R)

P A (Q V R)e(P A Q) V(P A R)

1. 摩根定律：

「(P V Q)0「P A

「(P A Q)<=^*P* V「Q

1. 吸收律：

P V (P A Q)eP, P A (P V Q)0F

1. 补余律：

*P* V ~>P0T, *P* A "0F

1. 连词化归律：

*P* f Qo「P V Q

F〜Q0(Pf Q)a (Qf P)

P^>qo(P A Q)V (「Q A ")

1. 量词转化律：

-> (3 jr)P(j?)<=>( V z) (「P(x))

->(v *z)p&)0(* m z) (「*P&))*

1. 量词分配律：

(V x) (P(j?) *A* Q(z) )0( V jc)P(i) A (Vz)Q(z)

(D(P&) V Q(z))0( DP(z) V(MQ&)

|  |
| --- |
| **2.**永真蕴含式 |
| 谓词永真蕴含式可定义如下。 |
| 定义4. 6对谓词公式*P*和Q,如果P-Q永真，则称*P*永真蕴含Q,且称Q为F的逻 |
| 辑结论，P为Q的前提，记作F0Q。 |
| 常用的永真蕴含式如下。 |
| (1)化简式： |
| *P* A *Q=>P, P* A QnQ |
| (2)附加式： |
| *P》P* V *Q, Q=>P* V *Q* |
| (3)析取三段论： |
| V Q=>Q |
| (4)假言推理： |
| P,Pf QnQ |
| (5)拒取式： |
| 「Q,P -► Q=> *P* |
| (6)假言三段论： |
| F-\* Q, *QfR=>P — R* |
| (7)二难推理： |
| P V Q» *P f R, Q f R=>R* |
| (8)全称固化： |
| *(\/ x)P(x)=>P(y)* |
| 式中以是个体域中的任一个体，利用此永真蕴含式可消去谓词公式中的全称量词。 |
| (9)存在固化： |

(DFa)》P3)

式中点是个体域中某一个可以使P3)为真的个体，利用此永真蕴含式可消去谓词公式中

的存在量词。

上面给出的等价式和永真蕴含式是进行演绎推理的重要依据，因此这些公式也被称为 推理规则。除了这些公式以外，在4. 4节的归结演绎推理中，还需要将反证法推广到谓词公 式集，即*G为F*的逻辑推论，当且仅当FA「G是不可满足的。

**4.2.4**谓词公式的范式

范式是公式的标准形式，公式往往需要变换为同它等价的范式，以便对它们进行一般性 的处理。在谓词逻辑中，根据量词在公式中出现的情况，可将谓词公式的范式分为两种。

1. 前束范式

定义4.7设F为一个谓词公式，如果其中的所有量词均非否定地岀现在公式的最前 面，而它们的辖域为整个公式，则称*F*为前束范式。一般地，前束范式可写成

(QiZi ) (Q„j;n)M(jC] ,je2 , ,x„)

式中，Q,V=l,2,・・・，〃)为前缀，它是一个由全称量词或存在量词组成的单词串；MS】， 心，・・・，乃)为母式，它是一个不含任何量词的谓词公式。

例如，(Vz)( VjO( m z)(F(i) AQ3，z) VRCr,z))是前束范式。

任一含有量词的谓词公式均可化为与其对应的前束范式，其化简方法将在4. 4. 1节中 讨论。

1. **Skolem** 范式

定义**4.8**如果前束范式中的所有存在量词都在全称量词之前，则称这种形式的谓词 公式为Skolem范式。

例如心 z)O^)(3 ;y)(P&) VQ(v，z) 是 Skolem 范式。

任一含有量词的谓词公式均可化为与其对应的Skolem范式，其化简方法将在4. 4. 1 节中讨论。

**4.2.5**置换与合一

在不同的谓词公式中，往往会出现谓词名相同但其个体不同的情况，此时推理过程是不 能直接进行匹配的，需要先进行置换。例如，可根据全称固化推理和假言推理由谓词公式 WJA)和(Vz)(Wi(z)fW2(z))推出W2(A)o对谓词W】(A)可看做是由全称固化推理 推出的，其中A是任一个体常量。要使用假言推理，首先需要找到项A对变元z的置换，使 WJA)与WJz)一致。这种寻找项对变元的置换，使谓词一致的过程叫做合一的过程。下 面讨论置换与合一的有关概念与方法。

**1.**置换

置换(substitution)可以简单地理解为在一个谓词公式中用置换项替换变元。其形式 定义如下：

定义**4. 9**置换是形如

的有限集合。其中«2，…，L是项，心口2，…，1〃是互不相同的变元；』,/+表75用，，置换 宀，并且要求&与邛不能相同，乃不能循环地出现在另一个"中。

例如：

*{a/x.c/y,f{b}/z}*

是一个置换，但是

*{g(y)/x.f(x)/y}*

不是一个置换，原因在于h与了之间岀现了循环置换的现象。置换的目的本来是要将某些 变元用另外的变元、常量或函数取代，使其不在公式中出现。但在草3)/以，(工)/少中，它 用g(y)置换工，用/Xg3))置换y,既没有消去了，也没有消去贝 若改为

*{gCa^/jc.fCx^/y}*

就可以了，它将把公式中的z用g耸)来置换用/(g(a))来置换，从而消去了 1和V。 通常，置换用希腊字母们等来表示。

定义4.10设 *…，丄心*是一个置换，F是一个谓词公式，把公式*F*

中出现的所有工换成Z,3 = 1,2,・・・，〃)，得到一个新公式G,称*G为F*在置换。下的例示，记 作 *G=F0o*

一个谓词公式的任何例示都是该公式的逻辑结论。

定义**4.11**设

*0 =*修/了1，如/%2，…“”/工”}

A = *{ujyx ,u2/y2*

是两个置换。则。与人的合成也是一个置换，记作。・人。它是从集合

"2/心，…，，”/工” Hi/、】 *^u2/y2^- ^um/ym}*

中删去以下两种元素：

1. 当 *tiX — Xi* 时，删去 M/了,,(，= 1,2, •••,〃)；
2. 当 *y,* G {xi *9x2*，•- *,xn}*时，删去 *Uj/y)*, (j = 1,2, — ,m) o

最后剩下的元素所构成的集合。

例 4.4 设 *0={f(y')/x,z/y} {a/x,b/y,y/z}*,求。与人的合成。

解：先求出集合

*{f^b/y^/xAy/z^/y.a/x.b/y.y/z} = {f{b)/x,y/y,a/x,b/y.y/z)*

式中,/(6)/x中的八。)是置换以作用于/'(少的结果，“丁中的了是置换义作用于z的结 果。在该集合中点/j/满足定义中的条件(1),需要删除；和力'满足定义中的条件(2), 也需要删除。最后得

。•人=*{fCb)/x,y/z}*

**2.**合一

合一(unifier)可以简单地理解为寻找项对变量的替换，使得两个谓词公式一致。其形 式定义如下。

定义**4.12** 设有公式集F={F1 ,Fz，…，FQ ,若存在一个置换们可使*F}0=F20=- =* F/,则称。是F的一个合一，称F|,F2,…，F“是可合一的。

例如，设有公式集 F={P(H，3sr(G),PS,g(i),N)},则

A = {Q/z,gS)/)/(g(Q))/z}

是*F*的一个合一。

一般来说，一个公式集的合一不是唯一的。

定义4.13设3是公式集*F*的一个合一，如果对*F*的任一个合一*9*都存在一个置换人， 使得 *6=8* • 4,则称 8 是一个最一般合一(Most General Unifier,MGU) o

一个公式集的最一般合一是唯一的。若用最一般合一去置换那些可合一的谓词公式， 可使它们变成完全一致的谓词公式。

**4.3**自然演绎推理

从一组已知为真的事实出发，直接运用经典逻辑中的推理规则推出结论的过程称为自 然演绎推理。在这种推理中，最基本的推理规则为三段论推理，包括假言推理、拒取式推理、 假言三段论等。

在自然演绎推理中，需要避免两类错误：肯定后件的错误和否定前件的错误。所谓肯 定后件的错误是指，当P-Q为真时，希望通过肯定后件Q为真来推出前件P为真，这是不 允许的。原因是当*P-Q为*真及Q为真时，前件P既可能为真，也可能为假。所谓否定前 件的错误是指，当P-Q为真时，希望通过否定前件P来推出后件Q为假，这也是不允许 的。原因是当P-Q及P为假时，后件Q既可能为真，也可能为假。

例4.5设已知如下事实：

|  |  |
| --- | --- |
| 求证：Q为真。 | |
| 证明：因为C *=>C* | 假言推理 |
| *B,C=>B^C* | 引入合取词 |
| *B/\C,B/\CfD=> D* | 假言推理 |
| D,D->Q=〉*Q* | 假言推理 |

所以，Q为真。

例4.6设已知如下事实：

1. 如果是需要编写程序的课，王程就喜欢。
2. 所有的程序设计语言课都是需要编写程序的课。
3. C语言是一门程序设计语言课。

求证：王程喜欢C语言这门课。

证明：首先定义谓词。

Prog(j?) 7是需要编程序的课

Like(j?,j/) 7 喜欢;y

Lang(jr) z是一门程序设计语言课。

把上述已知事实及待求解问题用谓词公式表示如下： Prog(z)f Like(Wang, *jc)*

(V z) (Lang(n)—► Prog(z))

Lang(C)

应用推理规则进行推理：

Lang(>)->Prog(>) 全称固化

Lang(C) ,Lang(y)f Prog(y)=〉Prog(C) 假言推理

Prog(C) ,Prog(j')->Like( Wang, z) => Like(Wang,C) 假言推理

因此，王程喜欢C语言这门课。

一般来说，自然演绎推理由已知事实推出的结论可能有多个，只要其中包含了需要证明 的结论，就认为问题得到解。

自然演绎推理的优点是定理证明过程自然，易于理解，并且有丰富的推理规则可用。其 主要缺点是容易产生知识爆炸，推理过程中得到的中间结论一般按指数递增，对于复杂问题 的推理不利，甚至难以实现。

**4.4**归结演绎推理

归结演绎推理是一种基于鲁滨逊归结原理的机器推理技术，鲁滨逊归结原理也称为消 解原理.是鲁滨逊于1965年在海伯伦(Herbrand)理论基础上提出的一种基于逻辑反证法 的机械化定理证明方法。

在人工智能中，几乎所有的问题都可转化为一个定理证明问题，而定理证明的实质就是 要对前提*P*和结论Q证明*P-Q*永真。由4. 2节可知，要证明*P-Q*永真，就是要证明P-\* Q在任何一个非空的个体域上都是永真的。这将是非常困难的，甚至是不可能实现的。为 此，人们进行了大量的探索，后来发现可以釆用反证法的思想，把关于永真性的证明转化为 关于不可满足性的证明，即，要证明P-Q永真，只要能够证明PA「Q为不可满足即可，这 正是归结演绎推理的基本出发点•

**4.4.1**子句集及其简化

由于鲁滨逊归结原理是在子句集的基础上进行定理证明的，因此，在讨论这些方法之 前，需要先介绍子句集的有关概念。

1. 子句和子句集

定义，・14原子谓词公式及其否定统称为文字。

例如,P(x)、Q(z)、「P(z)、「Q(i)等都是文字。

定义4.15任何文字的析取式统称为子句。

例如,P(x) V Q(x) /(x)) V Q(z,g(z))都是子句o

定义，.16不包含任何文字的子句统称为空子句。

由于空子句不含有任何文字，也就不能被任何解释所满足，因此空子句是永假的，不可 满足的。空子句一般记为□或NIL。

定义4.17由子句或空子句所构成的集合称为子句集。

1. 子句集的化简

在谓词逻辑中，任何一个谓词公式都可以通过应用等价关系及推理规则化简成相应的 子集，其化简步骤如下：

1. 消去连接词—和〜。

反复使用如下等价公式：

*p — Q0「p \/ Q*

P<->Qo(P A Q)V (P A「Q)

即可消去谓词公式中的连接词f和J。

例如公式

(V 7)(( V ：y)P(z，V)—「( V *y) (QCx ,y) f R(i,y)))*

经等价变化后为

(Vj?)( -> ( *\/ y) PCx 9 y)* V ~\* ( V y) ( ~\* Q(z,;y)—\*• R(2r;y)))

1. 减少否定符号的辖域。

反复使用双重否定律：

摩根定律：

-• (P A Q)0「P V *-'Q*

「(P V Q)D「P A ->Q

量词转化律：

-•(V^)P(^)<=>(3jc)^P(x)

-» ( 3 z)P(z)<=4( V z)「P(x)

将每个否定符号「移到紧靠谓词的位置，使得每个否定符号最多只作用于一个谓词上。

例如，第(1)步所得公式经本步变换后为

(V1)(( m 夕)「P(z,jy) V ( - jy)(Q(z,jO A 「RCi,、)))

1. 对变元标准化。

在一个量词的辖域内，把谓词公式中受该量词约束的变元全部用另外一个没有出现过 的任意变元代替，使不同量词约束的变元有不同的名字。

例如，第(2)步所得公式经本步变换后为

(Vz)(( m 丁)「P(“以)V ( 3 (Q(j：,z) A、R(z,z)))

1. 化为前束范式。

化为前束范式的方法是把所有量词都移到公式的左边，并且在移动时不能改变其相对 顺序。由于第(3)步已对变元进行了标准化，每个量词都有自己的变元，这就消除了任何由 变元引起冲突的可能，因此这种移动是可行的。

例如，第(3)步所得公式化为前束范式后为

(Vz)( m jy)( m z)(「P(z以)V (Q(z,z) A (丁，z)))

1. 消去存在量词。

消去存在量词时，需要区分以下两种情况。

•若存在量词不岀现在全称量词的辖域内(即它的左边没有全称量词)，只要用一个新 的个体常量替换受该存在量词约束的变元，就可消去该存在量词。

•若存在量词位于一个或多个全称量词的辖域内，例如：

(V X1 ) ( V x2 ) ,,, ( V ) ( 3))F&】口2 9 …口“ ,y)

则需要使用Skolem函数/(幻，◎，…口”)替换受该存在量词约束的变元，然后消去该 存在量词。

例如，在第(4)步所得公式中存在量词(不了)和(Mz)都位于(Vz)的辖域内，因此都需 要用Skolem函数来替换。设替换y和z的Skolem函数分别是f(z)和g(z),则替换后的 公式为

(Vz)(「Pii/a)) v(Q&,g(z))a「R(z,ga))))

1. 化为Skolem标准形。

Skolem标准形为

(V j-i ) ( V ( V xn)M(jr1 ,工2，…

式中，M(zi ,了2，•••，］”)是Skolem标准形的母式，它由子句的合取构成。

把谓词公式化为Skolem标准形需要使用以下等价关系:

P V(Q A R)0(p V Q) A (F v R)

例如，第(5)步所得的公式化为Skolem标准形后为

(Vx)(( -1 V QM,ga))) A (「PG’/Xz)) V「R&’gG))))

1. 消去全称量词。

由于母式中的全部变元均受全称量词的约束，并且全称量词的次序已无关紧要，因此可 以省略。但对于剩下的母式，仍假设其变元是被全称量词量化的。

例如，第(6)步所得公式消去全称量词后为

(P(jt,/(j?)) V Q(z,g(z))) A (「V R(z,g&)))

1. 消去合取词。

在母式中消去所有合取词，把母式用子句集的形式表示出来。其中，子句集的每一个元 素都是一个子句。

例如，第(7)步所得公式的子句集中包含以下两个子句：

*P(x,f(x')*) V Q&,g(；r))

-> /(x)) V ->R(i,g&))

1. 更换变元名称。

对子句集中的某些变元重新命名，使任意两个子句中不出现相同的变元名。由于每一 个子句都对应母式中的一个合取元，并且所有变元都是由全称量词量化的，因此任意两个不 同子句的变元之间实际上不存在任何关系。这样，更换变元名不会影响公式的真值。

例如，对第(8)步所得公式，可把第二个子句集中的变元名亢更换为丁,得到如下子句集：

V Q(i,g(s))

1. 子句集的应用

通过上述化简步骤，可以将谓词公式化简为一个标准子句集。由于在消去存在量词时 所用的Skolem函数可以不同，因此化简后的标准子句集是不唯一的。这样，当原谓词公式 为非永假时，它与其标准子句集并不等价。但是，当原谓词公式为永假(即不可满足)时，其 标准子句集则一定是永假的，即Skolem化并不影响原谓词公式的永假性。这个结论很重 要，是归结原理的主要依据，可用定理的形式来描述。

定理**4.1**设有谓词公式F,其标准子句集为S,则*F*为不可满足的充要条件是S为不 可满足的。

在证明此定理之前，先进行如下说明。

为讨论问题方便，设给定的谓词公式F已为前束形：

(Qijt］ )\*--(Qrxr)--(QMj'w)M(a'1 口2，…，z”)

式中，MGnu2,…口“)已化为合取范式。由于将F化为这种前束形是一种很容易实现的等 值运算，因此这种假设是可以的。

又设(Qq「)是第一个出现的存在量词(刁右)，即F为

*F* = ( V Xi )•••( V )( 3 *xr) ••• (Qnjc n)* ,…,*,xr* ,1\* ,・•・，《r“)

为把F化为Skolem标准形，需要先消去这个(不万)，并引入Skolem函数，得到

F1 = ( V )•••( V ZlI)(QilZrH)・・.(Q”Zn)M(Hi，…,It，/'(幻,Tl1)，Zh4，…,Z”) 若能证明

F不可满足0F］不可满足

则同理可证

F1不可满足=F2不可满足

重复这一过程，直到证明了

Fk】不可满足*^Fm*不可满足

为止。此时,已已为F的Skolem标准形。而S只不过是F.的一种集合表示形式。因此有

F不可满足0S不可满足

下面开始用反证法证明

F不可满足不可满足

先证明

已知F不可满足，假设F】是可满足的，则存在一个解释J,使F】在解释［下为真。即 对任意的］】，・•・01在*I*的设定下有

((2宀1+ )・・・(。宀)河(工1，…，\*1 ,/(工］,…，ZlI)口+，…，n)

为真，即对任意的…，11都有一个 *心*,…,了―)使

(Qf 1^41 )...(。,&兀)同(飞1 ,...,Zl1，，••• ) ,ZrH , ，Z“)

为真，即在J下有

(V JT1 ),,,( V *JCr-\* ) ( 3 JCrXQr+lXH-l ) •,• ( Q„X B ) M( J：i，•-- , Xr ，…,右)

为真，即F在I下为真。

但这与前提F是不可满足的相矛盾，即假设Fl为可满足是错误的。从而可以得出“若 F不可满足，则必有H不可满足”。

再证明。。

已知F|不可满足，假设F是可满足的。于是便有某个解释I使F在I下为真。即对 任意的.，・・m 在I的设定下都可以找到一个使

(Q—I lr+1 )・・・(Qq~”)M&i，…，,"了宀 m”)

为真。若扩充I,使它包含一个函数，(e，・・・ui)，且有

*Xr* = /(X］,…，Zi)

这样，就可以把所有的(心，…口—)映射到工，从而得到一个新的解释n并且在此解释下 对任意的X1，— *,Xr-｝*都有

(Q^! )\*-\*(Q„J7„ )M(Xj，…，Zi，*f (工1*，…，了宀)】，…，工”)

为真。即在「下有

（Vii ）•••（ V *Xr-i* ）（3 7『）（＜2心1+ ）・・・（Qq“）M（«ti ,•••,工1,/（jci，…，:Ti）, ji，…，z”） 为真。它说明E在解释『下为真。但这与前提F】是不可满足的相矛盾，即假设F为可满 足是错误的。从而可以得出“若E不可满足，则必有F不可满足气

于是，定理得证。

由此定理可知，要证明一个谓词公式是不可满足的，只要证明其相应的标准子句集是不 可满足的就可以了。至于如何证明一个子句集的不可满足性，由鲁滨逊归结原理来解决。

**4.4.2**鲁滨逊归结原理

鲁滨逊归结原理通过对子句集的子句做逐次归结来证明子句集的不可满足性，它是对 定理自动证明的一个重大突破。

1. 基本思想

由谓词公式转化为子句集的方法可以知道，在子句集中子句之间是合取关系。其中，只 要有一个子句为不可满足的，则整个子句集就是不可满足的。另外，前面已经指出空子句是 不可满足的。因此，一个子句集中如果包含空子句，则此子句集就一定是不可满足的。

鲁滨逊归结原理就是基于上述认识提出来的。其基本思想是：首先把欲证明问题的结 论否定，并加入子句集，得到一个扩充的子句集S'。然后设法检验子句集S'是否含有空子 句，若含有空子句，则表明宁是不可满足的；若不含有空子句，则继续使用归结法，在子句集 中选择合适的子句进行归结，直至导出空子句或不能继续归结为止。鲁滨逊归结原理分为 命题逻辑归结原理和谓词逻辑归结原理。

1. 命词逻辑的归结

归结推理的核心是求两个子句的归结式，因此需要先讨论归结式的定义和性质，然后讨 论命题逻辑的归结过程。

下面给岀归结式的定义及性质。

定义4.18若P是原子谓词公式，则称P与为互补文字。

定义4.19设G和C2是子句集中的任意两个子句，如果G中的文字L】与G中的 文字切互补，那么可以从G和G中分别消去心和L?,并将G和C2中余下的部分按析 取关系构成一个新的子句G2,则称这一过程为归结，称Gz为G和C2的归结式，称G和 c2为G2的亲本子句。

例4.7 设G=PVQVR,C2=「PVS,求G和C2的归结式

解：这里L1=P,L2=-P,通过归结可以得到

C12 = Q V R V *S*

例4.8 设G = ~,Q,C2=Q,求G和C2的归结式C12o

解：这里L| =「Q,L2=Q,通过归结原理可以得到

C】2 = NIL

例 4.9 设 G =「PVQ,C2=「Q,C3 = P,求 *G、C?、G* 的归结式 C123o

解：若先对G、Cz归结，可得到

然后对C】2和C.3归结，可得到

C123 = NIL

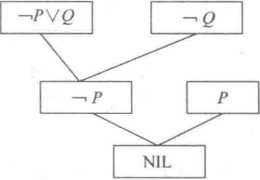
如果改变归结顺序，可以得到相同的结果，即其归结过 程是不唯一的。归结可用一棵树来表示，如例4. 9的归结过 程可用图4. 3来表ZK 0

图4. 3归结过程的树形表示

定理12 归结式C】2是其亲本子句G和C2的逻辑 结论。

证明：设C1=LVC；,C2=-LVC2关于解释/为真，只 需证明C|2= GVC关于解释I也为真。对于解释LL和 ~\*L中必有一个为假。

若L为假，则必有C；为真，不然就会使G为假，这将与前提假设G为真矛盾，因此只 能有C；为真。

同理，若为假，则必有C；为真。

因此，必有C12= C； VC；关于解释/也为真，即C是G和C?的逻辑结论。

这个定理是鲁滨逊归结原理中很重要的一个定理，由它可得到以下两个推论。

推论1设G和G是子句集S中的两个子句，C\*是G和C2的归结式，若用02代替

G和C2后得到新的子句集S"则由Si的不可满足性可以推出原子句集S的不可满足

性，即

的不可满足性DS的不可满足性

推论2设G和C2是子句集S中的两个子句，G?是G和C?的归结式，把加入S 中得到新的子句集S2,则S与S2的不可满足性是等价的，即

S2的不可满足性*DS*的不可满足性

推论1和推论2的证明可利用不可满足性的定义和解释*I*的定义来完成。

这两个推论说明，为证明子句集S的不可满足性，只要对其中可归结的子句进行归结， 并把归结式加入到子句集S中，或者用归结式代替它的亲本子句，然后对新的子句集证明 其不可满足性就可以了。如果经归结能得到空子句，根据空子句的不可满足性，即可得到原 子句集S是不可满足的结论。

在命题逻辑中，对不可满足的子句集S,其归结原理是完备的。这种不可满足性可用如 下定理描述。

定理4.3 子句集S是不可满足的，当且仅当存在一个从S到空子句的归结过程。

要证明此定理，需要用到海伯伦原理，正是从这种意义上说，鲁滨逊归结原理是建立在 海伯伦原理的基础上的。对此定理的证明从略，有兴趣者请查阅有关资料。最后还需要指 出，鲁滨逊归结原理对可满足的子句集S是得不出任何结果的。

归结原理给出了证明子句集不可满足性的方法。若假设F为已知的前提条件,G为欲 证明的结论，且F和G都是公式集的形式，根据前面提到的反证法：“G为F的逻辑结论， 当且仅当FA是不可满足的"，可把已知F证明G为真的问题转化为证明F A*「G为*不 可满足的问题。再根据定理4.1,在不可满足的意义上，公式集FA与其子句集是等价 的，又可把F A「G在公式集上的不可满足性问题转化为子句集上的不可满足性问题，这样，就可用归结原理来进行定理的自动证明。

应用归结原理证明定理的过程称为归结反演。在命题逻辑中，已知F,证明G为真的归 结反演过程如下：

1. 否定目标公式G,得
2. 把并入到公式集F中，得到｛F，「G｝。
3. 把｛F,「G｝化为子句集S。
4. 应用归结原理对子句集S中的子句进行归结，并把每次得到的归结式并入S中。 如此反复进行，若出现空子句，则停止归结，此时就证明了 G为真。

例4.10设已知的公式集为｛P,(FAQ)fR,(SVT)fQ,T｝,求证结论R。

解：假设结论*R*为假，即为真，将加入公式集，并化为子句集

S=｛P,「FV「Q V R,「S V Q，「T V Q，T，「R｝

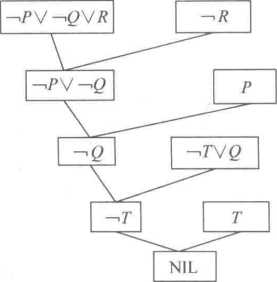
其归结演绎树如图4.4所示。在该树中，由于根部 出现空子句，因此命题*R*得到证明。

图4. 4 一个命题逻辑的归结演绎树

这个归结证明过程的含义为：开始假设子句集S中 的所有子句均为真，即原公式集为真，「択也为真；然后 利用归结原理，对子句集中含有互补文字的子句进行归 结，并把所得到的归结式并入子句集中；重复这一过程， 最后归结出空子句。根据归结原理的完备性，可知子句 集S是不可满足的，即开始时假设「択为真是错误的， 这就证明了択为真。

1. 谓词逻辑的归结

在谓词逻辑中，由于子句集中的谓词一般都含有变

元，因此不能像命题逻辑那样直接消去互补文字，而需要先用一个最一般合一对变元进行代 换,然后才能进行归结。可见，谓词逻辑的归结要比命题逻辑的归结复杂一些。

谓词逻辑中的归结可用如下定义来描述。

定义4.20设G和G是两个没有公共变元的子句，L］和L2分别是G和中的文 字。如果L］和「丄2存在最一般合一, CT,则称

C】2 = (｛Cg｝ 一 ｛Lg｝) u(｛。2招一妇｝)

为G和G的二元归结式，而L］和L2为归结式上的文字。

这里使用集合符号和集合的运算是为了说明问题的方便，即先将子句和Lm写成 集合的形式，并在集合表示下做减法和并集运算，然后再写成子句集的形式。

此外，定义中还要求G和c2无公共变元，这也是合理的。例如，G =P(n)，G = 「P(/(z)),而，=｛。，。2｝是不可满足的，但由于G和C2的变元相同，就无法合一了。没 有归结式，就不能用归结法证明S的不可满足性，这就限制了归结法的使用范围，如果对G 或G的变元进行换名，便可通过合一对G和*C2*进行归结。如例4.10,首先对G进行换 名，即G=「P(r3)),则可对G和C2进行归结，得到一个空子句，从而证明了 S是不可 满足的。事实上，在由公式集化为子句集的过程中，其最后一步就是进行换名处理。因此, 定义中假设G和C2没有相同变元是可以的。

例 4.11 设 G =P(q) VR(z),C2=「P3)VQ(。)，求 Ci2。

解：取Li = PJ),L2=则L,和的最一般合一是*o={aly}Q*根据定 义4.18可得

*c]2=*(而} 一 {星})u ({Cr} — {顷})

=({FS),R&)} — {P(Q}) U ((^P(a),Q(6)}-(-P(a)})

={R(力} U {Q(W) = {R(z),Q。)}

=R(z) V *Q(b)*

例 4.12 设 G *=P(x)* V Q(«) ,C2 = - *P(b)* V R(z),求o

解：由于G和C?有相同的变元了,不符合定义4. 18的要求。为了进行归结，需要修改 *C2*中变元的名字，令C2=「P。)VR(y),此时L1=P(jc),L2=-P(6),L1和L2的最一般 合一 *a={b/jc}*,则有

C12= ({Cg} — {Lg}) U*({C2a} - {L2j})*

=({P(b),Q(a)} — {F(。)}) U ({\*(D),R(jy)} — {「F(。)})

={Q(a)) (J *{R(y)}* = {Q(a) ,/?(>)}

=Q(a) V R(:y)

例 4.13 设 G =P(z) V - *Q〈b)* ,C2 = - *PCa)* V QG)V R(z)。

解：对G和通过最一般合一的作用可以得到两个互补对，但需要注意，求归结式不 能同时消去两个互补对，同时消去两个互补对的结果不是二元归结式。例如，在*a={a/x,* 力丁}下，若同时消去两个互补对，所得到的R(z)不是G和。2的二元归结式。

例 4.14 设 C] =PS)V P(/(a)) V QG) q =「P(;y) V *R(b),*求 C12。

解：对参加归结的某个子句，若其内部有可合一的文字，则在进行归结之前应先对这些 文字进行合一。本例的G中有可合一的文字P(z)与P(/(a)),若用它们的最一般合一 *a={f(a)/x}*进行代换，可得到

Cm= P(/(a)) V Q(/(a))

此时，可对Cp与C2进行归结。选择L1=P(/(a)),L2=-P(jz),L1和「L2d最一般合一 是*a={f(a)/y},^*可得到G和C?的二元归结式为

C12 = *R(b)* V Q(/(«))

在这个例子中，把G。称为G的因子。一般来说,若子句C中有两个或两个以上的文 字具有最一般合一“则称*Ca为子句C*的因子。如果*Ca*是一个单文字，则称它为C的单元 因子。应用因子概念，谓词逻辑中的归结原理给出如下定义。

定义4.21若G和是无公共变元的子句，则

1. G和的二元归结式。
2. G和C2的因子*C2a2*的二元归结式。
3. G的因子Gs和G的二元归结式。
4. G的因子Cm和C2的因子C2如的二元归结式。

这4种二元归结式都是子句G和C2的二元归结式，记为C】2。

例 4. 15 设 G *=P(y')* V P(/(x)) V Q(g(i)) (2 = - P(/(g(a))) V Q(6)，求 G?。

解：对G,取最一般合一。={/怎)/刃，得G的因子：

*Ci(j =* P(/(X)) V Q(g(z))

对G的因子和G的因子进行归结，可得到G和C2的二元归结式：

C12 = Q(g(g(a))) V *Q(b)*

对谓词逻辑，定理4. 2仍然适用，即归结式G2是其亲本子句G和C?的逻辑结论。用归 结式取代它在子句集S中的亲本子句，所得到的子句集仍然保持着原子句集S的不可满足性。

此外，对谓词逻辑，定理4. 3也仍然适用，即从不可满足的意义上说，一阶谓词逻辑的归 结原理也是完备的。

谓词逻辑的归结反演过程与命题逻辑的归结反演过程相比，其步骤基本相同，但每步的 处理对象不同。例如，在步骤(3)化简子句集时，谓词逻辑需要把由谓词构成的公式集化为 子句集；在步骤(4)按归结原理进行归结时，谓词逻辑的归结原理需要考虑两个亲本子句的 最一般合一\* O

例**4.16** 已知

*F*: ( V^r)(( 3 A B(3；)) -^ ( 3 >) (C(>) A *D(x,y)))*

*G：*「(DC(z)f (Vz)(3(A(w)f

证明：G是F的逻辑结论。

证明：先把G否定，并放入F中，得到的{F,「G}为

{(Vi)(( %)G4(W)A *B(y)) Oy)(C(y))* A *D(x,y))),*

「(「(DCO) f ( Vz)(

再把{F,「G}化为子句集，得到

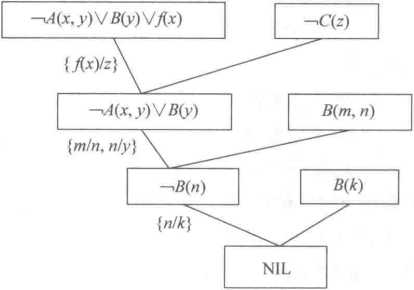
1. 「A(w)V「83)VCXg))
2. *->A(u9v)V* -'B(v)VD(u,/(w))
3. *-•C(z)*
4. *ACm,n)*
5. *B(k)*

其中，(1)、(2)是由F化出的两个子句,(3)-(5)是由化出的3个子句。

最后应用谓词逻辑的归结原理对上述子句集进行归结，其过程为

1. 「A&’jy) V「B3) (由(1)和(3)归结，取
2. -> *B(n)* (由(4)和(6)归结，取 *o= {m/x<,n/y})*
3. NIL (由(5)和(7)归结，取 *a={k/n})*

因此G是F的逻辑结论。上述归结过程可用图4. 5所示的归结树来表示。



为了进一步加深对谓词逻辑归结的理解，下面再给出两个经典的归结问题。 例4.17 “快乐学生”问题。

假设：任何通过计算机考试并获奖的人都是快乐的，任何肯学习或幸运的人都可以通 过所有考试，张不肯学习但是他是幸运的，任何幸运的人都能获奖。证明张是快乐的。

解：先将问题用谓词表示如下。

任何通过计算机考试并获奖的人都是快乐的：

(V 二)((Pass(z,computer) A Win&,prize)) f Happy(z))

任何肯学习或幸运的人都可以通过所有考试：

(V z) ( V y) (Study&) V Lucky(])—► Pass(z,j;))

张不肯学习但他是幸运的：

「 Study(Zhang) A Lucky(Zhang)

任何幸运的人都能获奖：

(V1)( Lucky(z) —► Win(z, prize))

目标“张是快乐的'‘的否定：

Happy(Zhang)

将上述谓词公式转化为子句集如下：

1. Pass(i,computer) V「Win(z,prize) V Happy(z)
2. 「StudyJ) VPass3，Q
3. Lucky(w) V Pass(z/,T7)
4. 「Study(Zhang)
5. Lucky(Zhang)
6. Lucky(s) V Win(s,prize)
7. - Happy(Zhang) (本子句为结论的否定)

按谓词逻辑的归结原理对此子句集进行归结，其归结反演树如图4.6所示。由于归结 出了空子句，这就证明了张是快乐的。

例4.18 “激动人心的生活”问题。

假设：所有不贫穷并且聪明的人都是快乐的。那些看书的人是聪明的。李明能看书且 不贫穷。快乐的人过着激动人心的生活。求证：李明过着激动人心的生活。

解：先将问题用谓词表示如下。

所有不贫穷并且聪明的人都是快乐的：

(V jc) ( ( Poor(j?) A Smart(o')) -► Happy(x))

那些看书的人是聪明的：

(V >) (Read(^) —► Smart(3/))

李明能看书且不贫穷：

Read(LiMing) *A* Poor(LiMing)

快乐的人过着激动人心的生活：

(V *z)* ( Happy(z) -► Exciting(z))

目标“李明过着激动人心的生活”的否定：

「Exciting(LiMing)

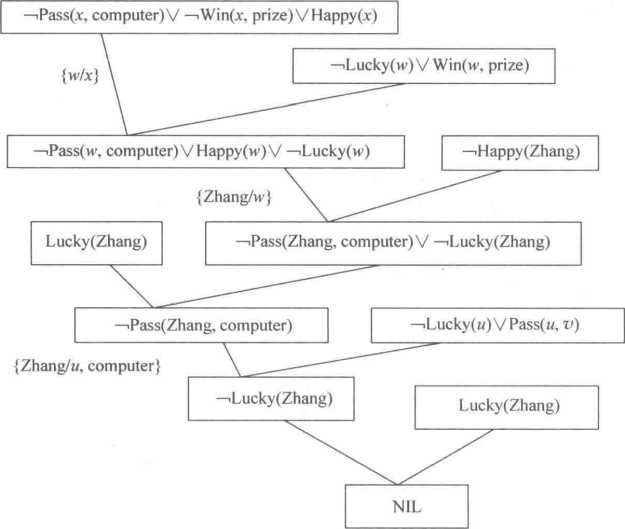
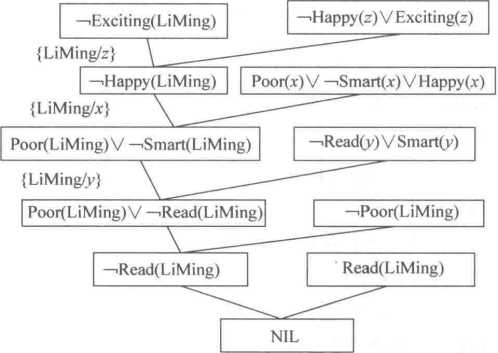


图4. 6 “快乐学生”问题的归结反演树

将上述谓词公式转化为子句集如下：

1. Poor(z) V「Smart(Z) V Happy(7)
2. 「Read(jz) V Smart(jO
3. Read(LiMing)
4. Poor(LiMing)
5. 「Happy(z) V Exciting(z)
6. 「Exciting(LiMing)

按谓词逻辑的归结原理对此子句集进行归结，其归结反演树如图4. 7所示。由于归结 出了空子句，这就证明了李明过着激动人心的生活。



**4. 4.3**归结演绎推理的归结策略

归结演绎推理实际上就是从子句集中不断寻找可进行归结的子句对，并通过对这些子 句对的归结，最终得出一个空子句的过程。由于事先并不知道哪些子句对可进行归结，更不 知道通过对哪些子句对的归结能尽快得到空子句，因此就需要对子句集中的所有子句逐对 进行比较，直到得出空子句为止。这种盲目地全面进行归结的方法不仅会产生许多无用的归 结式,更严重的是会产生组合爆炸问题。因此，需要研究有效的归结策略来解决这些问题。

目前，常用的归结策略可分为两大类，一类是删除策略；另一类是限制策略。删除策略 是通过删除某些无用的子句来缩小归结范围；限制策略是通过对参加归结的子句进行某些 限制来减少归结的盲目性，以尽快得到空子句。为了说明选择归结策略的重要性，在讨论各 种常用的归结策略之前，还是先提一下广度优先策略。

1. 广度优先策略

广度优先策略是一种穷尽子句比较的复杂搜索方法。设初始子句集为S，广度优先策 略的归结过程可描述如下：

1. 从S出发，对S。中的全部子句进行所有可能的归结，得到第一层归结式，把这些 归结式的集合记为slo
2. 对S及S中的全部子句进行所有可能的归结，得到第二层归结式，把这些归结式 的集合记为S2。
3. 对S°、S】及S2中的全部子句进行所有可能的归结，得到第三层归结式，把这些归 结式的集合记*为S3。*

如此继续，直到得出空子句或不能再继续归结为止。

例4.19设有如下子句集：

S = {「1(工)V R(z) U),「択3)V L(jO,「LS)} 用广度优先策略证明S为不可满足。

证明：从初始子句集S出发，依次构造S,S2,…，直到出现空子句结束。其归结树如 图4. 8所示。

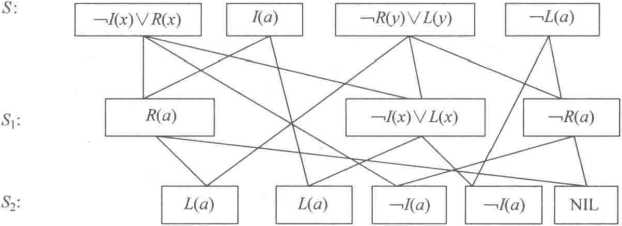


图4. 8广度优先策略的归结树

从这个例子可以看岀，广度优先策略归结岀了许多无用的子句，既浪费时间，又浪费空 间。但是，这种策略有一个有趣的特性，就是当问题有解时，保证能找到最短归结路径。因 此，它是一种完备的归结策略。广度优先对大问题的归结容易产生组合爆炸,但对小问题仍 是一种比较好的归结策略。

1. 支持集策略

支持集策略是沃斯(Wos)等人在1965年提出的一种归结策略。它要求每一次参加归 结的两个亲本子句中，至少应该有一个是由目标公式的否定所得到的子句或它们的后裔。 可以证明支持集策略是完备的，即当子句集为不可满足时，由支持集策略一定能够归结岀一 个空子句。也可以把支持集策略看成是在广度优先策略中引入了某种限制条件，这种限制 条件代表一种启发信息，因而具有较高的效率。

例**4.20**设有如下子句集：

S = {FCr) V R(z)U), 7(、)V L(j/),

式中，*「IM)NR(jc)*为目标公式的否定。用支持集策略证明**S**为不可满足。

证明：从S出发，其归结树如图4. 9所示。

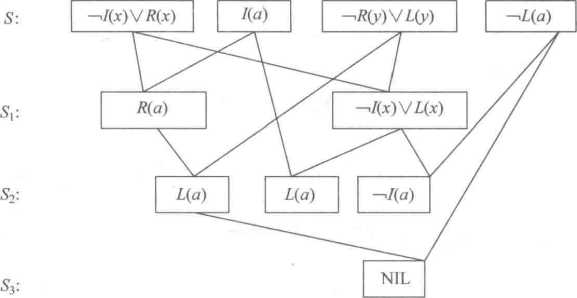


图4.9支持集策略的归结树

从上述归结过程可以看出，各级归结式数目要比广度优先策略生成的少,但在第二级还 没有空子句。就是说这种策略限制了子句集元素的剧增，却增加了空子句所在的深度。此 外，支持集策略具有逆向推理的含义，由于进行归结的亲本子句中至少有一个与目标子句有 关，因此推理过程可以看成是沿目标、子目标的方向前进的。

1. 删除策略

删除策略所依据的主要思想是：归结过程在寻找可归结子句时，子句集中的子句越多, 需要付出的代价就会越大。如果在归结时能把子句集中无用的子句删除，就会缩小搜索范 围，减少比较次数,从而提高归结效率。常用的删除方法有以下几种。

1)纯文字删除法

如果某文字L在子句集中不存在可与其互补的文字「L,则称该文字为纯文字。在归 结过程中，纯文字不可能被消除，用包含纯文字的子句进行归结也不可能得到空子句，因此 对包含纯文字的子句进行归结是没有意义的，应该把它从子句集中删除。对子句集而言，删 除包含纯文字的子句是不影响其不可满足性的。例如，对子句集

*S={PN Q\! R — QN R，Q，「R»*

式中，P是纯文字，因此可以将子句PVQVR从子句集S中删除。

1. 重言式删除法

如果一个子句中包含互补的文字对，则称该子句为重言式。例如：

P(x) V -P(x), *P&)* V *Q(工)*V --P(x)

都是重言式，不管P(z)的真值为真还是为假，P(z) V -P(x)和P(r) VQ(r) V」P(z)均 为真。

重言式是真值为真的子句。对一个子句集来说，不管是增加还是删除一个真值为真的 子句，都不会影响该子句集的不可满足性。因此，可从子句集中删除重言式。

1. 包孕删除法

设有子句G和G,如果存在一个置换”使得C9UC2,则称G包孕于C2。例如： P&)包孕于 F。)VQ(n)2= W/t}。

PCz)包孕于 *P(a)9a=(a/x}o*

P&)包孕于 P(O)VQ(N)S={〃Z}。

PCr) V QS)包孕于 *P(f(a))* V *Q(a)* V *RCy) {f(a)/x} o*

PCz) v Q(y)包孕于 P(«) V Q(G V *R(aj) 2= {a/x^/y} Q* 对子句集来说，把其中包孕的子句删除后，不会影响该子句集的不可满足性。因此，可 从子句集中删除那些包孕的子句。

1. 单文字子句策略

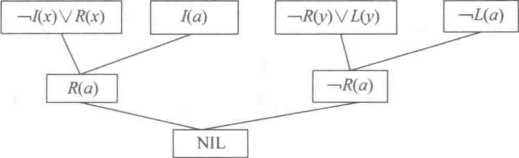
如果一个子句只包含一个文字，则称此子句为单文字子句。单文字子句策略是对支持 集策略的进一步改进，它要求每次参加归结的两个亲本子句中至少有一个子句是单文字 子句。

例，**21**设有如下子句集：

*S =* { --I(x) V R(h)J(q),槌(y) V L3),「L(。)}

用单文字子句策略证明S为不可满足的。

证明：从S岀发，其归结树如图4.10所示。



S:

S：

£：

图4. 10单文字子句策略的归结树

采用单文字子句策略，归结式包含的文字数将少于其亲本子句中的文字数，这将有利于 向空子句的方向发展，因此会有较高的归结效率。但这种策略是不完备的，即当子句集为不 可满足时，用这种策略不一定能归结出空子句。

1. 线性输入策略

这种策略要求每次参加归结的两个亲本子句中至少有一个是初始子句集中的子句。所 谓初始子句集是指开始归结时所使用的子句集。

例**4.22**设有如下子句集：

S = {「*1(工)*V R(i), U), *^R(y)* V L(y),「L(々)) 用线性输入策略证明S为不可满足。

证明：从S出发，其归结树如图4. 11所示。

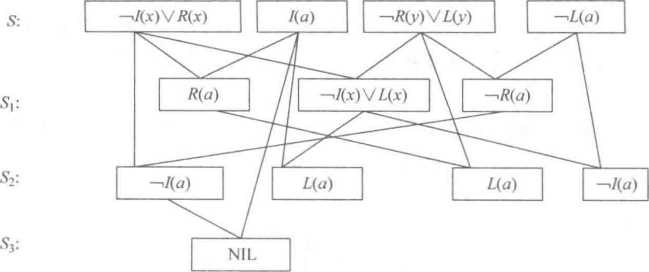


图4. 11线性输入策略的归结树

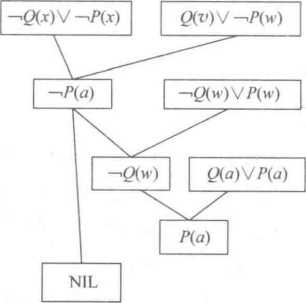


图4.12祖先过滤策略的归结树

线性输入策略可限制生成归结式的数目，具有简单和高效的优点。但是，这种策略也是 一种不完备的策略。例如,子句集

S= (Q(U)v P(a),「QJ) v P(G,「Q&) V「Pa),Q(；y) v *^P(y)}*

从S出发很容易找到一棵归结反演树.但不存在线性输入策略的归结反演树。

1. 祖先过滤策略

这种策略与线性输入策略有点相似，但是放宽了对子句的限制。每次参加归结的两个 亲本子句只要满足以下两个条件中的任意一个就可进行归结：

1. 两个亲本子句中至少有一个是初始子句集中的子句。
2. 如果两个亲本子句都不是初始子句集中的子句，则一个子句应该是另一个子句的 先辈子句。所谓一个子句c,是另一个子句a的先辈子句是指c2是由g与别的子句归结 后得到的归结式。

例**4.23** 设有如下子句集:

S={~>Q(«z) V P(^) *^Q(y)* V *P(y),*

*^Q(oj)* V P(G,QS) V *P(a))*

用祖先过滤策略证明S为不可满足。

证明：从*S*出发，祖先过滤策略归结树如图4. 12 所示。

可以证明祖先过滤策略也是完备的。

上面讨论了几种基本的归结策略，在实际应用中， 可以把几种策略结合起来使用。至于如何结合，需根据 具体需要而定。总之，在选择归结反演策略时，应考虑 其完备性和效率问题。

