



目录

CONTENTS

8.1

神经网络的发展历史

8.2

神经元与神经网络

8.3

BP神经网络及其学习算法

8.4

卷积神经网络

8.5

生成对抗网络

8.6

深度学习的应用





中國石油大學(华东)
CHINA UNIVERSITY OF PETROLEUM

计算机科学与技术学院
College of Computer Science & Technology

第九章 专家系统

人工智能课题组

智能科学系





目录

CONTENTS

9.1

专家系统概述

9.2

推理方法

9.3

一个简单的专家系统

9.4

非确定性推理

9.5

专家系统工具

9.6

专家系统的应用

9.7

专家系统的局限性



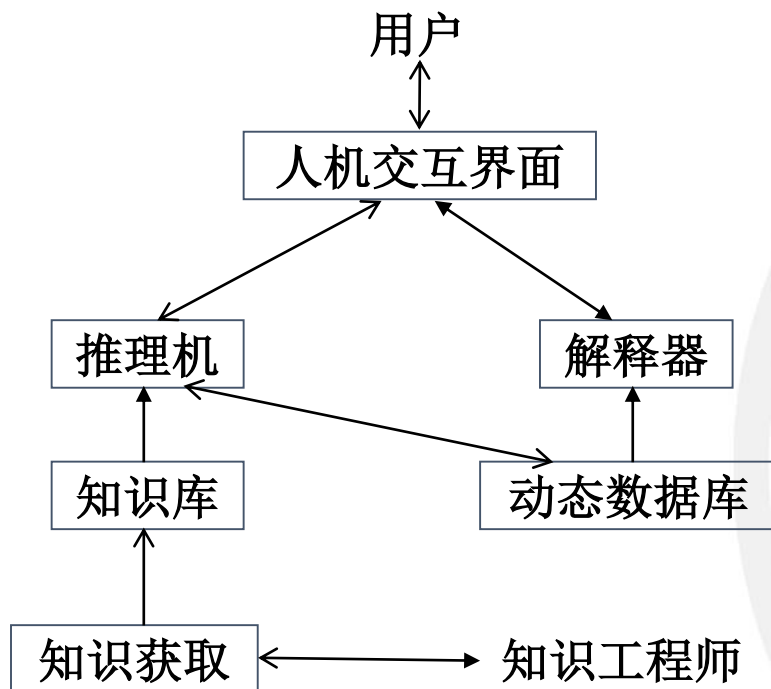


9.1 专家系统概述

- 定义：一种智能的计算机程序，它运用知识和推理来解决只有专家才能解决的复杂问题。（费根鲍姆，1982）
- 几个有名的早期系统
 - 1965年，斯坦福大学的费根鲍姆教授和化学家勒德贝格合作研发了世界上第一个专家系统DENDRAL，用于帮助化学家判断待定物质的分子结构。
 - 之后，费根鲍姆领导小组又研发了著名的专家系统MYCIN，该系统可以帮助医生对住院的血液感染患者进行诊断和选用抗生素HYPERLINK类药物进行治疗。
- 不追求问题的最佳解，利用知识得到一个满足解是系统的求解目标。

