

人工智能



知识表示与处理方法



知识表示与处理方法

- 知识和知识表示
- 逻辑表示法
- 产生式表示法
- 框架表示法
- 语义网络表示法



1. 知识和知识表示

概述

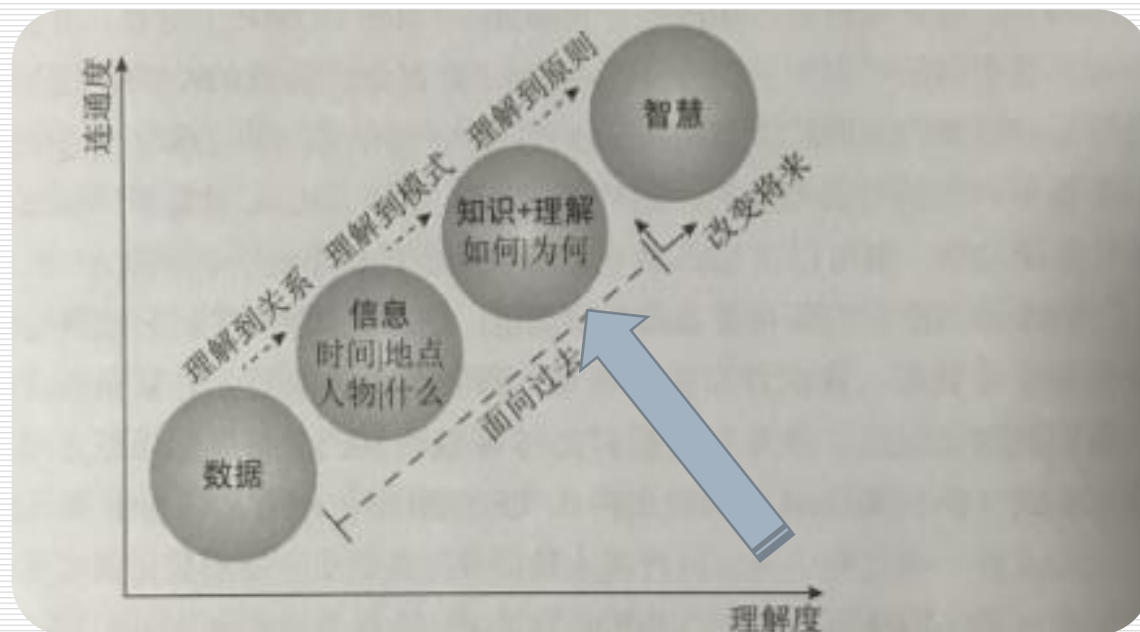
人工智能的求解以知识为基础，一个程序具备的知识越多，它的求解问题的能力就越强，所以，知识表示是人工智能研究的一个重要课题，也贯穿着人工智能发展历程。

1. 知识和知识表示

什么是知识？

- 知识是人类进行一切智能活动的基础。
- 知识是人们在长期的生活及社会实践中，在科学研究及实验中积累起来的，对客观世界的认识与经验。
- 把有关信息关联在一起所形成的信息结构称为**知识**。

DIKW 模型



数据分析

智能

认知

美国系统理论家罗素.艾可夫认为人的心智内容分为5个层次

1. 数据：承载信息的原始符号
2. 信息：通过关系连接在一起有特定含义的有用数据，能提供when,where,who,what类问题的答案；
3. 知识：按特定模式组合的信息集合，可以记忆和保存，能回答how之类的问题； ---记忆
4. 理解：然而知识还需进一步建立在已知知识基础上的理解，即从已知获取新知识，并综合起来回答why之类的问题，才算达到认知和分析性的水准； ---明白
5. 智慧：是一个不确定的非概率性外推过程，是建立原理和准则的过程，也是哲学探索的本质