**4. 4.4**用归结反演求取问题的解

归结原理除了可用于定理证明外，还可用来求取问题的解，其思想与定理证明相似。下 面给出求取问题解的一般步骤：

1. 把问题的已知条件用谓词公式表示出来，并化为相应的子句集。
2. 把问题的目标的否定用谓词公式表示出来，并化为相应的子句集。
3. 对目标否定子句集中的每个子句，构造该子句的重言式(即把该目标否定子句和该 目标否定子句的否定之间再进行析取所得到的子句)，用这些重言式代替相应的目标否定子 句，并把这些重言式加入到前提子句集中，得到一个新的子句集。
4. 对这个新的子句集，应用归结原理求出其证明树，这时证明树的根子句不为空，称 这个证明树为修改证明树。
5. 用修改证明树的根子句作为回答语句，则问题的解就在此根子句中。

下面通过一些例子来说明如何利用归结反演求取问题的答案。

例土 24已知：张和李是同班同学；如果7和丁是同班同学，则7的教室也是y的教 室；现在张在302教室上课。

问：现在李在哪个教室上课？

解：首先定义谓词。

*CCx.y)* ： 7和了是同班同学。

At(«z.以)：z在"教室上课。

把已知前提用谓词公式表示如下：

C(zhang,li)

(VI)( V y)( V Z，)(C(w,y) A At(j-,u) -► At(jz,u))

At (zhang,302)

把目标的否定用谓词公式表示如下：

-« ( 3 v) At( li

把上述公式化为子句集：

CCzhang, li)

-1 C(x,>) V At(jr,w) V At(y,«)

At(zhang,302)

把目标的否定化成子句式，并用重言式代替：

At(li,v) V At(li,v)

把此重言式加入前提子句集中，得到一个新的子句集，对这个新的子句集，应用归结原 理求出其证明树。其修改证明树如图4. 13所示。该证明树的根子句就是所求的答案，即 “李在302教室气

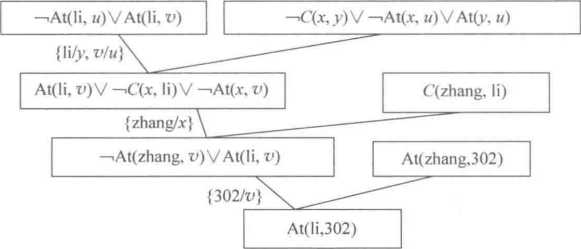


图4. 13例4.24的修改证明树

**4.5**基于规则的演绎推理

1. 4节讨论了归结演绎推理，需要把谓词公式化为子句集，尽管这种转化在逻辑上是等 价的，但是原来蕴含在谓词公式中的一些重要信息会在求取子句集的过程中丢失。例如，下 面的几个蕴含式

*-'A* A B -► C, A A「Cf B, AB V C，f A V C

都与子句

A V B V C

等价，但在AVBVC中，是根本得不到原逻辑公式中所蕴含的那些超逻辑的含义的。况 且，在不少情况下，人们大多希望使用那种接近于问题原始描述的形式来进行求解，而不希 望把问题描述转化为子句集。

规则是一种比较接近于人们习惯的问题描述方式，按照这种问题描述方式进行求解的 系统称为基于规则的系统，或者叫做基于规则的演绎系统。规则演绎系统的推理可分为正 向演绎推理、逆向演绎推理和双向演绎推理3种形式，本书主要讨论前面的两种。

**4.5.1**规则正向演绎推理

规则正向演绎推理和前面所讨论过的正向推理相对应。它是从已知事实出发，正向使 用规则，直接进行演绎，直至达到目标位置的一种证明方法。一个直接演绎系统不一定比反 演系统更有效，但其演绎过程容易被人们理解。

1. 事实表达式的与/或形变换

在一个基于规则的正向演绎系统中，其前提条件中的事实表达式是作为系统的初始综 合数据库来描述的。对事实的化简，只需转换成不包含蕴含符号-的与/或形式表示即可， 而不必化为子句集。把事实表达式化为非蕴含形式的与/或形的主要步骤如下：

（1） 利用规则Q消去蕴含符号。实际上，在事实表达式中很少有蕴含符 号出现，因为可把蕴含式表示成规则。

（2） 利用摩根定律及量词转化律把「移到紧靠谓词的位置，直到每个否定符号的辖域 最多只含有一个谓词为止。

1. 对所得到的表达式进行前束化。
2. 对辖域内的变元进行改名和标准化，使不同量词约束的变元有不同的名字，并利用 Skolem函数消去存在量词。
3. 消去全称量词，而余下的变元都被认为是全称量词量化的变元。
4. 对变元进行换名，使得各主合取式之间的变元名互不相同。

例如，表达式

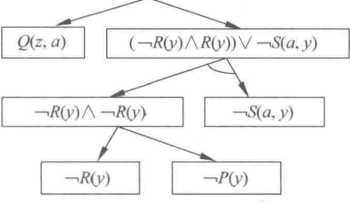
(D( Vy)(Q(jy,z) A「((R3)V P(>)) A S(w)))

可转化为

*Q(z9a)* A ((「R3)A *P(y))*V *S(a,y))*

这就是与/或形表示。

1. 事实表达式的与/或树表示

事实表达式的与/或形可用一棵与/或树表示出来。例如，对上例所给出的与/或形，可 用如图4.14所示的与/或树来表示。在该图中，每 个节点表示该事实表达式的一个子表达式，子表达 式之间的与、或关系规定如下：

0z, a)A((-/?(y))V 7心))

图4. 14 一个事实表达式的与/或树表示

当某个表达式为*k*个子表达式的析取时，如 *EZE/N… m，*其中的每个子表达式*Ej*均被表 示为VE2 V・・・V顼的后继节点，并由一个4线 连接符(将指向这*k*个后继节点的边连接起来的一 个圆弧)将这些后继节点都连接到其父节点，即表 示成与的关系。

当某个表达式为*k*个子表达式的合取时，如

*E\AEzA…AE-*其中的每个子表达式*E,*均被表示*为E\ l\E-2・・皿*的一个单一的后继 节点，并由左个单线连接符(即指向单个后继节点的有向边)将这些后继节点连接到其父节 点，即表示成或的关系。

这样，与/或树的根节点就是整个事实表达式，端节点均为事实表达式中的一个文字。 有了与/或树的表示，就可以求出其解树(结束于文字节点上的子树)集。并且可以发现，事 实表达式的子句集与解树集之间存在一一对应的关系，即解树集中的每个解树都对应着子 句集中的一个子句。例如，如图4. 14所示的与/或树有3个解树，分别对应以下3个子句： Q(z»a)

*-'R(y')* V S(a»>)

「P(j/) V「*S(a.y)*

与/或树的这个性质很重要。它可以把与/或树看成是对子句集的简捷表示。不过，表 达式的与/或树表示要比子句集表示的通用性差一些，原因是与/或树中的合取元素没有进 一步展开，因此不能像在子句集中那样对某些变量进行改名，这就限制了与/或树表示的灵 活性。例如，上面的最后一个子句，在子句集中，其变量丁可全部改名为丁，但无法在与/或 树中加以表示，因而失去通用性，并且可能带来一些困难。

还需要指岀，这里的与/或树是作为综合数据库的一种表示形式，其中的变量受全称量 词的约束，而在可分解产生式系统中所描述的与/或树则是搜索过程的一种表示形式，这两 种与/或树之间有着不同的目的和含义，因此应用时应加以区分。

1. 规则的与/或形变换

在与/或形正向演绎系统中，是以正向方式使用规则(F规则)对综合数据库中的与/或 树结构进行变换。为简化这种演绎过程，通常要求F规则应具有如下形式：L-W,式中，L 为单文字,W为与/或形公式，需要注意，这里假定出现在蕴含式中的任何变量全都受全称 量词的约束，并且这些变量已经被换名，使得它们与事实公式和其他规则中的变量不同。

如果领域知识的规则表示形式与上述要求不同，则应将它转换成要求的形式。其变换 步骤如下：

1. 暂时消去蕴含符号一。设有如下公式：

(V Jf) ((( 3 j/) ( V *z)* (P(j?/»z) ) f ( V

运用等价关系Q可将上式变为

(V b) (「(( - y)( V z) ,z) ) V ( V ")A(«r,〃)))

1. 把否定符号「移到紧靠谓词的位置上，使其作用域仅限于单个谓词。通过使用摩 根定律及量词转化律可把上式转化为

(V z)(( V 丁)( m z)(「(P(zV ( V *u)Q(t9u)))*

1. 引入Skolem函数，消去存在量词，上式变为

(V ^r)(( V *y) CPCjc f(jc 9y))')* V ( V u)Q(j: 9 u))

1. 化成前束式，消去全部全称量词，上式变为

P(jr9y,f(jr,y) ) V Q(z,〃)

此式中的变元都被视为受全称量词约束的变元。

1. 恢复蕴含式表示。利用等价关系」PVQoPf Q将上式变为

P(i，;y，顶(z，v))f Q(z,")

在前面对F规则的要求中之所以限制其前件为单文字，是因为在进行正向演绎推理时 要用F规则作用于表示事实的与/或树，而该与/或树的叶节点都是单文字，这样就可用F 规则的前件与叶节点进行简单匹配。对非单文字情况，若形式为L VL2-W,则可将其转 换成与之等价的两个规则Li->W与*L2^W*进行处理。

1. 目标公式的表示形式

与/或树正向演绎系统要求目标公式用子句形式表示。如果目标公式不是子句形式，则 要将其转化成子句形式。把一个目标公式转化为子句形式的步骤与4. 4节所述的化简子句 集步骤类似，但Skolem化的过程不同。目标公式的Skolem化过程是简化子句集的Skolem 过程的对偶形式，即把目标公式中属于存在量词辖域内的全称变量用存在量化变量的 Skolem函数来代替，使经Skolem化的目标公式只剩下存在量词，然后再对析取元进行变量 替换，最后把存在量词消掉。

例如，设目标公式为

(不、)(V 飞)(P&/) V *Q(x,y))*

用Skolem函数消去全称量词后有

(3y)(P(/(>),y) V Q(g”))

进行变量换名，使每个析取元具有不同的变量符号，于是有

(m jO(P(，(：y)，少 V ( 3 *z)QC*/(z) »2：))

最后，消去存在量词得

(P(/(>)^)V Q(f(z),z))

这样，目标公式中的变量都假定受存在量词的约束。

1. 规则正向演绎推理过程

规则正向演绎推理过程是从已知事实出发，不断运用F规则，推岀欲证明的目标公式 的过程。即先用与/或树把已知事实表示出来，然后再用F规则的前件和与/或树的叶节点 进行匹配，并通过一个匹配弧把匹配成功的F规则加入到与/或树中，依次使用F规则，直 到产生一个含有以目标节点为终止节点的解图为止。

下面分命题逻辑和谓词逻辑两种情况来讨论规则正向演绎过程。

1)命题逻辑的规则演绎过程

由于命题逻辑中的公式不含变元，因此其规则演绎过程比较简单。

例4. 25设已知事实为

A V B

F规则为

n ： A f C A D

r2: B f E *A G*

目标公式为

C v G

证明：先将已知事实用与/或树表示出来，然后再用匹配弧把门和a分别连接到事实 与/或树中与ri和立前件匹配的两个不同端节点，由于出现了以目标节点为终节点的解 树，故推理过程结束。这一过程如图4.15所示。在该图中，粗箭头表示匹配弧，它相当于一 个单线连接符。

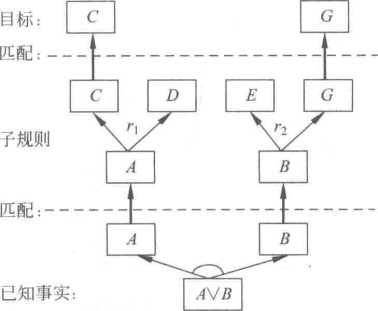


图4. 15命题逻辑的规则演绎过程

为了验证上述推理的正确性，下面再用归结演绎推理给予证明。由已知事实、规则及目 标的否定可得到如下的子句集：

*{A* V B, A V C, -1 A V D, ->B V V GlClG}

其归结树如图4.16所示。

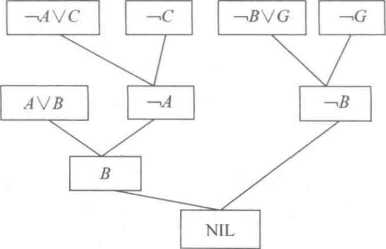


图4. 16例4. 25的归结树

由图4. 16可以看出，用归结演绎推理对已知事实、F规则集目标的否定所得到的子句 集进行归结，得到了空子句NIL,从而证明了目标公式。它与正向演绎推理所得到的结果是 一致的。

在谓词逻辑情况下，由于事实、F规则及目标中均含有变元，因此，其规则演绎过程还需 要用最一般合一对变量进行置换。

例**126**设已知事实的与/或形表示为

*P(x,y)* V (Q(z) A R(p,y))

F规则为

*P(u9v)* —► S(u) V *N(p)*

目标公式为

S(a) V *N(b)* V Q(c)

其规则演绎过程如图4. 17所示。

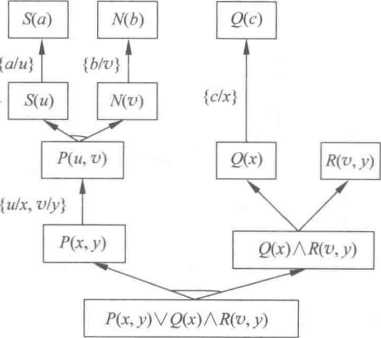


图4. 17谓词逻辑的规则演绎过程

**4.5.2**规则逆向演绎推理

基于规则的逆向演绎推理是从目标公式的与/或树出发，逆向使用规则(B规则)对目标 表达式的与/或树进行变换，直到得出含有事实节点的一致解图为止。

1. 目标公式的与/或形变换

逆向系统中的目标公式釆用与/或形表示，而不像正向系统那样需要化成子句形式。在 把任意目标公式化为与/或形表示时,采用正向系统中对事实表达式处理的对偶形式。即要 用存在量词约束变元的Skolem函数来替换由全称量词约束的相应变元，消去全称量词，再 消去存在量词，并进行变元换名，使主析取元之间具有不同的变元名。

例如，有如下目标公式：

O^)(Vjt)(P(x)->(Q(^,>) A「(R&) A S(y))))

Skolem化后为

V(Q(e)a A OR(处))v *-^S(y)))*

变元换名后为

「Pg)) v*(Q(f(y)^y)*A (須(少)V *S(y)))*

1. 目标公式的与/或树表示

目标公式的与/或形也可用与/或树表示出来，但其方法与正向演绎推理中事实的与/或 树表示略有不同。它规定子表达式之间的析取关系用单线连接符连接，表示为或的关系。 而子表达式之间的合取关系则用&线连接符连接，表示为与的关系。例如，对于上述目标公 式的与/或树，可用如图4. 18所示的与/或树来表示。

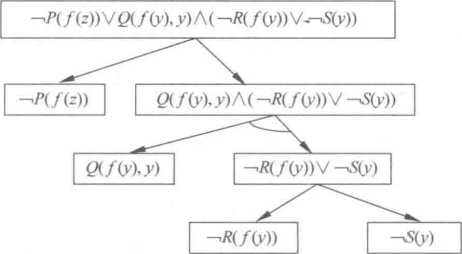


图4. 18目标公式的与/或树表示

在该图中，若把叶节点用它们之间的合取及析取关系连接起来，就可得到原目标公式的 3个子目标：

Q(六少,少A「《(脸)

*Q(f(y),y)*A「S3)

可见，子目标是文字的合取式。

**3. B**规则的表示形式

B规则的表示形式为

Wf L

式中，前项W为任一与/或形公式，L为一单文字。当B规则为W-L AL?时，则可化简为 两条规则W-心和Wf L?进行处理。

**4.**已知事实的表示形式

反向演绎系统的事实表达式均限制为文字合取式，即形如

Fj A F2 A A *F„*

式中，每个F,(z=l,2,・・・，〃)都为单文字，且都可以单独起作用，因此可表示为如下集合 形式：

｛F,,F2，-,FJ

1. 规则逆向演绎推理过程

规则逆向演绎推理过程是从目标公式的与/或树出发，不断用B规则的右部和与/或树 的叶节点进行匹配，并将匹配成功的B规则用最一般合一匹配弧加入到与/或树中，直到产 生某个终止在事实节点上的一致解图为止。所谓一致解图是指在推理过程中所用到的置换 应该是一致的。

在谓词逻辑下，用B规则的右部和与/或树的叶节点进行匹配时，需要使用它们之间的 最一般合一。下面通过一个例子来说明逆向演绎系统的推理过程。

例4.27设有如下事实和规则。

|  |  |
| --- | --- |
| 事实：  *f}* : DOG(Fido)  /2：「BARKS(Fido)  /3： WAGS-TAIL (Fido)  /4： MEOWSC Myrtle) 规则： | Fido是一只狗  Fido是不叫的  Fid。摇尾巴  猫咪的名字叫Myrtle |

门：(WAGS-TAIL(^) A DOG(幻))—FRIENDLYC^ ) 摇尾巴的狗是温顺的狗

r2： FRIENDLY(x2) A「BARKS(工2)一「AFRAIDCy?，互)

温顺又不叫的东西是不值得害怕的

r3： DOG&3)f ANIMAL(a) 狗为动物

r4： CAT(jc4)-\*ANIMAL(x4) 猫为动物

r5： MEOWS&5)f ANIMAL&5)猫咪是猫

问题：是否存在这样的一只猫和一只狗，使得这只猫不害怕这只狗？ 该问题的目标公式为

(3^)(3^)(CAT(^) A DOG(y) A --AFRAID(x^))

该目标公式经过变换后得到

CAT(z) A D()G(、)A「AFRAUMw)

用逆向推理求解的演绎过程如图4. 19所示。

该推理过程所得到的解图是一致解图。此解图中有8条匹配弧，每条匹配弧上都有一 个置换。其中，终止在事实节点上的置换为｛Myrtle/z｝和｛ Fido/刃。把它们应用到目标公 式，就得到了该问题的解：

CAT(Myrtle) A DOG(Fido) A「AFRAID(Myrtle,Fido)

它表示：有这样一只名叫Myrtle的猫和一条名叫Fido的狗，这只猫不怕那只狗。

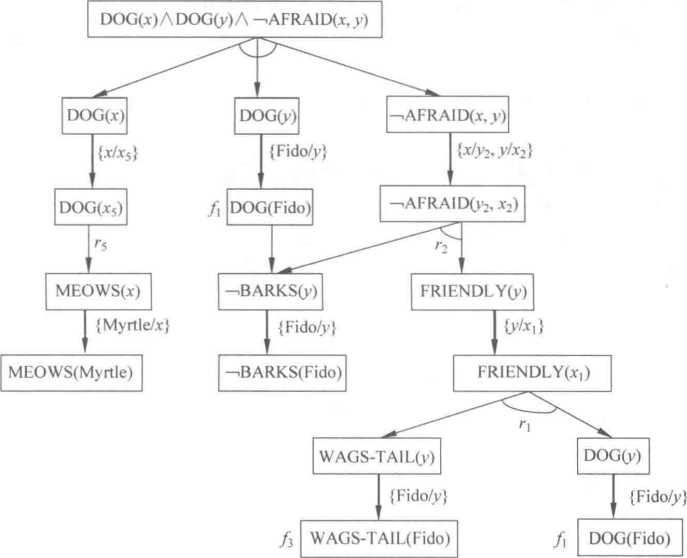


图4. 19用逆向推理求解的演绎过程

**4.6** 习题

1. 推理一般有几种方式？每种方式有什么特点？
2. 什么是正向推理？请画出正向推理一般过程的流程图。
3. 什么是逆向推理？请画出逆向推理一般过程的流程图。
4. 什么是双向推理？在哪些情况下需要进行双向推理？双向推理的主要问题是什么？
5. 把下列谓词公式化成子句集：
6. ( Vz)(。)(尸(工，jOAQ&,少)
7. ( Vz)( \/了)(卩(二，>)->Q(x, y))
8. ( V^r)( 3 ;y)(P(z, *y)* V (Q(z, *:y)fR(z, y)))*
9. *( V jt) ( V y)* ( 3 z) *(P(j:9* y)fQ&, y) *\/ R(.x, z))*
10. 判断下列子句集中哪些是不可满足的：
11. { "VQ,「Q,P, ”}
12. {PVQ, "VQ,PV「Q, "V「Q}
13. {P(jy) VQ(y),VRS)}
14. {->PG) VQ(z),「P(y) VR3), *p(a), S(a),* -'S(z) V
15. {-P(x) VQ(/(x),a), *~'P(h(y))\/Q(f(h(y))9* a) V *^P(z))*
16. {P(z) VQ(r) VRGr),「P(w)VR(y), *^Q(a)9「R(b)}*
17. 对下列各题分别证明G是否为F、F|、F2的逻辑结论：
18. F： (L)(\*)(P&, >))

G： (V3^)Ox)(PCjr,少)

1. F： ( Vz)(F(z) A(Q(々)VQ。)))

G： (D(P(z)AQ(i))

1. F： (3^)(3>)(P(/(^)) A(Q(/(j/)))

G： P(r(Q) AP(jO AQ3)

1. Fi： ( Vz)(F(z)f ( V；y)(Q(：y)f "(2)))

F2：(mz)(P&) A( Vw)(R3)fL&.：y)))

G： ( Vz)(R(Qf「Qa))

1. Fi： ( Vz)(P&)-> (Q&)/\R(i)))

*F2i* (D(P(z) AS&))

*G：* ( - «r) (S(z) A R(z))

1. 设已知
2. 能阅读的人是识字的。
3. 海豚不识字。
4. 有些海豚是很聪明的。

请用归结演绎推理证明：有些很聪明的人并不识字。

1. 应用归结演绎推理求解下列问题：

设张三、李四和王五3人中有人从不说真话，也有人从不说假话。某人向这3人分别提 出同一个问题：“谁是说假话者？”张三答：“李四和王五都是说假话者。”李四答：“张三和王 五都是说假话者。”王五答：“张三和李四中至少有一个说假话者。”求谁是说假话者，谁是说 真话者。

不确定性推理

**5.1**概述

在现实生活中遇到的问题通常都具有不确定性，能够进行精确描述的问题只占较少的 一部分。对于这些不确定性的问题,若采用前述确定性推理方法显然是无法解决的。因此， 为满足现实世界问题求解的需求，人工智能需要研究不确定性推理方法。可以将知识分成 确定性知识和不确定性知识，不确定性是智能问题的一个本质特征，是建立在不确定性知识 和证据的基础上的推理。而知识的不确定性主要体现在两个方面：随机性和模糊性。

随机性主要由概率知识来体现，因此研究随机性即通过概率和马尔可夫链等知识进行。 模糊性与随机性一样，在生活中几乎无处不在，我们常说的“高、矮、长、短、大、小”都体现了 一种模糊性现象，研究模糊性则通过模糊集理论，该理论以及模糊推理在5. 6节进行阐述。

**5.1.1**为什么要采用不确定性推理

采用不确定性推理是客观问题的需求，其原因主要包括以下几个方面。

1. 所需知识不完备、不精确。在很多情况下，解决问题所需要的知识往往是不完备、 不精确的。所谓知识的不完备是指在解决某一问题时，不具备解决该问题所需要的全部知 识。例如，医生在看病时，一般是从病人的部分症状开始诊断的。所谓知识的不精确是指既 不能完全确定知识为真，又不能确定知识为假。例如，专家系统中的知识多为专家经验，而 专家经验又多为不确定性知识。
2. 所需知识描述模糊。知识描述模糊是指知识的边界不明确。例如，平常人们所说 的“很好”“好”“比较好”“不很好”“不好”“很不好”等概念，其边界都是比较模糊的。那么，当 用这类概念来描述知识时，所描述的知识当然也是模糊的。例如“如果王刚这个人比较好， 那么我就把他当作好朋友”所描述的就是比较模糊的知识。
3. 多种原因导致同一个结论。在现实世界中，可由多种不同原因导出同一结论的情 况有很多。例如，引起人体低烧的原因至少有几十种，医生在看病时只能根据病人的症状， 低烧的持续时间以及病人的体质、病史等做岀猜测性的推断。
4. 解决方案不唯一。现实生活中存在的问题一般都存在着多种不同的解决方案，并 且这些方案之间又很难绝对地判断其优劣。对于这种情况，人们往往是优先选择主观上认 为相对较优的方案，这也是一种不确定性推理。

总之，在人类的知识和思维行为中，确定性只是相对的，而不确定性才是绝对的。人工 智能要解决这些不确定性问题，必须采用不确定性的知识表示和推理方法。

**5.1.2**不确定性推理要解决的问题

可将不确定性推理要解决的问题概括为以下5类。

1. 如何进行知识和证据表示的问题

不确定性表示要解决的问题包括知识的不确定性表示和证据的不确定性表示。

不确定性推理中，知识是否能够很好地被表示直接影响推理的运行效率。一般地，用数 值来刻画知识的不确定性，该数值称为知识的静态强度或者知识的可信度，描述了知识的不 确定性程度。静态强度可以用概率、可信度或者隶属度来表示，如果用概率或者隶属度来表 示，则其取值范围是［0,口。取值越接近1,说明确定性程度越“高”；反之，取值越接近0,说 明确定性程度越“低”。通常情况下，静态强度是由领域专家给出的或是由实验统计方法得 到的。

证据通常包括两部分，其一是求解问题时已有的初始证据；其二是将求解问题中得到 的中间结果放入综合数据库，作为后续推理的证据。为了使推理能对不确定性进行处理，证 据的不确定性表示通常与知识的不确定性表示一致，即证据的不确定性表示也是用数值来 表示，代表相应事实的不确定性程度，将之称之为动态强度。

1. 如何进行匹配的问题

在确定性推理中，当从规则库中取出的某规则的前提与综合数据库中的已知事实相一 致时，便认为是匹配成功。只有当匹配成功时，相关的规则才会被激活使用。在不确定性推 理过程中，同样要解决匹配问题，但由于知识和证据都是不确定的，所以确定性匹配方式不 能釆用。目前，常用的方法是事先设定一个阈值，用来衡量知识和证据的相似程度，如果知 识和证据的不确定性程度在阈值限度内，则认为可以进行匹配，否则不能匹配。这里比较关 键的地方就是如何算出相似程度，往往要视实际具体情况的不同而采用不同的算法。

1. 如何进行证据组合的问题

在不确定性推理过程中，证据可以是复杂的组合条件。此时则需要有合适的算法来计 算证据组合的不确定性。即已知证据耳 和E2的不确定性值C(E)和C(EQ,求证据E和 *E2*的析取和合取的不确定性，即定义函数£和，2,使得

*CCE.* A 或)=fl(C(E1)9C(E2))

*C(Ei* V E2) = /2(C(E1),C(E2))

具体常用来计算证据组合不确定性的方法有以下3种：

1. 最大最小方法：

C(E, A E2)= min(C(E|),C(EQ〉

C(Ei V *E2*) = max(C(E]) ,C(E2))

1. 概率方法：

*C(Ei* A E2) = C(Et) XC(E2)

*C(Ei* V E2) = C(Ej) + C(E2) —C(Ej) *X C(E2)*

1. 有界方法。
2. 不确定性的遗传问题

在不确定性推理中，存在两个主要的问题：一是如何用证据的不确定性去得到结论的 不确定性；二是如何在推理中把初始证据的不确定性传递给最终结论。

一方面，按照某种算法由证据和知识的不确定性计算出结论的不确定性，至于不确定性 推理方法的处理方式各有不同；另一方面，不同不确定性推理方法的处理方式基本相同，都 是把当前推出的结论及其不确定性作为新的证据放入综合数据库，方便以后推理使用，这样 就实现了将不确定性传递给后面的结论，重复这样的过程，可以将不确定性传递给最终 结论。

1. 如何合成结论的问题

在不确定性推理过程中，可能出现由多个不同知识推出相同的结论，且不确定性程度不 同的情况，需要采用相应算法对这些不同的不确定性进行合成，求岀该结论的综合不确 定性。

**5.1.3**不确定性推理类型

不确定推理方法的研究主要沿着两条不同的路线发展：模型方法和控制方法。

模型方法是对确定性推理框架的一种扩展。模型方法把不确定性证据和不确定性知识 分别与某种度量标准对应起来，并且给出了更新结论不确定性的算法，从而构成相应不确定 性推理的模型。一般来说，模型方法与控制策略无关，即无论使用何种控制策略，推理的结 果都是唯一的。

模型方法又分为数值方法和非数值方法两大类。数值方法是对不确定性的一种定量表 示和处理方法。对于该类情况下，根据其所依据的理论可以将其分为两种不同的类型。一 类是基于概率的方法，如确定性推理、主观贝叶斯方法、证据理论等；另一种是基于Zadeh 教授提出的模糊集理论及其在此基础上发展的可能性理论。非数值方法是指除数值方法外 的其他各种处理不确定性的方法，如框架推理、语义网络推理、常识推理等。

控制方法主要在控制策略一级处理不确定性。其特点是通过识别领域中引起不确定性 的某些特征及相应的控制策略来限制或者减少不确定性对系统产生的影响。这类方法没有 处理不确定性的统一模型，其效果极大地依赖于控制策略。目前常用的控制方法有启发式 搜索和相关性制导回溯等。

本章只对模型方法展开讨论，有兴趣的读者可自行查阅文献了解控制方法。

**5.2** 概率基础

概率论的对象是随机现象。在概率中，把随机现象的某些样本点组成的集合称随机事 件，简称事件，而概率描述的就是随机事件发生的可能性大小。下面给岀概率的公理化 定义。

定义5.1设口为一个样本空间，F为卩的某子集组成的一个事件域，P(A)是F上的 一个实值函数，对于任意一事件A6F,称P(A)为事件A的概率，如果满足以下3条性质：

1. 非负性。对于任意AGF,都有P(A)20。
2. 正则性。p(n)=io
3. 可列可加性。若A.A2,…，&.,・・・互不相容，有

4-00 +8

P U & =

1 1=1

在概率的学习中，通常还会给出概率的具体定义形式，包括统计概率、古典概率、集合概 率、条件概率等。

定义5.2(统计概率)若在大量重复试验中.事件A发生的频率稳定地接近于一个固 定的常数它表明事件A出现的可能性大小，则称此常数*p*为事件A发生的概率，记为 P(A),即

*p* = P(A)

定义5.3(古典概率)设一种试验有且仅有有限的N个等可能结果，即N个基本事 件，而A事件包含着其中的L个可能结果，则称右为事件A的概率，记为P(A)O

|  |  |
| --- | --- |
|  | 心)=若 |
| 定义**5.4(**集合概率) | 假设。是集合型随机试验的基本事件空间，F是。中一切可测 |

集的集合，则对于F中的任意事件A的概率P(A)为A与卩的体积之比，即

|  |  |
| --- | --- |
|  | *P(A) v(n)* |
| 定义5.5(条件概率) | 把事件B已经岀现的条件下事件A发生的概率记作*P(A\B).* |

并称之为在*B*岀现的条件下A出现的条件概率。

|  |  |
| --- | --- |
| 定理5. 1(加法定理) | 两个不相容(互斥)事件之和的概率等于两个事件概率之和，即  P(A + B) = P(A)+P(B) |

两个互逆事件A和AT的概率之和为1。即当A+AT=O,且A与AT互斥，则P(A) + P(AT) = 1,或 P(A) = 1 — P(AT)。

若A、B为两个任意事件，则P(A + B) = F(A) + P(B)-P(AB)成立。此定理可推广 到3个以上事件的情形：

P(A + B + C) = P(A) + P(B) + P(C) 一 P(AB) - P(BC) - P(CA) + P(ABC)

|  |  |
| --- | --- |
| 定理5.2(乘法定理)  概率的乘积，即 | 设A、B为两个不相容的非零事件，则其乘积的概率等于A和B |

P(A ・ *B)* = P(A)・ *P(B)* 或 *P(A* ・ B) = *P(B) - P(A)*

设A.B为两个任意的非零事件，则其乘积的概率等于A(或B)的概率与在A(或B)出 现的条件下B(或A)出现的条件概率的乘积。

P(A - B) = P(A)P(B I A) 或 P(A ・ *B) = P(B)P(A | B)*

此定理可以推广到3个以上事件的乘积情形，即当z个事件的乘积P(A1A2-Az\_1)> 0时，则乘积的概率为

*P(A1A2-Az) =* P(AQ - *P(A2* I AQ - *P(A3* I A】A2)..・P(Az I

当事件独立时，则有

*P(A}A2—Az)* = P(Ai). P(A2) - P(&)...P(A=)

下面给出还需要了解的其他概率知识。

1. 先验概率。指根据历史资料或主观判断所确定的各事件发生的概率，该类概率没 能经过实验证实，属于检验前的概率，所以称之为先验概率。先验概率一般分为两类，一是 客观先验概率，是指利用历史资料计算得到的概率；二是主观先验概率，是指在无历史资料 或历史资料不全的时候，只能凭借人们的主观经验来判断取得的概率。
2. 后验概率。一般是指利用贝叶斯公式，结合调查等方式获取了新的附加信息，对先 验概率进行修正后得到的更符合实际的概率。
3. 全概率公式。如果影响事件A的所有因素由，国，…,场 满足：3 -旦= 0,(z.尹

*t*

丿3 = 1,2,・・・，〃,顶=1,2,・・・，〃)，且、P(B)= 1 ,则必有

1 = 1

*t*

P(A) =、P(3)P(A I *B,)*

1. 贝叶斯公式。也称后验概率公式或逆概率公式。设先验概率为P(B),调查所获 的新附加信息为PCAJB,),其中1=1,2,…，么顶= 1,2,・・・/。则贝叶斯公式计算的后验概 率为

P(B. *1 AQ =* ~~了(旦出貞」B\*)~~

史P(3)P(& | 3)

r=I

**5.3**主观贝叶斯方法

**5.3.1**不确定性的表示

下面介绍在主观贝叶斯方法中如何进行知识和证据的不确定性表示。

1. 知识的不确定性表示

在主观贝叶斯方法中，采用产生式来表示知识，其规则为

IF *E* THEN (LS,LN)H

其中，E是知识的前提条件，可以是简单条件，也可以是复合条件。H是结论。(LS,LN)表 示知识的静态强度，LS表示充分性度量，即表示E对H的支持度量；LN表示必要性度量, 即表示对H的支持度量。它们的表现形式分别为

T §= P(E 丨 H)

*P(E* I」H)

P(「E | H)

LN

1-P(E | H)

1 — *P(E* |「*H)*

I「H)

LS和LN的取值范围均为［0,+8)。

1. 证据的不确定性表示

在主观贝叶斯方法中，证据的不确定性是用概率或几率来表示的，二者之间的关系为

0,

OO ,

(0, + 8),

当E为假时 当E为真时 其他

**5.3.2**组合证据不确定性的计算

组合证据包括合取和析取两种基本情形。 当组合证据是多个单一证据的合取时危卩

*E = E}* AND *E2* AND …AND *En*

如果已知

P(E | S),P(E2 I S),・・・，P(E” I S)

F(E | *S)* = min{P(E1 | S),P(E2 | S),…，P(旦 | S)} 当组合证据是多个单一证据的析取时，即

E = E OR *E2* OR …OR *En*

如果已知

P(Ej I S),P(E2 I S),…，P(E〃 I S)

*P(E* I *S)* = max(P(El | *S),P(E2* | S),…，P(E” | *S)}*

**5.3.3**不确定性的传递算法

主观贝叶斯方法推理的任务就是根据E的概率P(E)及LS和LN的值，把*H*的先验 概率P(H)或先验几率O(H)更新为后验概率或后验几率。由于一条知识所对应的证据可 能为真，也可能为假，还可能既非为真又非为假，因此，把H的先验概率或先验几率更新为 后验概率或后验几率时，需要根据证据的不同情况去计算其后验概率或几率。

1. 证据肯定为真

当证据E肯定为真时，有P(E) = P(E|S) = 1O将*H*的先验几率更新为后验几率的公 式为

0( H I E) = LSXO(H)

如果把*H*的先验概率P(H)更新为后验概率P(H|E),则可以得到几率和概率的对应 关系：

*P( ir* I *卩)=* LS X P(H)

1. (LS—1) XP(H) + 1
2. 证据肯定为假

当证据E肯定为假时，有*P(E) = P(E\S) = O.P(「E) = L* 将*H*的先验几率更新为 后验几率的公式为

0(H |*「E)* = LNXO(H)

如果把*H*的先验概率P(H)更新为后验概率P(H|E),则可以得到几率和概率的对应 关系：

1. 证据既非为真又非为假

当证据既非为真又非为假时，这时因为*H*依赖于证据E,而E基于部分证据S,则 P(H|S)是依赖于*S*的似然性。根据下面公式来计算证据不确定性的传递问题：

P(H | E) = P(H I E) XP(E I S) + P(H |*「E)* XP(「E | S)

这里不再讨论P(E|S) = 1和P(E|S)=O两种情况，只讨论剩下的两种情况：

1. P(E|S) = P(E)O此时E与S无关，根据全概率公式可得

P(H | S)= F(H | E) XP(E I S) + P(H |「E) XP(「E 丨 S)

=P(H I E) X P(E)+ F(H I「E) X POE) = P(H)

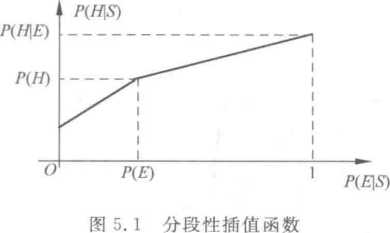
1. P(E|S)为其余情况。在此种情况下，依据3个特殊点(O,P(E),1)的分段线性插 值函数求得。该分段线性插值函数P(H|S)如图5.1所示，函数的解析表达式为

P(H |+ ~~「E)~~xp(E| s), 0<P(E I S) < P(E)

*l\b)*

P(H) + x [p(E I S)-P(E)], P(E) VP(E| S) < 1

1 — *1*



**5.3.4**结论不确定性的合成

现有77条知识都支持同一结论H,且每条知识的前提条件分别是〃个相互独立的证据 *E】，E2，・・・.E“,*而这些证据所对应的观察分别为S"S,,…，S”。此时，可以对每条知识分别 求出*H*的后验几率O(HIS),然后利用这些后验几率并按照下列公式可以求出所有观察 下*H*的后验几率。

cm3 q °(H I S) O(H | S) \* xO(H|S“)xcs)

°(H 1 $，&，…，S) — —o(h) X O(H)\* …X *O(H) \* O(H)*

下面通过实例进一步说明主观贝叶斯方法的推理过程。 例**5.1**设有下列规则： 厂 1 : IF Ei THEN (2,0.0001)

*r2*: IF Ei AND E2 THEN (100,0. 0001) *H,*

r3： IF THEN (200,0.001) *H2*

已知 P(E1) = P(E2)=0. 5,P(H1)=0. 092,P(H2) = 0. 01,用户提供的证据如下: |SQ = 0.76,P(E2|S2)= 0.68。求 P(H2|S1,S2)O

解：由已知知识得到的推理网络如图5. 2所示。

(1)计算 0(H)IS】)。

先把Hi的先验概率P(HQ更新为在Ei下的 后验概率F(孤旧)：

*p(H* I E )= LSi X )

尸(Hi '匕)一 (LSi —i)xPCHJ + l

2X0. 092

(2 - 1) X 0.092 + 1

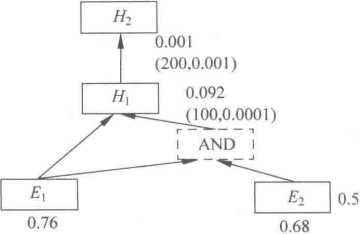
0.1685

图5.2推理网络

由于P(El|S])=O. 76>P(§),所以在当前观

察S下Hi的后验概率P(H】|SQ为

S) = P( W + ~~P%\*尸)~~x (P(E1 ! s)— P(Ei))

P(H,

=0. 092 + ~~]甲5 茶\_旺~~ x (0. 76 一 0. 5)

1 — 0. 5

=0. 1318

P(H1 I Sj)

。(如S) = ~~] W拓~~=瓮矗=。・1518

1. 计算 O(H1I(S1 AND S2))o

由于的前件是E|、E2合取关系，且已知*P(E】* |S,)=0. 76,P(E2 |S2)=0. 68,即 *P(E2IS2)<P(E]IS1)O*按合取取最小的原则，这里仅考虑E2对Hi的影响，即把计算 PCHi |(Sj AND S2))的问题转化为计算O(H」S2)的问题。

把8的先验概率P(HQ更新为在旦 下的后验概率*PCHl\E2)：*

LS2 X P(H1)

100 X 0.092 =知 2 (LS2 — 1) X P(H】)+ 1 (100 - 1) X 0.092 + 1

又由于P(E2|S2)>P(E2),得到在当前观察S2下的后验概率P(H」成)：

P(Hi I E2)-P(H1) x 頒(风丨 S2) -P(E2))

P(Hi I S2)

*0( H】*丨 S2)=

*1-P(E2) 0* Qif)9 — o no?