9.1 专家系统概述

- 以**知识库**和**推理机**为核心，可以处理非确定性的问题。
- 一个专家系统的基本结构如图所示：



9.1 专家系统概述

- **知识库**：用于存储求解问题所需要的领域知识和事实等。

知识一般以规则表示。

- **规则**：IF <前提> THEN <结论>/<动作>

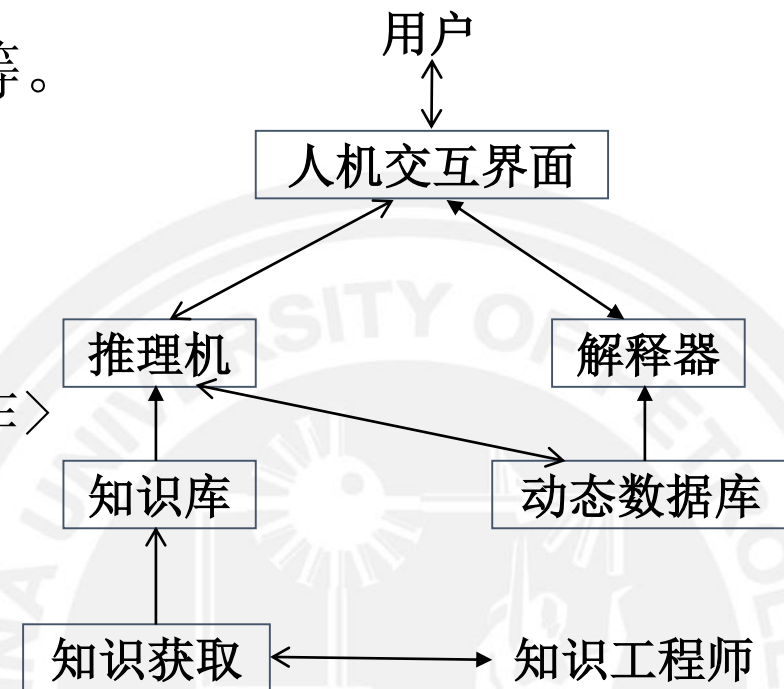
- 表示：当<前提>被满足时，可以得到<结论>或<动作>

- 例如：

IF 阴天 and 湿度大 THEN 下雨

IF 天黑 THEN 打开灯

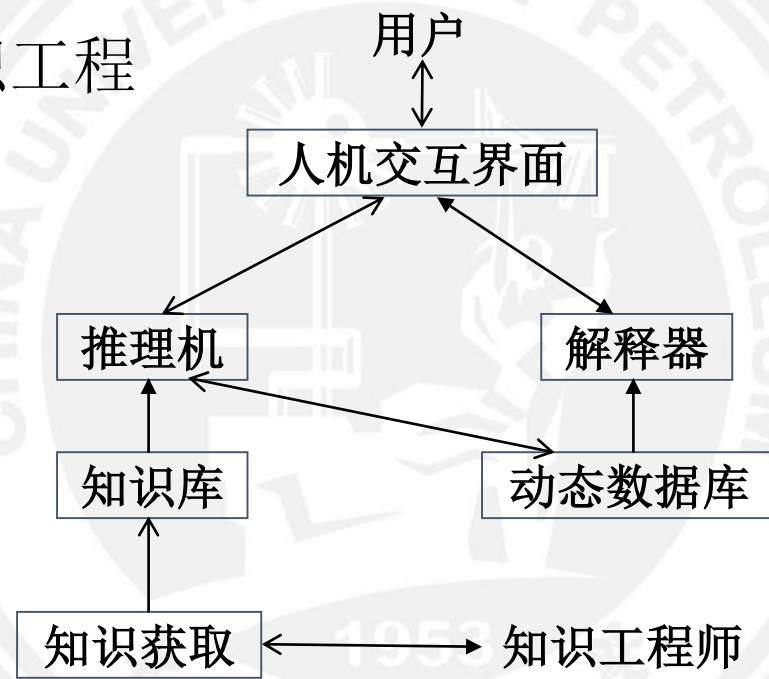
- **推理机**：是一个执行结构，负责对知识库中的知识进行解释，利用知识进行推理。





9.1 专家系统概述

- **人机交互界面**：是系统与用户的交互接口，系统在运行过程中需要用户通过该交互接口输入数据到系统中，系统则将需要显示给用户的信息通过该交互接口显示给用户。
- **知识获取**：是专家系统与知识工程师的交互接口，知识工程师通过知识获取模块将整理的领域知识加入知识库中。
- **动态数据库**：是一种工作存储区，用于存放初始已知条件、已知事实、推理过程中得到的中间结果以及最终结果等。