1. 知识和知识表示

1. 什么是知识



数据：一些无关联的**事实 (fact)**

信息：建立了事实间的联系后形成信息，提供对 what, who, when, where 等问题的回答

知识：当能建立模式之间的联系后便涌现了知识，提供对 how 的回答

智慧：能描述模式之间关系的规律，提供对 why 的回答

例：下雨了

例：温度下降了 15 度后就下雨了

例：如果湿度非常大并且温度下降显著，大气无法保留湿气便下雨了

例：理解下雨、蒸发、空气状况、温度等级及它们的变化之间的相互作用

1. 知识和知识表示

2. 知识的特性

- 相对正确性
- 不确定性
 - ✓ 由随机性引起的不确定性
 - ✓ 由模糊性引起的不确定性
 - ✓ 由经验引起的不确定性
 - ✓ 由不完全性引起的不确定性
- 可表示性与可利用性

1. 知识和知识表示

3. 知识表示

知识表示是指将知识形式化或数学描述，以便在计算机中表示，并能让计算机程序自动处理。

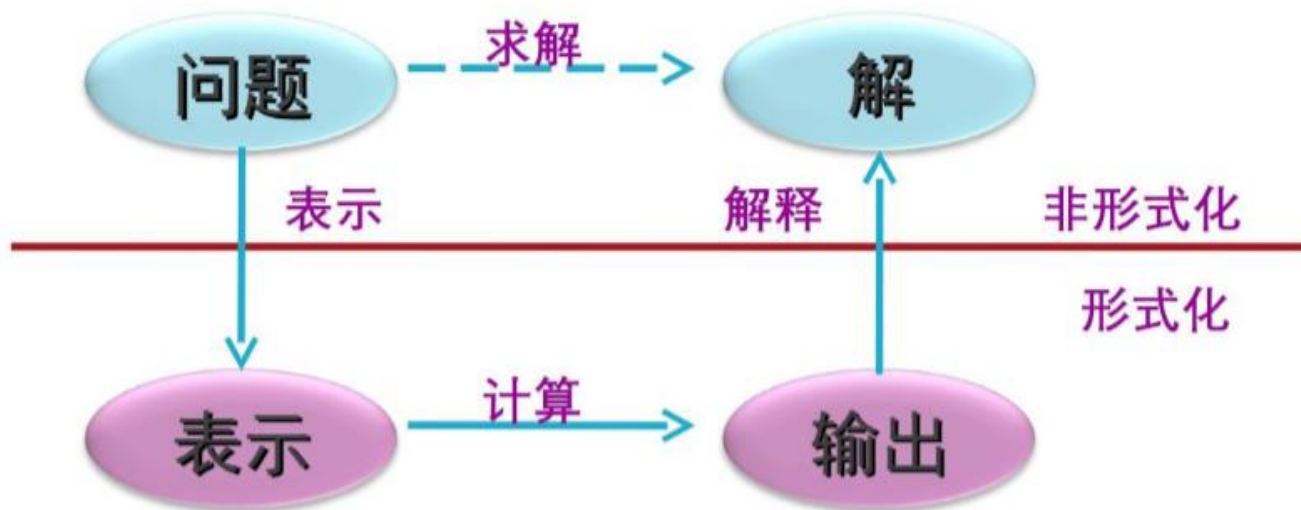
在**AI**领域，研究知识表示方法的目的是用知识来改善程序的性能，**AI**的研究集中在如何使程序拥有知识而具有智能行为。