0. 092 + ~~• \* M 4~~ x (0. 68 -0.5) = 0. 3866

1 — 0. 5

*P(H}* I S2)

0・ 3866 n

*1-PCH,* I *S2) ~* 1 —0.3866 — ,

(3)计算 O(Hi|S】，S2)。

先将*H｝*的先验概率转换为先验几率:

。凹)=\*^ =沽牒=。・1013

然后根据合成公式计算*H｝*的后验几率:

0(H. | S”S2)= ~~X。牛卩？)~~XCKHQ

0.1518

0.1013

0.6303

0.1013

X 0.1013

=0.9445

再将后验几率转换为后验概率:

P(M

0(3 丨 S】，S2)

1+O(H】丨 S「S2)

0. 9445

1 +0. 9445

=0.4857

1. 计算 P(H2|S1,S2)O

对厂3,Hi相当于已知事实，H2为结论。将H?的先验概率F(H2)更新为在下的后

200 X 0. 001 八 1SQ

*p(h2* I =

ls3 xp(h2)  
(ls3 -1)xp(h2)+ i

(200-DX0.001 + 1 = °・ 1668

由于F(孤|S[,S2)=0. 4857>P(H1),得到在当前观察St.S2下*H2*的后验概率

P(H2|S],S2)：

*P(H2* I SnS2)= ~~吃二，也}~~ X (P(H I S1,S2)-P(W1))

1 — *rkrl})*

=0. 0] + ~~Q ?66七匚?；01~~ x(0. 4857 一 0. 092) = 0. 0780

1 — 0. 09Z

从上例可以看出，H2先验概率是0.01,通过运用知识厂1/2、厂3及初始证据的概率进行 推理，最后推岀的*H2*的后验概率为0.0780,相当于概率增加到7倍多。

主观贝叶斯方法的主要优点是理论模型精确，灵敏度高，不仅考虑了证据间的关系，而 且考虑了证据存在与否对假设的影响，因此是一种较好的方法。其主要缺点是所需要的主 观概率太多，专家不易给出。

**5.4**可信度方法

可信度方法是由肖特里菲(Shortliffe)等人于1975年在确定性理论上结合概率论等理 论提出的一种不确定性推理模型。在专家系统等领域有广泛的应用。

本节主要介绍CF模型，它是基于可信度概念和产生式规则构建的不确定性推理模型。

**5.4.1**不确定性的表示

下面介绍在CF模型中如何进行知识和证据的不确定性表示。

1. 知识的不确定性表示

在CF模型中，不确定性推理规则的一般形式为

IF *E* THEN *H* (CF(H,E))

其中上表示前提条件，H表示知识的结论，CF(H,E)表示该规则的可信度，也称可信度因 子或规则强度。可信度是指人们根据以往经验对某个事物或现象为真的程度的一个判断, 其值为［―1,1：］,CF(H,E)>0表示该证据对结论为真的支持度，CF值越趋近1,则该证据 对结论为真的支持度就越大,CF(H,E) = 1则表示该证据使结论成立为真；CF(H,E)VO 表示该证据对结论为假的支持度，CF值越趋近一1,则该证据对结论为假的支持度就越大, CF(H,E) = — 1则表示该证据使结论成立为假；CF(H,E) = O则表示证据和结论没有 关系。

1. 证据的不确定性表示

证据的不确定性也是用可信度CF(E)来表示的。CF(E)所描述的是证据的动态强度。 区别于知识的静态强度CF(H,E)表示的是规则的强度。证据的可信度来源有以下两种情 况：如果是初始证据，则其可信度是由提供证据的用户给出的；如果是先前推出的中间结 论又作为当前推理的证据，则其可信度是原来在推出该结论时由不确定性的更新算法计算 得到的。

可信度取值范围是［-1,1］。其典型取值如下：

•当证据E肯定为真时，CF(E) = 1°

•当证据E肯定为假时，CF(E) = — 1。

•当证据E一无所知时，CF(E)=O。

**5.4.2**组合证据不确定性的计算

组合证据包括合取和析取两种情形。

1. 当组合证据是多个单一证据的合取时，即

E=E, AND *E2* AND …AND *En*

若已知 CF(E ) ,CF(E2),…，CF(E”)，则

CF(E) = min{CF(E1) ,CF(E2)，- ,CF(E„))

1. 当组合证据是多个单一证据的析取时，即

*E=E}* OR E2 OR …OR *En*

若已知 CF(Ei) ,CF(E2),-,CF(EJ,则

CF(E) = xnax{CF(Ei),CF(E2)，…,CF(E”)}

**5.4.3**不确定性的传递算法

CF模型中的不确定性推理实际上是从不确定性的初始证据出发，不断运用相关的不 确定性知识，逐步推出最终结论和该结论的可信度的过程。而每一次的不确定性推理都需 要由证据的不确定性和知识的不确定性去计算结论的不确定性，其计算公式为

CF(H) = CF(H,E) Xmax(0,CF(E)}

从上式可以看出，当CF(E) = 1时，有CF(H) = CF(H,E),这表明，规则强度CF(E)实 际上就是在前提条件对应的证据为真时结论*H*的可信度。而当CF(E)<0时，则相应的证 据以某种程度为假，若CF(H)=0,则表明在该模型中没有考虑证据为假时对结论H所产 生的影响。

**5.4.4**结论不确定性的合成

设有如下两条规则：

IF E] THEN *H* (CF(H,Ei))

IF *E2* THEN *H* (CF(H,E2)) 则结论*H*的综合可信度可按照如下步骤求得：

1. 分别求岀

CFJH) = CF(H,EQ Xmax{O,CF(E1)}

CF2(H) = CF(H,E2) Xmax{O,CF(E2)}

1. 求出E1和E2的综合可信度CF(H)：

]CFi(H)+CF2(H)—CF】(H) • CF2(H), CFJH) >0,CF2(H) >0

CF(H) = JcFi(H)+CF2(H)+CF】(H) - CF2(H), CFJH) <0,CF2(H) <0 CF](H)+CF2(H), CFJH) • CF2(H) <0

例**5.2**设有如下一组知识：

门:IF *E.* THEN *H* (0. 95)

r2： IF E2 THEN *H* (0. 65)

r3： IF E3 THEN *H* (-0.51)

r4: IF E4 AND (E5 OR E6) THEN E】(0. 81)

已知：CF(E2) = 0. 81,CF(E3) = 0. 65,CF(E4)= 0. 51,CF(E5) = 0. 65,CF(E6) =

0. 81,求 CF(H) 0

解：由r4得

CF(E]) = 0. 81 X max(0,CF(E4 AND (E5 OR ))}

=0. 81 X max{0,min(CF(E4)>CF(E5 OR E6)}}

=0. 81 X max(0 ,min{CF(E4 ) ,max{CF(F5) ,CF(E6)}}}

=0. 81 X max{0 ,min{CF(E4 ) ,max{0,65,0. 81}}}

=0. 81 X max(0,min(0. 51,0. 81}}

=0. 81 X max(0,0. 51}

=0.4131

由门得

CFJH) = CF(H,E\) X max{0,CF(E1)} = 0. 95 X max{0,0. 4131) = 0. 392 445 由广2得

CF2(H) = CF(H,E2) X max{0,CF(E2)} = 0. 65 X max{0,0. 81} = 0. 5265

由r3得

CF3 (H) = CF(H,E3) X max{0,CF(E3)) =— 0. 51 X max{0,0. 65} =— 0. 3315 根据结论不确定性的合成算法得

CF1,2(H) = CFJH) +CF2(H) —CF](H) XCF2(H)

=0. 392 445 + 0. 5265 - 0. 392 445 X 0. 5265 = 0. 712 323

rF ( = CFl2(H) + CF3(H)

U2-3 — CFMH) I，| CF3(H) |}

0. 712 323 一 0. 3315 \_ 八

-l-min{0. 712 323,0. 3315} - ”緖

这就是所求的综合可信度，即CF(H)=0.569 668。

**5.5**证据理论

证据理论是由G. Shafer拓展了 A. P. Dempster的工作而来的，因此也称为DS理论， 该理论实则是对简单概率的一种推广，称为广义概率，能够处理由“不知道”所引起的不确定 性，并且由于辨别框的子集可以是多个元素的集合，因而知识的结论部分不必限制在由单个 元素表示的最明显的层次上，而可以是一个更一般的不明确的假设，这样更有利于领域专家 在不同细节、不同层次上进行知识表示。

**5.5.1**理论基础

在证据理论中，常用的概念有概率分配函数、信任函数、似然函数以及类概率函数等。 下面一一介绍这些概念。

1. 概率分配函数

定义**5.6**设函数且满足

m ( 0 ) = 0

习 zn(A) = 1

则称*m*是2。上的概率分配函数，m(A)称为A的基本概率数。其中，2。表示由0的所有子 集构成的集合(下同)。

对概率分配函数有以下两点说明：

其一，概率分配函数的作用是把*Q*的任意的一个子集都映射为[0,1]上的一个数 “(A)。当A由单元素组成时,〃(A)表示对A的精确信任度；当A由多元素组成，且不为 全集时,m(A)也表示对A的精确信任度，但不知道这部分信任度该分配给A中的哪一个元 素；当A是全集时，则m(A)是对全集的各个子集进行信任分配后剩下的部分，表示不知道 该如何对它进行分配。

其二，概率分配函数不是概率。

1. .信任函数

定义**5.7** 信任函数(belief function)如下：

Bel： *2a* [0,1]

对 VAgO,Bel(A) =

BgA

Bel函数又称为下线函数，表示当前环境下对假设集A的信任度，其值为A的所有子集 的基本概率之和，表示对A的总的信任度。

1. .似然函数

定义 **5. 8** 似然函数(plausibility function)如下：

P1： 2" — [0,1]

对 VAUO,P1(A) = 1-Bel(-«A)

其中，A=O—A。

似然函数又称为上限函数或不可驳斥函数。由于Bel(A)表示对A的真的信任度， Bel(-A)表示对的信任度，即对A为假的信任度，因此,P1(A)表示对A为非假的信 任度。

信任函数与似然函数有如下性质：

1. Bel(0) = O,Bel(O) = l,Pl(0)=O,Pl(O) = l。
2. IF THEN Bel(A)<Bel(B) ,P1(A)<P1(B) o
3. V AO^,Pl(A)>Bel(A)0
4. VA^n,Bel(A) + Bel( -A)<1,PI(A)+P1(-^A)>1O

另外，P1(A)—Bel(A)表示既不信任A也不信任「A的程度，即是真是假不知道的 程度。

**5.5.2**不确定性表示

下面介绍在证据理论下如何进行知识和证据的不确定性表示。

1. 知识的不确定性表示

在证据理论中，不确定性知识的表示形式为

IF *E* THEN *H{hi ,h2, •,- ,hn}* (CF= {c】*,c2 ,•••,&})*

其中，E为前提条件，它既可以是简单条件,也可以是用合取或析取词联结起来的复合条件。 H是结论，它用样本空间中的子集表示，儿浦2,…，九是该子集中的元素。CF是可信度因 子，用集合形式表示，该集合中的元素ci心，…，&用来表示A 1 , /i2 , *••• ,h„*的可信度，c,*与h，i* 一一对应，并且&应满足如下条件：

*Ci* 2 0,

/ = 1,2,

(1，

1. 证据的不确定性表示

证据A的不确定性用类概率函数f(A)表示，类概率函数的定义可以由信任函数和似 然函数得到，原始证据的/(A)是由用户给出的，如果是推理过程中得到的中间结论，则其 确定性是由推理得到的。

下面给出类概率函数的概念。

定义**5.9**设。是有限域，对VAWO,A的类概率函数为

*f(A)* = Bel(A) + X (PKA) 一 Bel(A))

其中，|A|和|。|分别是A和。中元素的个数。

类概率函数/(A)具有以下性质：

1. 、/({$})= lo

1 = I

1. 对 V AU。,有 Bel(A)W，(A)WPl(A)°
2. 对 VAUO,有 f(「A) = l —/(A)。

**5.5.3**组合证据不确定性的计算

组合证据包括合取和析取两种情形。当组合证据是多个单一证据的合取时，即

*E = E]* AND *E2* AND …AND *En*

若已知 /(E1),/(E2),-,/(E„),则

/(E) = min{/(El),/(E2),-,/(EJ)

当组合证据是多个单一证据的析取时，即

*E = E.* OR *E2* OR …OR *En*

若已知 /(E1),/(E2),-,/(Em),则

/(E) = max{/(E1),/(E2),-,/(En)}

当有多条规则支持同一结论时，如果A = {0，s，・・・g,},则

IF *Ei* THEN *H* CF, (CF, {cn ，…，c,”}) 3 = 1,2,m)

如果这些规则相互独立地支持结论*H*的成立，可以先计算

四({a】…，修”}) *= (.f(Ei)*• c,i, *f(Ei) • ci2*，…,/(ED • *cim)(i* = 1,2,・・・ 然后可以根据求解正交和的方法，对g求正交和，以组合所有规则对结论*H*的支持。一旦 组合的正交和m(H)计算出来，就可以计算Bel(H)、Pl(H)J(H)。

下面介绍求正交和的方法。

定义5. 10设四和m2是两个不同的概率分配函数，则其正交和*m = m}©fn2*满足

m(0) = 0

m(A) = K-1 • 习 *m\* (x)m2 *(y)*

其中，K = 1 一、叫(工)秫2(了)= 习 四*(x)m2(y)*。

*x(}y=0*

如果K/0,则正交和*m*也是一个概率分配函数；如果K = 0,则不存在正交和s,称皿 和*rn2*矛盾。

**5.5.4**不确定性的更新

设有知识 IF *E* THEN H(A1,A2,-,AJ(CF={c],c2,-,cJ)定义

772 ( {a,}) *= f(E) • c, Ci* = 1,2，…，m)

或表示为

m( {\* }, {。2}，…，也,”})=*(f(E)* • cj , *f(E) • c2*/(E) • *cm)*

规定

tw(iQ) = 1 — *y^m({a,})*

*i =*

而对于*Q*的所有其他子集H,均有m(H)=O。

当*H为Q*的真子集时，有

Bel(H) =n(B)

B£H

进一步可以计算PKH)和/(H)o

例5.3有如下规则：

r1； IF *E}* AND *E2* THEN A={山，a2) ,CF= (0. 3,0. 5}

r2： IF E3 AND (E4 OR E5)THEN B= {^ } ,CF= (0. *7}*

*r3*: IF *A* THEN *H={hx, h2, h3*} ,CF= {0. 1, 0. 5, 0. 3} r4 ： IF B THEN *H={h1 , h2,*炳}，CF= {0. 4, 0. 2, 0. 1} 已知用户对初始证据给出的确定性为：CERCE, )=0. 8,CER(E2)=0. 6,CER(E3)=

0. 9,CER(EJ=0. 5,CER(E5) = 0. 7,并假设 O 中的元素个数 |。|=10,求 CER(H)。

解：由给定知识形成的推理网络如图5.3所示。

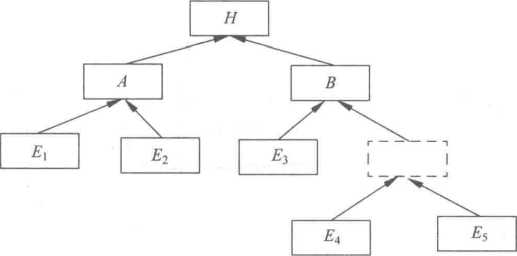


图5. 3推理网络

/(B) = Bel(B)+| B| / |r? | X(PKB)-BeKB)) =0.49 + 1/10X(1-0. 49) =0.541 所以 CER(B) = MD(B/E，)X/(B)=0. 541

1. 求 CER(H)。

由r3可得

*m}({h]},* {方2}，{儿3 })= {CER(A) X 0. 1, CER(A) X 0. 5, CER(A) X 0. 3}

={0. 584 X 0. 1, 0. 584 X 0. 5, 0. 584 X 0. 3}

={0.058, 0. 292, 0. 175}

mt (<Q) = 1 —[四(Ml } ) + 皿(統2 })+ 皿({方3 })]

=1 一 [0. 058 + 0. 292 + 0. 175] = 0. 475

再由r4可得

(h2}9 (A3}) = (CER(B) X0.4, CER(B) X0. 2, CER(B) XO. 1}

={0. 541 X 0. 4, 0. 541 X 0. 2, 0. 541 X 0. 1}

={0. 216, 0. 108, 0.054}

*m2* (.□) = 1 — [m2({Ai }) *+ m2C{h2})* + m2({A3})]

=1 一 [0. 216 + 0. 108 + 0. 054] = 0. 622

求正交和*m = mx®m2*。

*K =* (I?) X *m2 (Q)*

十 ({方i} ) X ( {Ai)) + mi ( {A] ) ) X *m2 (f2) + mx* (12) X m2 ({Ai})

+ Tn】({方2})X 死({先2})+ (統2})X *m2 (fl) + mx* (£2) X m2 ({A2})

+ mi ({人3}) X m2 (修3}) + ({人3}) X *m2 (fi) + mx (Q)* X m2({A3})

=0. 475 XO. 622

+ 0. 058 *X* 0. 216 + 0. 058 X 0. 622 + 0.475 X 0. 216

+ 0. 292 X 0. 108 + 0. 292 X 0. 622 + 0. 475 X 0. 108

+ 0. 175 X 0. 054 + 0. 175 X 0. 622 + 0. 475 X 0. 054

=0. 855 *m{h\* )=丄 X (mi ( {hi } ) X m2 ( (Ai } ) + mi ( {/i] } ) X *m2 (Q) -\~mY (Q) X LX*

=X(°・ °58 X 0. 216 + 0. 058 X 0. 622 + 0. 475 X 0. 216) U. obb

=0. 178

*m(h2*) = % X (mi ({A2}) *X m2(.{h2}') + m}({h2}') X m2 (□) +* (O) X m2({/i2}))

=X(°・ 292 *X* 0. 108 + 0. 292 X 0. 622 + 0.475 X 0. 108)

0. 855

= 0.309

*mg)*=点 X(772] ({/i3}) X 死({人3})+ mi ({心}) X *m2* (O) + *mx* (O) X m2({A3})) =t-Izf X (0. 175 X 0. 054 + 0. 175 X 0. 622 + 0. 475 X 0. 054)

U. obb

=0.168

m(0) = 1 — (m(仍1}) + ?n( {/12})+ m{ (/13 ) })

= 1 -(0.178 + 0. 309 + 0. 168)

=0.345

再根据m可得

Bel(H) = m({龙1 }) +m(也2}) +m({人3}) = 0. 178 + 0. 309 + 0. 168 = 0. 655

Pl(H) = 772(0)+ Bel(H) = 0. 345 + 0. 655 = 1

CER(H) = MD(H/E') X /(H) = 0. 759

尽管该理论运用较为广泛，但由于要求〃中的元素满足互斥条件，这在系统中不容易 实现，并且概率分配数构成的维数空间很大，计算比较复杂。

**5.6**模糊推理

**5.6.1**模糊知识的表示

1. 语言变量

模糊逻辑中所使用的变量可以是语言变量，所谓语言变量是指用自然语言中的词表示 并可以取语言值的变量。例如，变量“年龄”在普通集合中一般取0〜150的数值，而在模糊 集合中可以使用语言值“年轻、很年轻、不很年轻、老、很老、不很老”等。这些语言值可看成 是论域［0,150］上模糊集的集合名。

通常，语言变量的值可以由一个或多个原始值再加上一组修饰词和连词组成。例如，上 面给出的语言变量“年龄”，其原始词为“年轻”“老”，若加上修饰词“不很”可得到“不很年轻” “不很老”，若再加上连词“且”，则可得到“不很年轻且不很老气

1. 模糊命题的描述

模糊逻辑是通过模糊谓词、模糊量词、模糊概率、模糊可能性、模糊真值、模糊修饰语等 对命题的模糊性进行描述的。

1. 模糊谓词

设z为在U中取值的变量，F为模糊谓词，即*U*中的一个模糊关系，则命题可表示为

•z is F

其中的模糊谓词可以是“大、小、年轻、老、冷、暖、长、短”等。

1. 模糊量词

模糊逻辑中使用了大量的模糊量词，如“极少、很少、几个、少数、很少、多数、大多数、几 乎所有”等。这些模糊量词可以使我们很方便地描述类似下面的命题：

大多数成绩好的学生学习都很刻苦。

很少有成绩好的学生特别贪玩。

1. 模糊概率、模糊可能性和模糊真值

设义为模糊概率，兀为模糊可能性，匸为模糊真值，则对命题还可以附加概率限定、可能 性限定和真值限定：

*(jc* is F) is A

(j? is F) is *7c*

& is F) is r

式中，人可以是“或许、必须”等，兀可以是“非常可能、很不可能”等，〔可以是“非常真、有些 假”等。

4)模糊修饰语

模糊修饰语有以下4种：

1. 求补。表示否定，如“不、非”等。
2. 集中。表示“很、非常”等，其效果是减少隶属函数的值。
3. 扩张。表示“有些、稍微”等，其效果是增加0.5以上隶属函数的值。
4. 加强对比。表示“明确、确定”等，其效果是增加0.5以上隶属函数的值，减少0. 5 以下隶属函数的值。

**3** .模糊知识的表示方法

在扎德的推理模型中，产生式规则的表示形式是

IF x is F THEN *y \s G*

其中u和丁是变量，表示对象；F和G分别是论域U及V上的模糊集，表示概念。条件部 分(IF)可以是多个而is F,的逻辑组合，此时，诸隶属函数间的运算按照模糊集的运算进行。

模糊推理中所用的证据都是用模糊命题来表示的，其一般形式为

*X* is F'

其中，F'是论域U上的模糊集。

**5.6.2**模糊概念的匹配

模糊概念的匹配是指对两个模糊概念相似程度的比较与判断，而两个模糊概念的相似 程度又称为匹配度。本节主要讨论语义距离和贴近度这两种计算匹配度的方法。

1. 语义距离

语义距离刻画的实际上是两个模糊概念之间的差异，常用的计算语义距离的方法有多 种，这里主要介绍汉明距离。

设口={%工2,…，如}是一个离散有限论域，F和G分别是论域U上的两个模糊概念的 模糊集，则F和G的汉明距离定义为

*d(F,G) ==* —2 I 必f(%)—俱S)I

n i-i

例5.4设论域U={ —10,0,10,20,30}表示温度，模糊集

F = 0. 8/- 10 + 0. 5/0 + 0. 1/10

G = 0. 9/- 10 + 0. 6/0 + 0. 2/10

分别表示“冷''和“比较冷”，则

*d(F,G)* = 0. 2X (| 0.8-0. 9 | + | 0. 5 - 0. 6 | + | 0. 1 - 0. 2 |) = 0. 2 X 0. 3 = 0. 06

即F和G的汉明距离为0.06。

对求出的汉明距离，可通过下式

*l — d（F,G）*

将其转换为匹配度。当匹配度大于某个事先给定的阈值时，认为两个模糊概念是相匹配的。 当然，也可以直接用语义距离来判断两个模糊概念是否匹配。这时，需要检查语义距离是否 小于某个给定的阈值，距离越小，说明两者越相似。

**2.**贴近度

贴近度是指两个概念的贴近程度，可直接用来作为匹配度。

设F和G分别是论域

*U = {ui 9u2*，…必}

上的两个模糊概念的模糊集，则它们的贴近度定义为

（F,G） = y（F.G+（l-F©G））S

式中，F・G为F与G的内积*,FOG为F*与G的外积。

**5.6.3**模糊推理

模糊推理是按照给定的推理模式通过模糊集的合成来实现的。而模糊集的合成实际上 又是通过模糊集与模糊关系的合成来实现的。可见，模糊关系在模糊推理中占有重要的位 置。为此，在讨论模糊推理方法之前，先对模糊关系的构造问题进行简单的介绍。

**1.**模糊关系的构造

前面曾经介绍过模糊关系的概念，这里主要讨论由模糊集构造模糊关系的方法。目前 已有多种构造模糊集关系的方法，下面仅介绍其中最常用的几种。

1）模糊关系*Rm*

模糊关系Rm是由扎德提出的一种构造模糊关系的方法。设F和G分别是论域*U*和 V上的两个模糊集，则定义为

**Rm = [ （/ZF（U）A** 〃**G（P））V （1 —#F**（以））/（〃，R）

UXV

其中，X号表示模糊集的笛卡儿乘积。

例**5.5** 设*U=V=* {1,2,3} ,F和**G**分别是U和**V**上的两个模糊集，并设

*F* = 1/1+0. 6/2 + 0. 1/3

G = 0. 1/1+0. 6/2+ 1/3

则*Rm*为

|  |  |  |  |
| --- | --- | --- | --- |
|  | ■0. 1 | 0. 6 | 1 一 |
| Rm = | 0.4 | 0. 6 | 0. 6 |
|  | 0.9 | 0. 9 | 0. 9\_ |

下面以Rm（2,3）为例说明*Rm*中元素的求法。

Rm（2,3）=（所（以2）A *心3））*V （1-^f（u2））

=(0. 6 A 1) V (1-0.6) = 0.6 V 0. 4 = 0. 6

2）模糊关系R

模糊关系**R**是由麦姆徳尼提出的一种构造模糊关系的方法。设**F**和**G**分别为论域*U* 和V上的两个模糊集，则Re定义为

*Rc =* A 〃G（w））/（〃e）

*uxv*

对例5.5所给出的模糊集，其*Rc*为

|  |  |  |  |
| --- | --- | --- | --- |
|  | \_o. 1 | 0. 6 | 1 一 |
| *Rc =* | 0. 1 | 0. 6 | 0. 6 |
|  | \_0. 1 | 0. 1 | 0. 1\_ |

下面以R.（3,2）为例来说明R中元素的求法:

七（3,2）=伊（以3）A 衍（血）=0.1 A 0. 6 = 0. 1

3）模糊关系R

模糊关系R是米祖莫提提出的一种构造模糊关系的方法。设F和G分别是论域U和 卩上的两个模糊集，定义为

*Rg =* （//F （"）f /ZG （汾）/（“戒）

LJXV

对例5.5给出的模糊集，其七为

|  |  |  |  |
| --- | --- | --- | --- |
|  | '0. 1 | 0. 6 | T |
| *R& =* | 0. 1 | 1 | 1 |
|  | -1 | 1 | 1 |

1. 模糊推理的基本方法

与自然演绎推理相对应,模糊推理也有相应的3种基本模式，即模糊假言推理、拒取式 推理及模糊三段论推理。

1）模糊假言推理

设F和G分别是*U*和V上的两个模糊集，且有知识

IF z is F THEN *y is G*

若有U上的一个模糊集F‘，且F可以和F'匹配，则可以推出丁 isG',且G'是V上的一 个模糊集。这种推理模式称为模糊假言推理,其表示形式为

知识：IF 7 is F THEN :y isG

证据：=is *F'*

结论：*y* is *Gf*

在这种推理模式下，模糊知识

IF z is F THEN *y is G*

表示在F与G之间存在确定的模糊关系，设此模糊关系*为R。*那么，当已知的模糊事实F' 可以和F匹配时，则可通过F'与R的合成得到G'。

例**5.6**对例5. 5所给出的F、G以及所求出的択宀设有已知事实

*x* is较小

并设“较小"的模糊集为

较小=1/1 + 0. 7/2+ 0.2/3

求在此已知事实下的模糊结论。

解：本例的模糊关系在前文中已求出，设已知模糊事实“较小”为F',F'与Rm的合成即

为所求**G\**

*G'* = F'F'。*Gm* = {1,0.7,0.2}*。*

~0. 1

0. 4

\_0. 9

0. 6

0. 6

0. 9

0. 6

0. 9

={0. 4,0. 6,1}

即所求出的模糊结论G'为

*G'* = 0.4/1+ 0.6/2+ 1/3

2）模糊拒取式推理

设F和G分别是*U*和V上的两个模糊集，且有知识

IF z is F THEN *y is G*

若有V上的一个模糊集且G'可以与G的补集匹配，则可以推出z/s 且F'是U上的

一个模糊集。这种推理模式成为模糊拒取式推理。它可以表示为

知识：IFjt isFTHENy isG

证据：y is *G*

结论：*T* is *Ff*

在这种推理模式下，模糊知识

IF T is F THEN *y is G*

也表示在F与G之间存在确定的模糊关系，设此模糊关系为*Ro*那么•当已知的模糊事实 G'可以和匹配时，则可以通过R与G'的合成得到F',即

*F‘ = R。G'*

式中的模糊关系*R*可以是*Rm、R*或R中的任何一种。

例5.7设F、G如例5.5所示，已知事实为

*y* is较大

且模糊概念“较大”的模糊集*C为*

*G'* = 0. 2/1 + 0. 7/2+ 1/3

若G'与匹配，以模糊关系*R。*为例，推出F'。

解：本例中的模糊关系R已在前面求出，通过R与G'的合成即可得到所求的F'。

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | ro. 1 | 0. 6 | 1 - |  | ■0. *2-* |  | r 1 I |
| *F，= Rc。G’ =* | 0. 1 | 0. 6 | 0. 6 | O | 0. 7 | = | 0. 6 |
|  | \_0. 1 | 0. 1 | 0. 1\_ |  | \_ 1 \_ |  | \_0. 1\_ |

即所求出的F'为

F' = 1/1 + 0. 6/2 + 0. 1/3

模糊拒取式推理和模糊假言推理类似，也可以把计算Rm、Rc的公式代入到求F'的公式 中，得到求F’的一般公式。对Rm有

F' = RmG' j V A 〃G（P））V（1 —（以））］A “G（〃）｝/h

同理，对模糊关系R也可推出求F‘的一般公式:

*=RcG =* V {[/ZF（U）A A *^tG（V）}/v*

F' *= RCG*

在实际应用中，也可以直接利用这些公式由F、G和G'求岀F'。

3）模糊三段论推理

设F、G、H分别是U、V、W上的3个模糊集，且由知识

IF z is F THEN 丁 is G

IF 3； is G THEN *zis H*

则可推出

IF 1 is F THEN *zis H*

这种推理模式称为模糊假言三段论，可以表示为

知识：IF z is F THEN 了 is G

证据：IF yis G THEN z is H

结论：IF is F THEN *z is H*

在这种推理模式下，模糊知识

门：IF X is F THEN *y isG* 表示在F和G之间存在确定的模糊关系，设此模糊关系*为*模糊知识

r2 ： IF > is G THEN z is H

表示在G和H之间存在确定的模糊关系，设此模糊关系为*R2* 0

若模糊假言三段论成立，则,3的模糊关系R可由R与R的合成得到，即

*Ra = R\ ° R2*

这里的关系R、死、R：,可以是前面所讨论过的Rm、R与*R&*中的任何一种。为说明这种方 法，下面讨论一个例子。

例 5.8 设 U=W=V={1,2,3}。

*E=* 1/1 + 0. 6/2 + 0. 2/3

*F* = 0. 8/1 + 0. 5/2 + 0. 1/3

*G=* 0. 2/1+0. 6/2+ 1/3

按R求EXFXG上的关系択劇:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 択心= | \_0. 8  1  \_ 1 | 0. 5  1.5  1 | 0. r  0. 1  0. 1\_ |  |
| 再求FXG上的关系Rs |  |  |  |  |  |
|  |  | ~0. 2 0.6 r | | |  |
|  | **Rg2 =** | 0. | 2 1 | 1 |  |
|  |  | \_ 1 | 1 | 1\_ |  |
| 最后求EXFXG±的关系R： |  |  |  |  |  |
|  |  |  | '0. 2 | 0. 6 | 0. 8\_ |
| *R* | =七**1&2** | *=* | 0.2 | 0. 6 | 1 |
|  |  |  | \_0. 2 | 1 | 1 \_ |

**5.7** 习题

i.什么是不确定性推理？有哪几类不确定性推理方法？不确定性推理中需要解决的 基本问题有哪些？

2 .什么是可信度？

1. 设有如下一组推理规则：

rl： IF *E}* THEN E2(0. 6)

r2 ： IF *E2* AND E3 THEN E4(0. 7)

r3： IF E THEN *H* (0.8)

r4： IF E5 THEN *H* (0. 9)

且已知 CFCEj )=0. 5,CF(E3) = 0. 6,CF(E5) = 0. 7o 求 CF(H)0

1. 设有如下推理规则；

小:IF THEN

(2, 0. 00001)

(100, 0. 0001)

(200, 0. 001)

(50, 0. 1) *H2*

H1

Hi

h2

r2: IF *E2* THEN

r3: IF E3 THEN

r4: IF *Hi* THEN

且已知 P(E ) = P(E.2)= P(H3)= 0. 6,P(H|)=0.09LP(H2)= 0. 01,又由提供证据 的用户告知 *P(Ei* |SQ = 0. 84,P(E2|S2) = 0. 68,P(E3 |S3)=0. 36o

用主观贝叶斯方法求*p(h2isI9* S2, S3)o

1. 设有如下一组推理规则：

r1: IF Ei AND *E2* THEN *A={a}* (CF=(0. 9})

r2： IF *E2* AND *(E3* OR *EQ* THEN *B={b} , b2}* (CF={0. 5, 0.4})

r3 ： IF A THEN *扇,h3}* (CF={0.2, 0.3, 0.4})

r4 ： IF B THEN *H={hi, h2*, A3} (CF= (0. 3, 0. 2, 0. 1))

且已知初始证据的确定性分别为：CER(EQ=0. 6,CER(E2)=0. 7,CER(E3)=0. 8, CER(E4) =0. 9。

假设 |X2|=10,求 CER(H)。

1. 简述模糊概念、模糊集和隶属函数三者之间的关系。
2. 设U=V={1,2,3,4},且有如下推理规则：

IF jc is 少 THEN yis 多

其中，“少”与“多，，分别是U与V上的模糊集，设

少=0. 9/1+0. 7/2 + 0. 4/3

多=0.3/2+0. 7/3 + 0. 9/4

已知事实为

*x* is较少

“较少”的模糊集为

较少=0. 8/1 + 0.5/2 + 0. 2/3

请用模糊关系求出模糊结论。

8.设有论域U=3i，"2,以3,爲4, “5},并设尸、。是［7上的两个模糊集，且有

*F =* 0. 9/*u\* + 0. 7/u2 +0. 5/u3 +0. *3/u4*

*G =* 0. 6/u3 + 0. 8/*uA* + 1/*u5* 请分别计算F"FUG、「F。

9.设有如下两个模糊关系:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | ■0. 3 | 0.7 | 0. *2~* |  | -0. 2 | 0. 8\_ |
| *Ri =* | 1 | 0 | 0.4 | 択**2 =** | 0. 6 | 0.4 |
|  | \_ 0 | 0. 5 | 1 \_ |  | \_0, 9 | 0.1\_ |

请写出**R**与死的合成*RrR2o*

10.设**F**是论域**U**上的模糊集,**R**是UXV上的模糊关系，**F**和**R**分别为

*F =* (0.4,0.6,0.8}

「0.1

0. 3

0.4

0. 6

0. 5\_

0. 8

L0. 6 0. 3 0

求模糊变换**FoR**。

机器学习

**6.1**概述

机器学习是人工智能的重要研究领域之一。本章主要介绍机器学习的基本概念以及常 用的几种机器学习方法。

**6.1.1**机器学习的基本概念

要了解什么是机器学习，就要从人类的“学习”说起。可以说人们每天都在学习，可是终 究什么是学习，至今都没有一个统一的定义。综合众多观点，可以这样认为，学习是一个有 特定目的的知识获取和能力增长的过程，其内在行为是获取知识、积累经验、发现规律等，其 外部表现是改进性能、适应环境和实现系统的自我完善。以下是关于学习且比较有影响的 定义：

（1） 西蒙认为，学习就是系统中的适应性变化，这种变化使系统在做重复工作或类似工 作时能够做到更好或效率更高。

（2） 米哈尔斯基认为，学习是对经历描述的建议和修改。

（3） 蔡普金认为，学习是一种过程，通过对系统重复输入各种信号，并从外部校正该系 统,从而系统对特定的输入具有特定的响应。自学习就是不具有外来校正的学习，即不具有 奖罚的学习，它不给系统响应正确与否的任何附加信息。

机器学习的定义是基于人的学习的，由于学习没有统一的定义，机器学习无法给岀严格 的定义。从学科角度来讲，机器学习是研究如何让计算机来模拟人类学习活动的一门学科。

机器学习的研究工作主要从以下3个方面进行：一是认知模型的研究，通过对人类学 习机理的研究和模拟，从根本上解决机器学习方面存在的种种问题；二是理论学习的研究， 从理论上探索各种可能的学习方法，并建立起独立于具体应用领域的学习算法；三是面向

任务的研究，主要目的是根据特定任务的要求建立相应的学习系统。

**6.1.2**机器学习的发展历史

机器学习的发展大致可以分为4个时期，即热烈时期、冷静时期、复兴时期以及蓬勃发 展时期。

机器学习作为人工智能应用研究领域重要分支，它的兴起时间与人工智能的诞生时间 几乎是一致的。通常认为20世纪50年代中叶到60年代初期是热烈时期。F. Rosenblatt 是这一时期的代表人物，他于1957年提出了感知器模型。该时期研究的是“没有知识”的学 习。其主要研究目标是各种自组织系统和自适应系统，基本思想是：如果给系统一组刺激、 一个分馈源以及修改自身的足够自由度，那么系统将能自适应地趋向最优组织。该时期所 釆用的方法主要是不断修改控制参数，以改进系统的执行能力，而不涉及与具体任务有关的 知识。该时期所依据的主要理论基础是早在20世纪40年代就开始研究的神经网络模型。 在这个时期，我国研究了数字识别学习机。

20世纪60年代中叶至70年代中叶属于机器学习的冷静时期。该时期的研究目标是 模拟人类的概念学习过程，并釆用逻辑结构或图结构作为机器内部描述。温斯顿 （Winston）的结构学习系统和海斯（Hayes）、罗斯（Roth）等人提出的基于逻辑的归纳学习系 统是该时期的代表性工作。

虽然这类学习系统取得了较大的成功，但只能学习单一的概念，而且未能投入实际应 用。另外，神经网络学习机因理论缺陷未能达到预期效果,机器学习的研究转入低潮。

20世纪70年代中叶到80年代中叶称为复兴时期。在该时期，人们开始从单一概念的 学习扩展到多个概念的研究。机器的学习过程一般都建立在大规模的知识库上，实现知识 强化学习。值得庆幸的是，在本阶段人们开始将学习系统与各种应用结合起来，并取得了很 大的成就，很好地促进了机器学习的发展。在出现第一个专家学习系统后，示例归纳学习系 统成为研究主流，自动知识获取成为机器学习的应用研究目标。

1980年，在美国卡内基-梅隆大学召开了第一届计算机学习国际研讨会，它标志着机 器学习的研究已经在全世界兴起。此后，机器归纳学习进入应用阶段。1986年，国际杂志 *Machine Learning*创刊，迎来了机器学习蓬勃发展的新时期。20世纪70年代末，中国科学 院自动化研究所进行了质谱分析和模式文法推断研究，表明我国的机器学习研究得到恢复。

自1986年起，机器学习进入了最新阶段。神经网络的研究再度兴起，使得机器学习进 入了链接学习的研究阶段，与此同时，传统的符号学习研究也取得了很大的发展。实际上, 链接学习和符号学习各有所长，并具有很大的互补性。因此，二者结合起来的混合型学习系 统研究已成为机器学习研究的一个新热点。如果能将二者很好地融合在一起，就可以在一 定程度上有机地模拟人类逻辑思维和直觉思维，这将是人工智能的一个重大突破。

机器学习进入新阶段的主要表现如下：

（1） 机器学习已经成为新的边缘学科，并在高校形成一门课程。

（2） 结合各种学习方法，取长补短的多种形式的集成学习系统研究正在兴起。

（3） 机器学习与人工智能在各种基础问题上的统一观点正在形成。

（4） 各种学习方法的应用范围不断扩大，一部分已经成为商品。归纳学习的知识获取 工具已在诊断分类专家系统中广泛使用。链接学习在声音、图文识别中占据优势。遗传算 法与强化学习在工程控制中有较好的应用前景。

1. 知识发现和数据挖掘的研究已形成热潮，并在生物学、金融管理、商业销售等领域

得到成功的应用，给机器学习注入了新的活力。 :

1. 与机器学习有关的学术活动空前活跃。国际上除了每年一次的机器学习研讨会 ♦

外，还有计算机学习理论会议以及遗传算法会议。

**6.1.3**学习系统的基本模型

机器学习的实现依赖于学习系统，学习系统能够利用过去与环境作用时得到的信息并

提高自身的性能。学习系统的基本模型结构如图6.1所示。



图6.1学习系统的基本模型结构

从学习系统的基本模型结构可以看出，学习系统不仅与环境和知识库有关，而且还包含 学习与执行两个环节。学习系统中的环境是指学习系统进行学习时的信息来源。学习环节 是机器先从环境获取外部信息，然后通过对获取信息的分析、综合、类比和归纳等过程形成 知识，所生成的知识被放入知识库，即学习是将外界信息加工成知识的过程。知识库是以某 种形式表示的知识的集合，用来存放学习环节所得的知识。执行环节是利用知识库中的知 识完成某种任务的过程，并把完成任务过程中所获得的一些信息反馈给学习环节，以提高学 习性能。

适当的学习环境是建立学习系统模型的第一重要因素，环境所提供的信息水平与质 量都影响着机器的性能。即如果没有很好的环境，提供的信息杂乱无章，则学习部分不 容易处理，必须从足够的数据中提取原则，然后放入知识库中，这增加了学习环境的设计 负担。

知识库是设计学习系统的另一重要因素，常用的知识表示有多种，如谓词、产生式、语义 网络等。选择合适的表示方法也是很重要的，选择表示方法时，应当遵循以下一些原则：其 一，所选择的表示法要能够很好地表达相关的知识，因为不同的知识表示方法适用于不同的 对象；其二,尽可能地使得推理容易些；其三，要考虑知识库的修改难易程度；其四，要考虑 知识是否易于扩展，随着系统学习能力的提髙，单一的知识表示法已不能满足需要，有的时 候还需要几种知识表示同时使用，以适应外部环境需要。

**6.1.4**学习策略

机器学习的学习过程与推理过程是密切相关的，按照学习中所使用的推理方法分类，可 以将机器学习的学习策略分为记忆学习、归纳学习、类比学习、传授学习、演绎学习和联结学 习等。本章将主要介绍几种常用的学习策略。

**6.2**记忆学习

记忆学习又称为机械学习，是最简单的机器学习方法。该方法主要是凭借记忆，即存储 学习过的知识，供需要时检索调用，其特点是不再需要重新计算或推理。在记忆学习系统 中，知识的获取是以较为稳定和直接的方式进行的，不需要系统进行过多的加工。

当给定一个问题并由记忆学习解决后，该系统就记住了这个问题及其解。现将系统的 执行元件抽象成一个函数八输入自变量值（⑥，幻，・・・，］”），经过该函数计算后得到的函数 值（乂，力，…，乂）若经过评价确定是正确的，则把以下的输入输出模式

［（^1…，,'2，..・，必）］

存入知识库中，在以后的学习过程中，如果重新计算（幻，Z2,・•・，¥）的值，该系统就自动从知 识库中将（y，必，…，、”）搜索岀来而不是重新计算。记忆学习系统的模型如图6.2所示。

（可，入,・・・，—，...，》）=> ［（寻，乂2,……，以）］

图6.2记忆学习系统的模型结构

**6.3** 归纳学习

归纳学习是应用归纳推理进行学习的一种方法，即从一系列的事例中归纳出一般性的 知识描述的过程。根据学习过程是否有教师指导可分为示例学习以及观察与发现学习两种 形式。

**6.3.1**示例学习

示例学习又称实例学习，它是通过环境中若干与某概念有关的例子,经归纳得出一般性概 念的一种学习方法。这种学习方法给学习者提供某一概念的一组正例和反例，学习者从这个 例子中归纳出一个总的概念描述，并使这个概念描述适合于所有的正例,并排除所有的反例。

1. 示例学习的模型

示例学习的过程是：首先从示例空间中选择合适的训练示例，然后经解释过程得到一 般性的知识，最后再从示例空间中选择更多的示例对它进行验证，直到得到可实用的知识为 止。示例学习的空间模型如图6. 3所示。



图6, 3示例学习的学习过程

该空间模型是示例学习的基本模型，该模型包括两个主要空间和两个主要过程，它们分 别是示例空间、规则空间、解释过程和验证过程。

示例空间是所有可对系统进行训练的示例集合。与示例空间有关的主要问题是示例的 质量、数量以及它们在示例空间中的组织和示例空间的搜索方法。示例的质量和数量直接 影响学习的质量，而示例的组织方式将影响到学习效率。规则空间是事物所具有的各种规 律的集合。

解释过程的主要任务是从搜索到的示例中抽象出所需要的信息，并对这些信息进行综 合、归纳，形成一般性的知识。这种形成知识的过程实际上是一个归纳推理的过程。解释过 程是示例学习最主要的组成部分，其常用的解释方法有把常量换成变量、去掉条件、增加选 择和曲线拟合4种。

验证过程的主要任务是从示例空间中选择新的示例，对刚刚归纳出的规则做进一步的 验证和修改。其中，最主要的问题是选择哪些新的示例和怎样得到这些示例。

1. 执行过程描述

依据双空间模型建立的归纳学习系统，其执行过程可以大致描述为：首先由施教者给 示例空间提供一些初始示教例子，由于示教例子在形式上往往和规则形式不同，因此需要对 这些例子进行转换，解释为规则空间接受的形式。然后利用解释后的例子搜索规则空间，由 于一般情况下不能一次就从规则空间中搜索到符合要求的规则，因此还要寻找一些新的示 教例子，这个过程就是选择例子。程序会选择对搜索规则空间最有用的例子，对这些示教例 子重复上述循环。如此循环多次，直到找到所要求的例子。

1. 示例空间

在双空间模型中，示例空间所要考虑的主要问题包括两个：一是示教例子的质量，另一 个是示例空间的搜索方法。解释示教例子的目的是从例子中抽取出用于搜索规则空间的信 息，也就是把示教例子变换成易于进行符号归纳的形式。选择例子就是确定需要哪些新的 例子和怎样得到这些例子。

1. 规则空间

规则空间的目的是指定表示规则的操作符和术语，用以描述和表示规则空间中的规则， 与之相关的两个问题是对规则空间的要求和规则空间的搜索方法。所谓规则空间是用指定 的描述语言可以表示的所有规则(概念假设)的集合。对规则空间有3个方面的要求：规则 表示方法应适应归纳推理；规则的表示与例子的表示一致；规则空间应包含要求的规则。

归纳学习方法可分为以下两大类：

1. 单概念学习方法。典型的单概念学习系统包括米切尔(Tom Mitchell)的基于数据 驱动的变型空间法、昆兰(J.R. Quinlan)的ID3方法、狄特利希(T.G. Dietterich)和米哈尔 斯基(R. S. Michalski)提出的基于模型驱动的Induce算法等。
2. 多概念学习方法。典型的多概念学习方法和系统有米哈尔斯基的AQ11、 DENDRAL和AM程序等。多概念学习任务可以划分成多个单概念学习任务来完成。

多概念学习与单概念学习的差别在于多概念学习方法必须解决概念之间的冲突问题。

变形空间学习方法(learning by version space),也称为变形空间学习法。变形空间法 是米切尔于1977年提出的一种数据驱动型的学习方法。该方法以整个规则空间为初始的 假设规则集合Ho依据示教例子中的信息，系统对集合H进行一般化或特殊化处理，逐步缩小集合H,最后使得*H*收敛到只含有要求的规则。由于被搜索的空间*H*逐渐缩小，故称 为变形空间法。

在规则空间中，表示规则的点与点之间存在着一种由一般到特殊的偏序关系，将其定义 为覆盖。例如，color (X,Y)覆盖 color(ball,Z)，于是又覆盖 color (ball, red) o

作为一个简单的例子，考虑有以下属性和值的对象域：

Sizes = {large, small}

Colors= (red, white,blue}

Shapes= (ball, brick,cube}

这些对象可以用谓词obj(Sizes,Color,Shapes)来表示。用变量替换常量这个泛化操作定义。 图6. 4展示了一个规则空间偏序关系的一部分。可以把归纳学习看成是对所有训练示 例相一致的概念空间的搜索。在搜索规则空间时，使用一个可能合理的假设规则的集合 H,是规则空间的子集。

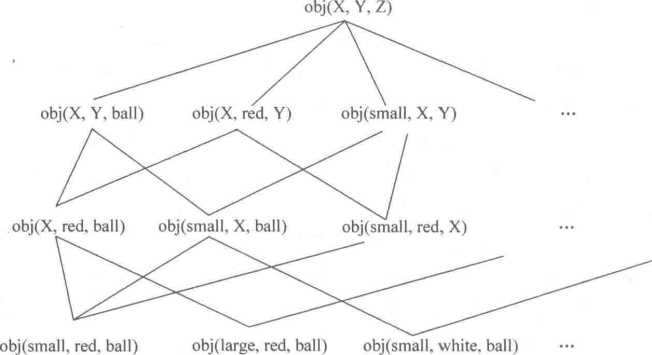
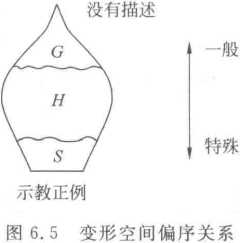


图6.4规则空间偏序关系



集合*H*由两个子集G和*S*所限定，子集*G*中的元素表示 *H*中的最一般的概念，子集*S*中的元素表示*H*中的最特殊的 概念，集合H由G、S及G与S之间的元素构成，即

H = GUSU{^|S<K<G}

式中V表示变形空间中的偏序关系，如图6.5所示。

米切尔的学习算法称为候选删除算法。在这种算法中，把 尚未被数据排除的假设称为可能假设，把所有可能假设构成的 集合H称为变形空间。算法一开始，变形空间H包含所有的 概念。随着向程序提供示教正例后，程序就从变形空间中删除 候选概念。当变形空间仅包含一个候选概念时，就找到了所要求的概念。

该算法分为4个步骤：

1. 把H初始化为整个规则空间。这时G仅包含空描述。S包含所有最特殊的概念。 实际上，为避免S集合过大,算法把S初始化为仅包含第一个示教正例。
2. 接受一个新的示教例子。如果这个例子是正例，则从G中删除不包含新例的概念,

然后修改S为由新正例和S原有元素共同归纳出最特殊化的泛化。这个过程称为对集合S 的修改过程。如果这个例子是反例，则从S中删去包含新例的概念，再对G作尽量小的特 殊化，使之不包含新例。这个过程称为集合G的修改过程。

1. 重复步骤(2),直到G=S,且使这两个集合都只含有一个元素为止。
2. 输出*H*中的概念(即输出G或S)。

下面给出一个实例。

用特征向量描述物体，每个物体有两个特征：大小和形状。物体的大小可以是大的 (也)或小的(sm)。物体的形状可以是圆的(cir)、方的(squ)或三角的(tri)。要教给程序“圆” 的概念，这可以表示为G,cir),其中］表示任何大小。

初始H集是规则空间，如图6. 6所示。G和S集分别是

*G* = {(z,>)}

*S* = { (sm squ), (sm cir) , (sm tri), (lg squ) , (lg cir) , (lg tri) }

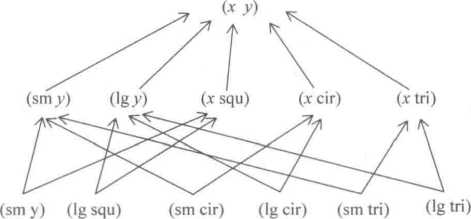


图6. 6初始变形空间

第一个训练例子是正例(sm cir)，这表示小圆是圆。经过修改S算法后得到G= {(1 y ) }, S= { (sm cir)},如图 6. 7 所示。

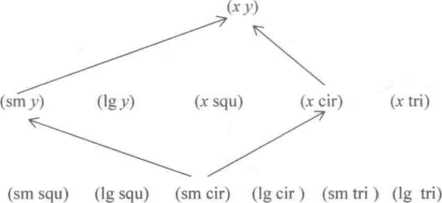
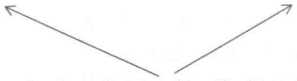


图6. 7修改S算法

第二个训练例子是反例(lg tri),这表示大三角不是圆。 得到 *G=* { (x cir) ,(sm >)} ,S= {(sm cir)}，如图 6. 8 所示。

这一步对G集进行特化处理,



(xtri)

*(xy)*

(sm squ) (Ig squ) (sm cir) (Ig cir) (sm tri)

(sm\*) (lg») (x squ) (x cir)

(lg tri)

第三个训练例子是正例(lg cir),这表示大圆是圆。这一步首先从*G*中去掉不满足此正 例的概念(smjO。再对S和该正例作泛化，得到

G = ｛怂 cir)｝

S = ｛ (z cir)｝

这时算法结束，输岀概念(z cir)0

该方法的主要缺点是：学习正例时，对S进行泛化，这往往扩大*S。*学习反例时，对G 进行特化，这往往扩大G。G和S的规模过大会给算法的实用造成困难。算法是在训练例 子引导下，对规则空间进行宽度优先搜索。对大的规则空间，算法慢得无法接受。

**6.3.2**观察与发现学习

观察与发现学习分为观察学习与发现学习两种。前者用于对事例进行概念聚类，形成 概念描述；后者用于发现规律，产生相应的规则。

1. 概念聚类

概念聚类是一种观察学习，是由米卡尔斯基(R・S. Michalski)在1980年首先提出来的。 其基本思想是把事例按一定的方式和准则进行分组，如划分为不同的类，不同的层次等，使 不同的组代表不同的概念，并且对每一个组进行特征概括，得到一个概念的语义符号描述。

例如对下列事例：

麻雀、乌鸦、喜鹊、鸡、鸭、鹅……

可根据它们是否家禽分为如下两类：

鸟=｛麻雀，乌鸦，喜鹊，…｝

家禽=｛鸡，鸭，鹅，…｝

这里，“鸟”“家禽”就是由聚类得到的新概念，并且根据相应动物的特征还可得知：

“鸟有羽毛，有翅膀，会飞，会叫，野生”

“家禽有羽毛，有翅膀，会飞，会叫，家养”

如果它们的共同特征提取出来了，就能得到“鸟类”的概念。

1. 发现学习

发现学习是从系统的初始知识、观察事例或经验数据中归纳出规律或规则。这是最困 难且最富创造性的一种学习。它可分为经验发现与知识发现两种，前者指从经验数据中发 现规律和定律，后者是指从已观察的事例中发现新的知识。