9.1 专家系统概述

- **解释器**：是专家系统特有的模块，也是与一般计算机软件系统的区别之一。
- 解释一般分为：
 - Why解释：回答为什么
 - 例如，在一个医疗专家系统中，系统给出患者验血的建议，如果患者想知道为什么让自己去验血，用户只有通过交互接口输入Why，系统就会根据推理结果给出让患者验血的原因，让用户明白验血的意义。
 - How解释：回答如何得到
 - 例如，假设专家系统最终诊断患者患有肺炎疾病，如果患者想了解专家系统是如何得出这个结论结果的，只要通过交互接口输入How，专家系统就会根据推理结果给用户解释，根据什么症状判断其患有肺炎。



9.2 推理方法

按照推理的方向，推理一般分为：**正向推理**和**逆向推理**

- **正向推理**：正向地使用规则，从已知条件出发向目标进行推理。（数据驱动的推理）
- 基本思想：检验是否有规则的前提被动态数据库中的已知事实满足，如果被满足，则将该规则的结论放入动态数据库中，再检查其他的规则是否有前提被满足；反复该过程，直到目标被某个规则推出结束，或者再没有新结论被推出为止。
- 例子：

设有规则：

r1: IF A and B THEN C

r2: IF C and D THEN E

r3: IF E THEN F

并且已知A、B、D成立，求证F成立。

答： $r1 \rightarrow C$, $r2 \rightarrow E$, $r3 \rightarrow F$



9.2 推理方法

- **逆向推理**：从假设求解目标成立、逆向使用规则进行推理，又称为目标驱动的推理。

- 基本思想：

先将目标作为假设，查看是否有某条规则支持该假设，即规则的结论与假设是否一致，然后看结论与假设一致的规则其前提是否成立。如果前提成立（在动态数据库中进行匹配），则假设被验证，结论放入动态数据库中；否则将该规则的前提加入假设集中，一个一个地验证这些假设，直到目标假设被验证为止。

- 例子：

设有规则：

r1: IF A and B THEN C

r2: IF C and D THEN E

r3: IF E THEN F

并且已知A、B、D成立，求证F成立。

答： F?

r3: E?, r2: C? D?, r1: A? B?

已知A、B、D



9.3 一个简单的专家系统

你的朋友不知道某种动物是什么，给你打电话咨询，对话如下：

你：你看到的动物有羽毛吗？

FD：有羽毛。

你：会飞吗？

FD：（经观察后）不会飞。

你：有长腿吗？

FD：没有。

你：会游泳吗？

FD：（看到该动物在水中）会。

你：颜色是黑白吗？

FD：是。

你：这个动物是企鹅。





9.3 一个简单的专家系统

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

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

```
(rule <规则名>
  (if <前提>
    (then <结论>)))
```

- 例如：

“如果有羽毛则是鸟类”可以表示为：

```
(rule r3
  (if (same 有羽毛 yes))
  (then (类 鸟类)))
```

其中，r3是规则名，(same 有羽毛 yes)是规则的前提，(类 鸟类)是规则的结论。



9.3 一个简单的专家系统

• 例如:

“如果是鸟类且不会飞且会游泳且是黑白色则是企鹅” 可以表示为:

```
(rule r12
  (if (same 类 鸟类)
      (notsame 会飞 yes)
      (same 会游泳 yes)
      (same 黑白色 yes))
  (then (动物 企鹅)))
```