1. 知识和知识表示

3. 知识表示

▶ 知识表示的过程

- 非形式化的自然语言描述 → 形式化的易于被计算机理解



1. 知识和知识表示

4. 知识表示方法

- ① 一阶谓词逻辑
- ② 产生式
- ③ 框架
- ④ 语义网络（知识图谱）
- ⑤ 状态空间
- ⑥ 问题归约
- ⑦ 遗传编码
- ⑧ 人工神经网络

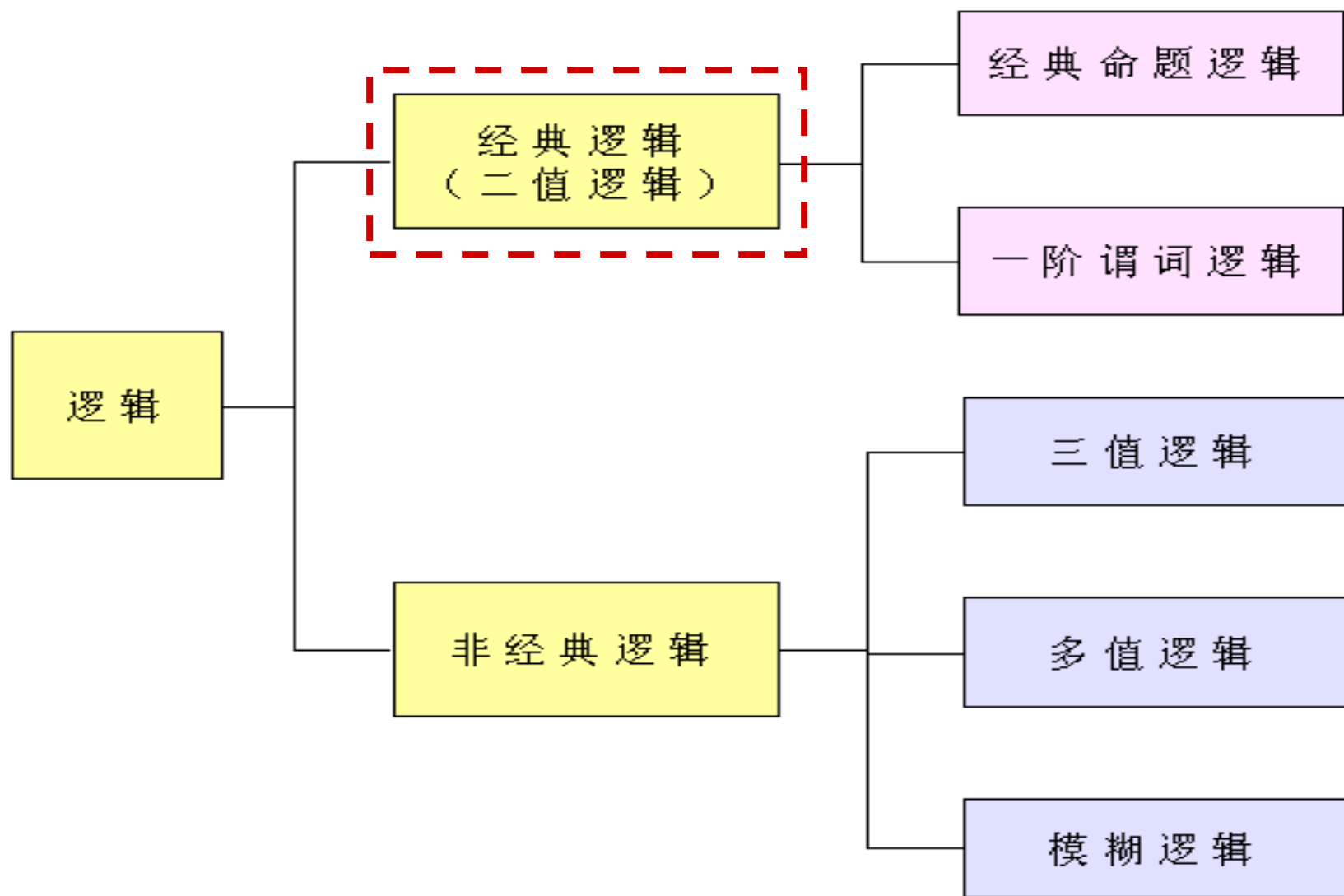


2. 一阶谓词逻辑表示法

逻辑表示法

- 逻辑是一种比较常见的知识表示法；
- 命题逻辑和谓词逻辑最先应用于人工智能的两种逻辑，在知识的形式化表示方面，特别在定理的自动证明方面，发挥了重要作用，在AI的发展史中占有重要地位。

2. 一阶谓词逻辑表示法



2. 一阶谓词逻辑表示法

逻辑表示法

- **命题逻辑**，常用大写英文字母表示，如：**P**表示“武汉是第七届世界军人运动会的主办城市”这个命题。
- 命题逻辑表示法有较大的局限性：
 - 其无法把它描述的事物的结构及逻辑特征反映出来，“老李是小李的父亲”——字母**P**表示不出二李怎样的父子关系；
 - 也不能把不同事物间的共同特征表述出来，如：两个命题“李白是诗人”，“杜甫也是诗人”——用**P**和**Q**表示两个命题，则整个命题表示为 **$P \wedge Q$** 共同特征（都是诗人）无法形式化的表示出来。