发现学习使用归纳推理，在学习过程中除了初始知识外，教师不进行指导，所以，它是无 教师指导的归纳学习。

**6.4**决策树学习

决策树又称为判定树，是常用于分类和预测的一种树形结构，是应用最为广泛的归纳推 理算法之一，决策树学习算法有很多，常用的有ID3、ID4、C4.5、CART等。

决策树是一种由节点和边构成的用来描述分类过程的层次数据结构。每个节点代表对 某一属性的一次测试，每条边代表一个测试结果，叶节点代表某个类或类的分布。在决策树 中，从根节点到叶节点的每一条路径都代表一个具体的实例，并且同一路径上的所有属性之 间为合取关系，不同路径为析取关系。



图6. 9决策树

决策树还可以表示成规则形式，如下所示。

IF age= youth AND income=high THEN 该客户会购买计算机

IF age= youth AND income=low THEN该客户不会购买计算机

IF age= middle aged THEN该客户会购买计算机

IF age= senior AND credit\_rating = fair THEN 该客户不会购买计算机

IF age= senior AND credit\_rating = excellent THEN 该客户会购买计算机

以下为决策树的经典学习算法——ID3算法。

ID3算法是J. R. Quinlan首先提出的o ID3算法通过使用信息增益(information gain) 来选择测试属性。这种度量称为属性选择度量或分裂的优良性度量。选择具有最高信息增 益的属性作为当前节点的测试属性。ID3的数学基础是信息嫡和条件嫡。而嫡实际上就是 系统信息量的加权平均。

定义6. 1对于数据集D,若对任意一个数据d(d属于D)有。个不同取值选项，那么 数据集D对于这c个状态的信息嫡为

*C*

Entropy (D)=习—P.logP,

1=1

其中，P,是数据集中*D*取值为z•的数据的比例。在ID3中，通常釆用以2为底的对数。

定义6.2在概率论和信息论中，信息增益是非对称的，用以度量两种概率分布P和Q 的差异。信息增益描述了当使用Q进行编码时再使用P进行编码的差异。通常P代表样 本或观察值的分布，也有可能是精确计算的理论分布。Q代表一种理论、模型、描述或者对 *P*的近似。当它用于文本数据的特征选择时，衡量的是某个词的出现与否对判断一个文本 是否属于某个类所提供的信息量，定义为某一特征在文本中出现前后的信息嫡之差。信息 增益的公式描述如下：

InfGain(w) = P(u>)力 P(c, | u>)logP(矛)寸 P(q I 矛Nog~~，了)~~

F(q) 切 *PCci)*

其中，P(s)是词s在文本中出现的概率,P(q)是g类文本在文本集中出现的概率,P(g|w)表 示文本包含词条侄时属于c类的条件概率，P(矛)表示文本中不包含词条w的文本的概 率,P(Ct |w)表示文本不包含词条B时属于C,类的条件概率，|C|为类别总数。

任何一个决策树算法，其核心步骤都是为每一次分裂确定一个分裂属性，即究竟按照哪 一个属性来把当前数据集划分为若干个子集，从而形成若干个“树枝气ID3算法采用信息 增益为度量来选择分裂属性的。哪个属性在分裂中产生的信息增益最大，就选择该属性作 为分裂属性。那么什么是信息增益呢？这需要首先了解“嫡”这个概念。嫡是数据集中的不 确定性、突发性或随机性的程度的度量。当一个数据集中的记录全部属于同一类的时候，则 没有不确定性，这种情况下的嫡为0。决策树分类的基本原则是，数据集被分裂为若干个子 集后，要使每个子集中的数据尽可能“纯”，也就是说子集中的记录要尽可能属于同一个类 别。如果套用炳的概念，即要使分裂后各子集的嫡尽可能小。

ID3算法步骤如下：

第1步，创建根节点。

第2步，根节点数据集为初始数据集。

第3步，根节点属性集包括全体属性。

第4步，当前节点指向根节点。

第5步，在当前节点的属性集和数据集上，计算所有属性的信息增益。

第6步，选择信息增益最大的属性作为当前节点的决策属性。

第7步，如果当前最大信息增益小于等于0,则当前节点是叶子节点，标定其类别，并标 记该节点已处理。执行第14步，否则执行第8步。

第8步，对属性A的每一个可能值生成一个新节点。

第9步，把当前节点作为新节点的父节点。

第10步，从当前节点数据集中选择属性A等于某值的数据，作为该值对应新节点的数据集。

第11步，从当前节点属性集中去除属性A,然后作为新节点的属性集。

第12步，如果新节点数据集或者属性集为空，则该新节点是叶子节点，标定其类别，并 标记该节点已处理。

第13步，标记当前节点已处理。

第14步，令当前节点指向一个未处理节点。如果无未处理节点，则算法结束，否则执行 第5步。

例6.2表6.1是一个顾客买车意向的训练集。

表6.1顾客买车意向

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| sample | age | income | health | buy |
| 1 | <30 | <3000 | good | no |
| 2 | <30 | <3000 | bad | no |
| 3 | <30 | >3000 | bad | yes |
| 4 | <30 | 23000 | good | yes |
| 5 | 30 〜60 | <3000 | good | yes |
| 6 | 30 〜60 | >3000 | good | yes |
| 7 | 30 〜60 | >3000 | bad | yes |
| 8 | >60 | <3000 | good | yes |
| 9 | >60 | <3000 | bad | no |
| 10 | >60 | >3000 | bad | no |

根据表6. 1首先计算最大信息蝸如下：

Entropy(D) =一 & X log2 濫一备 X Iog2 *% = 0.* 9710

然后计算每个属性的嫡。对于属性age,有3种取值，即3个子集，这样可以计算按照属 性age得到的嫡为

Entropy(age)=会(\_奇 \* "疫 y ~ y X 晶 3)+ 会(-亭 ' logz 言)

+ 泓—§ X 1狗§ — § X 晶奇)=0. 6755

信息增益为

InfGain(age) = Entropy(D) — Entropy(age) = 0. 2955

类似地，可以得到其余两个属性的信息增益：

InfGain( income) = 0. 1246

InfGain(health) = 0. 1246

由此可知，age的信息增益值最髙，故选之为测试属性。

从图6. 10可知，分支30〜60的样本都属于同一类，因此该分支节点为一个叶节点， 然后对另外两个节点子集进行属性选择，创建分支(方法同上)，直至分支节点全部为叶

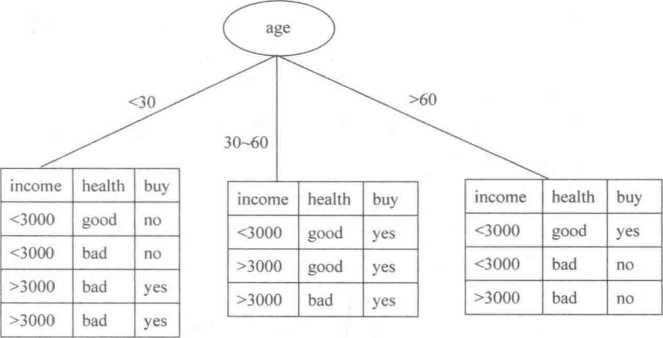


图6. 10基于信息增益生成的决策树

**6.5** 类比学习

机械学习是一种单纯依靠记忆学习材料，而避免去理解其复杂内部和主题推论的学习 方法。平时多称为死记、死背或死记硬背。美国心理学家奥苏伯尔提出与机械学习相对的 有意义学习概念，指符号所代表的新知识与学习者认知结构中已有的知识建立非实质性的 和人为的联系。这个理念也可以被描述为类比学习，这是一种很重要、很有效的学习方法。 运用类比，可迅速地把新旧知识进行对比、联系，可迅速发现同中的异，更加清晰地理解知 识；找出异中的同，构建知识网络。在日常生活中，类比现象是普遍存在的，它是寻找类似 事物并加以比较的一个过程，是一种重要的认知方法，是认识新事物的一种有效的思维 方式。

**6.5.1**类比学习的基本过程

类比学习的基本过程如下：

第1步，搜索匹配。当给定一个新问题时，首先搜索与之相似的已知规则。找出的规则 可能不唯一，此时可以依据其相似度从高到低进行排序。

第2步，选择规则。在第1步中可能存在不唯一的相似规则，此时选择最为相似的有关 规则。相似度越高，越有利于提高推理的可靠性。

第3步，建立对应的关系。该过程是将第2步得到的最相似规则与当前的问题之间建 立相似元素的对应关系，并建立相应的映射。

第4步,更新知识库。求解出新问题解之后，将此新问题及其解放入知识库。

**6.5.2**属性类比学习

属性类比学习是根据两个相似事物的属性实现类比学习。在该学习系统中，采用框架 来表示事物，其中，已知事物的框架称为源框架，目标事物的框架称为目标框架；使用框架 的槽来表示事物的属性。该学习过程是把源框架中的某些槽值传递到目标框架的相应槽中 去。该传递过程通常分为两步：

第1步，利用源框架产生若干个候选槽。

候选槽是指其值有可能传递给目标框架的槽。选择候选槽可用如下启发式规则：

•选用那些用极值填写的槽。

•选用那些确认为重要的槽。

•选用那些目标框架中没有的槽。

•选用那些目标框架中没有相应槽值的槽。

•选用源框架中的所有槽。

选择候选框架的过程是相继使用以上几条启发式规则，直至找到一组相似槽为止。

第2步，利用目标框架中的已有信息来筛选由第1步推荐的相似性。

选出一组候选框架之后，还必须用目标框架中的已有知识对这组候选框架进行筛选。 筛选时所使用的启发式规则如下：

•选择在目标框架中尚未填入的槽。

•选择在目标框架中为“典型”实例的槽。

•若上一步无槽可选，则选那些与目标有密切关系的槽。

•若仍无槽可选，则选那些与目标中的槽相似的槽。

•若仍无槽可选，则选那些与目标框架有密切关系的槽。

完成以上的选择后，可得到一组槽值，再把这组槽值分别填入目标框架的相应槽中，就 实现了源框架的某些槽值向目标框架的传递。

**6.5.3**转换类比学习

转换类比学习方法是基于“中间一结局分析”法发展起来的，是纽厄尔、西蒙等人在其完 成的通用问题求解程序(General Problem Solver,GPS)中提出的一种问题求解模型。其求 解问题的基本过程如下：

第1步，把问题的当前状态与目标状态进行比较，找出它们之间的差异。

第2步，根据第1步所得到的差异找出一个可减少差异的算符。

第3步，若该算符可以作用于当前状态，则该算符把当前状态改变为另一个更接近目标 的状态；若该算符不能作用于当前状态，即当前状态所具备的条件与算符要求的条件不一 致，则保留当前状态，并生成一个子问题，然后对此子问题用此法。

第4步，当子问题被求解以后，恢复保留的状态，继续处理原问题。

转换类比学习方法由外部环境获得与类比有关的信息，学习系统找出与新问题相似的 旧问题的有关知识,对这些知识进行转换，使之适用于新问题，从而获得新的知识。它主要 由回忆过程和转换过程两个过程组成。

回忆过程用于寻找新旧问题的差别，具体准则如下：

•新旧问题初始状态的差别。

•新旧问题目标状态的差别。

•新旧问题路径约束的差别。

•新旧问题求解问题可应用度的差别。

根据以上准则可以求出新旧问题的差别度，差别度越小，表示两者越相似。

转换过程是对旧问题的解进行适当的变换，使之成为求解新问题的求解方法。变换时, 其初始状态是与新问题类似的旧问题的解，即一个算符序列，目标状态是新问题的解。变换 中通过“中间一结局分析”法来减少目标状态与初始状态之间的差异，使初始状态逐步过渡 到目标状态,即求岀新问题的解。

转换类比的过程如图6.11所示。

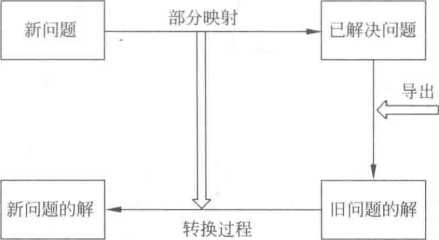


图6. 11转换类比

当遇到新问题时，将新问题映射到原先已经解决的问题中，如果部分映射，并且从已解 决问题中可以引导出解决新问题的方法，则在该方法的基础上通过匹配和转换得到新问题 的解决方法。

**6.5.4**派生类比学习

遇到新问题，将新问题映射到原问题中，在原有问题的基础上抽象出解决方法；同时, 新问题又能重新引导出另一个原先已解决的问题，即派生出另一个问题，而又能从该问题中 得出新的解决方法，此时便可以类比两个已解决的问题的解决方法，找岀相似之处，得出新 问题的解决方法。派生类比的过程如图6. 12所示。

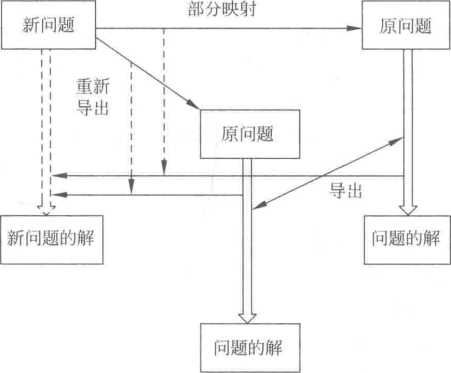


图6. 12派生类比

**6.5.5**联想类比学习

联想类比学习是把已知领域（源系统）的知识联想到未知领域（目标系统）的类比方法. 是一种综合的类比推理方法。

联想类比条件：

（1） 同构相似联想。

（2） 同态相似联想。

（3） 接近联想。

（4） 对比联想。

（5） 模糊联想。

类比学习法按原理可分为直接类比、拟人类比、象征类比、幻想类比、因果类比、对称类 比、仿生类比和综合类比8种。

（1） 直接类比。是从自然界或者人为成果中直接寻找出与创意对象相类似的东西或事 物，进行类比创意。例如，鲁班发明锯子，是从带齿的草叶把人手划破和长有齿的蝗虫板牙 能咬断青草获得直接类比实现的。

（2） 拟人类比。使创意对象“拟人化”，这种类比就是创意者使自己与创意对象的某种 要素认同、一致，自我进入“角色”，体现问题，产生共鸣，以获得创意。例如，凯库勒梦见一条 蛇咬住自己的尾巴，由此提出了苯分子环状结构理论。

（3） 象征类比。是一种借助事物形象或象征符号表示某种抽象概念或情感的类比。例 如，麦克斯韦用数学公式表示出了法拉第的电磁变化理论。

（4） 幻想类比。是在创意思维中用超现实的理想、梦幻或完美的事物类比创意对象的 创意思维法。例如，在凡尔纳的小说中有霓虹灯、可移动的人行道、空调机、摩天大楼、坦克、 电子操纵潜艇、导弹，在20世纪，这些东西都成为现实。

（5） 因果类比。两个事物之间可能存在着同一种因果关系。例如，在合成树脂中加入 发泡剂，可得到质轻、隔热和隔音性能良好的泡沫塑料，于是有人就用这种因果关系，在水泥 中加入一种发泡剂，结果发明了既质轻又隔热、隔音的气泡混凝土。

（6） 对称类比。自然界和人造物中有许多事物或东西都有对称的特点。例如，物理学 家狄拉克从描述自由电子运动的方程中得出正负对称的两个能量解。知道了电荷正负的对 称性，狄拉克又从对称类比中提出了存在正电子的对称解，其结果被实践证实。

（7） 仿生类比。人在创意、创造活动中,常将生物的某些特性运用到创意、创造上。例 如，仿鸟类展翅飞翔，造出了具有机翼的飞机。

（8） 综合类比。事物属性之间的关系虽然很复杂，但可以综合它们相似的特征进行类 比。例如，设计一架飞机，先做一个模型放在风洞中进行模拟飞行试验，就是综合了飞机飞 行中的许多特征进行类比。

历史上，许多重大的科学发现、技术发明和文学艺术创作是运用类比创意技法的硕果。 在科学领域，惠更斯提出的光的波动说就是与水的波动、声的波动类比而发现的；欧姆将其 对电的研究和傅里叶关于热的研究加以类比，建立了欧姆定律；医生詹纳发现种牛痘可以 预防天花，是受到挤牛奶女工感染牛痘而不患天花的启示，等等。在技术领域，控制论创始 人维纳通过类比把人的行为、目的等引入机器，又把通信工程信息和自动控制工程的反馈概 念引入活的有机体，从而创立了控制论；皮卡尔父子利用平流层理论先设计平流层气球飞 过15 690m髙空，又通过类比设计出世界上下潜深度最大的深潜器，下潜深度达到 19 168m；而仿生学的迅猛发展更说明了类比学习的重要性。

**6.6**解释学习

1983年，美国伊利诺伊大学的Dejong提出了解释学习，3年后由米切尔等人又提出了 基于解释的概括化的统一框架，即通过运用相关的领域知识，对当前的一个实例进行分析， 构造解释结构，然后对该解释结构进行概括化，得到相应知识的一般性描述。

1. 解释学习的基本原理

这里主要讨论米切尔等人提出的解释泛化学习方法。其框架的一般性描述如下。

已知：目标概念、训练实例、领域理论、操作性准则。

求出：满足操作性准则的关于目标概念的充分条件的概念描述。

其中，目标概念是需要学习的概念；学习实例是为学习系统提供的一个实例，在学习系 统中有着很重要的作用，它应能充分说明目标概念；领域理论是指相关领域的事实与规则， 在学习系统中作为背景知识；操作性准则用于指导学习系统对用来描述目标的概念进行取 舍等，使得通过学习产生的关于目标概念的一般性描述成为便于以后使用的一般性知识。

1. 解释学习的基本过程

从以上框架可知，解释学习的一般过程可以分为构造解释结构和获取一般性知识描述 两个步骤。

第1步，在该步骤过程中主要是解释提供给系统的实例为什么是满足目标概念的一个 实例。当用户输入实例后，系统首先进行问题求解，从目标开始方向推理，通过领域知识获 得目标匹配，有了解释的结果，便得到了一个解释结构。

例**6.3**假设要学习的目标是“一个物体。可以安全地放置在另一个物体厶的上面”， 即目标概念为

Safe-to-Stack (a, 6)

首先对学习的目标概念进行逻辑描述:

物体1在物体2的上面

On(obji »obj2)

Isa(obji , book)

Isa(obj2 *9* table) VolumeCobji , 1)

Density(obji , 0. 1)

物体1是书

物体2是桌子

物体1的体积是1

物体1的密度是0.1

领域知识是把一个物体安全地放置在另一个物体上面的准则:

「Fragile(6)-\*Safe-to-stack(a, *b)*表示如果*b*不是易碎的，则q可以安全地放到*b*的上 面；Lighter(a, 6)-\*Safe-to-stack(a, b)表示如果q比方轻，则。可以安全地放到。的上面； Volume(*p*, 〃)A Density//>, *d')* A Product(v, *d*, w)—►WeightC*p,* b)表示如果力的体积是 涉，密度是注,w乘以d的积是b,则*p*的重量是3； Is-a(p, table)->Weight(p, 5)表示如果 力是桌子，则 *P* 的重量是 5 ； Weight ( *p{ ,* w1 ) A Weight ( » w2 ) A Smaller *Ctv} ,* w2) -\*

Lighter(力I，p2)表示如果0的重量是Wi *,p2*的重量是w2 ,Wi比％小，则》i比力2轻。

其证明过程是一个由目标引导的逆向推理，这样我们最终可以得到一个解释树，树的结

构如图6. 13所示。

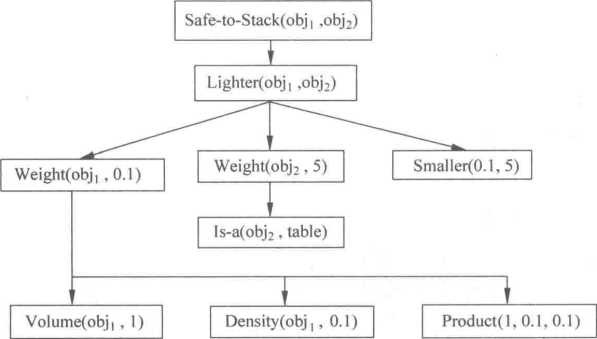


图 6. 13 Safe-to-Stack(obji ,obj2)解释树

第2步，主要是从第1步所得到的解释结构进行一般性知识描述。这样的过程实际上 是将特殊问题抽象化的过程。根据以上事例所得到的概括性解释结构如图6.14所示。由

此将一般化解释结构的所有叶节点合取作为前件，以顶点的目标概念为后件，略去解释结构 的中间部件，就生成如下一般性知识：

VolumeCO! ,5 ) A Density (Q *A* Product (5 0 ,叫)A Is-a(Q , table) A Smaller (wi ,5)-> Safe-to-stack(Oi *9O2)o*

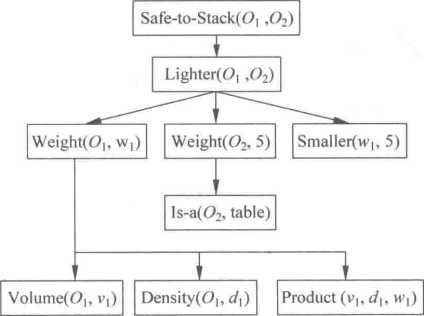


图 6. 14 Safe-to-StackCO] ,。2）解释树

**6.7**神经学习

神经学习是一种基于人工神经网络的学习方法。神经学习主要是神经网络的训练过 程，其主要表现为联结权值的调整。本节主要讨论感知器学习、BP网络学习和Hopfield网 络学习。

**6.7.1**感知器学习

感知器模型由美国学者F. Rosenblat于1957年提出，是一种早期的神经网络模型，感 知器模型中第一次引入了学习的概念，也就是说，可以用基于符号处理的数学方法来模拟人 脑所具备的学习功能。根据网络中所拥有的计算节点的层数，将其分为单层感知器和多层 感知器。本节主要讨论单层感知器学习模型。

1. 单层感知器学习算法思想

单层感知器学习基于迭代的思想，通常采用误差校正学习规则的学习算法，其主要思想 是利用神经网络的期望输出与实际输出之间的偏差作为联结权值和阈值调整的参考。

设XS)和W3)分别表示学习算法在第*n*次迭代时输入向量和权值向量，通常，可以 将阈值作为神经元权值向量WS)的第一个分量加到权值向量中，也可以设置其值为0,对 应地把一1固定地作为输入向量X3)的第一个分量，这样输入向量和权值向量可分别写成 如下的形式：

X(") = (― 1 口1 (〃)，了2 3)，…，有(〃))，其中 Zo(〃)=+ 1

W(〃)= (0(n) ,wi(7?) ,w2 (n) ,••• ,wm(w)),其中 w0 (n) =9(〃)

定义如下激活函数：

r+i, e a

g)=〈  
〔一1, *jc 尘 A*

. 下面给出单层感知器学习算法描述：

*I* (1)设置变量和参数。f(z)为激活函数以3)为网络实际输出，为期望输出w为 1 学习速率，〃为迭代次数/为实际输出与期望输出的误差。

1. 初始化联结权值和阈值。给权值向量祯,(0)3 = 0,1，…，m)分别赋一个较小的非 零随机数作为初值。其中，叫(0)是第0次迭代时输入向量中第*i*个输入的连接权值，。(0) 是第0次迭代时输出节点的阈值。
2. 输入一组新的样本X3) = ( — 1,]](/2),\*23)，・・・，&(〃))，并给出期望输出*d3)。*
3. 计算网络的实际输出：=八脖叫(〃)与3)一。3))°
4. 计算期望输出与实际输出差：*e=cl(n)— y(n) o*
5. 判断当前误差是否满足终止条件。若满足，则算法结束；否则，将〃值加1,并用下 式调整权值：叫(〃 + 1)=叫3)+ ][d(〃)一、(〃)]円(〃)，然后转步骤(3)。
6. 单层感知器学习算法实例

例**6.4**用单层感知器实现逻辑与运算。

解：逻辑与的真值如表6. 2所示，其问题转换如图6. 15所示。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 表6.2逻辑与真值表 | | | | 问题转换 | 输入向量：  Xi=(0,0,l,l)； *X2* = (0,1,0,1) |
| Xi | x2 | Y | |
| 0 | 0 |  | 0 | I > |  |
|  |  |  | 输出向量：  K = (0,0,0,1) |
| 0 | 1 |  | 0 |  |
| 1 | 0 |  | 0 |  |
| 1 | 1 |  | 1 | 图 | 6.15问题转换 |

设初始联结权值、阈值以及学习速率取值如下：

W] (0) = 0. 4, w2 (0) = 0. 8, 。(0) = 0.6,，^=0. 4 算法的具体学习过程如下：

设两个输入为⑥(0) = 0和互(0)=0,其期望输出为d(0)=0,实际输出为

)(0)= /(wi (0)x1 (0) + 用2(0)互(0) 一。(0))

=/(0.4X0 + 0.8X0-0.6) = /(-0. 6) = 0

实际输出与期望输出相同，不需要调节权值。再取下一组输入：心(0)= 0和互(0)= 1,期望输出』(0)=0,实际输出为

、(0)= /(W] (0)(0) +w2(0)x2(0) —(9(0))

=/(0. 4 X 0 + 0. 8 X 1 -0. 6) = /(0. 2) = 1

实际输出与期望输出不同，需要调节权值，其权值调整如下：

。(1) =。(0) + 7(^(0) 一 贝0)) X (-1) = 0. 6 + 0.4 X (0- 1) X (-1) = 1

Wi (1)=叫(0) + 卩((7(0) — j/(0))zi(0) = 0. 4 + 0. 4 X (0 — 1) X 0 = 0. 5

w2(1) = 32(0) + Nd(°)— 丁(0)处2(0) = 0.8 + 0. 4X (0 一 1) X 1 = 0. 4

取下一组输入：R (1) = 1和x2(l) = O,其期望输出为*d⑴=0,*实际输出为

夕(1)= *f(wi* (1)X1 (1) + w2 (1 )x2(l) —0(1))

*=f(0.* 5 X 1+0.4X0-1) = /(- 0. 5) = 0

实际输出与期望输出相同，不需要调节权值。再取下一组输入：⑥(1) = 1和及(1)= 1，其期望输出为。(1) = 1,实际输出为

>(!)=八叫(1)]】(1) +叫(1\*2(1)一 0(1))

=/(0. 5X1+0. 4X1-1)= /(-0. 1) = 0

实际输出与期望输出不同，需要调节权值，其权值调整如下：

*0(2)* = 9(1)+ ?以(1)-丁(1))X(—1) = 1+0. 4 X (1-0) X (-1) = 0.6

W] (2) = Wi (1) + *7j(d(l) —* (1) = 0. 5 + 0. 4X(1 — 0)X1 = 0. 9

*w2(2) =* w2(l) *+ 7)(. d(.l)* — >(l))x2 (1) = 0. 4 + 0. 4X (1—0) XI = 0. 8

取下一组输入：约(2)=。和如(2)=0,其期望输出*为d(2)=0,*实际输出为

3/(2) = *f(0.* 9X0 + 0. 8X0-0. 6) = /(- 0. 6) = 0

实际输出与期望输出相同，不需要调节权值。

再取下一组输入：心(2) = 0和x2(2) = 1,期望输出为*d⑵=0,*实际输出为

*y(2)* = /(0. 9X0 + 0. 8 X 1—0. 6) = *f(0.* 2) = 1

实际输出与期望输出不同，需要调节权值,其调整如下：

0(3) = *0(2) + 7/(d(2) - y(2)) X* (- 1) = 0. 6 + 0. 4 X (0 — 1) X (—1) = 1

也 1(3)=的(2) + ?以(2) — *y(2))xl(2)* = 0. 9 + 0. 4 X (0 - 1) X 0 = 0. 9 w2(3) = w2(2) + *cl(2) — y(2))j：2(2)* = 0. 8 + 0. 4 X (0 — 1) X 1 = 0. 4 此时的阈值和连接权值满足结束条件，算法可以结束。

对此,可检验如下：

对输入“0。”有 ^ = /(0. 9X0 + 0. 4X0-1) =/(-l)=0o

对输入“0 1"有 3/ = /(0. 9X0+0. 4X0. l-l)=/(-0.7)=0o

对输入“1。"有 J = /(O. 9X1+0. 4X0-1) = /(-0. l) = 0o

对输入T 1"有 *y=f(0.* 9X1+0. 4X1-1) ^=y(0. 3)=0o

**6.7.2**反向传播网络学习

反向传播网络(Back-Propagation Network,简称BP网络)学习算法也称误差反向传播 算法，是由Rumelhart和Meclelland于1985年提出的，实现了明斯基的多层网络设想，解决 了前馈神经网络的学习问题，即自动调整网络全部权值的问题。

BP网络是将W-H学习规则一般化，对非线性可微分函数进行权值训练的多层网络。 BP网络是一种多层前向反馈神经网络，其神经元的变换函数是S形函数，因此输出量为 0〜1的连续量，它可以实现从输入到输出的任意非线性映射。

由于其权值的调整采用反向传播的学习算法，因此称为反向传播网络。

W-H学习规则即Widrow-Hoff学习规则，是纠错学习规则的一种，于1962年由 Widrow和Hoff提岀，这种学习规则使神经元的期望输出与实际输出之间的平方差最小， 因此也称为最小均方差(LMS)学习规则。

BP网络主要用于以下方面：

（1） 函数逼近，用输入矢量和相应的输出矢量训练一个网络逼近一个函数。

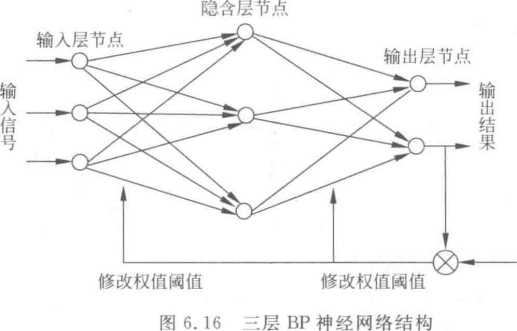
（2） 模式识别，用一个特定的输出矢量将它与输入矢量联系起来。

（3） 分类，把输入矢量以所定义的合适方式进行分类。

（4） 数据压缩，减少输出矢量维数以便于传输或存储。

**1. BP**网络模型与结构

神经网络模型与结构如图6. 16所示。



其中，左=1, 1= 1,2,si,项=1,2,• • m

一个具有厂个输入和一个隐含层的神经网络模型结构感知器和自适应线性元件的主要 差别在激活函数上：前者是二值型的，后者是线性的。BP网络具有一个或多个隐含层，除 了在多层网络上与前面已介绍过的模型有不同外，其主要差别也体现在激活函数上。

BP网络的激活函数必须是处处可微的，所以它不釆用二值型的阈值函数｛0,1｝或符号 函数（-1,1｝,BP网络经常使用的是S形的对数或正切激活函数或线性函数，如图6. 17 所示。

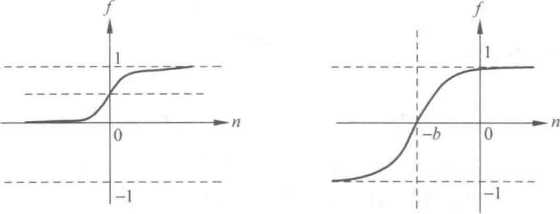


图6.17 BP网络S形激活函数

因为S形函数具有非线性放大系数功能，它可以把输入从负无穷大到正无穷大的信号 变换成一1〜1的输出，对较大的输入信号，放大系数较小；而对较小的输入信号，放大系数 则较大，所以采用S形激活函数可以处理和逼近非线性的输入输出关系。

只有当希望对网络的输出进行限制，如限制为。〜I,那么在输出层应当包含S形激活 函数，在一般情况下，均是在隐含层采用S形激活函数，而输出层采用线性激活函数。

BP网络的特点如下：

（1） 输入和输出是并行的模拟量。

（2） 网络的输入输出关系由各层连接的权因子决定，没有固定的算法。

（3） 权因子是通过学习信号调节的，因此学习越多，网络越聪明。

（4） 隐含层越多，网络输出精度越高，且个别权因子的损坏越不容易对网络输出产生大 的影响。

**2. BP**学习规则

BP算法属于S算法，是一种监督式的学习算法。其主要思想为：对于q个输入学习样 本P「P2,…，P”已知与其对应的输出样本为匸，取，…，学习的目的是用网络的实际 输出A ,4,…①，与目标矢量*T}*工，…，L之间的误差来修改其权值，使4 （1=1,2,…， g）与期望的Tl尽可能地接近，即，使网络输岀层的误差平方和达到最小。

BP算法由两部分组成：信息的正向传递与误差的反向传播。在正向传播过程中，输入 信息从输入经隐含层逐层计算传向输出层，每一层神经元的状态只影响下一层神经元的状 态。如果在输出层没有得到期望的输出，则计算输出层的误差变化值，然后转向反向传播， 通过网络将误差信号沿原来的连接通路反传回来，修改各*P/ J* 叫〃 z 层神经元的权值，直至达到期望目标。

其中以=1,2,…，S23 = 1,2,・・・，S]〃 = 1,2,・・・，C

设输入为输入神经元有“个，隐含层内有si个神经 -

元，激活函数为F],输出层内有毎个神经元，对应的激活函 图6.18具有一个隐含层的

数为戸2,输出为A,目标矢量为To 简化网络图

**3** .信息的正向传递

（1） 隐含层中第/个神经元的输出为

*S = fl* （ + 缶J， 2 = 1,2,-- ,51

（2） 输出层第为个神经元的输出为

*a2k = ft* I + 缶J， & = 1,2, •・・，S2

（3） 定义误差函数为

E（W,B） =!寸（益一妃尸

1. 利用梯度下降法求权值变化及误差的反向传播

1）输出层的权值变化 对从第z个输入到第&个输出的权值有

△切2妇*=—7/ =— T）* X :""一 =,（益—*alk}fz*

*1 1 3a2k 3 侦 2玳*

其中：

*Ski* =（讣—*a2k*）= *^kf* 2

*ek = tk— a2k*

同理可得

厶 \_\_ *3E* \_\_ 3E 3知  
△皈I依一依X减

2）隐含层权值变化

对从第顶个输入到第Z个输出的权值，有

A \_ 3E \_ *dE da2k dau*△叫L「顽一宙**F><**顽

$2

*=— Sk) 饥=r)"Pj*

为=】

其中：

同理可得

s2

*&j = ^if* **1**， 七=£ **3**血也**2**如

*k=\*

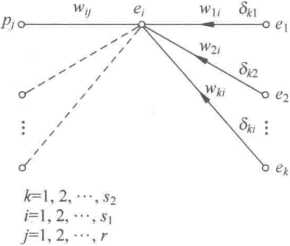


图6. 19误差反向传播法的图形解释

△如*=T]8ij*

1. 误差反向传播的流程图与图形解释

误差反向传播过程实际上是通过计算输出层的误差％，然后将其与输出层激活函数的 一阶导数兄相乘来求得如。由于隐含层中没有直接给 出目标矢量，所以利用输出层的反向传递来求出隐含

s2

层权值的变化量△^2如。然后计算*们=网2K，*并同

\*=1

样通过将e,与该层激活函数的一阶导数呂相乘，而求得 禺，以此求出前层权值的变化量△叫八如果前面还有 隐含层，沿用上述同样方法以此类推，一直将输出误差 久一层一层地反推算到第一层为止。误差反向传播法 的图形解释如图6. 19所示。

**6. BP**网络的训练过程

为了训练一个BP网络，需要计算网络加权输入矢量以及网络输出和误差矢量，然后求 得误差平方和。当所训练矢量的误差平方和小于误差目标时，训练则停止；否则在输出层 计算误差变化，且采用反向传播学习规则来调整权值，并重复此过程。当网络完成训练后, 对网络输入一个不是训练集合中的矢量，网络将以泛化方式给出输出结果。为了能够较好 地掌握BP网络的训练过程，以两层网络为例来叙述BP网络的训练步骤。

（1）用小的随机数对每一层的权值W和偏差B初始化，以保证网络不被大的加权输入 饱和；并进行以下参数的设定或初始化：

1. 期望误差最小值error\_goalo
2. 最大循环次数max\_epoch0
3. 修正权值的学习速率丘,一般情况下*k = 0.* 01-0. 7O
4. 从1开始的循环训练：for epoch=l: max\_epocho

(2)计算网络各层输岀矢量A和A?以及网络误差E：

A] = tansig(Wi \* )

*A2* =purelin(W2 \* Aj ,B2)

*E=T~A*

1. 计算各层反传的误差变化*D2*和D.并计算各层权值的修正值以及新权值：

D2 =deltalin(A2 ,E)

*Di* =deltatan(Aj *,D2* ,W2)

[dW| ,dB1] = learnbp(P,D1 ,lr)

*[\_dW2* »dB2] = learnbp(Ai ,D2, lr)

W】=W]+dW】； B1=B1+dB1

*W2=W2+dW2；* B2 = B2 + dB2

1. 再次计算权值修正后误差平方和：

SSE = sumsqr(*T —* purelin(W2 \* tansig( \* P,Bj ) ,B2 ))

1. 检查SSE是否小于err\_goal,若是，训练结束；否则继续。

以上所有的学习规则与训练的全过程仍然可以用函数trainbp. m来完成。它的使用同 样只需要定义有关参数：显示间隔次数、最大循环次数、目标误差以及学习速率，而调用后 返回训练权值、循环总数和最终误差：

TP= [disp\_freq max\_epoch err\_goal lr]

*[W, B* 9 epochs, errors] = trainbp(W *,B',F',P,T,* TP)

例如，有21组单输入矢量和相对应的目标矢量，试设计神经网络来实现这对数组的函 数关系。

P=-l： 0. 1： 1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| r=[-0. 96 | 0.577 | —0.0729 | 0. 377 | 0. 641 | 0. 66 | 0. 461 |
| 0.1336 | —0. 201 | -0.434 | —0. 5 | —0. 393 | -0. 1647 | 0.0988 |
| 0. 3072 | 0. 396 | 0.3449 | 0.1816 | —0.0312 | —0.2183 | -0. 3201] |

测试集：

p2 = -l： 0. 025： 1

泛化性能：使网络平滑地学习函数，使网络能够合理地响应被训练以外的输入。

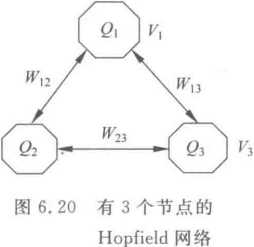
要注意的是，泛化性能只对被训练的输入-输出对最大值范围内的数据有效，即网络具 有内插值性，不具有外插值性。超出最大训练值的输入必将产生大的输出误差。

**6.7.3 Hopfield** 网络学习

Hopfield网络学习的过程实际上是一个从网络初始状态向其稳定状态过渡的过程。 为了描述Hopfield网络的稳定性，本节首先介绍Hopfield网络的能量函数。

**1. Hopfield**网络的能量函数

离散Hopfield网络的能量函数可定义为

*E =-*

林1

式中/是网络中的神经元个数，是第I个神经元和第j个神经元之间的连接权值，且有 凹=凹；叫和巧分别是第i个神经元和第丿个神经元的输出；*0.*是第z个神经元的阈值。 可见，无论神经元的状态由0变为I还是由I变为0,都总 有△EV0。它说明离散Hopfield网络在运行中，其能量 函数总是在不断降低，最终将趋于稳定状态。

例**6.5** 图6. 20为3个节点的Hopfield网络，若给

定的初始状态为卩。={1,0,1},各节点之间的联结权值为 \*

W]2 = W21 = 1 , W13 = W31 = — 2 , W23 = *侦32* = 3

各节点的阈值为

*0\ =—* 1 > 。2 = 2 , = 1

请计算在此状态下的网络能量。

解：*E =—*血+凹13勒必+32]寸2以1 +w23V2 *V3* +w3| V3V1 +TV32P3 也）

*+ 9\ V\* +但戏 +。3 *力3*

=—(W12V1 *V2* +wi3 V] *V3* +S23P2S)+仇 *5 +02V2* +83

= -(lXlX0 + (-2)XlXl + 3X0Xl) + ( —DX1 + 2X0 + 1X1

=2

**2. Hopfield**网络学习算法

（1）设置互连权值：

X/ Kt， z尹顶

= < 5 = 1

Io, *x = j* 其中*送为S*类样例（即记忆模式）的第，个分量，它可以为1或0,样例类别数为S，节点数 为*no*

（2） 对未知类别的样例初始化：

*yi（t） = Xi,* 1 < C W 以

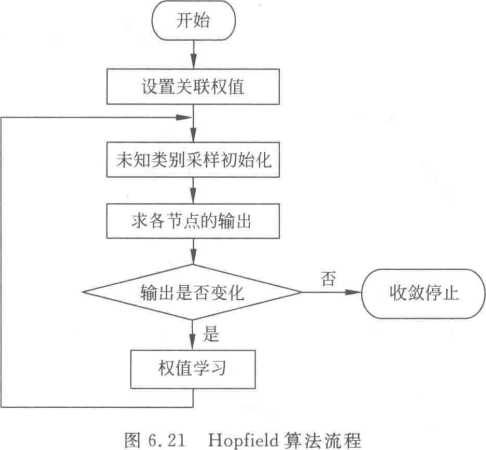
其中，y（Q为节点，时刻z的输出，y（0）是节点的初值；幻为输入样本的第/个分量。

（3） 迭代运算：

y （, + 1） = /、（ »叫必（/）） , 1 < J — 1

其中，函数f为阈值型。重复这一步骤，直到新的迭代不能再改变节点的输出，即收敛为 止。这时，各节点的输出与输入样例达到最佳匹配。否则转第（2）步继续。

Hopfield算法流程如图6.21所示。



**6.8** 贝叶斯学习

**6.8.1**贝叶斯定理

贝叶斯定理用于解决以下问题：已知某条件概率，如何得到两个事件交换后的概率，也 就是在已知P(A|B)的情况下如何求得P(B|A)O这里先解释什么是条件概率。

*P(B\A)*表示事件*B*已经发生的前提下事件A发生的概率，称为事件B发生的条件下 事件A的条件概率。其基本求解公式为

*P(A* | B)=

*P(AB)*

P(B)

贝叶斯定理之所以有用，是因为人们在生活中经常遇到这种情况：可以很容易直接得 岀P(A|B),P(B|A)则很难直接得出，但我们更关心P(B|A),贝叶斯定理打通了从P(A|B) 获得P(B|A)的道路。

|  |  |
| --- | --- |
| 贝叶斯定理如下： | F(B | A)= ~~四孔從~~ |

式中P(B|A)是后验概率，是在条件A发生的情况下B发生的概率；P(H)是先验概率，或 称为*H*的先验概率，即不考虑其他的情况下*H*发生概率；P(A|B)和P(B| A) 一样，也是 后验概率。

**6.8.2**朴素贝叶斯分类算法

朴素贝叶斯分类算法是基于贝叶斯定理的，它的工作过程如下：

(1)每个数据样本用一个〃维特征向量X={.,丑，…,石}表示，分别描述对72个属性

**Al** ,人**2,…，A”**样本的〃个度量。

**(2)**假定有**m**个类**G，C2，**・・・，**G**。给定一个未知的数据样本**X(**即没有类标号)，分类 法将预测**X**属于具有最髙后验概率(条件**X**下)的类，即，朴素贝叶斯分类将未知的样本分 配给类**G，**当且仅当

**P(C, I X)> P(C7 I X),** *1 Wj Wm打殆 i*

这样即最大化**P(G lx)**。其**P(GIX)**最大的类**G**称为最大后验假定。根据贝叶斯定 理，有

**P(C,**

**X)=**

**P(x I G)P(C)**

**PCX)**

1. 由于**p(x)**对于所有类为常数,只需要**P(X|G)P(G)**最大即可。如果类的先验概率 未知，则通常假定这些类是等概率的，即**P(G) = P(G)= ・・・ = P(C”)**。并据此只对**P(X|C)** 最大化。否则，最大化**P(X|G)P(C,)**。注意，类的先验概率可以用**P(C,) = s,/s**计算，其中 **5,**是类**G**中的训练样本数*，*而**5**是训练样本总数。
2. 给定具有许多属性的数据集，计算**P(x|G)**的开销可能非常大。为降低计算**P(x| G)**的开销，可以作类条件独立的朴素假定。给定样本的类标号，假定属性值相互条件独立，

即在属性间不存在依赖关系。这样，*P(X* **| Q) = [[PS**丨**G),**概率**P3 |C,),P(i2**丨 *k=\*

**G),…，P(MG)**可以由训练样本估值，其中：

1. 如果**A.**是分类属性，则其中电是在属性上具有值孔的类**G** 的样本数，而$是**C,**中的训练样本数。
2. 如果① 是连续值属性，则通常假定该属性服从高斯分布，因而有

] W

| G)= *g(.xk* s(. ) = e 2\*

J2jt 气

其中，给定类**C**的训练样本属性**A\***的值*,g(m，&*，吒)是属性&的高斯密度函数，而&、 气分别为平均值和标准差。

**(5)**为对未知样本**x**分类，对每个类**G，**计算**P(X|G)P(G)**。样本**x**被指派到类**G，** 当且仅当

*PCX* **I** *G)P(Cj) > PCX* **I** *Cj)P(Cj),* **1 < j <** *?n9j i*

换言之,**X**被指派到其**F(X|G)P(G**)最大的类**C1O**

朴素贝叶斯分类算法的工作流程如图**6. 22**所示：

可以看到，整个朴素贝叶斯分类算法分为**3**个阶段。

第一阶段一一准备工作阶段。这个阶段的任务是为朴素贝叶斯分类做必要的准备，主 要工作是根据具体情况确定特征属性，并对每个特征属性进行适当划分，然后由人工对一部 分待分类项进行分类，形成训练样本集合。这一阶段的输入是所有待分类数据，输出是特征属 性和训练样本。这一阶段是整个朴素贝叶斯分类中唯一需要人工完成的阶段，其质量对整个过 程将有重要影响，分类器的质量很大程度上由特征属性、特征属性划分及训练样本质量决定。

第二阶段——分类器训练阶段。这个阶段的任务就是生成分类器，主要工作是计算每 个类别在训练样本中的岀现频率及每个特征属性划分对每个类别的条件概率估计，并将结 果记录。其输入是特征属性和训练样本，输出是分类器。这一阶段是机械性阶段，根据前面

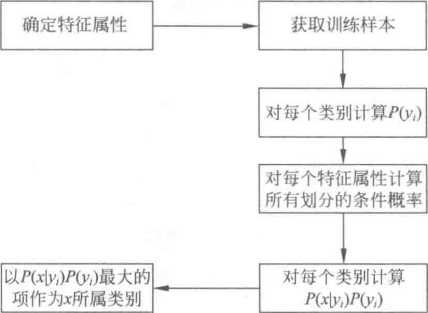


图6.22朴素贝叶斯分类算法流程图

讨论的公式可以由程序自动计算完成。

第三阶段一一应用阶段。这个阶段的任务是使用分类器对待分类项进行分类，其输入 是分类器和待分类项，输岀是待分类项与类别的映射关系。这一阶段也是机械性阶段，由程 序完成。

例**6.6**购买计算机实例的数据集如表6. 3所示。

Class：

Cl: buys\_computer= "yes\*'

C2: buys\_computer= "no"

Data sample

X=(age< = 30,

Income = medium,

student = yes,

credit\_ra t ing=Fair )

表6.3购买计算机实例的数据集

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| age | income | student | credit\_rating | buys\_computer |
| <30 | high | no | fair | no |
| <30 | high | no | excellent | no |
| 31 〜40 | high | no | fair | yes |
| \* >40 | medium | no | fair | yes |
| >40 | low | yes | fair | yes |
| >40 | low | yes | excellent | no |
| 31 〜40 | low | yes | excellent | yes |
| <30 | medium | no | fair | no |
| <30 | low | yes | fair | yes |
| >40 | medium | yes | fair | yes |
| <30 | medium | yes | excellent | yes |
| 31 〜40 | medium | no | excellent | yes |
| 31 〜40 | high | yes | fair | yes |
| >40 | medium | no | excellent | no |