9.3 一个简单的专家系统

2、推理机是如何**利用这些知识进行推理**的呢？假设采用逆向推理进行求解。

• 假定系统首先提出的假设是鸵鸟，则推理过程如图所示：

你：你看到的动物有羽毛吗？

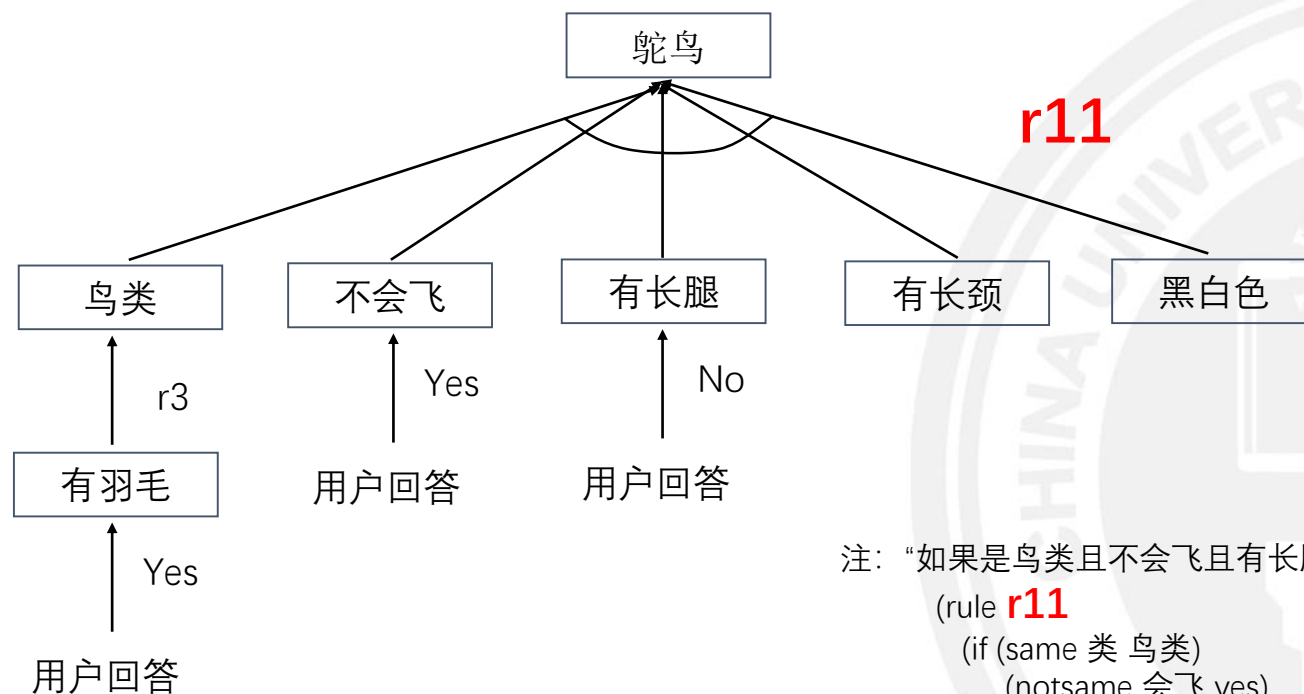
FD：有羽毛。

你：会飞吗？

FD：（经观察后）不会飞。

你：有长腿吗？

FD：没有。



注：“如果是鸟类且不会飞且有长腿且有长颈且是黑白色则是鸵鸟”：

```
(rule r11
  (if (same 类 鸟类)
    (notsame 会飞 yes)
    (same 有长腿 yes)
    (same 有长颈 yes)
    (same 黑白色 yes))
  (then (动物 鸵鸟)))
```

