



人工智能

Artificial Intelligence

主讲：相明

西安交通大学电信学院计算机系

E_mail: mxiang@mail.xjtu.edu.cn



第二章 知识工程

2.1 概述

什么是知识：

1. 数据与信息

数据是信息的载体和表示；信息是数据的语义。

2. 知识

一般来说，把有关信息关联在一起所形成的信息结构称为知识。

雪为白色的（事实）

如果头痛且流涕，则可能患了感冒（规则）

知识的特性

1. 相对正确性

知识是经验的总结，有一定的适用条件。

2. 不确定性

1) 随机性：如果头痛且流涕，则可能患了感冒

2) 模糊性：高个子适合于打篮球。

3) 不完全性：对事物认识上的不完全、不准确导致知识的不确定性。

4) 经验性：经验性知识本身就具有不确定性。

专家系统中大部分知识都具有不确定性。

3. 可表示性与可利用性

知识的分类

◆ 按作用范围：

常识性知识：人们普遍知道的知识，即所谓常识。

领域性知识：具体应用领域中的专业性知识。

◆ 按作用及表示：

事实：用于描述领域内有关概念、事物的属性及状态
（即对事实的描述：雪是白色的）

规则：反应事物之间的因果关系

◆ 按确定性：

确定性知识：是指可指出其真值为真或假的知识，它是精确性的知识。

不确定性知识：是指具有“不确定”特性的知识，它是对不精确、不完全、及模糊性知识的总称。

2.2 知识表示方法

- ◆ 知识表示：就是对知识的一种描述，一种计算机可以接受的用于描述知识的数据结构。
- ◆ 知识的两大类表示方法：
 - 符号表示法：主要用来表示逻辑性知识。（包括本章讨论的各种方法）
 - 连接机制表示法：是用神经网络表示知识的一种方法。
- ◆ 常用的知识表示法：
 - 一阶谓词逻辑表示法，产生式表示法，框架表示法，语义网络表示法，脚本表示法，过程表示法，**Petri**网表示法，面向对象表示法。

2.2.1 经典逻辑表示法

◆谓词逻辑是一种形式语言，也是到目前为止能够表达人类思维活动规律的一种最精确的语言。它与自然语言比较接近，又可方便地存储到计算机中并被计算机做精确处理。所以它成为最早应用于人工智能中表示知识的一种语言。

命题逻辑

定义2.1: 命题是具有真假意义的语句。

- ◆ 在命题逻辑中命题通常用大写英文字母表示。
- ◆ 命题逻辑无法把客观事物的结构及逻辑关系反映出来，也不能把不同事物间的共同特点表述出来。

例如：

- $P =$ “老李是小李的父亲”。

看不出老李和小李的关系。

- $P =$ “李白是诗人”， $Q =$ “杜甫也是诗人”。

无法形式地表示出二者的共同特点（都是诗人）。

- $P =$ “每个人都是要死的”。

$Q =$ “孔子是人”。

$R =$ “孔子是要死的”。

写成命题形式： $P \wedge Q \rightarrow R$ (R 是 P ， Q 的逻辑结论)

谓词逻辑

1. 一个谓词分为谓词名与个体两个部分。
 - ◆ 谓词名刻画个体的性质、状态或个体间的关系。
 - ◆ 个体表示独立存在的事物或者概念。

例如：

Teacher(zhang), Greater(5,3)

- ◆ 谓词的一般形式

$$P(x_1, x_2, \dots, x_n)$$

其中， P 是谓词名， x_1, x_2, \dots, x_n 是个体。谓词名通常用大写的英文字母表示，个体通常用小写的英文字母表示。

2. 个体可以是常量、变元或者函数。

例如：

$\text{Less}(x, 5)$, x 是一个变元。

$\text{Teacher}(\text{father}(\text{wang}))$, 其中 $\text{father}(\text{wang})$ 是一个函数。

个体变元的取值范围称为个体域。

3. 谓词的语义由人指定。

例如：

$S(x)$ 可以表示 x 是一个人；也可以表示 x 是一朵花。

4. 谓词的取值为 **true** 或 **false**.