1. 数据样本属性：age, income,credit\_ratingo
2. 类别属性：buys\_computero

*yx* : buys\_computer="yes"

*y2*: buys\_computer= "no"

1. 计算每个类的先验概率P(y)：

P3i)= 9/14 = 0. 643

P(W2)= 5/14 = 0. 357

1. 计算每个特征属性对于每个类别的条件概率：

P(age^"30" | buys\_computer= "yes") = 2/9 = 0. 222

P(income^"medium" | buys\_computer= "yes") =4/9 = 0. 444

*P*(student^Hyes" | buys\_computer= "yes") = 6/9 = 0. 667

*P*(credit\_rating^"fair" | buys\_computer= "yes") =6/9 = 0. 667

P(age<"30" I bttys\_computer = "no") = 3/5 = 0. 600

*P*(income^Hmedium" | buys\_computer= "no") =2/5 = 0. 400

*P*(student^\*'yesn | buys\_computer=nno") = 1/5 = 0. 2

P(credit\_rating^"fair" | buys\_computer= "no") = 2/5 = 0. 400

1. 计算条件概率P(X|y)：

PCX I buys\_computer=" yesH) =0. 222X0. 444X0. 667X0. 667 = 0.044

P(X|buys\_computer=”no”)=0. 600X0. 400X0. 200X0. 400 = 0. 019

1. 计算对于每个类y的P(X|y)P(y)：

P(X| buys\_computer= "yes")P(buys\_computer= \*'yes") =0. 044X0. 643 = 0. 028 P(X| buys\_computer= " no") P( buys\_computer = \*'no") = 0. 019 X 0. 357 = 0. 007 因此，对于样本X,朴素贝叶斯分类算法预测buys\_computer =”yes”。

特别要注意的是：朴素贝叶斯分类算法的核心在于它假设向量的所有分量之间是独立的。 朴素贝叶斯分类算法的优点如下：

•算法逻辑简单，易于实现。

•分类过程中时空开销小。

•算法稳定，对于不同的数据特点其分类性能差别不大，健壮性比较好。

**6.9**在线机器学习

在线机器学习(online learning)是指每来一个样本，就利用迭代方法更新模型变量，使 得当前的期望损失最小，因此需要及时处理收集的数据，并给出预测或建议结果，更新模型。 现在的在线机器学习常用到逻辑回归(logistic regression),在线机器学习算法中主要用到 在线梯度下降(OGD)和随机梯度下降(SGD)。

在线机器学习在一个时间点只处理一个样本，处理完后便可丢弃，避免了重复使用同一 个样本。但在线机器学习也有一个局限性，它很难产生真正稀疏的解。

下面介绍几种提升模型稀疏性的在线最优化求解算法，包括截断梯度法、前向后向切分 算法、正则对偶平均算法、FTRL算法。

**6.9.1**截断梯度法

为了得到稀疏的特征权重W,最简单的一个方式就是设定一个阈值，当W某纬度上系 数小于这个阈值时将其设置为0。这种方法简单，也易于实现，但在实际中（尤其是在OGD 中），W的某个系数比较小有可能是因为该维度训练不足而引起的，简单进行截断会造成这 部分特征的丢失。

截断梯度法（Truncated Gradient,TG）是由 John Langford、Lihong Li 和 Tong Zhang 在2009年提出的，实际上是对简单截断的一种改进。下面首先描述L1正则化和简单截断 的方法，然后再来看TG对简单截断的改进以及这3种方法在特定条件下的转化。

1. **L1**正则化法

权重更新方式为

= w⑺*-/G"）-*矿。sgn（W⑺）

注意，这里AER是一个标量，且义20,为L1正则化参数。sgn（汾为符号函数，如果V= [如戏，…是一个向量，，是向量的一个维度，那么有

sgn（V） = [sgn（0）,sgn（w2），…，sgn（pN）] G

出）为学习率，通常将其设置为1亦的函数。G”）=V^（W⑺，Z"））代表了第*t*次迭代中损失 的梯度。由于OGD每次仅根据观测到的一个样本进行权重更新，因此也不再使用区分样 本的下标丿。

1. 简单截断法

简单截断法以为为窗口。当”A不为整数时，釆用标准的SGD迭代；当*t/k为*整数时， 釆用如下权重更新方式：

俨中）=*T0（W{ty \_《9。,。）*

[0,丨 4 IW。

*Tq <*

成其他

注意，其中*0. R*是一个标量，且。2。；如果V=Ei»v2 ,•••,%]£R"是一个向量*泓* 是 向量的一个维度，那么有

丁0 （p,。）= ETq （勺，。）]，To（々2，。），…，丁0 （wn，。）£ RN

1. 截断梯度法

截断梯度法是在简单截断法基础上的改进，同样是采用截断的方式，其权重更新方式为 W（S = Tj （w（r） - /G"），/双，。）

maxCO,^,- —a） , q £ [0,硏

*Ti* = y min（0+ □） , v, G [—9，°]

、4， 其他

其中A 6R且『20。TG同样是以，为窗口，每上步进行一次截断。当不为整数时 A（/）=0,当*t/k*为整数时义⑺=以。义和。决定了 W的稀疏程度，这两个值越大，则稀疏性越 强。尤其当令*X = 0*时，只需要通过调节一个参数就能控制稀疏性。

根据阶段梯度法的权重更新方式，可以很容易设计出TG的算法逻辑：

1. input *0*
2. initialize WE RN
3. for c= 1,2,3,•••do

4

1. refresh *W* according to

max(0,叫一/g, —产人⑴),(啊一舟g,) £ [0,硏  
w,=< min(0,w, —yz) *g：* + 产 V) , (sl gi)£ [ —。，。]

判—武g” 其他

1. end
2. return *W*

**6.9.2**前向后向切分算法

1. 算法原理

前向后向切分(Forward-Backward Splitting, FOBOS)是由 John Duchi 和 Yoram Singer提出的。从英文全称上来看，该方法应该叫FOB AS,但是由于一开始作者管这种方 法叫FOLOS(Forward Looking Subgradients),为了减少读者的困扰，作者干脆只修改了一 个字母，叫FOBOS0

在FOBOS中，将权重的更新分为两个步骤:

W(")

W(中)=argminjy || W\_W(")|| 2 + ^(W) |

前一个步骤实际是一个标准的梯度下降步骤，后一个步骤可以理解为对梯度下降的结 果进行微调。观察第二个步骤，发现对W的微调也分为两部分：前一部分保证微调发生在 梯度下降结果的附近；后一部分则用于处理正则化，产生稀疏性。

如果将这两个步骤合二为一，即将wQ++)的计算代入中，有

矿“ =argminj-j- || W-W(0 + V(Z)GO) || 2

令F(W)=§|| W —W^+产G”)||2 + ?(心)0(W)。如果存在一个最优解，那 么可以推断0向量一定属于F(W)的次梯度集合：

0 e 3F(W) *= W -* W(0 +/G⑴ 4-7?<r+i)a^(W)

由于=argminF(W),那么有

w

0 = {W\_W⑴ + 舟逆)+v(T)叩(W)} |w=w"D

上式实际给出了 FOBOS中权重更新的另一种形式：

W(s = 俨)—产G® —.井+”欧彼中)) 可以看出，不仅与迭代前的状态W⑵有关，而且与迭代后的欧(W"+D )有关，可能这就 是FOBOS名称的由来。

1. **L1-FOBOS**

在这里主要看FOBOS如何在L1正则化下取得比较好的稀疏性。

在L1正则化下，有欧（W）F || *W* || lo为了简化描述，用向量V = …:

来表示W（，I），用标量AER来表示？（，++〃，并将上面的FOBOS的权重更新公式等号的 ♦ 右边按纬度展开：

*N [*

W（/+1） = argmin" （w,- — v,）2 + A | *w,-* | ）

*N*

可以看到，在求和公式£ （号（叫一叫尸+丄丨的|）的每一项都是大于等于0的，所 以上述公式可以拆解成对特征权重W每一维度单独求解：

*N [*

= argmin〉2 （万（b,—叫T+）丨 s, | ）

FOBOS在L1正则化条件下，特征权重各个纬度更新的方式为

=sgnCx;, ）max（0, | 勺 | — A）

=sgn（w-n —武.g；" ）max{0, | *w-l） — .*|—丁什扣腭

其中，g\*为梯度G⑺在纬度，上的取值。

根据FOBOS在L1正则化条件下的特征权重各个纬度更新的方式可以很容易设计出 L1-FOBOS的算法逻辑：

1. input A
2. initialize W£ RN
3. for z= 1,2,3…do
4. G=VuZ（W,X（W））
5. refresh *W* according to

w, =sgn（w, —7y（,）g, ）max（0, | w, — *z）g, |* ）人}

1. end
2. return *W*
3. **Ll-FOBOS**与**TG**的关系

从LI-FOBOS的特征权重各个纬度的更新方式可以看出,Ll-FOBOS在每次更新W 的时候，对W的各个纬度都会进行判定，当满足|讪。一产药）|一"（，++）人《0时对该纬度 进行截断•令讪由）=0。那么怎么去理解这个判定条件呢？如果把判定条件写成I w!n - 矿）普,丨《"3扣\*那么这个含义就很清晰了：如果一个样本产生的梯度不足以令对应纬 度上的权重值发生足够大的变化（？（，++〃），则认为在本次更新过程中该纬度不够重要，应 当令其权重为0。

对于L1\_FOBOS特征权重的各个纬度的更新公式也可以写为如下形式：

5 仲， 丨讪。一矿侄牧

*w'iu = < .*

〔3归一*寸件*）—v（"〃sgn（M> — / # ）,其他

比较上式与TG的权重更新公式可知，如果令0=83 = 1,人終则L1- FOBOS与TG完全一致。可以认为L1-FOBOS是TG在特定条件下的特殊形式。

**6.9.3**正则对偶平均算法

1. 算法原理

正则对偶平均(Regularized Dual Averaging,RDA)是微软公司10年的研究成果，RDA是 简单对偶平均方案(Simple Dual Averaging Scheme)的一个扩展，由Lin Xiao发表在2010年。

在RDA中，特征权重的更新策略为

W(f+1> = argminjyj] <G(r) ,w> +^(W) *+^-h(W)[*

其中表示梯度g⑴对w的积分平均值,欧(W)为正则项％(W)为一个辅助的严格 凸函数，*｛叶*1｝是一个非负且非自减序列。

1. **L1-RDA**

L1-RDA特征权重的各个维度更新的方式为

0,

—专(gW — asgn(gW)),其他

可以发现，当某个维度上累积梯度平均值的绝对值| g-G) |小于阈值人的时候，该维度权 重将被置0,特征权重的稀疏性由此产生。

根据L1-RDA特征权重的各个维度更新的方式，可以很容易设计出L1-RDA的算法逻辑:

1. input y,A
2. initialize W £ R' , G= 0 £ RN
3. for『=1,2,3…do
4. G=Lyk? + yVxtZ(W,X<，),yz))
5. refresh *W* according to

‘0， I *gi* I <A

w!，+1> =< /7

-Asgn(gi)),其他

1. end
2. return *W*

**6.9.4 FTRL**

FTRL(Follow the Regularized Leader)是由 Google 公司的 H. Brendan McMahan 在 2010年提出的，后来他在2011年发表了一篇关于FTRL和AOGD、FOB()S、RDA比较的 论文，他在2013年又和Gary Holt.Michael Young等人发表了一篇关于FTRL工程化实现 的论文。

**1. FTRL**算法原理

FTRL综合考虑了 FOBOS和RDA对于正则项和W限制的区别，其特征权重更新公

式为

l|W|U+心 || W — W⑴ || 可

将上述公式的最后一项展开，等价于求下面这样一个最优化问题：

伊中）=argmin]（G（EW— %⑴W⑴）W + 厶 || W || 】  
+ 号。2 + £舟）*\\W\\l+^^（s）* || 护||"

由于§立护II " 偿相对于W来说是一个常数,并且令Z（z）=逆"W —史b⑴W⑴, . Z s=] S=[

上式等价于

矿"=argmin^^ + A. || W || i +y（A2 + || W || "

针对特征权重的各个维度将其拆解成N个独立的标量最小化问题：

minimize^ Z!r）Vr, +Ai || 叫 || 】+扌（义2 + "如

'0, I 蛇 IVA

—（“2 + gb⑴） W — Aisgn（^-Z> ））,其他

1. **FTRL**学习率

在FTRL中，每个维度上的学习率都是单独考虑的，在一个标准的OGD中使用的是一 个全局的学习率策略*/=\/后*这个策略保证了学习率是一个正的非增长序列，对于每一 个特征维度都是一样的。

考虑特征维度的变化率：如果特征1比特征2变化得更快，那么在维度1上的学习率 应该下降得更快。很容易就可以想到可以用某个维度上的梯度分量来反映这种变化率。在 FTRL中，维度*i*上的学习率是这样计算的：

折）— «

孕\_ r

这里的。和0是需要输入的参数。

1. **FTRL**算法逻辑

到此为止，已经得到了 FTRL的特征权重维度的更新方法和每个特征维度的学习率计 算方法，那么很容易写出FTRL的算法逻辑。

1. input a,/?,Ai »A2
2. initialize WE **RN** ,Z=0G **RN** ,Q=0E **RN**
3. for Z= 1,2,3…do

4

1. for in 1,2，…，N do

6 ^r,q,=/+gf

|  |  |  |  |
| --- | --- | --- | --- |
| 7 | *Zi=Zi + gi—(JiWi* | | |
|  | f0, | I^KAi |
| 8 | Wt =Y | rh | +直WSZ) (zt —A] sgn(2：, )), 其他 *a /* |
| 9 | end |  |  |
| 10 | end |  |  |
| 11 | return *W* | |  |

FTRL中的4个参数需要针对具体的问题进行设置。

**6.10** 习题

1. 什么是机器学习？机器学习的研究目标是什么？
2. 简述机器学习的发展历史。
3. 什么是记忆学习？其基本思想是什么？
4. 什么是归纳学习？归纳学习一般又可分为哪两种学习形式？
5. 简述决策树学习，并编程实现ID3算法。
6. 简述神经学习，并编程实现BP算法。
7. 简述贝叶斯学习，并编程实现朴素贝叶斯分类算法。
8. 试编程实现基于信息嫡进行划分选择的决策树算法，并为下表中的西瓜数据集生成 一棵决策树。

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 色泽 | 根蒂 | 纹理 | 脐部 | 敲声 | 触感 | 密度 | 含糖率 | 好瓜 |
| 1 | 青绿 | 卷缩 | 清晰 | 凹陷 | 浊响 | 硬滑 | 0. 697 | 0.460 | 是 |
| 2 | 乌黑 | 卷缩 | 清晰 | 凹陷 | 沉闷 | 硬滑 | 0. 774 | 0. 376 | 是 |
| 3 | 乌黑 | 卷缩 | 清晰 | 凹陷 | 浊响 | 硬滑 | 0. 634 | 0. 264 | 是 |
| 4 | 青绿 | 卷缩 | 清晰 | 凹陷 | 沉闷 | 硬滑 | 0. 608 | 0. 318 | 是 |
| 5 | 浅白 | 卷缩 | 清晰 | 凹陷 | 浊响 | 硬滑 | 0. 556 | 0. 215 | 是 |
| 6 | 青绿 | 稍卷 | 清晰 | 稍凹 | 浊响 | 硬滑 | 0.403 | 0.237 | 是 |
| 7 | 乌黑 | 稍卷 | 清晰 | 稍凹 | 浊响 | 软粘 | 0. 481 | 0. 149 | 是 |
| 8 | 乌黑 | 稍卷 | 清晰 | 稍凹 | 浊响 | 软粘 | 0.437 | 0.211 | 是 |
| 9 | 乌黑 | 稍卷 | 稍糊 | 稍凹 | 沉闷 | 硬滑 | 0. 666 | 0. 091 | 否 |
| 10 | 青绿 | 硬挺 | 清晰 | 平坦 | 清脆 | 软粘 | 0. 243 | 0. 267 | 否 |
| 11 | 浅白 | 硬挺 | 模糊 | 平坦 | 清脆 | 硬滑 | 0. 245 | 0. 057 | 否 |
| 12 | 浅白 | 卷缩 | 模糊 | 平坦 | 浊响 | 软粘 | 0. 343 | 0.099 | 否 |
| 13 | 青绿 | 稍卷 | 稍糊 | 凹陷 | 浊响 | 硬滑 | 0. 639 | 0. 161 | 否 |
| 14 | 浅白 | 稍卷 | 稍糊 | 凹陷 | 沉闷 | 硬滑 | 0. 657 | 0. 198 | 否 |
| 15 | 乌黑 | 稍卷 | 清晰 | 稍凹 | 浊响 | 软粘 | 0. 360 | 0. 370 | 否 |
| 16 | 浅白 | 卷缩 | 模糊 | 平坦 | 浊响 | 硬滑 | 0. 593 | 0. 042 | 否 |
| 17 | 青绿 | 卷缩 | 清晰 | 稍凹 | 沉闷 | 硬滑 | 0. 719 | 0. 103 | 否 |

1. 试设计一个BP改进算法，能通过动态调整学习率显著提升收敛速度。并在http：// archive, ics. uci. edu/上选择两个UCI数据集与标准的BP算法进行实验比较。
2. 用题8中的西瓜数据集训练一个朴素贝叶斯分类器，对测试例“测1”进行分类：

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 编号 | 色泽 | 根蒂 | 纹理 | 脐部 | 敲声 | 触感 | 密度 | 含糖率 | 好瓜 |
| 测1 | 青绿 | 卷缩 | 清晰 | 凹陷 | 浊响 | 硬滑 | 0. 697 | 0. 460 | ? |

数据挖掘

**7.1**数据挖掘概述

**7.1.1**数据挖掘概念与发展

随着科学技术的飞速发展，使得各个领域或组织机构积累了大量的数据。如何从这些 数据中提取有价值的信息和知识以帮助做出明智的决策成为巨大的挑战。计算机技术的迅 速发展使得处理并分析这些数据成为可能，这种新的技术就是数据挖掘(Data Mining, DM),又称为数据库知识发现(Knowledge Discovery in Database, KDD) 0

数据挖掘概念首次出现在1989年举行的第11届国际联合人工智能学术会议上，其思 想主要来自机器学习、模式识别、统计和数据库系统。国内对数据挖掘的研究起步较晚， 1993年，国家自然科学基金首次支持该领域的研究。此后，国家和各省自然科学基金、国家 社科基金、“863”“973”项目、国家和各省的科技计划每年都有相关项目支持，众多研究机构 和大学成立了专门的项目组，从事数据挖掘研究与应用的人员越来越多。现今，数据挖掘的 基本理论问题逐步得到了解决,更多的是数据挖掘的应用。

数据挖掘是一种将传统的数据分析方法与处理大量数据的复杂算法相结合的技术。目 前对其并没有统一的定义，但是众多教材中大多釆用的是韩家炜先生给出的关于数据挖掘 的定义：

数据挖掘就是从大量的、不完全的、有噪声的、模糊的、随机的数据中提取隐含在其中 的、人们事先不知道的，但又是潜在有用的信息和知识的过程。

上述定义的含义有以下几个方面：第一，数据源必须是大量的、真实的，真实的数据往 往含有噪声或缺失；第二，发现的是用户感兴趣的知识；第三，发现的知识要可接受，可理 解，可运用，能支持特定的问题发现，能够支持决策，可以为企业带来利益，或者为科学研究 寻找突破口。

**7.1.2**数据挖掘的任务

数据挖掘的任务可以分为预测型任务和描述型任务。预测型任务就是根据其他属性的 值预测特定属性的值，如回归、分类、离群点检测等。描述型任务就是寻找、概括数据中潜在 联系的模式，如聚类分析、关联分析、演化分析、序列模式挖掘。

1. 分类分析

分类分析就是通过分析示例数据库中的数据，为每个类别做出准确的描述，或建立分析 模型，或挖掘出分类规则，然后用这个分类模型或规则对数据库中的其他记录进行分类。分 类分析已广泛用于用户行为分析、风险分析、生物分析、生物科学领域等。

1. 聚类分析

物以类聚，人以群分，聚类分析技术试图找出数据集中的数据的共性和差异，并将具有 共性的对象聚合在相应的簇中。聚类分析已广泛应用于客户细分、定向营销、信息检索等 领域。

聚类与分类是容易混淆的两个概念。聚类是一种无指导的观察式学习，没有预先定义 的类。

1. 关联分析

关联分析是发现特征之间的相互依赖关系，通常是在给定的数据集中发现频繁岀现的 模式知识（又称关联规则）。关联规则广泛用于市场营销、事务分析等领域。

**7.1.3**数据挖掘的应用

数据挖掘就是为大数据而生的，有大量数据的地方就有数据挖掘的用武之地。目前，应 用较好的领域或行业有生物信息学、电信业、零售业以及保险、银行、证券等金融领域。

生物信息学是数据挖掘应用的新领域，是21世纪生物学的产物。科学家们在试图了解 和认识自身的过程中产生了大量的生物分子数据。充分利用这些数据，通过数据分析、处理 揭示这些数据的内涵，从而得到对人类有用的信息，在指导试验、精心设计试验方面发挥着 重要作用，是生物学家、数学家和计算机科学家所面临的一个严峻挑战。

零售业收集了关于销售、顾客购物史、消费等大量数据，是数据挖掘很好的应用领域之 一。零售数据挖掘可以帮助识别顾客购买行为，发现顾客的购买模式和趋势，改进服务质 量，取得更好的顾客保持度和满意度，提高货品消费比，设计更好的货品运输与分销策略，降 低成本。

金融领域存有大量的客户信息记录、自身服务记录等，可用数据挖掘技术分析客户需求 和兴趣，银行方面可以预测存、贷款趋势等，可以更好地服务客户。为了避免不良贷款的风 险，银行管理者可能希望挖掘银行对客户的记录信息来发现一些有意义的模式等。

随着网络的发展，大量的文档数据涌现在网上，文本挖掘显得更加重要，这也是数据挖 掘应用的一个重要子领域。文本挖掘主要是寻找自然语言文本中的规律、模式或者趋向，并 且通常是为了特定的目的进行关于文本的分析。文本挖掘继承了数据挖掘的很多方法，但 又异于数据挖掘本身，因为文本挖掘着力于从非结构化或者半结构化的文本中抽取有用的 知识，而数据挖掘则主要是从结构化的数据库中发现数据的主要模式。

**7.1.4**数据挖掘过程与方法

数据挖掘只是数据挖掘过程的一部分，完整的挖掘过程还应包括以下步骤：定义业务 目标、甄别数据源、收集数据、选择数据、数据质量检查、数据转换和结果解释。

一般来说,许多商业问题能够用比较广泛的数据挖掘技术来解决。此外，使用不同的数 据挖掘技术也能够帮助审核数据挖掘结果的稳健性。对于市场细分来说，可以使用基于统 计学的聚类分析，也可以使用神经网络聚类分析。其结果可能是一致的，也可能是不一致 的。在结果一致的情况下，可以认为结果是稳健的；在结果不一致的情况下，在要基于这些 结果做出商业决策之前,应该仔细地检查分析结果。

在数据挖掘阶段，概括而言，数据挖掘分析员可以使用的数据挖掘方法主要有如下 几个：

（1） 预估模型，包括分类和预估两种类型。

（2） 聚类技术。

（3） 连接技术。

（4） 时间序列分析。

在本章中将介绍几种具体的数据挖掘方法。

**7.2**分类

分类的任务就是确定对象属于哪个预定义的目标类。分类问题是一个普遍存在的问 题,有许多不同的应用。例如，根据电子邮件的标题和内容检查出垃圾邮件，对一大堆照片 区分岀哪些是猫，哪些是狗。分类任务就是通过学习得到一个目标函数，把每个属性集7 映射到一个预先定义的类标号目标函数也称分类模型。

分类模型可以作为解释性的工具，用于区分不同类中的对象。分类模型还可以预测未 知记录的类标号，分类模型可以看做一个黑箱，当给定未知记录的属性集上的值时，它自动 地赋予未知样本类标号。

**7.2.1**决策树分类法

有关决策树学习在第6章中已经提到，我们已经知道决策树分类法是一种简单但广泛 的分类技术。

从原则上讲，对于给定的数据集，可以构造的决策树的数目达指数级。尽管某些决策树 比其他决策树更为准确，但是由于搜索空间是指数规模的，找出最佳决策树在计算上是不可 行的。现在的许多算法都采取贪心算法，采取一系列局部最优决策来构造决策树，比如 Hunt算法。

设*D,是与节点£*相关联的训练记录集，而必，…，oU是类标号.Hunt算法的递 归定义如下：

1. 如果*Dt*中所有记录都属于同一类乂，则：是叶节点，用y标记。
2. 如果D,中包含属于多个类的记录，则选择一个属性测试条件，将记录划分成较小 的子集。对于测试条件的每个输出，创建一个子女节点，并根据测试结果将D中的记录分 布到子女节点中。然后，对于每个子女节点，递归地调用该算法。

**7.2.2**基于规则的分类器

基于规则的分类器是使用一组if-then…规则来对记录进行分类的技术。为了建立基 于规则的分类器，需要提取一组规则来识别数据集的属性和类标号之间的关键联系。提取 分类规则的方法有两大类：直接方法和间接方法。直接方法是直接从数据中提取分类规 则，间接方法是从其他分类模型中提取分类规则。

顺序覆盖算法经常被用来直接从数据中提取规则，规则对于某种评估度量以贪心的方 式增长。该算法从包含多个类的数据集中一次提取一个类的规则。决定哪一个类的规则最 先产生的标准取决于多种因素，如类的普遍性，或者给定类中误分类记录的代价。

顺序学习规则：对每个给定的类**G，**希望规则可以覆盖该类的大多数元组，但不包括其 他类的元组(或很少)。

1. 初始值为空规则集。
2. 使用Learn-One=Rule函数得到一条新规则。
3. 从训练集中删除被新产生的规则所覆盖的实例。
4. 重复步骤(2)和步骤(3),直到满足停止标准为止。

**7.2.3**朴素贝叶斯分类器

朴素贝叶斯分类方法是基于统计的学习方法，利用概念统计进行学习分类，如预测一个 数据属于某个类别的概念。主要算法有朴素贝叶斯分类算法、贝叶斯信念网络分类算法等。

贝叶斯分类方法的主要特点如下：

1. 充分利用领域知识和其他先验知识，显式地计算假设概率，分类结果是领域知识和 数据样本信息的综合体现。
2. 利用有向图的表示方式，用弧表示变量之间的依赖关系，用概率分布表示依赖关系 的强弱。表示方法非常直观,有利于对领域知识的理解。
3. 能进行增量学习，数据样本可以增量地提高或降低某种假设的估计概率并且能方 便地处理不完整数据。

贝叶斯定理是朴素贝叶斯分类算法和贝叶斯信念网络分类算法的基础，在6. 8节讨论 过，在此不再重复。下面介绍朴素贝叶斯分类算法。

根据朴素贝叶斯分类的原理，算法基本描述如下。

函数名：NativeBayes

输入：类标号未知的样本**X=** ，血，•••

输出：未知样本**X**所属类别号

1. for *j — 1* to *m*
2. 计算x属于每个类别G的概率p(x|G)= P5IG)p&2IG)・・・p(z」g)
3. 计算训练集中每个类别G的概率*P(Cj)*
4. 计算概率值**9=P(X|G)XP(G)**
5. End for

6选择计算概率值。最大的作为类别输出

**7.2.4**基于距离的分类算法

给定一个数据库。=｛如，如，・・・《”｝和一组类C=｛G(2,…，GJ。假定每个元组包括 一些数值型的属性值—=｛如，5…，3,每个类也包含数值性属性值：。=｛勺，弓2,…， *Cjk*驾则分类问题是要分配每个*tt*到满足如下条件的类C；:

sim(z, ) 2 sim(L ,CQ , V C” £ C, *Cp ^=- C)* 其中sim(L，C)称为相似性。

在实际的计算问题中往往用距离来表征相似性。距离越近，相似性越大；距离越远，相 似性越小。为了计算相似性，应首先得到表示每个类的向量。最常用的是通过计算每个类 的中心来完成。

基于距离的分类算法如下：

输入：每个类的中心g,c2,・・・，g；待分类的元组，

输出：每个类别。

1. dist = 8
2. for 1 = 1 to *m* do
3. if dist(c, U)<Cdist then begin
4. *c^~i*
5. dist—dist(q ,/)
6. end

该算法通过对每个元组和各个类的中心进行比较，从而可以找出它的最近的类中心，得 到确定的类别标记。

**7.3**聚类

**7.3.1**概念

聚类分析的核心是聚类，聚类是一种无监督学习，实现的是将整个数据集分成不同的 “簇”，在相关的文献中，也将之称为“对象”或“数据点”。聚类要求簇与簇之间的区别尽可能 大，而簇内数据的差异尽可能小。与分类不同，不需要先给出数据的类别属性。

**7.3.2**聚类分析的基本方法

聚类分析的研究主要基于距离和基于相似度的方法。经过长时间的发展，形成不少聚 类算法。根据不同的数据类型和聚类的目的可以选择不同的聚类算法。主要的聚类算法可 以划分为如下4类。

1. 划分聚类的方法

给定一个数据集，将构建数据集的有限个划分，每个划分都是一个簇，且每一个划分应 当满足如下两个条件：

（1） 每个划分中至少包含一个样本。

（2） 每个样本只能属于一个簇。

奸Means和人Medoids就是典型的划分聚类算法，下面介绍如Means的具体算法。

妇Means算法是一种最常用的基于划分的聚类方法。其基本思想是：把数据集划分成 &个簇以由用户指定），每个簇内部的样本非常相似，但不同簇之间样本则又差异很大。在 给定初始*k*个簇之后，算法根据某个距离函数反复地把数据分入*k*个聚类中，直到满足终止 条件为止。

下面给出如Means的算法思想：

奴聚类中心个数,数据样本。

1确定*k*个数据点作为初始聚类中心

1. repeat
2. for对于数据样本D中的每个数据7
3. 计算z到每个聚类中心的距离
4. 将了分配到最近的那个聚类中心所属的类
5. End for

*7*计算当前每个类的均值,并作为新的聚类中心

8满足终止条件结束，否则执行循环部分

终止（收敛）条件可以是以下的任何一个：

•没有（或只有最小数目的）数据点被重新分配给不同的聚类。

•没有（或只有最小数目的）聚类中心再发生变化。

•误差平方和（SSE）局部最小。

SSE= tS II#一叫 II2

其中上表示聚类数目，G表示第j个聚类，叫是聚类G的聚类中心（G中所有数据点的均 值向量）II II表示数据点z和聚类中心之间的距离。

这里只是强调SSE局部最小，妇Means算法并不能保证找到对应于最小化全局目标函 数的最优解。该算法对于随机选取的初始聚类中心非常敏感，但是可以通过多次执行该算 法来清除初始中心敏感的影响。

妇Means并不适合所有的数据类型，比如它不能处理非球形簇、不同尺寸和不同密度的 簇，尽管指定足够大的簇个数时，它通常可以发现纯子簇。对包含离群点的数据进行聚类 时Means也有问题。最后-Means仅限于具有中心（质心）概念的数据。

下面举例说明该算法的实现过程。

例7.1现有一个数据集（1,2, 30, 15, 10, 18, 3, 9, 8, 25）,用卜Means算法将这些

数据进行聚类。

解：首先给出力=3,即将数据集聚成3类。随机选取后3个数作为初始簇均值：皿= 9, 7712=8, 7713=25,开始迭代。

第一次迭代：分别计算其余每个数据到这3个均值的距离，并将其分给距离最近的均 值所代表的簇。这里采用的距离值为两个数的差的绝对值。这样可以得到3个簇为

*Ki* = {1,2,3,8}, K2 = {9,10,15}, *K3* = {18,25,30}

对这个结果重新计算每个簇的均值。则均值将更新为四=3.5,秫2 = 11.3,四=24.3。

第二次迭代：重复第一次迭代中的方法，得到新的3个簇为

Ki = {1,2,3}, *K2* = {8,9,10,15), K3 = {18,25,30}

新的均值为四=2,死=13,?«3=24.3。

第三次迭代：重复第一次迭代中的方法，得到新的3个簇为

Ki = {1,2,3}, *K2 =* {8,9,10,15,18}, K3 = {25,30}

新的均值为 四=2,”2 = 12,皿=27.5。

第四次迭代：重复第一次迭代中的方法，得到新的3个簇为

Kj = {1,2,3}, *K2* = {8,9,10,15,18), *K3* = {25,30}

由此可以看出，每个簇中的数据不再被重新分配，数据达到稳定，算法终止。

1. 层次聚类的方法

层次聚类技术是第二类重要的聚类方法。与Means 一样，与许多聚类方法相比，这 些方法相对较老，但是它们仍然被广泛使用。在这些方法中，采用的是某种标准对给定的数 据集进行层次的分解。其结构实际上就是层次树。可以通过两种方法来构造层次树，即自 底向上的方法和自顶向下的方法，它们分别又称为凝聚的方法和分裂的方法。凝聚的方法 是最初假设所有项属于一个单独的簇，然后寻找最佳配对并合并成一个新簇，聚类的过程从 底部开始，最终的结果显示在最上面；分裂的方法与之相反，开始时将所有数据看做一个 簇，按照某种标准,将每一簇分裂为更小的簇。直到最终每个样本单独的出现在一个簇中， 或者达到一个指定的阈值终止条件。到目前为止，凝聚层次聚类技术最为常见。

层次聚类常常使用称作树状图(dendrogram)的类似树的图显示。该图显示簇-子簇联 系和簇合并或分裂的次序。对于二维点的集合，层次聚类也可以使用嵌套簇图(nested cluster diagram)表示。

层次聚类的方法尽管简单,但经常会遇到合并或分裂点选择的困难。这样的决定是非 常关键的，因为一旦一组对象合并或分裂，下一步的处理将对新生成的簇进行。而且它不具 有很好的可伸缩性，因为合并或分裂的决定需要检查和估算大量的对象或簇。层次聚类缺 乏全局目标函数，凝聚层次聚类技术使用各种标准，在每一步局部地确定哪些簇应当合并或 分裂，这种方法产生的聚类算法避开了解决困难的组合优化问题。对于合并两个簇，凝聚层 次聚类算法倾向于作出好的局部决策，因为它们可以使用所有点的逐对相似度信息，然而一 旦作岀合并两个簇的决策，以后就不能撤销，这种方法阻碍了局部最优标准变成全局最优标 准。就计算量和存储需求而言,凝聚层次聚类算法是昂贵的，所有合并都是最终的，对于噪 声、高维数据(如文档数据)，这也可能造成很多问题。先使用其他技术进行聚类，这两个问 题在某种程度上都可以加以解决。

给定聚类簇G和G，可以通过以下公式计算它们的最小距离、最大距离和平均距离。 最小距离：dmin（G，G）= min dist（«r,z） *xeci,zec{*

最大距离：dmax（G，G）= max dist（jr,z） x6C.,»6C.

卩 G II", ||溢潭dist（E）

平均距离：*df（Ci，Cj）=*

当算法使用最小距离刁床（G，G）衡量簇间距离时，有时称它为最近邻聚类算法。此 外，如果当最近的簇之间的距离超过某个任意的阈值时聚类过程就会终止，则称其为单连接 算法。

当算法使用最大距离*dm“（G，G）*衡量簇间距离时，有时称它为最远邻聚类算法。此 外，如果当最近簇之间的最大距离超过某个任意的阈值时聚类过程就会终止，则称其为全连 接算法。

层次聚类里有如下几种常见的算法：Chameleon、CURE、ROCK、BIRCH、DIANA和

AGNES0下面主要介绍AGNESO

AGNES是一种采用自底向上聚合策略的层次聚类方法。它先将数据集中的每个样本 看做一个初始聚类簇，然后在算法运行的每一步找出距离最近的两个聚类簇进行合并，该过 程不断重复，直至达到预设的聚类簇个数。

AGNES算法描述如下：

输入：样本集D={.,五，…口詩；聚类簇距离度量函数由 聚类簇数E

1. for j = 1 »2,••• do
2. Gf}
3. end for
4. for *i=* 1,2, •,- *<m* do
5. for，=1,2,•・•，?《 do

6

*1*

1. end for
2. end for

10设置当前聚类簇个数：*q = m*

1. while *q^>k* do
2. 找出距离最近的两个聚类簇C,-和*C广*
3. 合并G・和G・：G・=G・uc广
4. for ,=厂 +1, j •+2,…，q do
5. 将聚类簇G重新编号为Gt
6. end for
7. 删除距离矩阵M的第厂行与第厂列
8. for j = 1,2,…，/n do
9. M（r ,j）=d（G・，G）
10. M（j,r ）=M（广,;）
11. end for
12. *q = q*—1
13. end while

输出：簇划分C={Cl,C2,-,CJ

1. 基于密度的方法

大部分划分方法基于对象之间的距离进行聚类。这样的方法只能发现球状簇，而在发 现任意形状的簇时遇到了困难。目前已经开发了基于密度概念的聚类方法，其主要思想是： 只要“领域”中的密度超过了某个阈值，就继续增长给定的簇。也就是说，对给定簇中的每个 数据点，在给定半径的领域中必须至少包含最少数目的点。这样的方法可以用来过滤噪声 或离群点，发现任意形状的簇。

基于密度的聚类代表算法有DBSCAN. OPTICS. DENCLUE算法等。下面介绍 DBSCAN 算法。

DBSCAN（Density-Based Spatial Clustering of Applications with Noise）是一个比较有 代表性的基于密度的聚类方法。与层次聚类不同，它将簇定义为密度相连的点的最大集合， 能够把具有足够高密度的区域划分为簇，并可在有“噪声”的空间数据库中发现任意形状的 聚类。给定数据集*D=* {円，五，…口詩，定义下面这几个概念。

£邻域：对务6D,其£邻域包含样本集D中与可的距离不大于E的样本，即

Ne（x；） = *{xi* G *D* I dist（而 口丿）We}

.核心对象：如果一个对象的£邻域至少包含最小数目（MinPts）个对象，则称该对象为 核心对象。

密度可达：给定一个对象集合D，如果力在Q的e邻域内，而q是一个核心对象,就说对 象》从对象q岀发是密度可达的。

密度相连：对1,与可，若存在*Xk*使得与*JCj*均由*Xk*密度可达，则称R与密度 相连。

DBSCNA使用簇的基于密度的定义，因此它是相对抗噪声的，并且能够处理任意形状 和大小的簇。这样,DBSCNA可以发现使用奸Means不能发现的许多簇。然而，当簇的密 度变化太大时，DBSCNA就会有麻烦。对于高维数据，它也有问题，因为对于这样的数据， 密度定义更为困难。最后，当近邻计算需要计算所有的点对邻近度时，DBSCNA的开销可 能是很大的。

下面介绍DBSCAN的算法描述：

输入：包含〃个对象的数据库，半径e，最少数目MinPts

输岀：所有生成的簇，达到密度要求

1. REPEAT
2. 从数据库中抽取一个未处理过的点
3. IF抽出的点是核心点THEN找出所有从该点密度可达的对象，形成一个簇
4. ELSE抽出的点是边缘点（非核心对象），跳出本次循环，寻找下一点
5. UNTIL所有点都被处理

DBSCAN的算法步骤如下：

输入：数据集D,参数MinPts.e

1将数据集*D*中的所有对象标记为unvisited

1. do
2. 从D中随机选取一个unvisited对象并将*p*标记为visited
3. if 0的€邻域包含的对象数至少为MinPts个
4. 创建新簇C,并把力添加到c中
5. 令N为力的e邻域中对象的集合
6. for *N*中每个点久
7. if 是 unvisited
8. 标记。，是 visited
9. if *p,*的e邻域至少有MinPts个对象，把这些对象添加到*N*
10. if *P,*还不是任何簇的对象，将*pi*添加到簇C中
11. end for
12. 输出C
13. Else标记*p*为噪声
14. Until没有标记为unvisited的对象

输出：簇集合

1. 基于模型的聚类

基于模型的聚类方法试图将给定数据与某个数学模型达成最佳拟合。此类方法经常假 设数据是根据潜在的概率分布生成的。基于模型的聚类方法主要包括统计学方法、概念聚 类方法和神经网络方法。

AutoClass方法是一种基于贝叶斯理论的数据聚类算法，通过对数据进行处理，计算岀 每条数据属于每个类别的概率值，对数据进行聚类。AutoClass能对复杂数据进行精确的 自动聚类，可以事先设定好类别数目让AutoClass自动寻找，在寻找结束后，能够得到每一 条数据分别属于每一类别的几率。AutoClass的程序是由Cheeseman和Stutz在1995年开 发出来的。

AutoClass具有以下优点：

1. 聚类的数据不需要预先给定数据类别，但是定义了每个数据成员。
2. 可以处理连续型或离散型数据。在AutoClass中，每一组数据都以一个向量来表 示，其中每个分量分别代表不同的属性，这些属性数据可以是连续型或离散型。
3. AutoClass要求将数据存成Data File(纯数据文件)与Header File(描述数据的文 件)两部分，这样可以让使用者自由搭配Data File和Header File而节省输入数据的时间。
4. 可以处理缺值数据。当一组数据中的某些属性值缺失时,AutoClass仍可对此组数 据进行聚类。

AutoClass也存在以下缺点：

1. AutoClass概率模型的前提是各属性相互独立，而这个假设在许多领域中是不成立的。
2. AutoClass不是一个完全自动化的聚类算法，需要主观地决定数据的适当群数范 围，而此问题却是聚类的一大难题。
3. 使用AutoClass处理数据时，必须不断地重复假设与测试，并结合专业知识与程 序，才能得出良好的结果，因而要花费大量的时间。

（4）没有提供一个先验标准来预测一组数据是否能够聚类，因而带有一定的臆断性。 没有提供一个后验方法来评估分类的结果是否可以信赖。

概念聚类是一种机器学习聚类方法，给定一组未标记的对象，产生对象的分类模式。与 传统的聚类不同.概念聚类除了确定相似的对象分组外，还找出每组对象的特征描述，其中 每组对象代表一个概念或类。因此，概念聚类是一个两步的过程：首先进行聚类，然后给出 特征描述。

COBWEB是一个常用且简单的增量式概念聚类方法。它的输入对象采用符号-值对 （属性-值对）来描述。该方法采用分类树的形式创建一个层次聚类。

SOM是神经网络方法的典型代表，SOM采用WTA（Winner Takes All）竞争学习算 法，其聚类过程通过若干单元对当前单元的竞争来完成，与当前单元权值向量最接近的单元 成为贏家或获胜单元，同时抑制距离较远的神经元。SOM可以在不知道输入数据任何信息 结构的情况下学习到输入数据的统计特征。其优点在于：可以实现实时学习，网络具有自 稳定性，无须外界给出评价函数，能够识别向量空间中最有意义的特征，抗噪声能力强。缺 点是时间复杂度较高，难以用于大规模数据集。

**7.4**关联规则

关联规则是数据中所蕴含的一类重要规律，用关联规则进行挖掘是数据挖掘的一项根 本任务，甚至可以说是数据库和数据挖掘领域中所发明并被广泛研究的最为重要的模型。 关联规则的目标是在数据项目中找岀所有的并发关系，这种关系也称为关联。

关联规则挖掘的经典应用是购物篮的数据分析，通过数据找出顾客在商场所选购的商 品之间的关联。例如，如果一位顾客购买了产品A,该顾客还有多大的可能会同时购买产品 B,或是某顾客在买了产品A和产品B的情况下还可能会买什么产品。诸如此类的问题都 能从关联分析中找到答案。

**7.4.1**基本概念

**1.**关联规则的形式

设*I={i}* u2，•••，"是一个项目集合，T是一个（数据库）事务（Transaction）集合，其中 每个事务庇是一个项目集合，并满足那么，一个关联规则可以表示成如下形式的蕴 含关系：

X —Y,其中 XOY,Y^ / 且 X *HY= 0*

X或丫是一个项目集合，称作项集，并称X为前件，丫为后件。

/表示一个商场中岀售的所有商品。D是一个事务，即可以认为是一位客户一次购买 的商品集合。例如，一个具体事务可以是

{鸡肉，啤酒，奶酪}

表示一位客户一次购买了鸡肉、啤酒和奶酪这3件商品。由此可以得到的一条规则是

鸡肉，啤酒—奶酪

其中，鸡肉、啤酒构成的集合就是X,奶酪就是丫。

1. 关联规则强度指标

由事务(transaction)得到的关联可能很多，而我们关心的是找出强关联。这时就需要 有相应的指标来衡量，用来衡量规则强度的指标不唯一，例如支持度、置信度和增益等。支 持度和置信度是两个常用的衡量关联规则强度的指标。

关联规则X-Y的支持度是数据库中包含XUV的事务占全部事务的百分比。它是概 率 P(XUV)，记作 support(X-\*Y) = P(XUV)o

关联规则X-y的置信度是包含XUY的事务与包含X的事务数的比值。它是概率 P(Y|X),记作 confidence(X^y)*= P(Y* | X)。

在进行关联规则前，由用户预先定义最小支持度阈值(min\_sup)和最小置信度阈值 (min\_conf)0对于那些支持度和置信度分别大于或等于min\_sup和min\_conf的规则，将其 称为强规则。

1. 频繁项集

每个属性由多个元素组成，这里的元素称为项(item),多个项组成的集合称为项集 (itemset) o根据项集中包含的项的数量，项集可以是1项集、2项集或者龙项集。例如｛啤 酒｝、｛牛奶)就是1项集；｛牛奶，啤酒，奶酪)则是3项集。如果某个项集的支持度大于或等 于预先设定的最小支持度阈值(min\_sup),则将这个项集称为频繁项集或大项集，所有的频 繁*k*项集组成的集合通常记为*Lko*

**7.4.2**关联规则挖掘算法

关联规则挖掘算法中，以Agrawal等人提出的Apriori算法最为著名，它是常用的关联 规则挖掘算法，其挖掘的过程主要包含两个阶段：第一阶段先从数据集中找岀所有的频繁 项集，它们的支持度大于等于最小支持度阈值(min\_sup)o第二阶段由这些频繁项集产生 关联规则，计算它们的置信度，然后保留那些置信度大于等于最小置信度阈值(min\_conf)的 关联规则。

**1. Apriori**算法中候选集合的产生

Apriori算法中候选集合的产生由连接和剪枝两个步骤组成：

1. 连接。为了找如，通过L-i与自己连接产生候选人项集的集合，该候选，项集记为

Go 中的两个项集九和人可以执行连接操作*ly^lz*的条件是门表示项集中的第*i*

个元素)：

(狂口 =虹口) A (ZiE2]=虹2]) A …A (狂为一2]=皿一2])

A(狂龙一口 =心一 口)

1. 剪枝。G是丄的超集，即它的成员可能不是频繁的，但是所有频繁的上项集都在 G中。因此可以通过扫描数据库并计算每个4项集的支持度来得到Lx

为了减少计算量，可以利用Apriori性质剪枝，即如果一个吏项集中包含的互一1个元素 的子集不在侦7中，则该候选集不可能是频繁的，可以直接从G中删除。

**2. Apriori**算法过程

输入：事务数据库D;最小支持度阈值min\_sup；最小置信度阈值min\_conf0 输出：事务数据库D中的所有频繁项目集L和关联规则AR。

算法描述：

1. Li = find\_frequent\_l\_itemsets(D)；
2. for以=2； *Lk-^0； & + + ) {*
3. *Ck* = apriori\_gen(L(t\_1 ,min\_sup);
4. for each transaction zG D{

〃扫描D以计数

5 *Ct=* subset(Q ；

〃得到：的候选子集

1. for each candidate ；
2. c. count+ + ；

9 *Lk = {cE Ck\* c, count 2 min\_sup}

1. }
2. return L = U丄\*;

子程序 aprior^genCL\*-! ; frequent(&—1)itemset):