9.3 一个简单的专家系统

2、推理机是如何**利用这些知识进行推理**的呢？假设采用逆向推理进行求解。

• 假设鸵鸟不能成立。系统再次提出新的假设“企鹅”，推理过程如图所示：

你：你看到的动物有羽毛吗？

FD：有羽毛。

你：会飞吗？

FD：（经观察后）不会飞。

你：有长腿吗？

FD：没有。

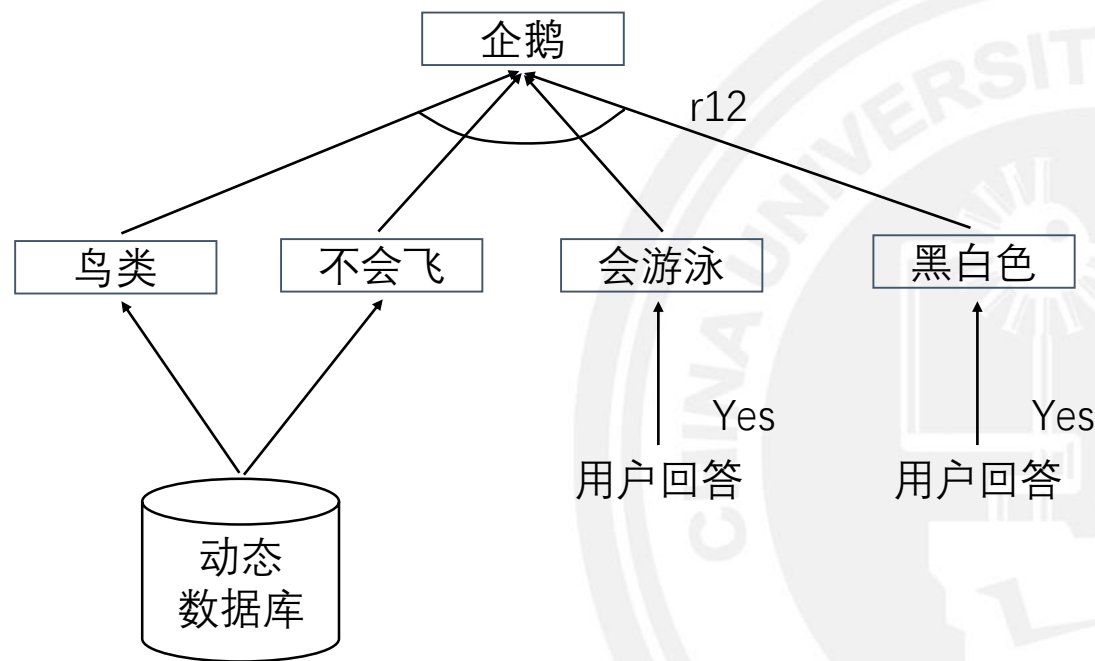
你：会游泳吗？

FD：（看到该动物在水中）会。

你：颜色是黑白吗？

FD：是。

你：这个动物是企鹅。





9.4 非确定性推理

- 在给出的专家系统的简单例子中，每个规则都是确定性的，但现实生活中很多实际问题是非确定性问题，需要非确定推理方法。

例：如果阴天则下雨

- 不确定性推理中要解决的问题：

9.4.1 事实、规则的表达

9.4.2 规则的运算

9.4.5 规则的合成





9.4.1 表示

一种简单的不确定推理方法——著名的专家系统MYCIN中的可信度方法CF (Certainty factor) 方法

1、事实的表示

事实A为真的可信度用CF(A)表示，取值范围为 $[-1, 1]$ 。

- 当CF等于1时，表示A肯定为真。
- 当CF等于-1时，表示a为真的可信度为-1，也就是a肯定为假。
- 例如：CF(阴天)=0.7，表示阴天的可信度为0.7



9.4.1 表示

2、规则的表示

具有可信度的规则表示为：IF A THEN B CF (B, A)，规则的可信度，CF取值范围也是 $[-1, 1]$ 。

- A是**规则的前提**
- B是**规则的结论**
- CF是**规则的可信度**，又称规则的强度，表示当前提A为真时，结论B为真的可信度。
- **CF取值大于0**，表示规则的前提和结论是正相关的。
- **CF取值小于0**，表示规则的前提和结论是负相关的。（前提越是成立，则结论越不成立。）
- 例子：如果阴天，则下雨的可信度为0.7
IF 阴天 THEN 下雨 0.7



9.4.1 表示

3、规则前提可以是复合条件，复合条件可以通过逻辑运算表示。常用的逻辑运算有：

- ① **与**： $CF(A \wedge B) = \min(CF(A), CF(B))$ ，表示A and B的可信度，等于CF (A) 和CF (B) 中最小的一个。
- ② **或**： $CF(A \vee B) = \max(CF(A), CF(B))$ ，表示A or B的可信度，等于CF (A) 和CF (B) 中最大的一个。
- ③ **非**： $CF(\sim A) = -CF(A)$ ，表示not A的可信度，等于A的可信度的负值。