4. 连接词：可以使用连接词构造简单的谓词公式。

非： \neg ；析取： \vee ；合取： \wedge ；蕴含： \rightarrow ；

等价： \Leftrightarrow , \leftrightarrow ；

谓词逻辑真值表

P	Q	$\neg P$	$P \vee Q$	$P \wedge Q$	$P \rightarrow Q$	$P \Leftrightarrow Q$
T	T	F	T	T	T	T
T	F	F	T	F	F	F
F	T	T	T	F	T	F
F	F	T	F	F	T	T

5. 谓词公式 (well formed formulas)

定义：按下述规则得到的合式公式：

- (1) 单个谓词是合式公式，称为原子公式；
- (2) 若 A 是合式公式，则 $\neg A$ 也是合式公式；
- (3) 若 A, B 是合式公式，则 $A \wedge B, A \vee B, A \rightarrow B, A \leftrightarrow B$ 都是合式公式；
- (4) 若 A 是合式公式， x 是任一个体变元，则 $(\forall x)A, (\exists x)A$ 都是合式公式；
- (5) 运用有限步上述规则得到的公式是合式公式。

一些重要的等价式

交换律: $P \vee Q \Leftrightarrow Q \vee P$

结合律: $(P \vee Q) \vee R \Leftrightarrow P \vee (Q \vee R)$

分配律: $P \vee (Q \wedge R) \Leftrightarrow (P \vee Q) \wedge (P \vee R)$

德. 摩根律: $\neg(P \vee Q) \Leftrightarrow \neg P \wedge \neg Q$

双重否定律: $\neg\neg P \Leftrightarrow P$

吸收律: $P \vee (P \wedge Q) \Leftrightarrow P$

补余律: $P \vee \neg P \Leftrightarrow T, P \wedge \neg P \Leftrightarrow F$

连接词化归律: $P \rightarrow Q \Leftrightarrow \neg P \vee Q,$

$P \leftrightarrow Q \Leftrightarrow (P \rightarrow Q) \wedge (Q \rightarrow P)$

一些重要的永真蕴含式

化简式: $P \wedge Q \Rightarrow P, P \wedge Q \Rightarrow Q$

附加式: $P \Rightarrow P \vee Q, Q \Rightarrow P \vee Q$

析取三段论: $\neg P, P \vee Q \Rightarrow Q$

假言推理: $P, P \rightarrow Q \Rightarrow Q$

拒取式: $\neg Q, P \rightarrow Q \Rightarrow \neg P$

假言三段论: $P \rightarrow Q, Q \rightarrow R \Rightarrow P \rightarrow R$

二难推论: $P \vee Q, P \rightarrow R, Q \rightarrow R \Rightarrow R$

全称固化: $(\forall x)P(x) \Rightarrow P(y)$

存在固化: $(\exists x)P(x) \Rightarrow P(a)$

推理规则

上述等价式和永真蕴含式可以作为推理规则。此外，谓词逻辑中还有如下一些推理规则：

1. P规则：在推理的任何步骤都可以引入前提。
2. T规则：推理时，如果前面步骤中有一个或者多个公式永真蕴含公式S，则可把S引入推理过程中。
3. CP规则：如果能从R和前提集合中推出S来，则可从前提集合推出 $R \rightarrow S$ 。
4. 反证法： $P \Rightarrow Q$ ，当且仅当 $P \wedge \neg Q \Leftrightarrow F$ 。即Q为P的逻辑结论，当且仅当 $P \wedge \neg Q$ 是不可满足的。

定理：Q为 P_1, P_2, \dots, P_n 的逻辑结论，当且仅当
 $(P_1 \wedge P_2 \wedge \dots \wedge P_n) \wedge \neg Q$ 是不可满足的。

5. 逻辑推理：运用等价式、永真蕴涵式、及上述推理规则进行推理。

基于谓词逻辑的知识表示

- ◆ 谓词逻辑可以表示事物的状态、属性等事实性的知识，也可以用来表示事物间确定的因果关系，即规则。

事实：用谓词公式的与/或形表示，例如：

$$A \vee B \vee C, \quad A \wedge B \wedge C$$

规则：用蕴涵式表示，例如：

如果X，则Y

$$X \rightarrow Y$$

- ◆ 用谓词公式表示知识时，需要首先定义谓词，指出每个谓词的确切语义，然后再用连接词把有关的谓词连接起来，形成一个谓词公式表达一个完整的意义。

刘欢比他父亲有名。

高扬是计算机系的学生，但他不喜欢编程。

人人爱劳动。

定义如下谓词：

Famous(x,y):x比y有名。

Computer(x): x是计算机系的学生

Like(x,y): x喜欢y **Love(x,y):** x爱y **Man(x):** x是人。

然后用谓词公式表示：

Famous(liuhuan,father(liuhuan))