1 for each itemset *lx* C *Lk-*1

1. for each item *l2* EL—
2. if (狂口=&邸 A(，[2]=人[2]) A …A(4隹一口=我住一口)
3. C = "8/2 //连接步；产生候选
4. if has\_infrequent\_subset(c,Li\_] ) then
5. delete c；
6. else add c to Q ；
7. return *Ck*

子程序 has\_infrequent\_subset(c： candidate *k* itemset； 侦一】：frequent(互一1)itemset): //使用先验知识

1. for each 以一1)subset s of c
2. if *s 任以\_\* then
3. return true
4. return false

Apriori算法的计算复杂度主要受支持度阈值、项数（维度）、事务数和事务的平均宽度 影响。降低支持度阈值通常导致更多的频繁项集，这给算法的计算复杂度带来不利影响，因 为必须产生更多候选项集并对其计数。随着支持度阈值的降低，频繁项集的最大长度将增 加，随之算法需要扫描数据集的次数也将增多。随着项数的增加，需要更多的空间来存储项 的支持度计数。由于Apriori算法反复扫描数据集，事务的平均宽度可能很大，而频繁项集 的最大长度随事务平均宽度的增加而增加，并且随着事务宽度的增加，事务中将包含更多的 项集，这将增加支持度计数时Hash树的遍历次数。

下面举例说明该算法规则。

例7.2现有一个事务数据库如表7.1所示。找出其所有满足最小支持度计数的关联 规则。

表 **7.1 AllElectronics** 数据库

|  |  |
| --- | --- |
| TID | List of item\_ID |
| T100 | 11,12,15 |
| T200 | 12,14 |
| T300 | 12,13 |
| T400 | 11,12,14 |
| T500 | 11,13 |
| T600 | 12,13 |
| T700 | 11,13 |
| T800 | 11,12,13,15 |
| T900 | 11,12,13 |

表的每一行表示一条交易，共有9行，左边表示顾客ID,右边表示商品ID,为了方便计 算，这里给出最小支持度计数为min\_sup=2（等于最小支持度为22%）。

首先，扫描数据库，识别所有1项集和它们的支持度计数，将它们称作候选1项集，记作 G，然后选择其支持度大于或等于min\_sup的项，将这些项称为频繁1项集,并记作L）。这 样就识别了所有的频繁1项集。

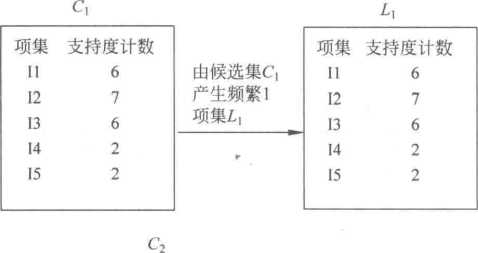
下面需要做类似的工作，产生所有可能频繁2项集，称作候选2项集，记作C2o这可以 通过从L产生所有可能的2项集来实现。扫描数据库，确定G中每个项集的支持度，再从 G中选择那些满足支持度大于或等于min\_sup的项集，得到*L2*。

产生关联规则的过程如图7. 1所示。

其中 q 分两步得到。第一步是连接，即 L2ooL2 = ｛ ｛II, 12,13｝ .｛11,12,15｝ .｛11,13, 15｝、｛12,13,14｝、｛12,13,15｝、｛12、14、15｝｝。第二步是利用 Apriori 性质剪枝：

* ｛11,12,13｝的2项子集为（11,12｝、｛12,13｝、｛11,13｝,它的所有2项子集都是乙中的 元素，因此保留这一项。
* ｛11,12,15｝的2项子集为｛11,12｝、｛12,15｝、｛11,15｝,它的所有2项子集都是L?中的 元素，因此保留这一项。
* ｛11,13,15｝的 2 项子集为｛11,13｝、｛11,15｝、｛13,15｝,其中｛13,15｝不是马中的元素, 因此删除这一项。
* ｛12,13,14｝的 2 项子集为｛12,13｝、｛12,14｝、｛13,14｝,其中｛13,14｝不是乙中的元素， 因此删除这一项。
* （12,13,15｝的 2 项子集为｛12,13｝、（12,15｝、｛13,15｝,其中｛13,15｝不是丄中的元素， 因此删除这一项。
* ｛12,14,15｝的 2 项子集为｛12,14｝、｛12,15｝、｛14,15｝,其中｛14,15｝不是 3 中的元素, 因此删除这一项。

经过上述的过程则可以得到图7. 1中的C3｛｛I1,I2,I3｝ JI1,I2,I5｝｝O这样就得到了全 部的频繁项集。



由频繁1项 集厶产生候 选集

| 项集 | 支持度计数 | | 项集 | 支持度计数 |
| --- | --- | --- | --- | --- |
| 11, | 12 | 4 | 12, 14 | 2 |
| 11, | 13 | 4 | 12, 15 | 2 |
| 11, | 14 | 1 | 13, 14 | 0 |
| 11, | 15 | 2 | 13 , 15 | 1 |
| 12, | 13 | 4 | 14, 15 | 0 |

由候选集

C2产生频 繁2项集3

| 项集 | 支持度计数 | |
| --- | --- | --- |
| 11, | 12 | 4 |
| 11, | 13 | 4 |
| 11, | 15 | 2 |
| 12, | 13 | 4 |
| 12, | 14 | 2 |
| 12, | 15 | 2 |

*L2*

图7.1产生关联规则的过程

**7.4.3**关联规则生成

得到所有的频繁项集后，关联规则就很容易了。对于置信度，可以用下面的公式计算： conHdenceCA^B) = P(A | B)=~~沖 port\_count(A [ B)~~

sup port\_count(A)

条件概率用项集的支持度计数表示，其中，sup port\_count(AUB)是包含项集AUB的 事务数，而sup port\_count(A)是包含项集A的事务数。由此，关联规则可以按以下的步骤 产生：

1. 对于每个频繁项集L,产生*L*的所有非空子集。
2. 对于L的每个非空子集S,如果~~‘叩P°r：M0unyq~~.2min conf(其中min conf是最

sup port\_count(s) ~ 一

小置信度阈值)，则输出规则*s=>(/ 一度。*

由于规则由频繁项集产生，因此每个规则都自动满足最小支持度。频繁项集和它们的 支持度可以预先放在散列表中，使得它们可以被快速访问。由前面的例子可以知道 AllElectronics 数据库包含频繁项集 X = ｛II, 12,15｝,其非空子集是｛II, 12｝、｛II, 15｝、｛15, 12｝、｛11｝、｛12｝、(15｝。产生关联规则如下：

｛II ,12｝ =>(15) ,confidence=2/4 = 50%

｛II, 15｝ => ｛12｝, confidence=2/2 = 100%

{12II} ,confidence=2/2 = 100%

{11} =>{12,15},confidence=2/6^33%

（Il）=>（12,15}, confidence = 2/633%

,15} ,confidence= 2/7^29%

{15 }n{Il ,12} ,confidence=2/2 = 100%

如果最小置信度值为70%,则只有第二个、第三个和最后一个可以输出，因为只有这些 产生强规则。

**7.5** 习题

1 .什么是数据挖掘？

1. 数据挖掘的主要内容是什么？
2. 数据分类和聚类有何不同？
3. 常用的数据挖掘的模型和算法有哪些？如何评价数据挖掘算法的优劣？
4. 简述数据挖掘的方法与过程。
5. 数据挖掘目前研究的热点是什么？谈谈你对数据挖掘研究发展趋势的看法。
6. 简述常用的分类算法，并编程实现朴素贝叶斯分类算法。
7. 简述常用的聚类算法，并编程实现人Means算法。
8. 简述常用的关联规则方法，并编程实现Apriori算法。
9. 对于下面每一个问题，请在购物篮领域举出一个满足下面条件规则的例子。此外， 指岀这些规则是否是主观上有趣的。

（1） 具有高支持度和高置信度的规则。

（2） 具有相当高的支持度却有较低置信度的规则。

（3） 具有低支持度和低置信度的规则。

（4） 具有低支持度和高置信度的规则。

1. 下表是西瓜数据集，试用奸Means算法对这些数据进行聚类，得到最终的簇划分。

|  |  |  |
| --- | --- | --- |
| 编 号 | 密 度 | 含糖率 |
| 1 | 0. 697 | 0.460 |
| 2 | 0.774 | 0.376 |
| 3 | 0. 634 | 0. 264 |
| 4 | 0. 608 | 0. 318 |
| 5 | 0. 556 | 0. 215 |
| 6 | 0. 403 | 0. 237 |
| 7 | 0. 481 | 0. 149 |
| 8 | 0. 437 | 0. 211 |
| 9 | 0.666 | 0.091 |
| 10 | 0. 243 | 0.237 |
| 11 | 0. 245 | 0.057 |

续表

|  |  |  |
| --- | --- | --- |
| 编 号 | 密 度 | 含糖率 |
| 12 | 0. 343 | 0. 099 |
| 13 | 0. 639 | 0.161 |
| 14 | 0.657 | 0.198 |
| 15 | 0.36 | 0. 370 |
| 16 | 0. 593 | 0. 042 |
| 17 | 0. 719 | 0.103 |
| 18 | 0. 359 | 0.188 |
| 19 | 0. 339 | 0.241 |
| 20 | 0. 282 | 0. 257 |
| 21 | 0. 748 | 0. 232 |
| 22 | 0. 714 | 0. 346 |
| 23 | 0.483 | 0. 314 |
| 24 | 0. 487 | 0. 432 |
| 25 | 0.525 | 0. 396 |
| 26 | 0. 462 | 0.472 |
| 27 | 0. 841 | 0. 376 |
| 28 | 0. 614 | 0. 445 |
| 29 | 0. 443 | 0.412 |
| 30 | 0. 325 | 0. 559 |

12.考虑下表中显示的购物篮事务。

|  |  |
| --- | --- |
| 事务ID | 购买项 |
| 1 | ｛牛奶，啤酒，尿布｝ |
| *2* | ｛面包，黄油，牛奶｝ |
| 3 | ｛牛奶，尿布，饼干｝ |
| 4 | ｛面包，黄油，饼干｝ |
| 5 | ｛啤酒，饼干，尿布｝ |
| 6 | ｛牛奶，尿布，面包，黄油｝ |
| 7 | ｛面包，黄油，尿布｝ |
| 8 | ｛啤酒，尿布｝ |
| 9 | ｛牛奶，尿布，面包，黄油｝ |
| 10 | ｛啤酒，饼干｝ |

（1） 从这些数据中能够提出的关联规则（包括零支持度的规则）的最大数量是多少？

（2） 找出一个具有最大支持度的项集（长度为2或更大）。

（3） 找出一对项。和。，使得规则*｛a｝^｛b｝*和｛a｝具有相同的置信度。

（4） 考虑下面的频繁 3 项集的集合：｛1,2,3｝,｛1,2,4｝,｛1,2,5｝,｛1,3,4｝,｛1,3,5｝,｛2, 3,4｝,｛2,3,5｝,｛3,4,5｝,假定数据集只有5个项。列出由Apriori算法的候选产生过程得 到的所有候选4项集。

（5） 对（4）中给出的频繁3项集的集合，列出由Apriori算法的候选剪枝步骤后剩下的 所有候选4项集。

大数据

**8.1**大数据概述

**8.1.1**大数据概念

大数据(big data)是一个抽象的概念，至今尚无确切、统一的定义，不同的研究机构与学 者对其有不同的定义。

全球最具权威的IT研究与顾问研究机构高德纳(The Gartner Group)咨询公司给出了 这样的定义：“大数据”是需要新处理模式才能具有更强的决策力、洞察发现力和流程优化 能力的海量、高增长率和多样化的信息资产。

麦肯锡全球研究所对大数据的定义是：一种规模大到在获取、存储、管理、分析方面大 大超出了传统数据库软件工具能力范围的数据集合，具有海量的数据规模、快速的数据流 转、多样的数据类型和价值密度低四大特征。

从狭义上讲，大数据主要是指大数据技术及其在各个领域中的应用，是指从各种各样类 ,型的数据中快速获得有价值的信息的能力。一方面，狭义的大数据反映的是数据规模非常 大，大到无法在一定时间内用一般性的常规软件工具对其内容进行抓取、管理和处理的数据 集合；另一方面，狭义的大数据主要是指对海量数据的获取、存储、管理、计算分析、挖掘与 应用的全新技术体系。

从广义上讲，大数据包括大数据技术、大数据工程、大数据科学和大数据应用等与大数 据相关的领域。即除了狭义的大数据之外，还包括大数据工程和大数据科学。大数据工程 是指大数据的规划建设运营管理的系统工程；大数据科学主要关注大数据网络发展和运营 过程中发现和验证大数据的规律及其与自然和社会活动之间的关系。对大数据进行广义分 类是为了适应经济时代发展需要而产生的科学技术发展的趋势。

大数据技术的战略意义不在于掌握庞大的数据信息，而在于对这些含有意义的数据进 行专业化处理。换而言之，如果把大数据比作一种产业，那么这种产业实现盈利的关键在于 提高对数据的“加工能力”，通过“加工”实现数据的“增值”。

从技术上看，大数据与云计算的关系就像一枚硬币的正反面一样密不可分。大数据必 然无法用单台的计算机进行处理，必须采用分布式架构。它的特色在于对海量数据进行分 布式数据挖掘，因此它必须依托云计算的分布式处理、分布式数据库和云存储、虚拟化技术。

随着云时代的来临，大数据也吸引了越来越多的关注。著云台的分析师团队认为，大数 据通常用来形容一个公司创造的大量非结构化数据和半结构化数据，这些数据在下载到关 系型数据库用于分析时会花费过多时间和金钱。大数据分析常和云计算联系到一起，因为 实时的大型数据集分析需要像MapReduce 一样的框架来向数十、数百或甚至数千的计算机 分配工作。

大数据需要特殊的技术，以有效地处理大量的容忍经过时间内的数据。适用于大数据 的技术包括大规模并行处理(MPP)数据库、数据挖掘电网、分布式文件系统、分布式数据 库、云计算平台、互联网和可扩展的存储系统。

**8. 1.2**特征

IBM公司认为大数据具有3V特点，即规模性(Volume).多样性(Variety)和实时性 (Velocity),但是这没有体现出大数据的巨大价值。而以IDC为代表的业界则认为大数据 具备4V特点，即在3V的基础上增加价值性(Value),具体表现为大数据虽然价值总量高但 其价值密度低。目前，大家公认的是大数据具有4个基本特征：数据规模大，数据种类多， 处理速度快以及数据价值密度低，即4V。

1. 数据规模大。数据量大是大数据的基本属性，随着互联网技术的广泛使用，互联网 用户急剧增多，数据的获取、分享变得相当容易。在以前，也许只有少量的机构会付出大量 的人力、财力成本，通过调查、取样的方法获取数据；而现在，普通用户也可以通过网络非常 方便地获取数据。此外，用户的分享、点击、浏览都可以快速地产生大量数据，大数据已从 TB级跃升到PB级。当然，随着技术的进步，这个数值还会不断变化。也许5年后，只有 EB级别的数据量才能够称得上是大数据了。
2. 数据种类多。除了传统的销售、库存等数据外，现在企业所采集和分析的数据还包 括像网站日志数据、呼叫中心通话记录.Twitter和Facebook等社交媒体中的文本数据、只 能由手机中内置的GPS(全球定位系统)产生的位置信息、时刻生成的传感器数据等。数据 类型不仅包括传统的关系数据类型，也包括未加工的、半结构化和非结构化的信息，例如以 网页、文档、E-mail、视频、音频等形式存在的数据。
3. 处理速度快。数据产生和更新的频率也是衡量大数据的一个重要特征。“1秒定 律”是大数据与传统数据挖掘相区别的最显著特征。例如，全国用户每天产生和更新的微 博、微信和股票信息等数据随时都在传输，这就要求处理数据的速度必须要快。
4. 数据价值密度低。数据量在呈现几何级数增长的同时，这些海量数据背后隐藏的 有用信息却没有呈现出相应比例的增长，反而是获取有用信息的难度不断加大。例如，现在 很多地方安装的监控使得相关部门可以获得连续的监控视频信息，这些视频信息产生了大 量数据，但是，有用的数据可能仅有一两秒钟。因此，大数据的4V特征不仅仅表达了数据

量大，而且在对大数据的分析上也将更加复杂，更看重速度与时效。

**8.1.3**发展历程

在1980年，著名未来学家阿尔文-托夫勒在《第三次浪潮》一书中，将“大数据”热情地 赞颂为“第三次浪潮的华彩乐章”。

最早提出“大数据”时代到来的是全球知名咨询公司麦肯锡。麦肯锡称：“数据已经渗 透到当今每一个行业和业务职能领域，成为重要的生产因素。人们对于海量数据的挖掘和 运用，预示着新一波生产率增长和消费者盈余浪潮的到来。”

《纽约时报》2012年2月的一篇专栏中称“大数据”时代已经降临，在商业、经济及其他 领域中，决策将日益基于数据和分析而作出，而并非基于经验和直觉。

哈佛大学社会学教授加里・金说：“这是一场革命,庞大的数据资源使得各个领域开始 了量化进程，无论学术界、商界还是政府，所有领域都将开始这种进程。”

亚马逊前首席科学家Andreas Weigend说：“数据是新的石油。”

2012年3月份美国奥巴马政府发布了“大数据研究和发展倡议”，投资两亿多美元，正 式启动“大数据发展计划”。计划在科学研究、环境、生物医学等领域利用大数据技术进行突 破。奥巴马政府的这一计划被视为美国政府继信息高速公路(Information Highway)计划 之后在信息科学领域的又一重大举措。

2012年5月，联合国发表名为《大数据促发展：挑战与机遇》的政务白皮书，指出大数据 对于联合国和各国政府来说是一个历史性的机遇，还探讨了如何利用包括社交网络在内大 数据资源造福人类。联合国的大数据白皮书还建议联合国成员国建设“脉搏实验室”(Pulse Labs)网络开发大数据的潜在价值。

随着2013年的一系列标志性事件的发生，人们越来越感觉到大数据时代的力量，因此 2013年被许多国外媒体和专家称为“大数据元年”。

2015年9月5日，国务院正式印发《促进大数据发展行动纲要》，这一行动纲要的出台 意味着大数据发展正式成为中国的国家战略。该纲要显示，国家将重点打造一批面向全球 的大数据龙头企业。大数据行业起飞的风口正在形成。

包括EMC、惠普、IBM、微软在内的全球IT巨头纷纷布局大数据。2015年最大的收购 案都与大数据有关，如Oracle对Sun、惠普对Autonomy。

**8.1.4**应用

大数据在人们生活各个方面都有所应用。

**1.**宏观经济领域

一些企业利用大数据分析实现对采购和合理库存量的管理，通过分析网上数据了解客 户需求，掌握市场动向。例如：

• IBM日本公司建立经济指标预测系统，从互联网新闻中搜索影响制造业的480项经 济数据，计算采购经理人指数的预测值。

•美国印第安纳大学利用谷歌公司提供的心情分析工具，从近千万条网民留言中归纳

出6种心情，进而对道琼斯工业指数的变化进行预测，准确率达到87%。

•有资料显示，全球零售商因盲目进货导致的销售损失每年达1000亿美元，这方面的 数据分析大有作为。

•华尔街对冲基金依据购物网站的顾客评论分析企业产品销售状况。

1. 农业领域

国内第一个农业大数据的研究和应用推广机构“农业大数据产业技术创新战略联盟”于 2013年6月18日在山东农业大学正式成立，标志着国内大数据技术在农业领域的应用有 了实质性突破。

农业大数据是大数据理念、技术和方法在农业的实践。农业大数据涉及耕地、播种、施 肥、杀虫、收割、存储、育种等各环节，是跨行业、跨专业、跨业务的数据分析与挖掘以及数据 可视化。

根据农业的产业链条划分，目前农业大数据主要集中在农业环境与资源、农业生产、农 业市场和农业管理等领域。

1. 农业自然资源与环境数据主要包括土地资源数据、水资源数据、气象资源数据、生 物资源数据和灾害数据。
2. 农业生产数据包括种植业生产数据和养殖业生产数据。其中，种植业生产数据包 括良种信息、地块耕种历史信息、育苗信息、播种信息、农药信息、化肥信息、农膜信息、灌溉 信息、农机信息和农情信息；养殖业生产数据主要包括个体系谱信息、个体特征信息、饲料 结构信息、圈舍环境信息、疫情情况等。
3. 农业市场数据包括市场供求信息、价格行情、生产资料市场信息、价格及利润、流通 市场和国际市场信息等。
4. 农业管理数据主要包括国民经济基本信息、国内生产信息、贸易信息、国际农产品 动态信息和突发事件信息等。
5. 商业领域

商业领域应用大数据实现商务智能。零售企业需要根据销售大数据供应有特色的本地 化商品并增加流行款式和生命周期短的产品，零售企业需要运用最先进的计算机和各种通 信技术对变化中的消费需求迅速做出反应。通过对大数据的挖掘，零售企业在选择上架产 品时，为确保提供新颖的商品，需要对消费者的消费行为以及趋势进行分析；在制定定价、 广告等策略时，需进行节假日、天气等大数据分析；在稳定收入源时，需要对消费群体进行 大数据分析。零售企业可以利用电话、Web、电子邮件等所有联络渠道的客户的数据进行分 析，并结合客户的购物习惯，提供一致的个性化购物体验，以提高客户忠诚度。同时，从微博 等社交媒体中挖掘实时数据，再将它们同实际销售信息进行整合，能够为企业提供真正意义 上的智能，了解市场发展趋势，理解客户的消费行为并为将来制定更加有针对性的策略。

1. 金融领域

国内金融业开始进入大数据竞争时代，大数据为金融机构提供了客户全方位信息，通过 分析和挖掘客户的交易和消费信息掌握客户的消费习惯，并准确预测客户行为，有针对性地 推销产品和服务，满足银行对潜在客户量身定制服务的需求。另外，在品牌管理和客户服务 反馈方面，通过大数据对人们在思想、情绪和通信方面的数据化情感分析，获取并汇总顾客 的反馈意见并对营销活动效果做出准确判断。

金融机构最为关注的是风险管理，而大数据在管理交易、信贷风险和合规方面大显神 通。许多金融机构早已釆用大数据技术防范欺诈,保持交易方面的合规，如在庞大的数据库 中核对黑名单中的名字，区别同名同姓。信用卡公司用大数据分析客户大规模的交易规律， 大大降低了风险。

虽然大数据时代为金融业带来的潜力和新商机还有待人们的进一步捕捉，但毋庸置疑 的是，大数据正在成为金融业运作中最有价值、最强大的决策辅助工具。

1. 医疗保健领域

大数据让专业医疗保健走入寻常百姓家。“谷歌流感趋势”项目依据网民搜索内容分析 全球范围内流感等病疫传播状况，与美国疾病控制和预防中心提供的报告对比，追踪疾病的 精确率达到97%。社交网络为许多慢性病患者提供临床症状交流和诊治经验分享平台，医 生借此可获得在医院通常得不到的临床效果统计数据。基于对人体基因的大数据分析，可 以实现对症下药的个性化治疗。

1. 社会安全领域

大数据提供更多的破案途径。通过对手机数据的挖掘，可以分析实时动态的流动人口 来源、出行，实时交通客流信息及拥堵情况。利用短信、微博、微信和搜索引擎，可收集热点 事件，挖掘舆情，还可追踪造谣信息的源头。美国麻省理工学院通过对十万多人手机的通 话、短信和空间位置等信息进行处理，提取人们行为的时空规律性，进行犯罪预测。

**8.2**数据获取

**8.2.1**网络爬山

**1.**概念与原理

网络爬虫（又称为网络蜘蛛，网络机器人，在FOAF社区中更经常称为网页追逐者）是 按照一定的规则自动抓取万维网信息的程序或脚本。另外一些不经常使用的名字还有蚂 蚁、自动索引、模拟程序或者蠕虫。它的定义有广义和狭义之分。狭义上指遵循标准的 HTTP协议，利用超链接和Web文档检索方法遍历万维网的软件程序；而广义上则凡是遵 循HTTP协议检索Web文档的软件都称为网络爬虫。

网络爬虫是一个功能很强的自动提取网页的程序，它为搜索引擎从万维网上下载网页，是 搜索引擎抓取系统的重要组成部分。整个搜索引擎系统主要包含4个模块，分别为信息搜索 模块、信息索引模块、信息检索模块和用户接口部分，而网络爬虫便是信息搜索模块的核心。

如果把互联网比作一个大昆虫织的网，网络爬虫就是在这张大网上爬来爬去的爬虫。 网络爬虫的主要目的是将互联网上的网页下载到本地形成一个互联网内容的镜像备份。

网络爬虫可以使用多线程技术，以具备更强大的抓取能力。

可以通过使用DNSCache技术减少爬虫对DNS的访问频率，避免DNS成为网络瓶 颈，提高抓取速度。

通过Java技术，以多线程方式可以大大增强爬虫抓取网页的效率。对于搜索引擎来

说，要想通过网络爬虫搜索到整个网络的页面是几乎不可能的，主要有两个原因：一是通过 现有的手段无法搜索到所有网站的网页，容量再大的搜索引擎系统也不能搜索到所有的网 页，这是一个技术瓶颈问题；二是存储问题和技术处理问题，简单地用一个数学计算方式可 以看出，比如一个普通网页大概有100KB(其中包含图片)，目前根据非官方的统计数据互 联网大概有1万亿个网页，数量这么庞大的网页再乘以网页的大小，对于任何搜索引擎来 说都是一个海量的数字。

网络爬虫还要完成信息提取任务，从抓取的网页中提取新闻、电子图书、行业信息等； 对于MP3、图片、Flash等各种不同内容，要实现自动识别、自动分类及相关属性测试(例如， MP3文件要包含的文件大小、下载速度等属性)。因为网页在网站内有自己所处的级数，网 络爬虫还要根据不同网页级数来提取抓取信息。网络爬虫的设计需能够自己判断出网页所 处的级数。

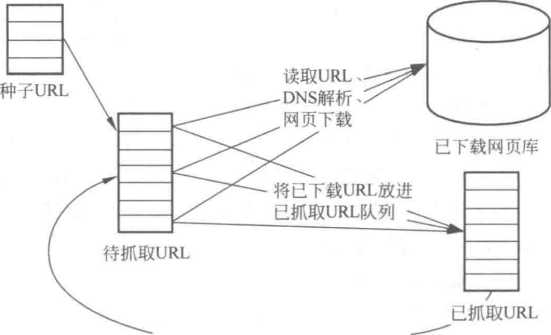
1. 系统架构

在网络爬虫的系统架构中，主过程由控制器、解析器、资源库3部分组成。

1. 控制器的主要工作是负责给多线程的各个爬虫线程分配工作任务。
2. 解析器的主要工作是下载网页，进行页面的处理，主要是将一些JavaScript脚本标 签、CSS代码内容、空格字符.HTML标签等内容处理掉，爬虫的基本工作是由解析器完成。
3. 资源库用来存放下载到的网页资源，一般采用大型的数据库存储，如Oracle数据 库，并对其建立索引。
4. 工作流程

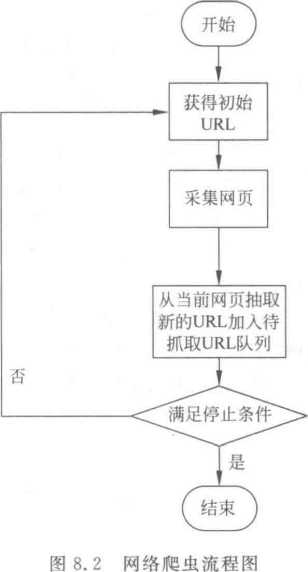
网络爬虫的工作流程如图8. 1所示，具体说明如下：

1. 选取一部分精心挑选的种子URLO
2. 将这些URL放入待抓取URL队列。
3. 从待抓取URL队列中取出待抓取的URL,解析DNS,得到主机的IP,并将URL 对应的网页下载下来，存储到已下载网页库中。此外，将这些URL放进已抓取URL队列。
4. 分析已抓取URL队列中的URL,分析其中的其他URL,并且将这些新的URL放 入待抓取URL队列，从而进入下一个循环。



抽取岀新的URL放入待抓取URL队列

图8.1网络爬虫的工作流程

网络爬虫是搜索引擎中最核心的部分，整个搜索引擎的素材库来源于网络爬虫的采集， 从搜索引擎整个产业链来看，网络爬虫是处于最上游的产业，其性能好坏直接影响着搜索引 擎整体性能和处理速度。

通常网络爬虫从一个或若干个初始网页上的URL 开始，获得初始网页上的URL列表，在抓取网页过程 中，不断从当前页面上抽取新的URL放入待抓取 URL队列，直到满足系统的停止条件，如图8. 2所示。

网络爬虫各个部分的主要功能如下：

1. 页面采集模块。该模块是爬虫和因特网的接 口，主要作用是通过各种Web协议(一般以HTTP、 FTP为主)完成对网页数据的采集，保存后将采集到的 页面交给后续模块做进一步处理。其过程类似于用户 使用浏览器打开网页，保存的网页供其他后续模块处 理，例如页面分析、链接抽取。
2. 页面分析模块。该模块的主要功能是对页面 采集模块采集的页面进行分析，提取其中满足用户要求 的超链接，加入超链接队列中。页面链接中给岀的 URL 一般是多种格式的.可能是完整的包括协议、站点 和路径的•也可能是省略了部分内容的，还可能是一个 相对路径。所以为处理方便，一般进行规范化处理，先 将其转化成统一的格式。
3. 链接过滤模块。该模块主要是用于对重复链接和循环链接的过滤。例如，相对路 径需要补全URL,然后加入到待采集URL队列中。此时，一般会过滤掉队列中已经包含的 URL以及循环链接的URL。
4. 页面库。用来存放已经采集下来的页面，以备后期处理。
5. 待抓取URL队列。从采集网页中抽取并作相应处理后得到的URL,当URL为空 时爬虫程序终止。
6. 初始URL0提供URL种子，以启动爬虫。
7. 抓取对象

网络爬虫的抓取对象可以分为以下4类：

1. 静态网页。网络爬虫在互联网上从一个网站的初始网页开始，获得网页上的链接, 在抓取过程中不断获得新的链接，直到达到系统指定的方式才会停止。
2. 动态网页。先通过程序分析一些非静态网页的参数，按一定的规则对所有要抓取 页面的链接进行整理，程序只会抓取这些特定范围内的网页。
3. 特殊内容。比如RSS.XML数据，由于情况特殊，需特殊处理。例如，新闻的滚动 页面需要爬虫不停地监控扫描，发现新内容马上就抓取。
4. 文件对象。目前网页上会有各种类型的文件，如图片.MP3.Flash.视频等文件，这 些都需要系统用一定的方式处理。例如，视频被抓取后，要知道其类型、文件大小、分辨 率等。
5. 抓取策略

网络爬虫在执行搜索任务时会采取一定的抓取策略，每种策略的抓取方式都不一样,执 行的效率也不一样。以下是常用的抓取策略。

1）深度优先策略

对于一些大网站和以静态网页为主的抓取内容，采取深度优先策略抓取，以便在最短时 间内获得最大量的内容。

深度优先策略是在开发爬虫早期使用较多的方法，它的目的是要达到被搜索结构的叶 节点（即那些不包含任何超链接的HTML文件）。采取深度抓取方式的时候，搜索引擎会 从网页的起始页开始，一个链接一个链接地跟踪下去，直至把这条线路追查完毕，然后再转 向另一个网页线路，如此不停地搜索循环下去。深度优先搜索沿着HTML文件上的超链 接走到不能再深入为止，然后返回到某一个HTML文件，再继续选择该HTML文件中的 其他超链接。当不再有其他超链接可选择时，说明搜索已经结束。

这种策略的优点是能遍历一个Web站点或深层嵌套的文档集合。缺点是因为Web结 构相当深，有可能造成一旦进去再也出不来的情况发生。

对图8. 3所示的网页结构使用深度优先策略抓取的顺序为：A-F-G、E-H-I、B、C、D。

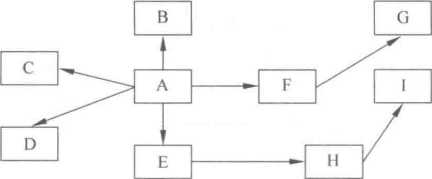


图8. 3网页结构示例

2） 广度优先策略

对于一些动态网页或小网站，采取广度优先策略抓取，搜索引擎会先抓取起始网页中链 接的所有网页，然后再选择其中的一个链接网页，继续抓取在此网页中链接的所有网页。在 抓取过程中，在完成当前层次的搜索后，才进行下一层次的搜索，逐层进行搜索。这是最常 用的方法，因为这个方法可以让网络爬虫并行处理，提高其抓取速度。广度优先搜索策略通 常是实现爬虫的最佳策略。因为它容易实现，而且具备大多数期望的功能。但是如果要遍 历一个指定的站点或者深层嵌套的HTML文件集，用广度优先搜索策略则需要花费较长 时间才能到达深层的HTML文件。

对图8. 3所示的网页结构使用广度优先策略抓取的顺序为：A-B、C、D、E、F-G、H-I°

3） 聚焦搜索策略

聚焦搜索策略只挑出某个特定主题的页面，根据“最好优先原则”进行访问，快速、有效 地获得更多的与主题相关的页面。聚焦爬虫在页面搜索时会对自己搜索到的页面进行评 价，在评价后给出分值，在对得分进行排序后会把排序表插入到一个队列中。在自己发起的 下一个搜索中会对弹岀队列的第一个页面进行分析，以这种策略来追踪目标页面的可能性 很大。聚焦搜索策略最关键的部分就是链接价值的计算方法，不同的计算方法会带来不同 的评分价值，得到的评价级别也不一样，这就决定了搜索策略的不同。

1. 最佳优先搜索策略。这种策略按照一定的网页分析算法，先计算出URL描述文本 的目标网页的相似度，设定一个值，并选取评价得分超过该值的一个或几个URL进行抓 取。它只访问经过网页分析算法计算出的相关度大于给定值的网页。这种策略存在的一个 问题是，在爬虫抓取路径上的很多相关网页可能被忽略.因为最佳优先策略是一种局部最优 搜索算法。因此需要结合具体的应用对搜索策略进行改进，以跳出局部最优点。有研究表 明，这样的闭环调整可以将无关网页数量降低30%〜90%。
2. 基于IP地址的搜索策略。先赋予爬虫一个起始的IP地址，然后根据IP地址递增 的方式搜索本IP地址段后的每一个WWW地址中的文档，它完全不考虑各文档中指向其 他Web站点的超级链接地址。优点是搜索全面，能够发现那些没被其他文档引用的新文档 的信息源，缺点是不适合大规模搜索。

搜索策略目前常见的是广度优先策略和最佳优先搜索策略。

1. 关键技术分析
2. 抓取目标的定义与描述
3. 针对有目标网页特征的网页级信息。对应网页库级垂直搜索，抓取目标网页，后续 还要从中抽取出需要的结构化信息。这种技术在稳定性和数量上占优，但成本高、灵活 性差。
4. 针对目标网页上的结构化数据。对应模板级垂直搜索，直接解析页面，提取并加工 出结构化数据信息。这种技术实施快，成本低，灵活性强，但后期维护成本高。
5. 网页的分析与信息的提取
6. 基于网络拓扑关系的分析算法。根据页面间超链接引用关系对与已知网页有直接 或间接关系的对象作出评价的算法，如网页粒度PageRank算法、网站粒度SiteRank算法。
7. 基于网页内容的分析算法。从最初的文本检索方法向涉及网页数据抽取、机器学 习、数据挖掘、自然语言等多领域综合的方向发展。
8. 基于用户访问行为的分析算法。有代表性的是基于领域概念的分析算法，涉及本 体论。
9. 发展趋势

随着网络的不断发展，大量有价值的网页会隐藏在深层网络中，现在的网络爬虫对深层 的网页中动态网页和数据库基本上是束手无策的。在现在搜索模式下如何跟上互联网这种 发展趋势变得异常重要，深层的网络爬虫研究变得更加迫切。

AJAX技术已在网页中经常被应用到。使用AJAX的最大优点是网站维护数据可以 不必更新整个页面，这样，Web应用程序可以更加快速地回应用户动作，并避免了在网络上 发送那些没有改变的信息。这样的无闪局部刷新可以加快网页的刷新速度。

随着网络的不断发展，各种多媒体信息都出现在网页上，比如海量的图片、动画、游戏、 视频等，这些都需要搜索引擎有应对之策。伴随着搜索引擎的发展，各种基于网络的多媒体 爬虫技术研究将会成为爬虫研究的新方向。

随着对等网络P2P技术的发展，网络不是将所有的压力都分布在服务器端，而是将压 力分担到每一台用户的计算机上，这样每台客户端的计算机将作为主机完成上传和下载工 作。网络成员可在网络数据库里自由搜索、更新、回答和传送数据。

**8.** *2. 2* **RSS**

维基百科对RSS的定义如下：RSS(Really Simple Syndication,简易信息聚合)是一种 消息来源格式规范，用以发布经常更新资料的网站，例如Blog文章、新闻、音讯或视讯的网 摘。网络摘要专业层面能够使网站自动地发布它们的资料，同时也使读者能够定期更新他 们喜欢的网站、电视剧和不同网站的网摘。

RSS简称聚合内容，目前广泛应用于各类型网站，其功能一般为最新信息的输出。

1. *RSS可以做什么*

RSS能实现以下功能：

1. 订阅BLOGo可以订阅工作中所需的技术文章，也可以订阅与自己有共同爱好的 作者的博客，总之，对什么感兴趣就可以订阅什么。
2. 订阅新闻。无论是奇闻怪事、明星消息还是体坛风云，只要想知道的，都可以订阅。 用户再也不用一个网站一个网站、一个网页一个网页去逛了。只要在一个RSS阅读器中订 阅需要的内容，这些内容就会自动出现在RSS阅读器中，用户也不必为了一个急切想知道 的消息而不断地刷新网页，因为一旦有了更新，RSS阅读器就会自动通知用户。
3. 订阅杂志文章。用户再也不用一个杂志一个杂志地去查看有没有新发表的论文 了，只要在一个RSS阅读器中订阅自己喜欢的杂志，每篇新出版的文章(甚至是刚接受的文 章)就会自动地岀现在用户的阅读器中。
4. 订阅最新搜索结果。订阅自己感兴趣的研究方向的最新论文的搜索结果，当该方 向有了新论文后，会自动地出现在RSS阅读器中。
5. 快速、高效地浏览。每一个条目都是以标题和摘要的形式出现的，方便用户快速浏 览，使用户可以在最短的时间内浏览海量信息，然后快速地从中找出自己感兴趣的内容。
6. **RSS**订阅

RSS阅读器基本可以分为3类。

1. 大多数阅读器是运行在计算机桌面上的应用程序，通过所订阅网站的新闻推送，可 自动、定时地更新新闻标题。在该类阅读器中，有Awasu.FeedDemon和RSSReader这3 款流行的阅读器，都提供免费试用版和付费高级版。
2. 新闻阅读器通常是内嵌于已在计算机中运行的应用程序中。例如,NewsGator内 嵌在微软公司的Outlook中，用户订阅的新闻标题位于Outlook的“收件箱”文件夹中； Pluck 内嵌在 Internet Explorer 浏览器中。
3. 在线的Web RSS阅读器，其优势在于不需要安装任何软件就可以获得RSS阅读 的便利，并且可以保存阅读状态，推荐和收藏自己感兴趣的文章。提供此服务的有两类网 站：一种是专门提供RSS阅读器的网站，例如国外的feedly,国内的有道、鲜果、抓虾；另一 种是提供个性化首页的网站，例如国外的netvibes.pageflakes,国内的雅蛙、阔地。

RSS订阅是站点用来和其他站点之间共享内容的一种简易方式，通常在时效性比较强 的信息上使用RSS订阅，如新闻、期刊、博客等。

订阅途径有博客站点、新闻发布站点、学术信息资源站点等。其中学术信息资源的 RSS发布包括：学术机构网站信息发布，如图书馆、大学院系、学术服务机构等；学术资源 网站的信息发布，如电子期刊网站最新目次、数据库(某一主题/关键词的更新论文、某种学 术期刊更新论文)等。

Rss订阅的步骤如下：

1. 第一次使用时，先下载安装RSS阅读软件。
2. 清理RSS阅读软件不必要的内置频道。
3. 右击“RSS信息订阅”，复制频道的链接地址(URL)O
4. 运行RSS阅读软件，从文件菜单中选择“添加新频道”，将链接地址(URL)粘贴到 输入框中，再按照提示操作，即完成了一个频道的定制。
5. 单击频道名即可查阅随时更新的信息。

**3.**第三方

第三方指两个相互联系的主体之外的某个客体。第三方可以和两个主体有联系，也可 以独立于两个主体之外。

所谓第三方支付，就是一些和产品所在国家以及国外各大银行签约并具备一定实力和 信誉保障的第三方独立机构提供的交易支持平台。在通过第三方支付平台的交易中，买方 选购商品后，使用第三方平台提供的账户进行货款支付，由第三方通知卖家货款到达，进行 发货；买方检验物品后，就可以通知第三方付款给卖家，第三方再将款项转至卖家账户。

购买者并非使用者，使用者并非最大的受益者，真正的受益者并非决策者，这就是第三 方买单的逻辑。你消费，不用自己买单，产品或服务的提供商根本不收你的钱，而且你消费 得越多，厂商还越高兴。这种消费模式之所以能够一直存在，是因为有第三方在替你买单， 替产品或服务的提供商支付费用。这种经济模式就被称为“第三方买单”。

下面介绍其实现原理。除了网上银行、电子信用卡等手段之外还有一种方式也可以相 对降低网络支付的风险，那就是正在迅猛发展起来的利用第三方机构的支付模式及其支付 流程,而这个第三方机构必须具有一定的诚信度。在实际的操作过程中这个第三方机构可 以是发行信用卡的银行本身。在进行网络支付时，信用卡号以及密码的披露只在持卡人和 银行之间转移，降低了因通过商家转移而导致的风险。

同样，当第三方是除了银行以外的具有良好信誉和技术支持能力的某个机构时，支付也 通过第三方在持卡人或者客户和银行之间进行。持卡人首先和第三方以替代银行账号的某 种电子数据的形式(例如邮件)传递账户信息，避免了持卡人将银行信息直接透露给商家，另 外也可以不必登录不同的网上银行界面，取而代之的是每次登录时都能看到相对熟悉和简 单的第三方机构的界面。

第三方机构与各个主要银行之间再签订有关协议，使得第三方机构与银行可以进行某 种形式的数据交换和相关信息确认。这样第三方机构就能实现在持卡人或消费者与各个银 行以及最终的收款人或者商家之间建立一个支付的流程。

**8.3**数据挖掘

**8.3.1**概述

从技术角度，数据挖掘(data mining)是从大量的、不完全的、有噪声的、模糊的、随机的 实际应用数据中提取隐含在其中的、人们事先不知道的、但又是潜在有用的信息和知识的过 程。与数据挖掘相近的同义词包括数据融合、数据分析和决策支持等。

这一定义包括好几层含义：数据源必须是真实的、海量的、含噪声的；发现的是用户感 兴趣的知识；发现的知识要可接受、可理解、可运用；并不要求发现放之四海皆准的知识, 仅支持特定的发现问题。

从商业角度，数据挖掘是一种新的商业信息处理技术，其主要特点是对商业数据库中的 大量业务数据进行抽取、转换、分析和其他模型化处理，从中提取辅助商业决策的关键性 信息。 •

简言之，数据挖掘其实是一类深层次的数据分析方法。因此，数据挖掘可以描述为：按 企业既定业务目标，对大量的企业数据进行探索和分析，揭示隐藏的、未知的或验证已知的 规律性，并进一步将其模型化的有效方法。

数据挖掘作为一门新兴的交叉学科，涉及数据库系统、数据仓库、统计学、机器学习、可 视化、信息检索和高性能计算等诸多领域。

此外，数据挖掘还与神经网络、模式识别、空间数据分析、图像处理、信号处理、概率论、 图论和归纳逻辑等领域关系密切。

数据挖掘与统计学有密切关系，近几年，人们逐渐发现数据挖掘中有许多工作是由统计 方法来完成的。甚至有些人（尤其是统计学家）认为数据挖掘是统计学的一个分支，当然大 多数人（包括绝大多数数据挖掘研究人员）并不这么认为。

但是，统计学和数据挖掘的目标非常相似，而且数据挖掘中的许多算法也源于数理统 计,统计学对数据挖掘发展的贡献功不可没。

数据挖掘与传统数据分析方法主要有以下两点区别：

首先，数据挖掘的数据源与以前相比有了显著的改变，包括数据是海量的，数据有噪声， 数据可能是非结构化的。

其次，传统的数据分析方法一般都是先给出一个假设，然后通过数据验证，在一定意义 上是假设驱动的；与之相反，数据挖掘在一定意义上是发现驱动的，模式都是通过大量的搜 索工作从数据中自动提取出来的。即数据挖掘是要发现那些不能靠直觉发现的信息或知 识，甚至是违背直觉的信息或知识,挖掘出的信息越是出乎意料，就可能越有价值。

在缺乏强有力的数据分析工具而不能分析这些资源的情况下，历史数据库也就变成了 “数据坟墓”一一里面的数据几乎不再被访问。也就是说，极有价值的信息被“淹没”在海量 数据堆中，领导者决策时还只能凭自己的经验和直觉。因此改进原有的数据分析方法，使之 能够智能地处理海量数据，也就演化为数据挖掘。

研究数据挖掘的目的，不再是单纯为了研究，更主要的是为商业决策提供真正有价值的 信息，进而获得利润。目前所有企业面临的一个共同问题是，企业数据量非常大，而其中真 正有价值的信息却很少，因此需要经过深层分析，从大量的数据中获得有利于商业运作、提 高竞争力的信息，就像从矿石中淘金一样，数据挖掘也由此而得名。

**8.3.2**数据挖掘工具

目前，世界上比较有影响的典型数据挖掘系统包括Enterprise Miner （ SAS公司）、

Intelligent Miner（ IBM 公司）、SetMiner （ SGI 公司）、Clementine （ SPSS 公司）、Warehouse Studio （ Sybase 公司）、See5 （ RuleQuest Research 公司）、CoverStory、EXPLORA、Knowledge Discovery Workbench^DBMiner^ Quest 等。

数据挖掘工具的选择可以考虑如下几点： J

（1） 商用数据挖掘系统各不相同。 ♦

（2） 不同的数据挖掘工具的功能和使用方法不同。

（3） 数据集的类型可能完全不同。例如：

•数据类型——是关系型的、事务型的、文本的、时间序列的还是空间的？

•系统问题——支持一种还是多种操作系统？是否釆用C/S架构？是否提供Web接 口且允许输入输出XML数据？

•数据源——是ASCII文件、文本文件还是多个关系型数据源？是否支持ODBC连 接（OLE DBJDBC）?