9.4.2 规则运算

4、规则运算：

在可信度方法中，规则运算的规则按照如下方式计算：

已知：

IF A THEN B CF(B, A)

CF(A)

求：CF(B)

则：CF(B) = $\max(0, CF(A)) * CF(B, A)$

• 例子：

IF 阴天 THEN 下雨 0.7

CF(阴天) = 0.5

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



9.4.5 规则的合成

5、规则合成:

已知:

IF A1 THEN B CF1

IF A2 THEN B CF2

求CF(B)

$$CF1(B) = \max(0, CF(A1)) * CF1$$

$$CF2(B) = \max(0, CF(A2)) * CF2$$

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



9.4.5 规则的合成

5、规则合成:

$$CF(B) = \begin{cases} CF1(B) + CF2(B) - CF1(B) \times CF2(B), & \text{当} CF1(B), CF2(B) \text{均大于} 0 \text{时} \\ CF1(B) + CF2(B) + CF1(B) \times CF2(B), & \text{当} CF1(B), CF2(B) \text{均小于} 0 \text{时} \\ CF1(B) + CF2(B), & \text{其他} \end{cases}$$

例如: 有以下两条规则:

IF 阴天 THEN 下雨 0.8

IF 湿度大 THEN 下雨 0.5

且已知: $CF(\text{阴天}) = 0.5$, $CF(\text{湿度大}) = 0.4$

那么, $CF(\text{下雨})$ 应该是多少呢?

这样, 上面例子合成后的结果为: $CF(\text{下雨}) = 0.4 + 0.2 - 0.4 \times 0.2 = 0.52$



9.5 专家系统工具

- 将描述领域知识的规则等“挖掉”，只保留知识表示方法和与领域无关的推理机等部分，就得到了一个专家系统工具，这样的工具保留了原有系统的主要框架，称为骨架型工具。
 - 例如EMYCIN (Empty MYCIN)
 - 骨架型专家系统工具使用简单方便，但是灵活性不够。
- 另一种专家系统工具是语言性工具，提供给用户的是构建专家系统需要的基础机制。除了知识库之外，使用者还可以使用系统提供的基本机制，根据需要构建具体的推理机等。
 - 著名的OPS5就是这样的系统。



9.6 专家系统的应用

- 专家系统是最早走向实用的人工智能技术。
- 世界上第一个实现商用并带来经济效益的专家系统是DEC公司的XCON系统，该系统拥有1000多条人工整理的规则，帮助新计算机系统配置订单。
- 1991年海湾危机中，美国军队使用专家系统用于自动的后勤规划和运输日程安排，这项工作同时涉及5万车辆、货物和人，而且必须考虑起点、目的地、路径以及解决所有参数之间的冲突。规划技术使得一个计划可以在几个小时内产生，旧的方法需要花费几个星期。
- 1996年，清华大学开发的一个市场调查报告自动生成专家系统也在某企业得到应用，该系统可以根据市场数据自动生成一份市场调查报告。



9.6 专家系统的应用

- 医学领域是较早应用专家系统的领域，像著名的MYCIN就是一个帮助医生对血液感染患者进行诊断和治疗的专家系统，我国的“关幼波胃脘病专家系统”是一个中医诊断专家系统。
- 在农业方面，我们有针对性地开发出一系列适合我国不同地区生产条件的实用经济型农业专家系统，为农技工作者和农民提供方便、全面、实用的农业生产技术咨询和决策服务，包括蔬菜生产、果树管理、作物栽培、花卉栽培、畜禽饲养、水产养殖、牧草种植等不同类型的专家系统。



9.7 专家系统的局限性

- 首先，知识获取的瓶颈问题一直没有得到很好的解决，基本都是依靠人工总结专家经验获得知识。
- 其次，知识库总是有限的，它不能包含所有信息。知识驱动的专家系统模型只能运用已有知识库进行推理，无法学习到新的知识。
- 另外，知识驱动的专家系统只能描述特定的领域，不具有共通性，难于处理常识问题。





THANKS