Computer(gaoyang) \wedge \neg Like(gaoyang, programming)

$\forall x (\text{Man}(x) \rightarrow \text{Love}(x,\text{labour}))$

一阶谓词逻辑表示法的特点

◆ 优点:

1. 自然性

接近自然语言的形式语言，用它表示的知识比较容易理解。

2. 精确性

谓词逻辑是二值逻辑，表示精确知识，可保证演绎推理结论的精确性。

3. 严密性

严格的形式定义及推理规则。

4. 容易实现

容易转换为计算机的内部形式，易于在计算机上实现。

◆ 缺点:

1. 不能表示不确定性的知识、启发性知识、元知识。

2. 组合爆炸

3. 效率低

2.2.2 产生式表示法

◆ 美国数学家E.Post在1943年首先提出“产生式”这一术语。他设计的产生式系统，称为Post系统。其目的是构造一种形式化的计算工具，并证明它具有和图灵机同样的计算能力。目前它已经成为人工智能中应用最多的一种知识表示方法。很多成功的专家系统都是用它来表示知识。例如费根保姆等人研制的分析化学分子结构的专家系统DENDRAL；肖特里菲等人研制的诊断传染性疾病的专家系统MYCIN等等。产生式形式上很简单，但在一定意义上模仿了人类思考的过程。

产生式的基本形式

1: 事实的表示

三元组: 老王年龄已40 表示为 (wang,age,40)

老王与老张是朋友: (friendship, wang,zhang)

四元组: 表示不确定性的知识

(friendship, wang, zhang, 0.8)

2、规则的表示

基本形式:

$P \rightarrow Q$

或者

If P Then Q

其中, **P**是前提, 用于指出该产生式是否可用的条件。**Q**是结论或者操作, 用于指出当前提**P**满足时, 应该得出的结论或者应该执行的操作。

◆ 例如: 下雨 \wedge 甲未打伞 \rightarrow 甲被淋湿

产生式与谓词逻辑蕴含式的区别

- ◆ 蕴含式只能表示精确知识；而产生式不仅可以表示精确知识，还可以表示不精确知识。

例如：在专家系统MYCIN中的一条知识，

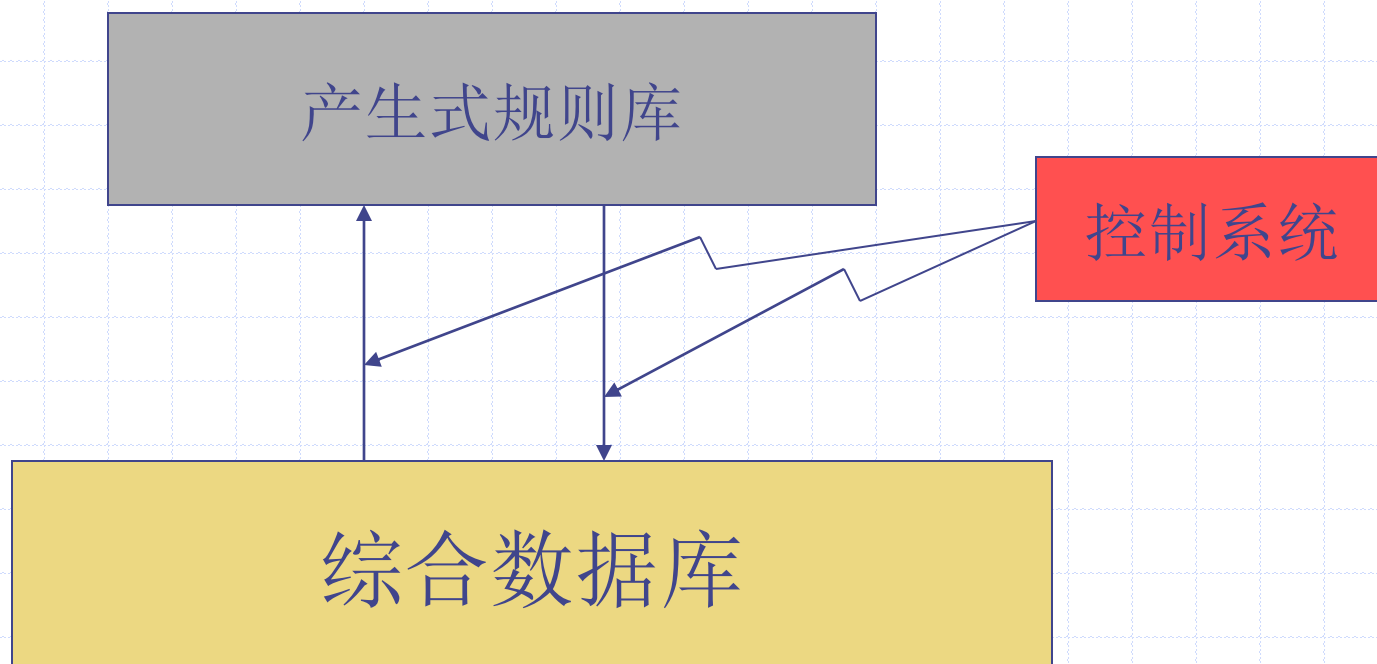
If 本微生物的染色斑是革兰氏阴性，
 本微生物的形状呈杆状，
 病人是中间宿主

Then 该微生物是绿脓杆菌，置信度为0.6

- ◆ 产生式中前提条件的匹配可以是精确的，也可以是非精确的；而谓词逻辑蕴含式总要求精确匹配。

产生式系统

- ◆ 一个产生式系统一般由三部分组成：规则库、综合数据库、控制系统。



规则库

◆ 用于描述相应领域知识的产生式集合。

建立规则库时，应该注意：

1. 有效地表达领域内的各种规则。
2. 便于对知识进行合理的组织与管理。

◆ 一个规则库的例子:

- R1: 动物有毛 \rightarrow 哺乳类
- R2: 动物产奶 \rightarrow 哺乳类
- R3: 哺乳类 \wedge 吃肉 \rightarrow 食肉类
- R4: 哺乳类 \wedge 吃草 \rightarrow 有蹄类
- R5: 食肉类 \wedge 黄褐色 \wedge 有斑点 \rightarrow 金钱豹
- R6: 食肉类 \wedge 黄褐色 \wedge 黑条纹 \rightarrow 虎
- R7: 有蹄类 \wedge 长脖 \rightarrow 长颈鹿
- R8: 有蹄类 \wedge 黑条纹 \rightarrow 斑马

综合数据库

- ◆ 又称为事实库、上下文、黑板等等。
- ◆ 存放已知的事实和推导出的事实;
- ◆ 数据结构:
符号串、向量、集合、数组、树、表格、文件等。

控制机制

- ◆ 又称为推理机构，负责整个产生式系统的运行。
- ◆ 控制机制完成的工作有：
 - 按照一定的策略，匹配规则的条件部分；
 - 当多于一条的规则匹配成功时（称为冲突），选择其中一条规则加以执行(冲突消解)；
 - 将匹配规则的结论部分放入综合数据库（直接添加到数据库中，或者替换其中的某些内容）；或者执行相应操作；
 - 计算结论的不确定性；
 - 决定系统何时终止运行。

动物识别的例子

◆ 已知事实：一动物{有毛，吃草，黑条纹}

- R1: 动物有毛 \rightarrow 哺乳类
- R2: 动物产奶 \rightarrow 哺乳类
- R3: 哺乳类 \wedge 吃肉 \rightarrow 食肉类
- R4: 哺乳类 \wedge 吃草 \rightarrow 有蹄类
- R5: 食肉类 \wedge 黄褐色 \wedge 有斑点 \rightarrow 猎狗
- R6: 食肉类 \wedge 黄褐色 \wedge 黑条纹 \rightarrow 虎
- R7: 有蹄类 \wedge 长脖 \rightarrow 长颈鹿
- R8: 有蹄类 \wedge 黑条纹 \rightarrow 斑马