本节介绍两种典型的数据挖掘工具 Amdocs和Predictive CRM0

1. **Amdocs**

在多年前电信行业已经开始利用数据挖掘技术进行网络出错预测等方面的工作，而近 年来随着CRM理念的盛行，数据挖掘技术开始在市场分析和决策支持等方面得到广泛应 用。市场上更出现了针对电信行业的包含数据挖掘功能的软件产品。比较典型的有 Amdocs 和 Predictive CRMO

Amdocs提供了整个电信运营企业的软件支撑平台。在其Clarify CRM产品组件中， 利用数据挖掘技术支持以下应用：客户流失管理（churn management） s终身价值分析 （lifetime value analysis）、产品分析（product analysis）、欺诈甄别（fraud detection） o

Amdocs产品中的数据分析和数据分析应用曾获得3届KDD杯奖。

1. **Predictive CRM**

Sip Infoware开发的Predictive CRM软件是一个面向电信行业的CRM平台软件，其 中应用了大量的数据挖掘和统计学技术。其数据挖掘部分实际上是把SAS Institute.SPSS 和UNICA等公司的数据挖掘产品加以二次开发以适应电信行业的需要。数据挖掘在P- CRM中的应用包括客户保持、交叉销售、客户流失管理、欺诈甄别等方面。

利用SAS软件技术进行数据挖掘可以有3种方式：

（1） 使用SAS软件模块组合进行数据挖掘。

（2） 将若干SAS软件模块连接成一个适合需求的综合应用软件。

（3） 使用SAS数据挖掘的集成软件工具SAS/EM。

SAS/EM是一个图形化界面、菜单驱动、对用户非常友好且功能强大的数据挖掘集成 软件，集成了数据获取工具、数据取样工具、数据筛选工具、数据变量转换工具、数据挖掘数 据库、数据挖掘过程、多种形式的回归工具、建立决策树的数据剖分工具、决策树浏览工具、 人工神经元网络、数据挖掘的评价工具。

目前，虽然已经有了许多成熟的商业数据挖掘工具,但这些工具一般都是一个独立的系 统，不容易与电信企业现有的业务支撑系统集成。而且由于数据挖掘技术本身的特点，一个 通用的数据挖掘系统可能并不适用于电信企业。

切实可行的办法是借鉴成熟的经验，结合自身特点开发专用的数据挖掘系统。

**8.3.3** 现状与未来

数据挖掘本质上是一种深层次的数据分析方法。

数据分析本身已有多年的历史，只不过在过去数据收集和分析的一般目的是用于科学 研究；另外，由于当时计算能力的限制，很难实现大量数据的复杂分析。

现在，由于各行业业务自动化的实现，商业领域产生了大量的业务数据，这些数据并不 是为了分析的目的而收集的，而是在商业运作过程中由于业务需要而自然产生的。

IEEE 的会刊 *Knowledge and Data Engineering* 率先在 1993 年出版了 KDD 技术专 刊。并行计算、计算机网络和信息工程等其他领域的国际学会、学刊也把数据挖掘和知识发 现列为专题和专刊讨论。数据挖掘已经成为国际学术研究的重要热点之一。

此外，在Internet上还有不少KDD电子出版物，其中以半月刊*Knowledge Discovery Nuggets* 最为权威(http：//www. kdnuggets. com/subscribe, html) o 在网上还有许多自由 论坛，如 DM Email Club 等。

自1989年KDD术语出现以来，由美国人工智能协会主办的KDD国际研讨会已经召 开了 10次以上，规模由原来的专题讨论会发展到国际学术大会。而亚太地区也从1997年 开始举行PAKDD年会。

与国外相比，国内对数据挖掘的研究起步稍晚，但发展势头强劲。1993年，国家自然科 学基金首次资助复旦大学在该领域的研究项目。目前，国内的许多科研单位和高等院校竞 相开展知识发现的基础理论及其应用研究。

近年来，数据挖掘的研究重点逐渐从发现方法转向系统应用，注重多种发现策略和技术 的集成以及多学科之间的相互渗透。

**8.4**数据分析

**8.4.1**概述

数据分析是指用适当的统计分析方法对收集来的大量数据进行分析，提取有用信息和 形成结论而对数据加以详细研究和概括总结的过程。这一过程也是质量管理体系的支持过 程。在产品的整个寿命周期，包括从市场调研到售后服务和最终处置的各个过程都需要适 当运用数据分析过程，以提升有效性。例如J.开普勒通过分析行星角位置的观测数据找出 了行星运动规律。又如，一个企业的领导人要通过市场调查分析所得数据来判定市场动向， 从而制定合适的生产及销售计划。因此数据分析有极广泛的应用范围。

数据分析的目的是把隐没在一大批看来杂乱无章的数据中的信息集中、萃取和提炼出 来，以找出所研究对象的内在规律。在实用中，数据分析可帮助人们作出判断，以便采取适 当行动。通过分析手段、方法和技巧对准备好的数据进行探索、分析，从中发现因果关系、内 部联系和业务规律，为商业目标提供决策参考。

仅仅知道怎么看数据是远远不够的，还要了解使用这些数据，怎么让数据显示出它本身

的威力。总结下来有以下几个方面：

1. 看历史数据，发现规律。
2. 从历史数据和现有数据中发现端倪，找出问题所在。在工作中，每天都会接触到大 量的数据，但是大部分时间人们看数据流于表面。数据就是我们的助手，能够帮助我们发现 问题，同时顺藤摸瓜找到问题的根源所在。这个能力是非常重要的。
3. 数据预测。通过分析数据，发现其中的规律，那么则可实现数据驱动运营，驱动产 品，驱动市场。
4. 学会拆解数据。要会对数据进行拆分，知道每个数据都是来自哪些方面，增高或者 降低的趋势是什么。

近几年来，数据分析在互联网领域非常受重视，无论是社区型产品、工具类产品还是电 子商务，都越来越把数据作为核心资产。确实，数据分析做得越深，越能够实现精细化运营, 在很多时候工作的重点才有据可依。但是要注意两方面的问题：

1. 不能唯数据论。数据有时候能够反馈一些问题，但是也要注意到，在有些时候数据 并不能说明所有问题，也需要综合各方面的情况整体来看。

同时要有数据分析的思维，不仅是互联网行业，几乎所有的行业每天都会产生大量的数 据。最重要的是知道怎么通过数据分析找出规律，发现问题，对将来做出预测。

1. 找到适合自己产品的数据指标。不同产品的特性和用户使用习惯都不一样，需要 找到适合自己产品的指标参数而不是随大流。

数据分析包括以下5个方面：

1. 可视化分析(analytic visualizations) o不管是对数据分析专家还是普通用户，数据 可视化都是数据分析工具最基本的要求。可视化可以直观的展示数据，让数据自己说话，让 观众听到结果。
2. 数据挖掘算法(data mining algorithms) 0可视化是给人看的，数据挖掘是给机器 看的。集群、分割、孤立点分析等算法让人们深入数据内部，挖掘价值。这些算法不仅要处 理大数据的量，也要处理大数据的速度。
3. 预测性分析能力(predictive analytic capabilities) 0数据挖掘可以让分析员更好地 理解数据，而预测性分析可以让分析员根据可视化分析和数据挖掘的结果做出一些预测性 的判断。
4. 语义引擎(semantic engines) o非结构化数据的多样性给数据分析提岀了新的挑 战，需要用一系列工具解析、提取、分析数据。语义引擎要设计成能够从“文档”中智能地提 取信息。
5. 数据质量和管理(data quality and management) o数据质量和数据管理是一些管 理方面的最佳实践。通过标准化的流程和工具对数据进行处理可以保证一个预先定义好的 高质量的分析结果。

**8.4.2**数据分析流程

数据分析流程概括起来主要包括明确分析目的与框架、数据收集、数据处理、数据分析、 数据展现和撰写报告6个阶段。

1. 明确分析目的与框架

明确分析目的与框架是进行数据分析的先决条件，为数据分析提供了方向。

**I** 一个分析项目，数据对象是什么？商业目的是什么？要解决什么业务问题？对这些问

♦ 题都要了然于心。

要基于商业的理解，整理分析框架和分析思路，例如减少新客户的流失、优化活动效果、 提高客户响应率等。不同的项目对数据的要求以及使用的分析手段都是不一样的。

1. 数据收集

数据收集是通过数据库和其他媒介按照确定的数据分析和框架内容，有目的地收集、整 合相关数据的过程，它是数据分析的基础。

1. 数据处理

数据处理是指对收集到的数据进行加工、整理，以便开展数据分析，它是数据分析前必 不可少的阶段。这个过程是数据分析整个过程中最占据时间的，也在一定程度上取决于数 据仓库的搭建和数据质量的保证。

数据处理主要包括数据清洗、数据转化、提取、计算等处理方法。

1. 数据分析

数据分析是指通过分析手段、方法和技巧对准备好的数据进行探索、分析，从中发现因 果关系、内部联系和业务规律，为商业目的提供决策参考。

到了这个阶段，要能驾驭数据，开展数据分析，就要涉及工具和方法的使用。首先要熟 悉常规数据分析方法，例如方差分析、回归分析、因子分析、判别分析、聚类、分类、时间序列 等，要了解这些方法的原理、使用范围、优缺点和对结果的解释。其次要熟悉数据分析工具， Excel是最常见的数据分析工具，一般的数据分析可以通过Excel完成；此外还要熟悉一些 专业的分析软件，如数据分析工具SPSS/SAS/Clementine/Matlab等，以便进行一些专业的 统计分析、数据建模等。

1. 数据展现

一般情况下，数据分析的结果都是通过图表、表格、文字的方式来呈现。借助数据展现 手段，能更直观地表述想要呈现的信息、观点和建议。常用的图表包括饼图、折线图、柱形 图/条形图、散点图、雷达图、金字塔图、矩阵图、漏斗图、帕累托图等。

1. 撰写报告

最后一'个阶段就是撰写数据分析报告，这是对整个数据分析成果的一个呈现。通过分 析报告，把数据分析的目的、过程、结果及方案完整地呈现出来。

一份好的数据分析报告要求框架清晰，结论明确，提出建议。首先需要有一个好的分析 框架，并且图文并茂，层次明晰，能够让阅读者一目了然。结构清晰、主次分明可以使阅读者 正确理解报告内容；图文并茂可以令数据更加生动活泼.提高视觉冲击力，有助于阅读者更 形象、直观地看清楚问题和结论，从而引发思考。另外，数据分析报告要有明确的结论、建议 和解决方案，而不仅仅是找出问题，这一部分是更重要的，否则称不上好的分析，同时也失去 了报告的意义，数据分析的初衷就是服务于商业目的，不能舍本求末。

**8.4.3**数据分析方法

常用数据分析方法有以下几种。

1. 聚类分析(cluster analysis) o指将物理或抽象对象的集合分组成为由类似的对象 组成的多个类的分析过程。聚类是将数据分类到不同的类或者簇的过程，所以同一个簇中 的对象有很大的相似性，而不同簇间的对象有很大的相异性。聚类分析是一种探索性的分 析，在分类的过程中，人们不必事先给出一个分类的标准，聚类分析能够从样本数据出发，自 动进行分类。聚类分析所使用的方法不同，常常会得到不同的结论。不同研究者对于同一 组数据进行聚类分析，所得到的聚类结果未必一致。
2. 因子分析(factor analysis) 0指研究从变量群中提取共性因子的统计技术。因子分 析就是从大量的数据中寻找内在的联系，减小决策的困难。因子分析的方法有十多种，如重 心法、影像分析法、最大似然解、最小平方法、阿尔发抽因法、拉奥典型抽因法等。这些方法 本质上都属近似方法，是以相关系数矩阵为基础的，所不同的是相关系数矩阵对角线上的 值，釆用不同的共同性估值。
3. 相关分析(correlation analysis) o这种方法研究现象之间是否存在某种依存关系， 并对具体有依存关系的现象探讨其相关方向以及相关程度。相关关系是一种非确定性的关 系，例如，以X和Y分别记一个人的身高和体重，或分别记每公顷施肥量与每公顷小麦产 量，则X与丫显然有关系，而又没有确切到可由其中的一个去精确地决定另一个的程度，这 就是相关关系。
4. 对应分析(correspondence analysis) o也称关联分析、R-Q型因子分析，通过分析 由定性变量构成的交互汇总表来揭示变量间的联系。可以揭示同一变量的各个类别之间的 差异以及不同变量各个类别之间的对应关系。对应分析的基本思想是将一个联列表的行和 列中各元素的比例结构以点的形式在较低维的空间中表示出来。
5. 回归分析(regression analysis) o是研究一个随机变量Y对另一个变量X或一组 变量X「X2,…，X\*的相依关系的统计分析方法。回归分析是确定两种或两种以上变数间 相互依赖的定量关系的一种统计分析方法。运用十分广泛，回归分析按照涉及的自变量的 多少，可分为一元回归分析和多元回归分析；按照自变量和因变量之间的关系类型，可分为 线性回归分析和非线性回归分析。
6. 方差分析(Analysis of Variance, A NOV A) o又称变异数分析或F检验,是R. A. Fisher发明的，用于两个及两个以上样本均数差别的显著性检验。由于各种因素的影响，研 究所得的数据呈现出波动。造成波动的原因可分成两类，一类是不可控的随机因素，另一类 是研究中施加的对结果形成影响的可控因素。方差分析是从观测变量的方差入手，研究诸 多控制变量中哪些变量是对观测变量有显著影响的变量。

数据分析常用的图表方法有以下几种：

(1)柏拉图(排列图)。是分析和寻找影响质量的主要因素的一种工具，其形式为双直 角坐标图，左边纵坐标表示频数(如件数、金额等)，右边纵坐标表示频率(如百分比)，折线表 示累积频率，横坐标表示影响质量的各项因素，按影响程度的大小(即出现频数多少)从左向 右排列。通过对排列图的观察分析可发现影响质量的主要因素。

1. 直方图(Histogram).将一个变量的不同等级的相对频数用矩形块标绘的图表(每 一矩形的面积对应于频数)。

直方图又称柱状图、质量分布图，是一种统计报告图，用一系列高度不等的纵向条纹或 线段表示数据分布的情况。一般用横轴表示数据类型，纵轴表示分布情况。

1. 散点图(scatter diagram) o表示因变量随自变量而变化的大致趋势，据此可以选择 合适的函数对数据点进行拟合。用两组数据构成多个坐标点，考察坐标点的分布，判断两变 量之间是否存在某种关联或总结坐标点的分布模式。
2. 鱼骨图(Ishikawa) o是一种发现问题“根本原因”的方法，它也可以称为“因果图”。 其特点是简捷实用，深入直观。它看上去有些像鱼骨，问题或缺陷(即后果)标在“鱼头”外。
3. FMEAO是一种可靠性设计的重要方法。它实际上是FMA(故障模式分析)和 FEA(故障影响分析)的组合。它对各种可能的风险进行评价、分析，以便在现有技术的基 础上消除这些风险或将这些风险减小到可接受的水平。

**8.4.4**数据分析工具

工欲善其事，必先利其器。要进行数据分析，需要学习和掌握各种分析手段和技能，特 别是要掌握分析软件工具。常用的数据分析工具有Matlab.SPSS.SAS.ExceKR等。下面 对Matlab.SPSS.SAS进行简单介绍。

1. **Matlab**

Matlab是matrix和laboratory两个词的组合，意为矩阵工厂(或矩阵实验室)。Matlab 是由美国MathWorks公司发布的主要面向科学计算、可视化以及交互式程序设计的高科 技计算环境。它将数值分析、矩阵计算、科学数据可视化以及非线性动态系统的建模和仿真 等诸多强大功能集成在一个易于使用的视窗环境中，为科学研究、工程设计以及必须进行有 效数值计算的众多科学领域提供了一种全面的解决方案，并在很大程度上摆脱了传统非交 互式程序设计语言(如C.FORTRAN)的编辑模式，代表了当今国际科学计算软件的先进水 平。Matlab主要应用于工程计算、控制设计、信号处理与通信、图像处理、信号检测、金融建 模设计与分析等领域。

Matlab具有以下特点：

1. 编程效率高。用Matlab编写程序犹如在演算纸上列出公式与求解问题，Matlab 语言也可通俗地称为演算纸式的科学算法语言。由于它编写简单，所以编程效率高，易学 易懂。
2. 用户使用方便。Matlab语言把编辑、编译、连接和执行融为一体，其调试程序手段 丰富，调试速度快，需要的学习时间少。它能在同一画面上进行灵活操作，快速排除输入程 序中的书写错误、语法错误以至语义错误,从而加快了用户编写、修改和调试程序的速度，可 以说在编程和调试过程中它是一种比VB还要简单的语言。
3. 扩充能力强。高版本的Matlab语言有丰富的库函数，在进行复杂的数学运算时可 以直接调用，而且Matlab的库函数同用户文件在形成上一样，所以用户文件也可作为 Matlab的库函数来调用。用户可以根据自己的需要方便地建立和扩充新的库函数，以便提 高Matlab使用效率和扩充它的功能。

（4） 语句简单，内涵丰富。Matlab语言中最基本、最重要的成分是函数，一般一个函数 由函数名、输入变量和输出变量组成，同一函数名，不同数目的输入变量（包括无输入变量） 及不同数目的输出变量，代表着不同的含义。这不仅使Matlab的库函数功能更丰富，而大 大减少了需要的磁盘空间，使得Matlab编写的文件简单、短小而高效。

（5） 高效方便的矩阵和数组运算。Matlab语言像BASIC.FORTRAN和C语言一样 规定了矩阵的一系列运算符，它不需定义数组的维数，并给出矩阵函数、特殊矩阵专门的库 函数，使之在求解诸如信号处理、建模、系统识别、控制、优化等领域的问题时显得极为简捷、 高效、方便，这是其他高级语言所不能比拟的。

（6） 方便的绘图功能。Matlab的绘图是十分方便的，它有一系列绘图函数（命令），使 用时只需调用不同的绘图函数（命令），在图上标出图题、XY轴标注，表格绘制也只需调用 相应的命令，简单易行。另外，在调用绘图函数时调整自变量可绘出不变颜色的点、线、复线 或多重线。

1. **SPSS**

SPSS 是 Statistical Package for the Social Science（社会科学统计软件包）的简称，是一 种集成化的计算机数据处理应用软件，是世界上最早的统计分析软件。

SPSS集数据录入、数据编辑、数据管理、统计分析、报表制作以及图形绘制为一体，功 能非常强大，可针对整体的大型统计项目提供完善的解决方案。

SPSS具有以下特点：

（1） 工作界面友好完善、布局合理且操作简便，大部分统计分析过程可以借助鼠标，通 过菜单命令的选择、对话框的参数设置，单击功能按钮来完成,不需要用户记忆大量的操作 命令。菜单分类合理，并且可以灵活编辑菜单以及设置工具栏。

（2） 具有完善的数据转换接口，可以方便地与Windows的其他应用程序进行数据共享 和交换。既可以读取Excel.FoxPro.Lotus等电子表格和数据软件产生的数据文件，也可以 读取ASCII数据文件。

（3） 提供强大的程序编辑能力和二次开发能力，可满足高级用户完成更为复杂的统计 分析任务的需要，具有丰富的内部函数和统计功能。

（4） 具有强大的统计图表绘制和编辑功能，图形美观大方，输岀报告形式灵活，编辑方 便易行。

1. **SAS**

SAS是美国SAS软件研究所研制的一套大型集成应用软件系统，它具有完备的数据存 取、管理、分析和展现功能。其创业产品——统计分析系统部分，以应有尽有、包罗万象和强 大精准的数据分析能力一直为业界推崇，被视为最权威的统计分析标准软件。

经过多年发展，SAS已被全世界多个国家和地区的科研机构和人员普遍采用，涉及教 育、科研、金融等各个领域。

SAS以强大的数据管理和同时处理大批数据文件的功能而受到高级用户的欢迎。也 正是因为这个特点，它是最难掌握的软件之一。使用SAS时，需要编写SAS程序来处理数 据，进行分析。如果在一个程序中岀现一个错误，找到并改正这个错误将是困难的。

SAS具有以下特点：

1. 数据管理。在数据管理方面，SAS是非常强大的，能让用户用任何可能的方式来处 理数据。它包含SQL(结构化査询语言)过程，可以在SAS数据中使用SQL査询。但是要 学习并掌握SAS软件的数据管理需要很长的时间。然而，SAS可以同时处理多个数据文 件，使这项工作变得容易。它可以处理的变量能够达到32 768个，其能够处理的记录条数

♦ 只受限于用户的硬盘空间所允许的最大数量。

1. 统计分析。SAS能够进行大多数统计分析(回归分析.logistic回归、生存分析、方 差分析、因子分析、多变量分析)。SAS的最优之处可能在于它的方差分析、混合模型分析 和多变量分析，而它的劣势主要是有序和多元logistic回归(因为这些命令很难)以及稳健 方法(它难以完成稳健回归和其他稳健方法)。
2. 绘图功能。在所有的统计软件中，SAS有最强大的绘图工具，由SAS/Graph模块 提供。然而,SAS/Graph模块是非常专业而复杂的，图形的制作主要使用程序语言。

**8.5 Hadoop**

1. **1** 简介

Hadoop是由Apache基金会开发的分布式系统基础架构。用户可以在不了解分布式 底层细节的情况下开发分布式程序。Hadoop设计理念之一是扩展单一的服务器为成千上 万台计算机的集群，且集群中每一台计算机同时提供本地计算能力和存储能力，充分利用集 群的威力进行高速运算和存储。

Hadoop实现了一个分布式文件系统(Hadoop Distributed File System, HDFS) o HDFS有高容错性，并且设计用来部署在低廉的硬件上；而且它提供高吞吐量来访问应用 程序的数据，适合那些有超大数据集的应用程序。HDFS放宽了 POSIX的要求，可以以流 的形式访问文件系统中的数据。Hadoop框架最核心的设计是HDFS和MapReduce。 HDFS为海量的数据提供了存储功能，而MapReduce为海量的数据提供了计算功能。

Hadoop目前主要应用于互联网企业，用于数据分析、机器学习、数据挖掘等。

1. **Hadoop** 框架

Hadoop框架是在应用层检测和处理硬件失效问题，而不是依赖于硬件自身来维持高 可用性。在Hadoop框架集群中硬件失效被认为是一种常态，集群的高可用性服务是建立 在整个集群之上的。

Hadoop整体框架如图8. 4所示。其中包括分布式文件系统(Hadoop Distributed File System, HDFS).并行计算模型(MapReduce).列式数据库(Hbase)、数据仓库(Hive)、数据分 析语言(Pig)、数据格式转化工具(Sqoop)、协同工作系统(Zookeeper)擞据序列化系统(Avro) o

Hadoop整体框架的特点如下：

• Hadoop主要应用在多节点集群环境下。

•以数据存储为基础。

•最大限度兼容结构化数据格式。

•以数据处理为目的。

•数据操作技术多样化。



BI报告

T *1*

RDBMS

ETL工具

图8. 4 Hadoop整体框架

*1*

RDBMS

**2. Hadoop**体系结构原理及角色组成

图8. 5是Hadhoop体系结构原理及角色组成。Hadoop使用主/从(Master/Slave)架 构，主要角色由NameNode>DataNode JobTracker^TaskTracker组成。在主节点的服务器 中会执行两套程序：一个是负责安排MapReduce运算层任务的JobTracker,另一个是负责 管理HDFS数据层的NameNode程序。而在Worker节点的服务器中也有两套程序：接受 JobTracker指挥，负责执行运算层任务的是TaskTracker程序；与NameNode对应的则是 DataNode程序，负责执行数据读写操作以及执行NameNode的副本策略。

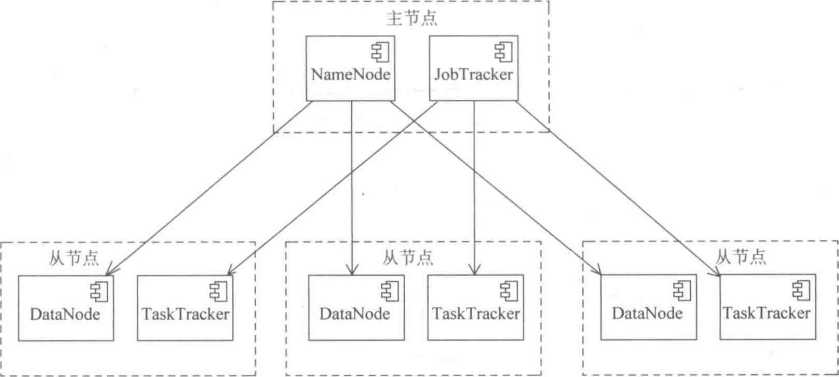


图8.5 Hadoop体系结构原理及角色组成

1. NameNode。是HDFS的守护程序，负责记录文件是如何分割成数据块的，以及这 些数据块被存储到哪些数据节点上。它的功能是对内存及I/O进行集中管理。
2. DataNodeo集群中每个从服务器都运行一个DataNode后台程序，后台程序负责 把HDFS数据块读写到本地文件系统。需要读写数据时，由NameNode告诉客户端去哪个 DataNode进行具体的读写操作。
3. Secondary NameNodeo是一个用来监控HDFS状态的辅助后台程序，如果 NameNode发生问题，可以使用Secondary NameNode作为备用的NameNode。
4. JobTracker0后台程序用来连接应用程序与Hadoop,用户应用提交到集群后，由

JobTracker决定哪个文件处理哪个任务(task)执行，一旦某个任务失败JobTracker会自动 开启这个任务。

1. TaskTrackero与负责存储数据的DataNode相结合，位于从节点，负责各自的 任务。

**3. Hadoop**家族成员

MapReduce是一种编程模型，用于大数据集的并行计算，可以使编程人员在不了解分 布式并行编程的情况下也可以将自己的程序运行在分布式系统上。

HDSF具有高容错性，通过流的方式访问文件系统中的数据。该系统由数百上千个存 储文件的服务器组成。

Hbase是一个分布式的面向列存储的数据库，与BigTable使用相同的数据模型，一个 数据行拥有一个可以选择的键和任意多列，主要用于随机访问和实时读写大数据。

Hive是建立在Hadoop基础上的数据仓库，支持类似传统SQL的查询语言，提供ETL 工具、数据存储管理和大型数据的查询、分析能力。

ZooKeeper是一个针对大型分布式系统的可靠协调系统，提供的功能包括配置维护、名 字服务、分布式同步、组服务等。ZooKeeper的目标就是封装好复杂易出错的关键服务，将 简单易用的接口和性能高效、功能稳定的系统提供给用户。

Common是为Hadoop其他子项目提供支持的常用工具，主要包括FileSystem.RPC和 串行化库。

Avro是用于数据序列化的系统。

Chukwa是开源的数据收集系统，用于显示、监视和分析数据结果。

Pig最大的作用就是对MapReduce算法(框架)实现了一套shell脚本，类似人们熟悉 的SQL语句，在Pig中称之为Pig Latin,在这套脚本中可以对加载的数据进行排序、过滤、 求和、分组(group by)、关联(joining), Pig也可以由用户自定义一些函数对数据集进行 操作。

**8. 5. 2**分布式离线计算框架**MapReduce**

**1.**概念

Map和Reduce是编程语言中的概念，都是处理数据集合的函数。两者的不同主要有 两点：第一，Map在处理数据序列的过程中只处理当前的数据信息，不需要与之前处理的状 态信息交互；而Reduce在处理过程中却依赖之前处理的结果，同时生成的结果也被后续的 处理使用。第二,Map只是遍历数据，数据处理无关先后；Reduce是在遍历的过程中生成 聚合信息。

MapReduce的基本原理为，将一个复杂的问题分成若干个简单的子问题来解决。然 后，对子问题的结果进行合并，得到原有问题的解。具体如下。

• Map：主节点读入输入数据，把它分成可以用相同方法解决的小数据块，然后把这些 小数据块分发到不同的工作节点上，每一个工作节点循环做同样的事，这就形成了 一个树形结构，而每一个叶子节点处理每一个具体的小数据块，再把这些处理结果

返回给父节点。

• Reduce：在处理过程中却依赖之前处理的结果，同时生成的结果也被后续的处理使 用。节点得到所有子节点的处理结果，然后把所有结果组合并返回到输出。

一个MapReduce任务会把一个输入数据集分割为独立的数据块，然后Map任务会以 完全并行的方式处理这些数据块。MapReduce系统自动对Map任务的输出分类，再把这 些分类结果作为Reduce任务的输入。无论是任务的输入还是输岀都会被存储在文件系统 中。MapReduce系统关注任务调度、任务检测和重新执行失败的任务。

MapReduce是一种用于在大型商用硬件集群（成千上万的节点）中对海量数据实施可 靠的、高容错的并行计算的软件系统，是最先由Google公司提出的分布式计算软件构架，具 有易于编程、高容错性、高扩展性的优点。

**2.**模型

用Map和Reduce方法来处理分布式计算问题时，应尽可能地实现数据处理的本地化， 降低由数据移动而产生的代价。每一个Map操作都是相对独立的，所有的Map操作都是 并行运行的，虽然实际中会受到数据源和CPU个数的影响。同样的，用一个Reduce集合 来执行Reduce操作，所有带有相同键的Map输出会聚集到同一个Reduce。MapReduce方 法能够处理一般服务器不能处理的大数据量处理问题。

MapReduce系统由单一的JobTracker主节点和若干个TaskTracker从节点组成，其中 每一个集群节点对应一个TaskTracker节点。主节点负责调度任务的各个组成任务到从节 点上，监控并且重新执行失败的组成任务；从节点执行主节点安排的组成任务。

MapReduce的Map和Reduce过程都定义了键值对（<key,value>）的数据结构，即系 统视任务的输入数据为键值对集合，并且产生键值对结合作为任务的输出。一次任务的输 入输出格式为

（input） <klrvl> - > map - > < k2z v2 > - > combine - Xk2, v2 > -> reduce - >< k3, v3 >（ output）

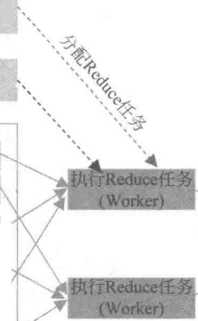
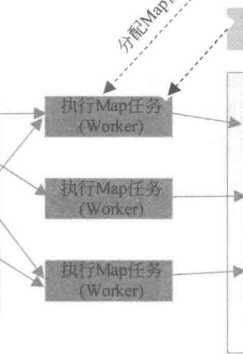
MapReduce通过工作状态的返回有效地处理了单点失效的问题。但是MapReduce是 隶属于大粒度的并行计算模式，并行节点间在Map阶段和Reduce阶段无法通信，也并非是 一种万能的数据处理模型。

MapReduce通过Map（映射）和Reduce（化简）来实现大规模数据（TB级）的并行计算。

如图8.6所示,MapReduce的运作方式就像快递公司一样。物流部门会将发往各地的 包裹先运送到各地的物流分站，再由分站派出快递员进行配送；快递员等每个包裹的用户 签单后将数据反馈给系统汇总，完成整个快递流程。在这里，每个快递员都会负责配送，所 执行的动作大致相同，且只负责少量的包裹，最后由物流公司的系统进行汇总。

在MapReduce运算层上，担任主节点的服务器负责分配运算任务，主节点上的 JobTracker程序会将Map和Reduce程序的执行工作指派给Worker服务器上的 TaskTracker程序，由TaskTracker负责执行Map和Reduce I作，并将运算结果回复给主 节点上的JobTracker。

开发人员先分析需求所提出问题的解决流程，找岀数据可以并发处理的部分（即化简）， 也就是那些能够分解为小段的可并行处理的数据，再将这些能够采用并发处理的需求写成 Map程序（即映射）。



用户

HDFS

本地磁盘

主节点

Map中间靖

Map中间结果2

Map中间结果3

Map中间结果3

Map中间结果1

Map中间结果］

输出结果］

输岀结果2

HDFS文件系统|

图& 6 MapReduce的运作方式

**8.5. 3 Hadoop**分布式文件系统

**1.**概述

Hadoop分布式文件系统（HDFS ）被设计成适合运行在通用硬件（commodity hardware）上的分布式文件系统。它和现有的分布式文件系统有很多共同点。但同时，它和 其他的分布式文件系统的区别也是很明显的。HDFS是一个高度容错性的系统，适合部署 在廉价的机器上。HDFS能提供高吞吐量的数据访问，非常适合大规模数据集上的应用。 HDFS放宽了一部分POSIX约束，以实现流式读取文件系统数据的目的。HDFS在最开始 是作为Apache Nutch搜索引擎项目的基础架构而开发的。HDFS是Apache Hadoop Core 项目的一部分。

HDFS的优点是具有高可靠性、高扩展性、高吞吐率。数据自动保存多个副本，副本丢 失后自动恢复；适合批处理，即移动计算而不会将数据、数据位置暴露给计算框架；适合大 数据处理，即GB级、TB级甚至PB级数据，百万规模以上的文件数量,10000以上节点规 模；流式文件访问，即一次写入，多次读取，保证数据一致性；可构建在廉价机器上，即通过 多副本提高可靠性，提供了容错和恢复机制。

HDFS的缺点有低延迟数据访问：比如毫秒级、低延迟与高吞吐率。小文件存储时，占 用NameNode大量内存，寻道时间超过读取时间；并发写入、文件随机修改时，一个文件只 能有一个写者，仅支持append（追加）。

HDFS的设计假设及其目标包括硬件失效、流式数据访问、大规模数据集、简单的一致 性模型、移动计算比移动数据的代价小、跨异构硬件和软件平台的可移植性。

•硬件失效。硬件失效是常态而不是特例。一个HDFS集群可能包含了成百上千的 服务器，每个都会存储文件系统的部分数据。而大量的组件就会导致组件出错的概 率非常高，而这也意味着HDFS的部分组件会经常不工作。因此，检查缺陷和快速

自动地恢复就成了 HDFS的核心架构目标。

•流式数据访问。运行在HDFS±的应用程序需要流式访问数据集的能力。它们不 是普通的运行在普通文件系统上的程序。HDFS被设计用来应对批量计算的场景， 而不是用来和用户交互。重点是数据访问的高吞吐而不是低延迟。POSIX引入了 大量的硬性需求来约束应用程序，而这些需求不是HDFS的目标需求。POSIX语 义在一些关键领域被认为可以提高数据吞吐率。

•大规模数据集。运行在HDFS±的程序拥有大规模的数据集。一个HDFS文件可 能是GB级别或是TB级别的存储。因此HDFS被调优为存储大文件。它应该提 供髙聚合的数据带宽并且可以在单个集群内扩展到其他的上百上千的节点。程序 应该支持在单实例中存在千万级别的文件。

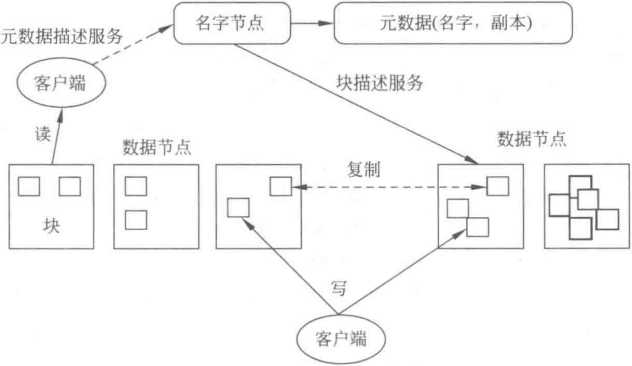
•简单的一致性模型。HDFS程序需要一个一次写入多次读出的文件访问模型。一 旦一个文件被创建、写入数据然后关闭，这个文件应该不再需要被改动。此假设简 化了数据一致性的问题，并且支持了数据的高吞吐。一个MapReduce程序或者一 个网络爬虫程序就非常符合这种模型。

•移动计算比移动数据的代价小。一个程序如果在运行计算任务时能更贴近其依赖 的数据，那么计算会更高效。尤其是在数据集规模很大时该效应更加明显。因为这 会最小化网络消耗而增加系统整体的吞吐能力。这一假设就是：把计算靠近数据 要比把数据靠近计算成本更低。HDFS提供给应用程序接口来做到移动程序使其 离数据更近。

•跨异构硬件和软件平台的可移植性。HDFS被设计为可以很容易地从一个平台移 植到另一个平台。这有利于推广HDFS,使其作为广泛首选的大数据集应用的 平台。

1. **HDFS**架构

图8.7是HDFS架构示意图。HDFS是主从(Master/Slave)体系结构，只含有一个 NameNode主服务节点，这个节点管理文件系统中的命名空间并调度客户端对文件的访问。 通常一台计算机就是一个DataNodeC数据节点),DataNode管理本节点上数据的存储。



在HDFS内部，一个文件被分割为一个或多个数据块，并且这些数据块被存储在一批 DataNode 中。

NameNode执行文件系统中命名空间的操作（打开、关闭、重命名文件和目录）， NamNode需要执行数据块到DataNode映射的决策。

DataNode负责响应来自客户端的文件读写要求，也要负责执行来自NameNode关于 数据块创建、删除和冗余存储的指令。

NameNode和DataNode都可以架设在普通商用计算机上，一个典型的HDFS集群中 部署一个专用机作为NameNode,其余的计算机部署为DataNode。虽然这个体系结构并不 排除把一个机器作为多个DataNode,但是这样的情况在实际部署中很少发生。单 NameNode结构极大地简化了集群的系统结构，NameNode主管并且存储所有的HDFS的 元数据，系统中用户数据绝不会流过NameNodeo

HDFS读文件的过程如图8.8所示。

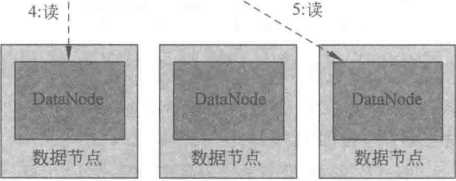
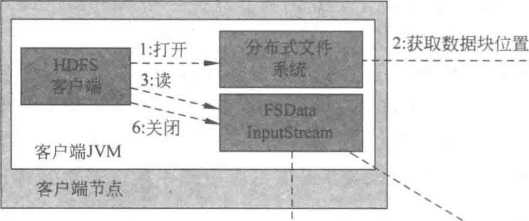


图8. 8 HDFS读文件过程

（1） 客户端用文件系统的openO函数打开文件。

（2） 分布式文件系统用RPC调用元数据节点，得到文件的数据块信息。

（3） 对于每一个数据块，元数据节点返回保存数据块的数据节点的地址。

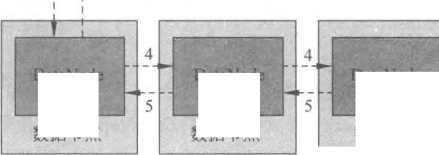
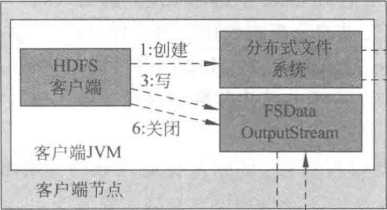
（4） 分布式文件系统返回FSDatalnputStream给客户端，用来读取数据。

（5） 客户端调用stream的readO函数开始读取数据。

DFSInputStream连接保存此文件第一个数据块的最近的数据节点。

（6） 数据从数据节点读到客户端，当此数据块读取完毕时,DFSInputStream关闭和此 数据节点的连接，然后连接此文件下一个数据块的最近的数据节点。当客户端读数据完毕 的时候，调FSDatalnputStream的close。函数。在读取数据的过程中，如果客户端在与数 据节点通信时出现错误，则尝试连接包含此数据块的下一个数据节点。失败的数据节点将 被记录，以后不再连接。

HDFS写文件的过程如图8. 9所示。



7：完成

NameNode

名字节点

DataNode

DataNode

DataNode

数据节点

图8. 9 HDFS写文件过程

数据节点 管道

4：写包| I 5:ACK(确认)包

数据节点

数据节点

1. 客户端调用create()函数创建文件。
2. 分布式文件系统用RPC调用元数据节点，在文件系统的命名空间中创建一个新的 文件。元数据节点首先确定文件原来不存在，并且客户端有创建文件的权限，然后创建新文 件。分布式文件系统返回FSOutputStream,客户端用于写数据。
3. 客户端开始写入数据,DFSOutputStream将数据分成块，写入数据队列。
4. 数据队列由Data Streamer读取，并通知元数据节点分配数据节点，用来存储数据 块(每块默认复制3块)。分配的数据节点放在一个管道中。

Data Streamer将数据块写入管道中的第一个数据节点。第一个数据节点将数据块发 送给第二个数据节点。第二个数据节点将数据发送给第三个数据节点。

1. DFSOutputStream \*发出去的数据块保存了应答队列，等待管道中的数据节点告 知数据已经写入成功。

如果数据节点在写入的过程中失败，则关闭管道，将应答队列中的数据块放入数据队列 的开始。当前的数据块在已经写入的数据节点中被元数据节点赋予新的标示，则错误节点 重启后能够察觉其数据块是过时的，会被删除。

失败的数据节点从管道中移除•另外的数据块则写入管道中的另外两个数据节点。

元数据节点则被通知此数据块复制的块数不足，将来会再创建第三个副本。

1. 当客户端结束写入数据时，则调用stream的close。函数。此操作将所有的数据块 写入管道中的数据节点，并等待应答队列返回成功。
2. 通知元数据节点写入完毕。
3. **4 HBase**大数据库

**1.**概述

HBase是一个分布式的、面向列的开源数据库，该技术来源于Fay Chang所撰写的 Google论文《BigTable： 一个结构化数据的分布式存储系统》。就像BigTable利用了 Google文件系统(Google File System)所提供的分布式数据存储一样,HBase在Hadoop之 上提供了类似于BigTable的能力。HBase是Apache的Hadoop项目的子项目。HBase与 一般的关系数据库有两点不同：它是一个适合于非结构化数据存储的数据库；它基于列的 模式而不是基于行的模式。

HBase可提供随机的、实时的大数据读写访问，目标是在商用硬件上存储非常大的 表——数十亿行，数百万列。

HBase Client 使用 HBase 的 RPC 机制与 HMaster 和 HRegionServer 进行通信。对于 管理类操作»Client与HMaster进行RPC；对于数据读写类操作,Client与HRegionServer 进行RPC。

Zookeeper Quorum 中除 了存储 ROOT 表的地址和 HMaster 的地址 ,HRegionServer 也会把自己以Ephemeral方式注册到Zookeeper中，使得HMaster可以随时感知到各个 HRegionServer的健康状态。此外» Zookeeper也避免了 HMaster的单点问题，见下文 描述。

HMaster没有单点问题，HBase中可以启动多个HMaster,通过Zookeeper的Master Election机制保证总有一个主节点运行,HMaster在功能上主要负责Table和Region的管 理工作：

1. 管理用户对Table的增、删、改、查操作。
2. 管理HRegionServer的负载均衡，调整Region分布。
3. 在Region Split后，负责新Region的分配。
4. 在 HRegionServer 停机后，负责失效 HRegionServer \_t的 Regions 迁移。

HRegionServer主要负责响应用户I/O请求，通过HDFS文件系统读写数据，是 HBase中最核心的模块。

H Store存储是H Base存储的核心，由两部分组成，一部分是MemStore,另一部分是 StoreFileSo MemStore 是 Sorted Memory Buffer,用户写入的数据首先会放入 MemStore, 当MemStore满了以后会刷新成一个StoreFile(底层实现是HFile),当StoreFile文件数量 达到一定阈值时，会触发Compact合并操作，将多个StoreFiles合并成一个StoreFile。

**2. HBase**的数据模型

HBase按预先定义好的列族(column family)结构来存储数据，即每一条数据有一个键 以及若干个列属性值组成，每列的数据都有自己的版本信息。数据是按列有序存储的，不同 于关系型数据库中按行存储。HBase有两种方式的数据操作：通过对有序键值进行扫描查 询，获取键值；或者借助强大的Hadoop进行MapReduce查询。HBase采用了强一致性的 读写保证，数据会在多个不同的域中保存。列族可以包含无限多个数据版本，每个版本可以 有自己的TTL(生命周期)。通过行级锁来保证写操作的原子性，但是不支持多行写操作的 事务性。数据扫描操作不保证一致性。

HBase下表的逻辑视图如图8. 10所示,其中包括行键(Row key)、时间戳(Time stamp)、 列族(Column family)、列(Column)。

在创建一张表时，必须定义行键名及所需列族的列族名，理论上一张表在创建时可以无 限制地定义列族个数，而时间戳会由系统自动生成。列无须在创建时定义，可以在使用时随

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 行键 | 时间戳 | 列族A | |  |
| 列a | ・・・ |  |
| 键 | *tn* |  | ― |  |
| … |  | … |  |
|  | 值1 |  |  |

图8. 10 HBase下表的逻辑视图

意定义并使用，一个列族下同样可以无限制地定义列的个数。HBase中可以任意地定义列 族个数及附属列的个数，只要能够保证其中的任意一列不为空，该行即为有效行。

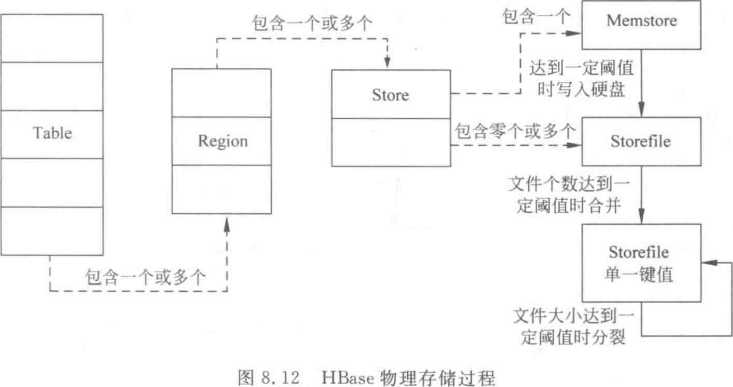
HBase下表的物理视图如图8. 11所示。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 行键 | 时间戳 | 列族 | 列 | 值 |
| 键 | *tn* | 列族A | … | … |
| 键 | 4 | 列族A | 列a | 值1 |

图8. 11 HBase下表的物理视图

在HBase中采用稀疏存储，物理存储过程中细化到一个单元（cell）。在逻辑视图中，任 意一行不为空的每一列都被称为一个单元。单元连同行键、时间戳、列族名、列名作为完整 的一行存储到文件系统中，并且这个存储过程中会自动排序，先对各行键按字母升序排列， 再在同一行键内按时间戳降序排列。

HBase物理存储过程如图8.12所示。



表在创建的初始阶段只含有一个Region,随着表中数据量的不断增大，一个Region会 分裂为两个Region,然后不断重复上述过程，并且Region会被存储到HDFS中不同的 DataNode上。Region包含一个或多个Store,其数量与表中的Region数量增长一致。

Store分为两个部分：第一个部分是Memstore,—个Store中只包含一个Memstore,并 且Memstore存储在内存空间中；第二个部分是Storefile,此部分由Memstore写入硬盘而 得。随着Memstore写入硬盘的次数增多,Storefile的数量也会增加，当文件个数增加到一定量时，系统会自动对Storefile文件进行合并。合并过程中主要完成以下几个工作：具有 相同行键的行存放在一个文件中；扔掉被标志为删除的行；扔掉时间戳过期的行，完成更 新操作。随着合并操作的频繁执行,Storefile会变得很大，达到一定文件大小时自动分裂文 件，以符合HDFS中对一个块数据大小的定义。

HBase的一张表中有多个列族，在物理存储上一个列族对 应一个文件夹，一个文件夹中可包含若干个Hfile文件。Hfile 是Storefile的底层文件格式,Hfile文件由数据块和元数据块 组成，其结构如图8.13所示。

数据块1

数据块N  
元数据块1

其中：

•记录(Record)数据块中是以键值对形式存放的用户数

据，一条记录保存一个键值对，或者说保存一个单元的 数据。

*元数据块N*

文件信息

数据块索引

元数据块索引

文件尾

•元数据块的主要作用是判断一个键值是否在当前Hfile

文件中。

•文件信息(Fileinfo)中保存了与该HFile相关的一些信

息，其中有系统保留的一些固定的值，也可以保存用户

自定义的一些值。

图8.13 Hfile的文件结构

•数据块索引(Data Block Index)保存的是每一个数据块

在HFile文件中的位置、大小信息以及每个块的第一个单元的键值。

•元数据块索引(Meta Block Index)的格式与数据块索引相同，元数据块索引保存的 是每一个元数据块在HFile文件中的位置、大小信息以及每个元数据块的键值。

•文件尾(Fixed file trailer)主要保存了该Hfile的一些基本信息、各个部分的偏移值 和寻址信息。

**8.6**数据可视化

数据可视化技术(IVT)可以理解成一种以图形或表格形式表示信息的方法，可以避免 出现信息描述上模棱两可的现象。可视化技术是由科学可视化、计算机图形学、数据挖掘、 统计学以及带有自定义扩展功能的、能够交互式地处理大规模多维数据集的机器学习等学 科发展而来的。