2. 一阶谓词逻辑表示法

逻辑表示法

- 谓词逻辑，是在命题逻辑的基础上发展而来的。
- 一个命题至少由一个主语和一个谓语组成。
 - 主语表示了独立存在的某个事物、实体或概念；
 - 谓语表示主语的性质、状态、关系等属性。
- **谓词逻辑：**将命题中的“谓词”抽象出来，作为知识的核心，而将主语作为谓词支配的部分，就要将知识形式化。

2. 一阶谓词逻辑表示法

1. 命题
2. 谓词
3. 谓词公式
4. 谓词公式的性质
5. 一阶谓词逻辑知识表示方法
6. 一阶谓词逻辑表示法的特点

2. 一阶谓词逻辑表示法 – 命题

□ 命题 (proposition) 例如: $3 < 5$ 非真即假的陈述句。

- 若命题的意义为真, 称它的真值为真, 记为 T 。
- 若命题的意义为假, 称它的真值为假, 记为 F 。
- 一个命题可在一定条件下为假。例如: 太阳从西边升起

P : 北京是中华人民共和国的首都

■ 命题逻辑: 研究命题及命题之间关系的符号逻辑系统。

■ 命题逻辑表示法: 无法把它所描述的事物的结构及逻辑特征反映出来, 也不能把不同事物间的共同特征表述出来。

P : 老李是小李的父亲

P : 李白是诗人
 Q : 杜甫也是诗人

2. 一阶谓词逻辑表示法 – 谓词

- 谓词的一般形式: $P(x_1, x_2, \dots, x_n)$
- 个体 x_1, x_2, \dots, x_n : 某个独立存在的事物或者某个抽象的概念;
- 谓词名 P : 刻画个体的性质、状态或个体间的关系。

(1) 个体是常量: 一个或者一组指定的个体。

- “老张是一个教师”: 一元谓词 $Teacher(Zhang)$
- “ $5 > 3$ ”: 二元谓词 $Greater(5, 3)$
- “Smith作为一个工程师为IBM工作”:
三元谓词 $Works(Smith, IBM, engineer)$

2. 一阶谓词逻辑表示法 – 谓词

(2) 个体是变元（变量）：没有指定的一个或者一组个体。

“ $x < 5$ ” : *Less*(x , 5)

(3) 个体是函数：一个个体到另一个个体的映射。

■ “小李的父亲是教师” : *Teacher* (*father* (*Li*))

(4) 个体是谓词

■ “Smith作为一个工程师为IBM工作” :
二阶谓词 *Works* (*engineer* (*Smith*), *IBM*)

2. 一阶谓词逻辑表示法 – 谓词公式

谓词公式：由谓词符号、常量符号、变量符号、函数符号以及括号、逗号等按一定语法规则组成的字符串的表达式。

1. 连接词 () “机器人不在2号房间” : $\neg \text{Inroom}(\text{robot}, r2)$

(1) \neg : “李明打篮球或踢足球” :
 $\text{Plays}(\text{Liming}, \text{basketball}) \vee \text{Plays}(\text{Liming}, \text{football})$

(2) \vee : “析取” (disjunction) ——或。

(3) \wedge : “合取” (conjunction) ——与。

“我喜欢音乐和绘画” :
 $\text{Like}(I, \text{music}) \wedge \text{Like}(I, \text{painting})$

2. 一阶谓词逻辑表示法 – 谓词公式

1. 连接词（连词）

(4) \rightarrow : “蕴含” (implication) 或 “条件”

“如果刘华跑得最快，那么他取得冠军。”：

$RUNS (Liuhua, faster) \rightarrow WINS (Liuhua, champion)$

(5) \leftrightarrow : “等价” (equivalence) 或 “双条件” (bicondition)。

$P \leftrightarrow Q$: “ P 当且仅当 Q ”。

2. 一阶谓词逻辑表示法 – 谓词公式

2. 量词 (quantifier)