产生式系统求解问题的一般步骤 (正向推理)

1. 初始化事实库；
2. 若规则库中存在尚未使用过的规则，且可匹配成功，则转第3步。否则转第5步；
3. 执行当前选中的规则，并把结论送入事实库；
4. 检查事实库中是否已经包含了解。若有则终止推理。若无则转第2步；
5. 要求用户增添事实。若有则转第2步。若无则终止推理。

产生式系统的分类

- ◆按推理方向划分：正向(数据驱动)、后向(目标驱动)和双向产生式系统
- ◆按确定性划分：确定性和不确定性产生式系统

产生式表示法的特点

◆ 优点：

1. 自然性

- ◆ “如果...则...”形式是人们常用的一种表示因果关系的知识表示形式，直观、自然、便于进行推理。

2. 模块性

3. 有效性

- ◆ 即可表示确定性知识，又可表示不确定性知识；既有利于表示启发式知识，又可便于表示过程性知识。

4. 清晰性

- ◆ 格式固定。

◆ 缺点：

1. 效率不高

反复执行“匹配——冲突消解——执行”过程，容易引起组合爆炸。

2. 不能表达具有结构性的知识

2.2.3 层次结构表示法

框架理论：

- ◆ 1957年美国著名的人工智能学者明斯基在其论文“A framework for representing knowledge”中提出了框架理论，并把它作为理解视觉、自然语言对话及其它复杂行为的基础。该理论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，当面临一个新事物时，就从记忆中找出一个合适的框架，并根据实际情况对其细节加以修改、补充，从而形成对当前事物的认识。

框架

- ◆ 框架是一种描述对象(一个事物、一个事件或一个概念)属性的数据结构。
- ◆ 一个框架由若干个“槽”组成，一个槽又可划分为若干个“侧面”。一个槽用于描述所论对象某一方面的属性，一个侧面用于描述相应属性的一个方面。

框架的一般表示形式

<框架名>

槽名1:

侧面名₁

值₁, 值₂, ..., 值_{p1}

侧面名₂

值₁, 值₂, ..., 值_{p2}

...

侧面名_{m1}

值₁, 值₂, ..., 值_{pm1}

侧面名₁

值₁, 值₂, ..., 值_{q1}

侧面名₂

值₁, 值₂, ..., 值_{q2}

...

侧面名_{m2}

值₁, 值₂, ..., 值_{qm2}

...

槽名n:

侧面名₁

值₁, 值₂, ..., 值_{r1}

侧面名₂

值₁, 值₂, ..., 值_{r2}

...

侧面名_{mn}

值₁, 值₂, ..., 值_{rmn}

约束:

约束条件₁

约束条件₂

...

约束条件_n

框架的例子

框架名： <假冒伪劣商品>

商品名称：

生产厂家：

出售商店：

处 罚：

处理方式：

处罚依据：

处罚时间： 单位（年、月、日）

经办部门：

框架的例子

框架名：<教师>

姓名：单位(姓、名)

年龄：单位(岁)

性别：范围(男、女)

缺省：男

职称：范围(教授、副教授、讲师、助教)

缺省：讲师

部门：单位(系，教研室)

住址：<住址框架>

工资：<工资框架>

开始工作时间：单位(年、月)

截止时间：单位(年、月)

缺省：现在

一个事例框架的例子

框架名：<教师-1>

姓名：夏冰

年龄：36

性别：女

职称：副教授

部门：计算机系软件教研室

住址：<addr-1>

工资：<sal-1>

开始工作时间：1988， 9

截止时间：1996， 7

框架的BNF描述

<框架> ::= <框架头> <槽部分> [<约束部分>]

<框架头> ::= 框架名 <框架名的值>

<槽部分> ::= <槽>, [<槽>]

<约束部分> ::= 约束 <约束条件>, [<约束条件>]

<框架名的值> ::= <符号名> | <符号名(<参数>, [<参数>]) >

<槽> ::= <槽名><槽值> | <侧面部分>

<槽名> ::= <系统预定义槽名> | <用户自定义槽名>

<槽值> ::= <静态描述> | <过程> | <谓词> | <框架名的值>
| <空>

<侧面部分> ::= <侧面>, [<侧面>]

<侧面> ::= <侧面名> <侧面值>

<侧面名> ::= <系统预定义侧面名> | <用户自定义侧面名>

<侧面值> ::= <静态描述> | <过程> | <谓词>

| <框架名的值> | <空>

<静态描述> ::= <数值> | <字符串> | <布尔值>

| <其它值>

<过程> ::= <动作> | <动作>, [<动作>]

<动作> ::= <动作名> | <动作名> [(<变元>, ...)]