根据所使用的数据源类型的不同，可以把可视化技术分为两种，即科学可视化和信息可 视化。

科学可视化主要集中于一些物理的数据，例如人体、地球和分子等。科学可视化方法也 处理多维数据，但是，这个领域的数据集主要是对数据的空间特征进行可视化，例如计算机 辅助影像技术和计算机辅助设计。

信息可视化主要集中于抽象的、非物理的数据，例如文本、层次和统计数据等。数据挖 掘技术主要面向信息的可视化。信息可视化技术可以分为传统方法和革新方法两种类型。 传统的方法主要采用2D和3D图像提供信息可视化，这些技术也经常用于科学可视化方面 的物理数据的展现。当信息被可视化以后，就可直观地理解和解释它了，可视化在数据探索

的初始阶段是极为重要的，图形化的数据表示包括图、茎叶图、直方图和散布图等形式。

**8.7** 习题

1. 简述什么是大数据（给出简单的定义描述），并举出一些身边的大数据的例子。
2. 简述大数据的4个特征分别代表的含义。
3. 简单描述Hadoop的核心技术内容。
4. 简述大数据的数据管理方式。
5. 支撑大数据业务的基础是（ ）0
6. 数据科学 B.数据应用
7. 数据硬件 D.数据人才

B.建立数据仓库逻辑模型

1. 为用户和业务部门提供决策支持

）。（多选题）

B.分布式并行计算

D.分布式数据库

1. 数据仓库的最终目的是（ ）o
2. 收集业务需求
3. 开发数据仓库的应用分析
4. 当前大数据技术的基础包括（

A.分布式文件系统

C.关系型数据库

E.非关系型数据库

1. 大数据的价值体现在（ ）。（多选题）
2. 大数据给思维方式带来了冲击
3. 大数据为政策制定提供科学依据
4. 大数据为智慧城市提供科学依据
5. 大数据实现了精准营销
6. 大数据的发力点在于预测
7. 在网络爬虫的爬行策略中，最基础的应用是（ ）。（多选题）

A.深度优先遍历策略

C.高度优先遍历策略

1. 广度优先遍历策略
2. 反向链接策略
3. 大站优先策略
4. 下列关于数据生命周期管理的核心认识中正确的是（ ）。（多选题）
5. 数据从产生到被删除销毁的过程中具有多个不同的数据存在阶段
6. 在不同的数据存在阶段,数据的价值是不同的
7. 根据数据价值的不同，应该对数据采取不同的管理策略
8. 数据生命周期管理旨在产生效益的同时降低生产成本
9. 数据生命周期管理最终关注的是社会效益
10. 简述数据可视化的常用技术与方法。

深度学习

深度学习框架，尤其是基于人工神经网络的框架可以追溯到1980年福岛邦彦提出的新 认知机，而人工神经网络的历史更为久远。“深度学习”这一概念从2007年前后开始受到关 注，自深度学习出现以来，它已在很多领域，尤其是计算机视觉和语音识别中，成为各种领先 系统的一部分。在通用的用于检验的数据集，例如语音识别中的TIMIT和图像识别中的 ImageNet.CifarlO上的实验证明，深度学习能够提高识别的精度。

人工智能就像长生不老和星际漫游一样，是人类最美好的梦想之一。虽然计算机技术 已经取得了长足的进步，但是到目前为止，还没有一台电脑能产生''自我”的意识。事实的确 如此，在人类和大量现成数据的帮助下，计算机可以表现得十分强大，但是离开了这两者，它 甚至都不能分辨岀一棵树和一条狗。图灵曾经在1950年的一篇论文中提出“图灵实验”的 设想，也就是“隔墙对话气 作为谈话者之一的人类不知道与自己对话的是人还是计算机，这 无疑给计算机，尤其是人工智能，预设了一个很高的期望值。但是半个世纪过去了，人工智 能的进展远远没有达到图灵试验的标准。这不仅让多年翘首以待的人们心灰意冷，认为人 工智能是忽悠，相关研究领域是“伪科学”。

自2006年以来，机器学习领域取得了突破性的进展。图灵试验至少不再是那么可望 而不可即了。至于技术手段，不仅依赖于云计算对大数据的并行处理能力，而且依赖于算 法。这个算法就是深度学习(deep learning) o借助于深度学习算法，人类终于找到了如何 处理“抽象概念”这个亘古难题的方法。

2012年6月，《纽约时报》披露了 Google Brain项目，吸引了公众的广泛关注。这个项 目是由著名的斯坦福大学的机器学习权威Andrew Ng教授和在大规模计算机系统方面的 世界顶尖专家JeffDean共同主导，用16 000个CPU Core的并行计算平台训练一种称为 “深度神经网络”(Deep Neural Networks,DNN)的机器学习模型(内部共有10亿个节点。 这一网络自然是不能跟人类的神经网络相提并论的。人脑中可是有150多亿个神经元，互 相连接的节点，也就是突触，更是如恒河沙数。曾经有人估算过，如果将一个人的大脑中所 有神经细胞的轴突和树突依次连接起来，并拉成一根直线，可从地球连到月亮，再从月亮返

回地球)，在语音识别和图像识别等领域获得了巨大的成功。

2012年11月，微软公司在中国天津的一次活动上公开演示了一个全自动的同声传译 系统，讲演者用英文演讲，后台的计算机自动完成语音识别、英中机器翻译和中文语音合成， 效果非常流畅。据报道，后面支撑的关键技术也是深度学习。

2013年1月，在百度公司年会上，百度创始人兼CEC)李彦宏高调宣布要成立百度研究 院，其中第一个成立的就是“深度学习研究所”。

2013年3月谷歌公司收购了加拿大神经网络方面的创业公司DNNresearch, DNNresearch公司是由多伦多大学教授Geoffrey Hinton与他的两个研究生 Alex Krizhevsky和Ilya Sutskever于2012年成立的,由于谷歌公司在本次收购中没有获得任何 实际的产品或服务，所以本次收购实质上属于人才性收购，收购的主体实为这个三人团队。

为什么拥有大数据的互联网公司争相投入大量资源研发深度学习技术？什么是深度学 习？为什么有深度学习？它是怎么来的？又能干什么呢？目前存在哪些困难呢？为了回答 这些问题,首先来了解一下机器学习(人工智能的核心)的背景。

**9.1**深度学习应用背景与概述

**9.1.1**应用背景

机器学习(machine learning)是一门专门研究计算机怎样模拟或实现人类的学习行为， 以获取新的知识或技能，重新组织已有的知识结构，使之不断改善自身的性能的学科。1959 年，美国的塞缪尔(Samuel)设计了一个下棋程序，这个程序具有学习能力，它可以在不断的 对弈中改善自己的棋艺。4年后，这个程序战胜了设计者本人。又过了 3年，这个程序战胜 了美国一个保持8年不败的冠军。这个程序向人们展示了机器学习的能力，提出了许多令 人深思的社会问题与哲学问题。

**9.1.2**概述

深度学习是多学科领域的交叉，包括神经网络、人工智能、图建模、最优化理论、模式识 别和信号处理。需要注意的是，深度学习是在信号和信息处理内容中通过学习获得一种深 度结构。它不是对信号和信息处理知识的理解，尽管从某些意义上说有些相似，但深度学习 重点在于通过学习获得一种深度网络结构，是实实在在地存在的一种计算机可存储结构，这 种结构表示了信号的某种意义上的内涵。

在图像识别、语音识别、天气预测、基因表达等方面，目前通过机器学习来解决这些问题 的思路如图9.1所示。

低级传感预处理 匚＞ 特征提取匚＞ 特征选择匸＞ 推理识费测'

\2°卜人工智能 :

开始时通过传感器来获得数据，然后经过预处理、特征提取、特征选择，再到推理、预测 或者识别。最后一个部分也就是机器学习的部分,绝大部分的工作代价要投入到这个部分。 中间的3部分概括起来就是特征表达。良好的特征表达对最终算法的准确性起了非常关键 的作用，而且系统主要的计算和测试工作都耗在这一大部分，但这些工作实际上一般都是人 ♦ 工完成的。

手工地选取特征是一件启发式的方法，非常费时，而且它的调节需要大量的时间。既然 手工选取特征不太好，那么能不能自动地学习一些特征呢？答案是能！深度学习就是用来 干这个事情的，它的一个别名为Unsupervised Feature Learning （非监督特征学习），而 Unsupervised在此处就是强调不要人参与特征的选取过程。

**9.1.3**人脑视觉机理

1981年的诺贝尔医学奖颁发给了 David HubekTorstenWiesel和Roger Sperryo前两 位的主要贡献是发现了视觉系统的信息处理机制，发现可视皮层是分级的，如图9.2所示。

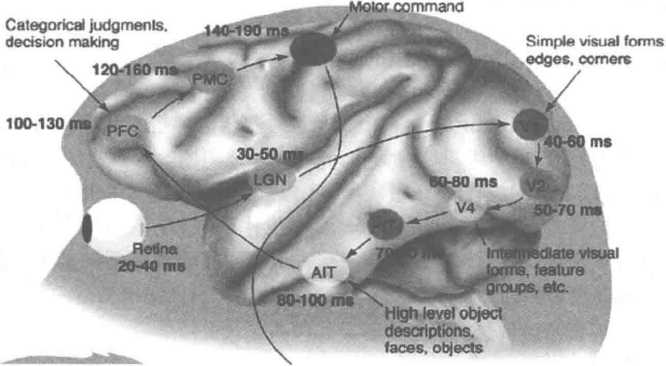
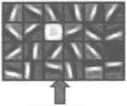
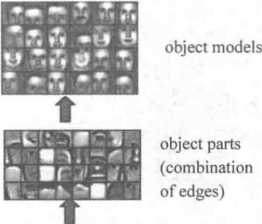


图9.2人脑视觉机理

1958年,David Hubei等研究了瞳孔区域与大脑皮层神经元的对应关系。他们在猫的 后脑头骨上开了一个3mm的小洞，向洞里插入电极，测量神经元的活跃程度。他们在小猫 的眼前展现各种形状、各种亮度的物体，并且在展现每一件物体时还改变物体放置的位置和 角度。他们期望通过这个办法让小猫瞳孔感受不同类型、不同强弱的刺激，目的是证明一个 猜测，那就是位于后脑皮层的不同视觉神经元与瞳孔所受刺激之间存在某种对应关系。一 旦瞳孔受到某一种刺激，后脑皮层的某一部分神经元就会活跃。经历了很多天反复的枯燥 的试验，同时牺牲了若干只可怜的小猫,David Hubei发现了一种被称为“方向选择性细胞” 的神经元细胞。当瞳孔发现了眼前的物体的边缘，而且这个边缘指向某个方向时，这种神经 元细胞就会活跃。这个发现激发了人们对于神经系统的进一步思考。神经-中枢-大脑的工 作过程或许是一个不断迭代、不断抽象的过程。

这里的关键词有两个，一个是抽象，另一个是迭代。从原始信号开始做低级抽象，逐渐



edges

向高级抽象迭代。人类的逻辑思维经常使用高度抽象的概念。例如，从原始信号摄入开始 （瞳孔摄入像素），接着做初步处理（大脑皮层某些细胞发现边缘和方向），然后抽象（大脑判 定眼前的物体是圆形的），然后进一步抽象（大脑进一步判定该物体是一只气球）。再比如人 脸识别，如图9.3所示。

Area V4 •♦♦♦ "十•切

Area VI

Retma ，‘ «

图9. 3人脸识别

这个生理学的发现促成了计算机人工智能在40年后的突破性发展。总的来说，人的视 觉系统的信息处理是分级的：从低级的VI区的边缘特征，到V2区的形状或者目标的部分 等,再到更高层的整个目标和目标的行为等。也就是说高层的特征是低层特征的组合，从低 层到高层的特征表示越来越抽象，越来越能表现语义或者意图。而抽象层面越高，存在的可 能猜测就越少，就越利于分类。例如，单词集合和句子的对应是多对一的，句子和语义的对 应又是多对一的，语义和意图的对应还是多对一的，这是一个层级体系。

到现在为止出现了一个关键词，那就是“分层”，而深度学习的“深度”是不是就表示存在 多少层，也就是多深呢？答案是肯定的。那深度学习又如何借鉴这个过程？终究还是由计 算机来处理，此时面对的一个问题就是怎么对这个过程进行建模。因为要学习的是特征的 表达，那么关于特征，或者说关于这个层级特征，我们就需要了解得更深入。因此，在具体介 绍深度学习之前，有必要解释一下特征。

*9.2*特征的概念

特征是机器学习系统的原材料，对最终模型的影响是毋庸置疑的。如果数据被很好地 表达成特征，通常线性模型就能达到满意的精度。那么对于特征，我们需要考虑什么呢？

**9.2.1**特征表示的粒度

学习算法在什么粒度上的特征表示才能发挥作用？就一个图片来说，像素级的特征根 本没有价值。例如，对一张摩托车的图片从像素级进行分析，根本得不到任何信息，即无法对摩托车和非摩托车进行区分。而如果特征具有结构性，比如是否具有车把手，是否具有车 轮，就很容易把摩托车和非摩托车区分开，这样学习算法才能发挥作用。

**9. 2. 2**初级（浅层）特征表示

既然像素级的特征表示方法没有作用，那么怎样的表示才有用呢？ 1995年前后,Bruno Olshausen和David Field两位学者任职于康奈尔大学，他们试图同时采用生理学和计算机 的手段，双管齐下地研究视觉问题。他们收集了很多黑白风景照片，从这些照片中提取岀 400个小碎片，每个小碎片的尺寸均为16X16（像素），不妨把这400个碎片标记为S［门， ，=0,1,・・・，399。接下来，再从这些黑白风景照片中随机提取另一个碎片，尺寸也是16X16 （像素），不妨把这个碎片标记为*T。*

他们提出的问题是，如何从这400个碎片中选取一组碎片S［＞］,通过叠加的办法合成 岀一个新的碎片，而这个新的碎片应当与随机选择的目标碎片T尽可能相似，同时，S［妇的 数量尽可能少。用数学的语言来描述就是Sum\_k（虹妇XS［妇）—丁,其中。［妇是在叠加碎 片S［妇时的权重系数。为解决这个问题，他们发明了一个算法——稀疏编码（sparse coding） o

稀疏编码是一个重复迭代的过程，每次迭代分两步：

（1） 选择一组S［妇，然后调整。［妇，使得Sum\_kS［妇XS［妇）最接近*To*

（2） 固定住妇，在400个碎片中选择其他更合适的碎片宁［妇替代原先的S［妇,使得 Sum\_kS［妇XS'［妇）最接近To

经过几次迭代后，最佳的S［妇组合被遴选出来。令人惊奇的是，被选中的S［妇基本上 都是照片上不同物体的边缘线，这些线段形状相似，区别在于方向。Bruno Olshausen和 David Field的算法结果与David Hubei和Torsten Wiesel的生理发现不谋而合！也就是 说，复杂图形往往由一些基本结构组成。

许多专家还发现，不仅图像存在这个规律，声音也是一样。他们从未标注的声音中发现 了 20种基本的声音结构，其余的声音可以由这20种基本结构合成，如图9. 4所示。

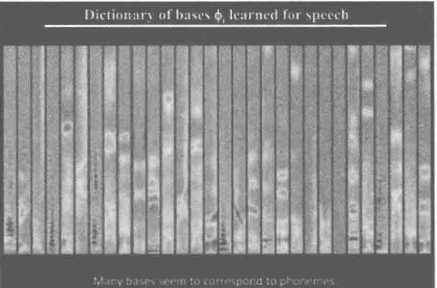


图9. 4基本声音结构及合成

图9.5为语音稀疏编码。

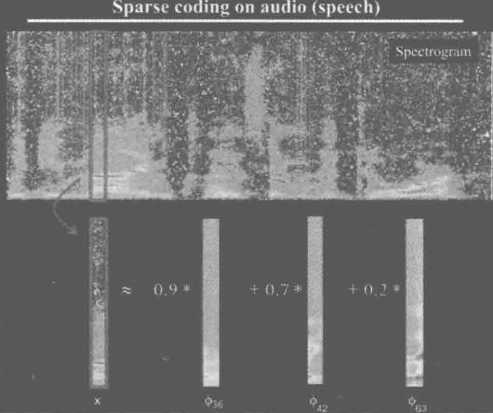


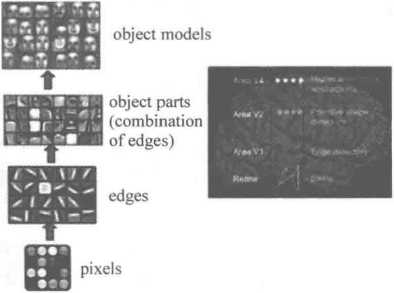
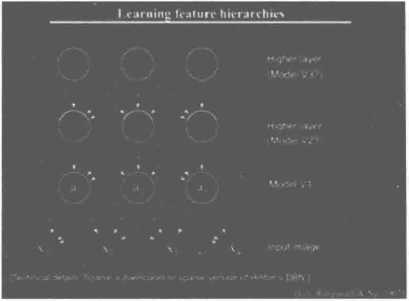
图9.5语音稀疏编码

**9.2.3**结构性特征表示

小块的图形可以由基本边缘构成，更结构化、更复杂的、具有概念性的图形如何表示呢? 这就需要更高层次的特征表示，比如V2、V4。因此VI看像素级是像素级，V2看VI是像 素级……这是层次递进的，高层表达由低层表达组合而成，用专业术语说就是基(basis)。 VI提出的基是边缘，V2是VI的这些基的组合，这时V2得到的又是V3的基，每一层都是 上一层的基组合的结果。

直观上说，就是找到有意义的小碎片(patch),再将其进行组合(combine),就得到了上 一层的特征(feature)，递归地向上学习特征(learning feature)。

在不同对象(object)±做训练时，所得的边缘基(edge basis)是非常相似的，但对象部分 (object part)和模式(model)完全不同，这样便于图像的准确识别，如图9. 6所示。



从文本来说，一个doc文档表示什么意思？我们描述一件事情，用什么来表示比较合 适？文档中的“字”就相当于像素级别，起码应该是词语（term）,换句话说，每个文档都是由 词语组成的。此时需要再向上一步达到话题（topic）级，之后才到达文档级。一个人在看一 个文档的时候，眼睛看到的是句子，由这些句子在大脑里自动切词形成词语，再按照概念组 织的方式先验地学习得到话题，然后再进行高层次的学习。

**9.2.4**特征数量

层次的特征构建需要由浅入深，但每一层该有多少个特征呢？无论哪一种方法，特 征越多，给出的参考信息就越多，准确性会得到提升。但特征多意味着计算复杂，探索的 空间大，可以用来训练的数据在每个特征上就会稀疏，都会带来各种问题，并不一定特征 越多越好。

**9.3** 深度学习基本思想

假设有一个系统S,它有*n*层（S1，S2，・・・，S”），它的输入是I,输出是O,形象地表示为 r-S|fS2f ・・・fS”fO,如果输出o等于输入I,即输入I经过这个系统变化之后没有任何 的信息损失。这意味着输入I经过每一层S,都没有任何信息损失，即在任何一层S，它都 是原有信息（即输入/）的另外一种表示。现在回到主题深度学习，我们需要自动地学习特 征，假设有一堆输入/（如一堆图像或者文本），并设计了一个系统S（有〃层），通过调整系统 中的参数，使得它的输出仍然是输入那么就可以自动地获取输入I的一系列层次特征， 即 S）*,S2*，…，S”。

对于深度学习来说，其思想就是堆叠多个层，也就是说这一层的输出作为下一层的输 入。通过这种方式就可以实现对输入信息进行分级表达了。另外，前面假设输岀严格地等 于输入，这个限制太严格，可以略微放宽这个限制，例如，只要使得输入与输出的差别尽可能 地小即可，放宽限制会导致另外一类不同的深度学习方法。上述就是深度学习的基本思想。

**9.4**浅层学习和深度学习

**9.4.1**浅层学习

浅层学习（Shallow Learning）是机器学习的第一次浪潮。20世纪90年代，各种各样的 浅层机器学习模型相继被提出，例如支撑向量机（Support Vector Machines, SVM ）、 Boosting.最大嫡方法（如LR（Logistic Regression,逻辑回归））等。这些模型的结构基本上 可以看成带有一层隐层节点（如SVM.Boosting）,或者没有隐层节点（如LR）O这些模型无 论是在理论分析还是应用中都获得了巨大的成功。

相比之下，由于理论分析的难度大，训练方法又需要很多经验和技巧，这个时期深度人 工神经网络反而相对沉寂。2006年，加拿大多伦多大学教授、机器学习领域的泰斗 Geoffrey Hinton和他的学生Ruslan Salakhutdinov在国际顶级期刊《科学》上发表了一篇

第9章深度学习 文章，开启了深度学习在学术界和工业界的浪潮。这篇文章有两个主要观点：

（1） 多隐层的人工神经网络具有优异的特征学习能力，学习得到的特征对数据有更本 质的刻画，从而有利于可视化或分类。

（2） 深度神经网络在训练上的难度可以通过逐层初始化（layer-wise pre-training）来有 效克服，在这篇文章中，逐层初始化是通过无监督学习实现的。

**9.4.2**深度学习

当前多数分类、回归等学习方法是浅层结构算法，其局限性是在有限样本和计算单元情 况下对复杂函数的表示能力有限，针对复杂分类问题，其泛化能力受到一定制约。深度学习 可通过学习一种深层非线性网络结构实现复杂函数逼近，表征输入数据分布式表示，并展现 了强大的从少数样本中集中学习数据集本质特征的能力。而多层的好处就是可以用较少的 参数表示复杂的函数。

深度学习的实质是通过构建具有很多隐层的机器学习模型和海量的训练数据来学习更 有用的特征，从而最终提升分类或预测的准确性。因此，“深度模型”是手段，“特征学习”是 目的。而区别于传统的浅层学习，深度学习的不同在于：

（1） 强调了模型结构的深度，通常有5层、6层，甚至十多层的隐层节点。

（2） 明确突出了特征学习的重要性，也就是说，通过逐层特征变换，将样本在原空间的 特征表示变换到一个新特征空间，从而使分类或预测更加容易。与人工规则构造特征的方 法相比，利用大数据来学习特征，更能够刻画数据丰富的内在信息。

深度学习是机器学习研究中的一个新领域，其动机在于建立、模拟人脑进行分析学习的 神经网络，它模仿人脑的机制来解释数据，例如图像、声音和文本。深度学习是无监督学习 的一种。深度学习的概念源于人工神经网络的研究。含多隐层的多层感知器就是一种深度 学习结构。深度学习通过组合低层特征形成更加抽象的高层表示属性类别或特征，以发现 数据的分布式特征表示。

**9.5**深度学习常用模型和方法

如果对所有层同时进行训练，时间复杂度会太高；如果每次训练一层，偏差就会逐层传 递。而深度网络的神经元和参数过多，就会面临过拟合的问题。2006年,Hinton提出了在 非监督数据上建立多层神经网络的一个有效方法，简单地说，分为两步，一是每次训练一层 网络，二是调优。深度学习训练过程具体如下：

（1）使用自底向上的监督学习。采用无标定数据（或有标定数据）分层训练各层参数， 这一步可以看做是一个无监督训练过程，是和传统神经网络区别最大的部分。具体过程是， 先用无标定数据训练第一层，训练时先学习第一层的参数（这一层可以看作是得到一个使得 输出和输入差别最小的三层神经网络的隐层），由于模型容量的限制以及稀疏性约束，使得 得到的模型能够学习到数据本身的结构，从而得到比输入更具有表示能力的特征；在学习 得到第n-1层后，将第7?-1层的输出作为第n层的输入，训练第n层，由此分别得到各层 的参数。

（2）使用自顶向下的监督学习。即通过带标签的数据去训练，误差自顶向下传输，对网 络进行微调，基于第一步得到的各层参数进一步微调整个多层模型的参数，这一步是一个有 监督训练过程。第一步类似神经网络的随机初始化初值过程，由于深度学习的第一步不是 随机初始化，而是通过学习输入数据的结构得到的，因而这个初值更接近全局最优，从而能 够取得更好的效果。所以深度学习效果好很大程度上归功于第一步的特征学习过程。

**9.5.1**自动编码器

深度学习最简单的一种方法是利用人工神经网络的特点，人工神经网络本身就是具有 层次结构的系统，如果给定一个神经网络,假设其输岀与输入是相同的，然后通过训练调整 其参数，得到每一层中的权重。自然地，就得到了输入*I*的几种不同表示（每一层代表一种 表示），这些表示就是特征。自动编码器就是一种尽可能复现输入信号的神经网络。为了实 现这种复现，自动编码器（AutoEncoder）就必须捕捉可以代表输入数据的最重要的因素，就 像PCA那样，找到可以代表原信息的主要成分。

具体过程如下：

（1） 给定无标签数据，用非监督学习学习特征。

在一些神经网络中，输入的样本是有标签的，这样根据当前的输出和目标之间的差去改 变前面各层的参数，直到收敛。但如果只有无标签数据，那么这个误差怎样得到？将数据输 入编码器，就会得到一个编码，这个编码也就是输入的一个表示，那么我们怎么知道这个编 码表示的就是输入的数据呢？加一个解码器，这时候解码器就会输出一个信息。如果输岀 的信息和一开始的输入信号是很像的（理想情况下就是一样的），那么就有理由相信这个编 码是正确的。所以，我们就通过调整编码器和解码器的参数，使得重构误差最小，这时候就 得到了输入信号的第一个表示，也就是编码。因为是无标签数据，所以误差就是直接重构后 通过与原输入相比得到的。

（2） 通过编码器产生特征，然后训练下一层，这样逐层训练。

得到第一层的编码，重构误差最小让我们相信这个编码就是原输入信号的良好表达，或 者粗略地说,它和原信号是一模一样的。那第二层和第一层的训练方式就没有差别了，将第 一层输出的编码当成第二层的输入信号，同样最小化重构误差，就会得到第二层的参数，并 且得到第二层输入的编码，也就是原输入信息的第二个表达。其他层用同样的方法训练就 可以了。

（3） 有监督微调。

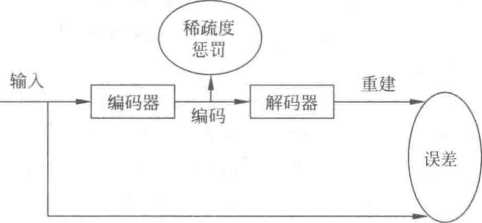
经过上面的方法，就可以得到很多层。至于需要多少层，要在试验中进一步确定。每一 层都会得到与原始输入不同的表达。当然，我们认为它越抽象越好，就像人的视觉系统一 样。到这里，这个自动编码器还不能用来分类数据，因为它还没有学习如何去连接一个输入 和一个类。它只是学会了如何去重构或者复现它的输入而已。或者说，它只是学习获得了 一个可以良好地代表输入的特征，这个特征可以在最大程度上代表原输入信号。那么，为了 实现分类，我们就可以在自动编码器最顶部的编码层添加一个分类器（例如罗杰斯特回归、 SVM等），然后通过标准的多层神经网络的监督训练方法（梯度下降法）去训练。

另一种方法是通过有标签样本微调整个系统。如果有足够多的数据，这种方法是最好

的。在研究中发现，如果在原有的特征中加入这些自动学习得到的特征，可以大大提高精确 度，甚至在分类问题上比目前最好的分类算法效果还要好。

**9.5.2**稀疏编码

继续加上一些约束条件得到新的深度学习方法，例如，如果在自动编码器的基础上加上 L1的限制(L1主要是约束每一层中的节点中大部分为0,只有少数不为0,这就是其名字中 Sparse的来源)，我们就可以得到稀疏自动编码器(Sparse AutoEncoder)法，如图9. 7所示。 人脑也是这样的，某个输入只是刺激某些神经元，其他的大部分神经元是受到抑制的。



输入：X.编码：*h=^X*

损失："X；呼)=11附力-\*|2+人£四

图9.7稀疏自动编码器法

降噪自动编码器(Denoise AutoEncoder,DA)是在自动编码器的基础上为训练数据加 入噪声，所以自动编码器必须学习去除这种噪声而获得真正的没有被噪声污染过的输入。 因此，这就迫使编码器学习输入信号的更加鲁棒的表达。DA可以通过梯度下降算法进行 训练，如图9. 8所示。

隐藏编码

(表达) KL(重建|原输入)

(僉。僉 (OOOO& (Foopb)

被污染的输入 原输入 重建

图9. 8降噪自动编码器

**9.5.3**深度信念网络

在最高两层，权值被连接到一起，这样更低层的输出将会提供一个参考的线索或者关联 到顶层，这样顶层就会将其联系到它的记忆内容。而我们最关心的，最后想得到的就是判别 性能。

在预训练后，深度信念网络(Deep Belief Networks, DBN)可以利用带标签数据用BP 算法对判别性能做调整。在这里，一个标签集将被附加到顶层(推广联想记忆)，通过一个自 下向上地学习到的识别权值获得一个网络的分类面。这个性能会比单纯的BP算法训练的 网络好。这可以很直观地解释，DBN的BP算法只需要对权值参数空间进行局部的搜索， 这相比前向神经网络训练得快，而且收敛的时间也少。

DBN的灵活性使得它的拓展比较容易。一个拓展就是卷积DBN(Convolutional DBN)O DBN并没有考虑到图像的二维结构信息，因为输入是简单地从一个图像矩阵一维 向量化的。而CDBN就是考虑到了这个问题，它利用邻域像素的空域关系，通过一个称为 卷积RBM(Restricted Boltzmann Machine,限制波尔曼机)的模型区达到生成模型的变换不 变性，而且可以容易得变换到高维图像。DBN并没有明确地处理对观察变量的时间联系的 学习上，虽然目前已经有这方面的研究，例如堆叠时间RBM,由此推广到有序列学习的刺激 额叶卷积机(dubbed temporal convolution machines),这种序列学习的应用给语音信号处 理问题带来了一个让人激动的未来研究方向。

目前，和DBN有关的研究包括堆叠自动编码器，它是通过用堆叠自动编码器来替换传 统DBN中的RBM,这就使得通过同样的规则来训练可以产生深度多层神经网络架构，但它 缺少层的参数化的严格要求。与DBN不同，自动编码器使用判别模型，这样这个结构就很 难采样输入采样空间，这就使得网络更难捕捉它的内部表达。但是，降噪自动编码器却能很 好地避免这个问题,并且比传统的DBN更优。它通过在训练过程中添加随机的污染并堆叠来 产生场泛化性能。训练单一的降噪自动编码器的过程和RBM训练生成模型的过程一样。

**9.5.4**卷积神经网络

卷积神经网络(Convolutional Neural Network)是一种有监督的深度学习模型，已成为 当前语音分析和图像识别领域的研究热点。它的权值共享网络结构使之更类似于生物神经 网络，降低了网络模型的复杂度，减少了权值的数量。该优点在网络的输入是多维图像时表现 得更为明显，使图像可以直接作为网络的输入，避免了传统识别算法中复杂的特征提取和数据 重建过程。卷积网络是为识别二维形状而特殊设计的一个多层感知器，这种网络结构对平移、 比例缩放、倾斜或者其他形式的变形具有高度不变性。卷积神经网络的结构如图9. 9所示。

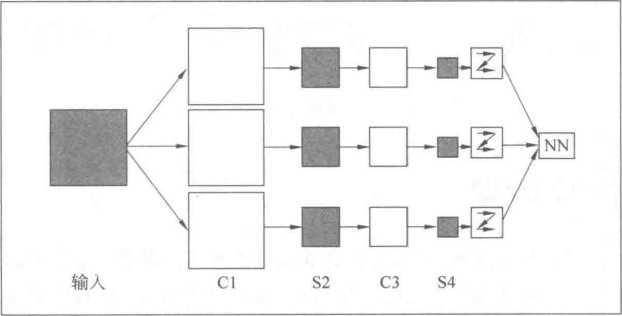
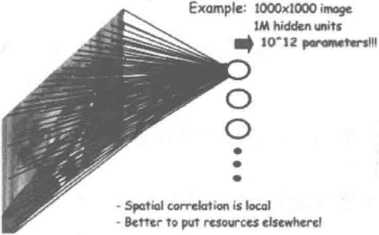
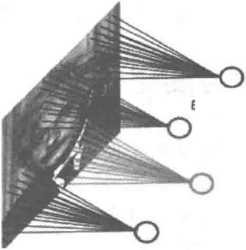


图9. 9卷积神经网络的结构

卷积神经网络是一个多层的神经网络，每层由多个二维平面组成，而每个平面由多个独 立神经元组成。输入图像通过和3个可训练的滤波器和可加偏置进行卷积，卷积后在C1 层产生3个特征映射图，然后对特征映射图中每组的4个像素再进行求和，加权值，加偏置, 得到3个S2层的特征映射图。这些映射图再通过滤波得到C3层。这个层级结构再和S2 一样产生S4。最终，这些像素值被光栅化,并连接成一个向量输入到传统的神经网络，得到 输出。

C层是特征提取层，每个神经元的输入与前一层的局部感受野(local receptive fields) 相连，并提取该局部的特征。S层是特征映射层，网络的每个计算层由多个特征映射组成, 每个特征映射为一个平面，平面上所有神经元的权值相等。卷积神经网络中的每一个特征 提取层(C层)都紧跟着一个用来求局部平均与二次提取的计算层(S层)，这种特有的两次 特征提取结构使网络在识别时对输入样本有较高的畸变容忍能力。

CNN的一个优势在于通过感受野和权值共享减少了神经网络需要训练的参数的个数。 例如，假设有一个1000X 1000(像素)的图像，有100万个隐层神经元，如果将它们全连接 (每个隐层神经元都连接图像的每一个像素点)，就有1000X1000X1 000 000= 1012个连接, 也就是10技个权值参数。图像的空间联系是局部的，就像人是通过一个局部的感受野去感 受外界图像一样，每一个神经元都不需要对全局图像进行感受，每个神经元只感受局部的图 像区域，然后在更高层，将这些感受不同局部的神经元综合起来就可以得到全局的信息。这 样就可以减少连接的数目，也就是减少神经网络需要训练的权值参数的个数。如图9. 10(b) 所示，假如局部感受野是10X10,隐层每个感受野只需要和这10X10的局部图像相连接, 所以100万个隐层神经元就只有一亿个连接，即1。8个参数，是原来的1/10 000,这样训练起 来就没那么费力了。但即便如此，数量还是相当巨大的，是否还有其他方法呢？

**LOCAUY *CONNECTED* NCURAL NET**

(b)局部连接神经网络

**FULLY CONNECTED NEURAL NET**

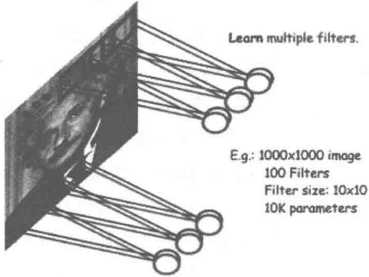
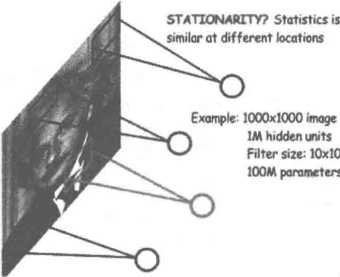
(a)全连接神经网络

Example； 1000x1000 IM hidden units Filter size： 10x10 100M parameters

图9.10全连接和局部连接

隐含层的每一个神经元都连接10X10的图像区域，也就是说每一个神经元存在10X 10=100个连接权值参数。如果每个神经元的这100个参数是相同的，也就是说每个神经 元用的是同一个卷积核去卷积图像，就只有100个参数。不管隐层的神经元个数有多少，两 层间的连接只需100个参数，这就是权值共享。例如，有一种滤波器，也就是一种卷积核，即 提出图像的一种特征，例如某个方向的边缘。此时如果要提取不同的特征，假设有100种滤 波器，每种滤波器的参数不一样，表示它提'取输入图像的不同特征，例如不同的边缘。这样每种滤波器对图像进行卷积就得到对图像的不同特征的反映，称为特征映射(Feature Map)o所以100种卷积核就有100个特征映射。

这100个特征映射就组成了一层神经元。每种卷积核共享100个参数，100种卷积核 也就是1万个参数。图9. 11(b)是不同的颜色表达不同的滤波器。



**LOCALLY CONNECTED NEURAL NET**

(a)局部连接神经网络

**CONVOLUTIONAL NET**

(b)卷积网络

图9. 11局部连接神经网络和卷积网络

图像可以直接作为网络的输入，避免了传统识别算法中复杂的特征提取和数据重建过 程。权值共享网络结构更类似于生物神经网络，降低了网络模型的复杂度，减少了权值的数 量，具有良好的容错能力、并行处理能力和自学能力。卷积网络可以处理环境复杂，对规则 不明确的问题进行推理。卷积网络较一般神经网络在图像处理方面有如下优点：

1. 输入图像和网络的拓扑结构能很好的吻合。
2. 特征提取和模式分类同时进行，并同时在训练中产生。
3. 权重共享可以减少网络的训练参数，使神经网络结构变得更简单,适应性更强。

**9.6**深度学习展望

深度学习目前仍有大量工作需要研究。目前的关注点还是从机器学习的领域借鉴一些 可以在深度学习中使用的方法，特别是降维领域。

1. 稀疏编码。通过压缩感知理论对高维数据进行降维，使得用非常少的元素的向量 就可以精确地代表原来的高维信号。
2. 半监督流行学习。通过测量训练样本的相似性，将高维数据的这种相似性投影到 低维空间。
3. 进化编程方法(Evolutionary Programming Approach) o它可以通过最小化工程能 量去进行概念性自适应学习和改变核心架构。

虽然深度学习已经被应用到尖端科学研究及日常生活当中，例如Google公司已经将这 种技术实际搭载在核心搜索功能之中，但其他知名的人工智能实验室对于深度学习技术的 反应并不一致。例如，艾伦人工智慧中心的执行官Oren Etzioni就没有考虑将深度学习纳 入正在开发的人工智慧系统中。该机构目前的研究是以小学程度的科学知识为目标，希望 能开发岀只须看学校教科书就能够轻松应付考试的智能程序。Oren Etzioni以飞机为例， 他表示，最成功的飞机设计都不是来自模仿鸟的结构，所以瞒神经的类比并不能保证人工智 能的实现，因此他们暂不考虑借用深度学习技术来开发这个系统。

现行的人工智能程序基本上都是将大大小小的各种知识写成一句一句的陈述句，再输 入系统。当输入问题进入智能程序时，就会搜寻自身的资料库，再选择最佳或最近解。2011 年，IBM公司有名的Waston智能计算机便是使用这样的技术在美国的电视益智节目中打 败了人类的最强卫冕者。虽然过去都是使用传统式的手工输入知识，然而Waston团队现 在也考虑将深度学习技术应用在一部分运算之中。IBM公司的首席科技主管Rob High表 示，他们现在已经在进行实验，研究深度学习能如何提高Waston的辨认图片的能力。

虽然各家人工智能实验室对于深度学习技术的反应不一，但科技公司与计算机科学家 已经看中它的潜在获利能力。有人已经开始尝试创立公司的可能性，而Facebook的人工智 能部门也开始招募相关领域的研究者。Andrew Ng表示，深度学习系统会随资料库的日渐 庞大而变得更加有效率。随着硬件与网络不断进化，各种影音资料急速积累，深度学习技术 将会吸引更多研究者发展它的各种可能性。

探索新的特征提取模型是值得深入研究的内容。此外，有效的可并行训练算法也是值 得研究的一个方向。通常的办法是利用图形处理单元加速学习过程。然而单个计算机的 GPU对大规模数据识别或相似任务数据集并不适用。在深度学习应用拓展方面，如何合 理、充分地利用深度学习增强传统学习算法的性能仍是目前各领域的研究重点。

**9.7**习题

1. 简述深度学习与神经网络的相同点和区别。
2. 深度学习目前有哪些成功的应用？简述原因。
3. 深度学习又称为特征学习，为何它的特征提取能力如此强？它的基本思想又是 什么？
4. 深度学习有哪些常用的模型和方法？
5. 简述深度学习中自动编译器的工作原理。
6. 卷积神经网络在图像识别上有哪些优点？
7. 卷积神经网络的训练过程中为何要使用权值共享？
8. 为什么加入L1正则化项可以防止过拟合，提高其泛化能力？
9. 训练模型需要多少数据？应该如何发掘这些数据？
10. 简述深度学习与人工智能的关系。

参考文献

El]

⑵

L3]

[4] ⑸

[6]

[7]

[8]

[9]

[10]

[11]

口2]

[13]

[14]

[15]

[16]

[17]

口8]

[19]

[20]

⑵]

[22]

[23]

[24]

[25]

[26]

[27]

[28]

[29]

[30]

[31]

李航.统计学习方法[M].北京：清华大学出版社,2012.

周志华.机器学习[M].北京：清华大学出版社,2016.

Stuart Russell,Peter Norvig.人工智能：一种现代方法[M]. 3版.北京：人民邮电出版社，2010.

蔡自兴，姜志明.基于专家系统的机器人规划口丄电子学报，1993, 21(5)： 88-90.

陈慧萍，赵跃华，钱旭.人工智能教程[M].北京：电子工业出版社，2001.

陈宗海.系统仿真技术及其应用[M],合肥：中国科学技术大学出版社,2009.

蔡自兴.人工智能研究的若干问题〃第五届中国人工智能联合会议论文集.西安：西安交通大学出 版社，1998： 527-528.

李陶深.人工智能[M].重庆：重庆大学出版社，2002.

焦李成.神经网络系统理论[M].西安：西安电子科技大学出版社，1990.

何华灿.人工智能导论[M].西安：西北工业大学出版社，1988.

陆汝铃.人工智能[M].北京：科学出版社,2000.

朱福喜，汤怡群，傅建明.人工智能基础[M].武汉：武汉大学出版社，2002.

杨祥全，蔡庆生.人工智能[M丄重庆：科技文献出版社重庆分社,1988.

尹朝庆，尹皓.人工智能与专家系统[M].北京：中国水利水电出版社，2002.

张緩，张铃.问题求解理论及应用[M].北京：清华大学岀版社，1990.

张文修，

赵瑞清.

徐立本，

阎平凡，

梁怡.遗传算法的数学基础[M].西安：西安交通大学出版社，2000.

专家系统原理[M].北京：气象出版社，1987.

姜云飞.机器学习及其应用[M].长春：吉林大学社会科学丛书编辑部，1988.

张长水.人工神经网络与模拟进化计算[M].北京：清华大学出版社，2000.

[32]

[33]

[34]

[35]

[36]

[37]

[38]

[39]

[40]

[41]

杨炳儒.知识工程与知识发现[M].北京：冶金工业出版社，2000.

涂序彦.人工智能及其应用[M].北京：电子工业出版社，1988.

李应潭.生命与智能[M].沈阳：沈阳出版社，1999.

李祖枢，涂亚庆.仿人智能控制[M].北京：国防工业出版社，2003.

廉师友.人工智能技术导论[M].2版.西安：西安电子科技大学出版社，2002.

蔡自兴.机器人学[M].北京：清华大学出版社，2000.

蔡自兴，徐光祐.人工智能及其应用[M].2版.北京：清华大学出版社，1996.

曾雪峰.论人工智能的研究与发展口丄现代商贸工业,2009,13(1)： 248-249.

元慧.议当代人工智能的应用领域与发展状态[J1 2008,5(33)： 15-16.

宋绍云，仲涛.BP人工神经网络的新型算法.人工智能及识别技术，2009, 5(5)： 1197-1198. Yann LeCun, Yoshua Bengio» Geoffrey Hinton. Deep LeamingCJ]. Nature\* 2015. 521(7553) : 436-44. Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks[J]. Advances in Neural Information Processing Systems♦ 2012, 25(2) : 1-9. 钟晓，等.数据挖掘概述[□.模式识别与人工智能，2001, 14(1)： 48-55.

史忠植.智能主体及其应用[M].北京：科学岀版社，2000.

蔡之华.模糊Petri网及知识表示[J].计算机应用与软件，1994,03： 30-36.

张科杰，袁国华，彭颖红.知识表示及其在机械工程设计中的应角探讨[J].机械设计，2004, 21(6)： 4-6,27.

刘晓霞.新的知识表示方法-一概念图[J].航空计算技术，1997,4(1)： 28-32.

王永庆.人工智能原理与方法[M].西安：西安交通大学出版社，1998.

王万森.人工智能原理及应用[M].北京：电子工业出版社,2012.

Jiawei Han.Micheline Kamber Jian Pei.数据挖掘概念与技术.北京：机械工业出版社,2012. 迈尔•舍恩伯格.大数据时代.杭州：浙江人民出版社，2013.

朱进云.大数据架构师指南.北京：清华大学出版社，2016.

图书资源支持

感谢您一直以来对清华版图书的支持和爱护。为了配合本书的使用，本书 提供配套的素材，有需求的用户请到清华大学出版社主页(<http://www.tup>. com.cn)上查询和下载，也可以拨打电话或发送电子邮件咨询。

如果您在使用本书的过程中遇到了什么问题，或者有相关图书出版计划， 也请您发邮件告诉我们，以便我们更好地为您服务。

我们的联系方式：

地 址：北京海淀区双清路学研大厦A座707

扫一扫

资源下载、样书申请 新书推荐、技术交流

邮 编：100084

电 话：010-62770175 - 4604

资源下载：http: //www. tup. com. cn

电子邮件：weijj@tup. tsinghua. edu. cn

QQ； 883604(请写明您的单位和姓名)

用微信扫一扫右边的二维码，即可关注清华大学出版社公众号“书圈



本书特色

•在作者多年讲授人工智能课程的基础上，对相关理论与技术进行了提炼与总结， 使全书结构严谨、逻辑性及前后章节的衔接增强，兼有普及与提高的双重功能。

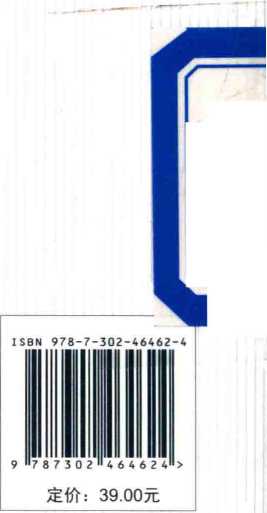
•传统人工智能部分深入浅出，通俗易懂，容易学习，主要讲述传统人工智能的基 本概念、原理、方法和技术。

•现代人工智能部分则体现其创新、精髓和尖端，主要讲述机器学习、数据挖掘、 大数据、深度学习的最新技术与方法。

•每章后面附有习题，以供读者练习；提供配套课件，读者可在清华大学出版社网 站本书页面下载。

扫一扫

课件下载、样书申请 教材推荐、技术交流



(1) 求 CER(A)。

因为 CER(£ AND E2) = min(CER(El),CER(E2)}=min(0. 8,0. 6}=0. 6 *mCia.},* {a2}) = {0. 6X0. 3, 0. 6X0. 5} = {0. 18, 0. 3}

Bel(A)=m((a1})+/72({a2})=0. 18+0.3 = 0.48

P1(A) = 1 —Bel( -1 2 A) = l —0 = 1

/(A) = Bel(A)+|A|/|lQ| X (Pl( A)-Bel(A)) =0. 48+2/10X (1—0. 48) =0. 584 所以 CER(A) = MD(A/E，)X/(A)=0. 584

(2) 求 CER(B)O

因为 CER(E3 and (E4 OR E5))

= min{CER(E3), max{CER(E4), CER(E5))}

= min{0. 9 , max{0. 5, 0. 7}} =min{0. 9, 0. 7} = 0. 7

*?n({bi }* ) = 0. 7X0. 7 = 0. 49

Bel(B) = m(仏) )=0. 49

P1(B) = 1-Bel(-«B) = 1-O = l