(1) 全称量词 (**universal quantifier**) ($\forall x$) : “对个体域中的所有 (或任一个) 个体 x ”。

“所有的机器人都是灰色的”：

$$(\forall x)[ROBOT(x) \rightarrow COLOR(x, GRAY)]$$

(2) 存在量词 (**existential quantifier**) ($\exists x$) : “在个体域中存在个体 x ”。

“1号房间有个物体”：

$$(\exists x) INROOM(x, r1)$$

2. 一阶谓词逻辑表示法 – 谓词公式

全称量词和存在量词举例：

- $(\forall x)(\exists y) F(x, y)$ 表示对于个体域中的任何个体 x 都存在个体 y ， x 与 y 是朋友。
- $(\exists x)(\forall y) F(x, y)$ 表示在个体域中存在个体 x ，与个体域中的任何个体 y 都是朋友。
- $(\exists x)(\exists y) F(x, y)$ 表示在个体域中存在个体 x 与个体 y ， x 与 y 是朋友。
- $(\forall x)(\forall y) F(x, y)$ 表示对于个体域中的任何两个个体 x 和 y ， x 与 y 都是朋友。

2. 一阶谓词逻辑表示法 – 谓词公式

全称量词和存在量词出现的次序将影响命题的含义。

例如：

- $(\forall x)(\exists y)(Employee(x) \rightarrow Manager(y, x)) :$

“每个雇员都有一个经理。”

- $(\exists y)(\forall x)(Employee(x) \rightarrow Manager(y, x)) :$

“有一个人是所有雇员的经理。”

2. 一阶谓词逻辑表示法 – 谓词公式

- 如果星期一不下雨，Tom会去登山。
- $\neg \text{weather}(\text{monday}, \text{rain}) \rightarrow \text{go}(\text{tom}, \text{mountains})$
- 所有篮球运动员都很高。
- $\forall X (\text{basketball_player}(X) \rightarrow \text{tall}(X))$
- 许多人喜欢三文鱼。
- $\exists X (\text{person}(X) \wedge \text{likes}(X, \text{Salmon}))$
- 没有人喜欢纳税。
- $\neg \exists X \text{ likes}(X, \text{taxes})$

2. 一阶谓词逻辑表示法 – 谓词公式

3. 谓词公式

□ 定义2.2 可按下述规则得到谓词演算的谓词公式：

- (1) 单个谓词是谓词公式，称为原子谓词公式。
- (2) 若A是谓词公式，则 $\neg A$ 也是谓词公式。
- (3) 若A，B都是谓词公式，则 $A \wedge B$ ， $A \vee B$ ， $A \rightarrow B$ ，
 $A \leftrightarrow B$ 也都是谓词公式。
- (4) 若A是谓词公式，则 $(\forall x) A$ ， $(\exists x) A$ 也是谓词公式。
- (5) 有限步应用 (1) – (4) 生成的公式也是谓词公式。

连接词的优先级别从高到低排列：

\neg ， \wedge ， \vee ， \rightarrow ， \leftrightarrow

2. 一阶谓词逻辑表示法 – 谓词公式

4. 量词的辖域

- 量词的辖域：位于量词后面的单个谓词或者用括弧括起来的谓词公式。
- 约束变元与自由变元：辖域内与量词中同名的变元称为约束变元，不同名的变元称为自由变元。

■ 例如：

$$(\exists x)(P(x, y) \rightarrow Q(x, y)) \vee R(x, y)$$

- $(P(x, y) \rightarrow Q(x, y))$ ： $(\exists x)$ 的辖域，辖域内的变元 x 是受 $(\exists x)$ 约束的变元， $R(x, y)$ 中的 x 是自由变元。
- 公式中的所有 y 都是自由变元。

2. 一阶谓词逻辑表示法 – 谓词公式

1. 谓词公式的解释

□ 谓词公式在个体域上的解释：个体域中的实体对谓词演算表达式中的每个常量、变量、谓词和函数符号的指派。

Friends (george, x)

Friends (george, susie) T

Friends (george, kate) F

■ 对于每一个解释，谓词公式都可求出一个真值（*T* 或 *F*）。

2. 一阶谓词逻辑表示法 – 谓词公式

2. 谓词公式的永真性、可满足性、不可满足性

■ 定义2.3 如果谓词公式 P 对个体域 D 上的任何一个解释都取得真值 T ，则称 P 在 D 上是永真的；如果 P 在每个非空个体域上均永真，则称 P 永真。