<参数> ::= <符号名>

- (1) 当槽值或侧面值是一个过程时，过程可以是一个<动作>串，也可以是主语言的某个过程的调用，从而将过程性知识表示出来。
- (2) 当槽值或侧面值是一个谓词时，其真值由当时谓词中变元的取值确定。
- (3) 当槽值或侧面值为<空>时，表示该值等待以后填入，当时还不能确定。
- (4) 约束条件是任选的，可有可无。
- (5) 主语言：框架表示法只是完成了知识的表示，但是为了实现推理还需要编写程序对这些知识进行处理，这一语言我们称为主语言。

框架网络

- ◆ 框架中的槽值或侧面值都可以是另外一个框架，这就在框架之间建立起了联系。这种联系称为横向联系。
- ◆ 框架之间还可以有继承关系，称为纵向联系。框架中可以有“继承”槽，指明上下关系。
- ◆ 具有横向联系和纵向联系的一组框架称为框架网络

继承关系的特点

- (1) 下层框架可以继承上层框架的属性及属性值。
- (2) 下层框架可以重新对上层框架的属性进行赋值（重载）。
- (3) 下层框架可以处理的属性包括：该框架本身定义的属性（槽、侧面等），及其所有上层框架中定义的属性（槽、侧面等）。其它没有定义的属性均属于非法。

框架中槽的设置与组织

- ⑩ 充分表达事物各有关方面的属性
- ⑩ 充分表达事物间的各种关系
 - ⑩ ISA槽
 - ⑩ AKO槽与Instance槽
 - ⑩ Subclass槽
 - ⑩ Part-of槽
 - ⑩ Infer槽与Possible-Reason槽
- ◆ 有利于进行框架推理

ISA槽: 是一个: 反映个体与概念的关系

是一种: 反映概念与更一般概念之间的关系

特点: 具有继承关系

框架名: <运动员>

姓名: 单位(姓, 名)

年龄: 单位(岁)

性别: 范围(男, 女)

缺省: 男

框架名: <棋手>

ISA: <运动员>

脑力: 特好



AKO槽： 是一种, 反映事物间的类属关系.

SUBCLASS槽： 反映子类与类之间的关系.

特点： 具有继承关系.

PART OF 槽: 反映部分与全体的关系.

框架名: <墙>

PART OF: <房间>

颜色:

门数:

窗数:

特点:不具备继承关系.

INFER槽:描述两个框架之间的逻辑推理关系.

POSSIBLE REASON槽:

特点: 不具备继承关系.

如果咳嗽、发烧、流鼻涕, 则八成是患了感冒, 需要服用感冒清, 一日三次, 多喝开水。

框架名: <诊断规则>

症状1: 咳嗽

症状2: 发烧

症状3: 流涕

INFER: <结论>

可信度: 0.8

框架名: <结论>

病名: 感冒

治疗方法: 服用感冒清

注意事项: 多喝开水

效果: 良好.

注意: 在此例中,也可用POSSIBLE REASON

框架表示法的特点

优点：

◆ 结构性

- 可以把知识的内部结构显示地表示出来。

◆ 继承性

- 减少了知识冗余，较好地保证了知识的一致性。

◆ 自然性

- 框架表示法体现了人在观察事物时的思维活动，与人的认识活动是一致的。

◆ 可以表示规则（**INFER**槽）。

2.3 知识获取与管理

2.3.1 知识获取的任务

2.3.2 知识获取的方式

2.3.3 知识管理

2.4 基于知识的系统

2.4.1 什么是知识系统

2.4.2 专家系统

2.4.3 专家系统举例