■ 定义2.4 如果谓词公式 P 对个体域 D 上的任何一个解释都取得真值 F ，则称 P 在 D 上是永假的；如果 P 在每个非空个体域上均永假，则称 P 永假。

■ 定义2.5 对于谓词公式 P ，如果至少存在一个解释使得 P 在此解释下的真值为 T ，则称 P 是可满足的，否则，则称 P 是不可满足的。

2. 一阶谓词逻辑表示法 – 谓词公式

3. 谓词公式的等价性

- 定义2.6 设 P 与 Q 是两个谓词公式， D 是它们共同的个体域，若对 D 上的任何一个解释， P 与 Q 都有相同的真值，则称公式 P 和 Q 在 D 上是等价的。如果 D 是任意个体域，则称 P 和 Q 是等价的，记为 $P \Leftrightarrow Q$ 。

2. 一阶谓词逻辑表示法 – 谓词公式

4. 谓词公式的永真蕴含

■ 定义2.7 对于谓词公式 P 与 Q ，如果 $P \rightarrow Q$ 永真，则称公式 P 永真蕴含 Q ，且称 Q 为 P 的逻辑结论，称 P 为 Q 的前提，记为 $P \Rightarrow Q$ 。

2. 一阶谓词逻辑表示法 – 谓词公式

□ 谓词逻辑的其他推理规则：

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。

2. 一阶谓词逻辑表示法

□ 谓词公式表示知识的步骤：

(1) 定义谓词及个体。

(2) 变元赋值。

(3) 用连接词连接各个谓词，形成谓词公式。

■ 例如：用一阶谓词逻辑可以表示关系数据库。



2. 一阶谓词逻辑表示法

- 用一阶谓词表示:

Occupant (Zhang , 201)

Occupant (Li, 201)

Occupant (Wang, 202)

Occupant (Zhao, 203)

Telephone (491, 201)

Telephone (492, 201)

Telephone (451, 202)

Telephone (451, 203)

2. 一阶谓词逻辑表示法

□ 优点:

- ① 自然性
- ② 精确性
- ③ 严密性
- ④ 容易实现

■ 局限性:

- ① 不能表示不确定的知识
- ② 组合爆炸
- ③ 效率低

□ 应用:

- (1) 自动问答系统 (Green等人研制的QA3系统)
- (2) 机器人行动规划系统 (Fikes等人研制的STRIPS系统)
- (3) 机器博弈系统 (Filman等人研制的FOL系统)
- (4) 问题求解系统 (Kowalski等设计的PS系统)

3. 产生式表示法

- 产生式
- 产生式系统
- 产生式系统——动物识别系统
- 产生式表示法的特点

3. 产生式表示法

- “产生式”：1943年，美国数学家波斯特（E. Post）首先提出。
- 1972年，纽厄尔和西蒙在研究人类的认知模型中开发了基于规则的产生式系统。
- 产生式通常用于表示事实、规则以及它们的不确定性度量，适合于表示事实性知识和规则性知识。

3. 产生式表示法

1. 确定性规则知识的产生式表示

- 基本形式: IF P THEN Q

或者: $P \rightarrow Q$

- 例如:

r_4 : IF 动物会飞 AND 会下蛋 THEN 该动物是鸟

2. 不确定性规则知识的产生式表示

- 基本形式: IF P THEN Q (置信度)

或者: $P \rightarrow Q$ (置信度)

例如: IF 发烧 THEN 感冒 (0.6)

3. 产生式表示法

3. 确定性事实性知识的产生式表示

- 三元组表示：（对象，属性，值）
或者：（关系，对象1，对象2）
- 例： 老李年龄是40岁： $(Li, age, 40)$
老李和老王是朋友： $(friend, Li, Wang)$

4. 不确定性事实性知识的产生式表示

- 四元组表示：（对象，属性，值，置信度）
或者：（关系，对象1，对象2，置信度）
- 例： 老李年龄很可能是40岁： $(Li, age, 40, 0.8)$
老李和老王不大可能是朋友： $(friend, Li, Wang, 0.1)$

3. 产生式表示法

□ 产生式与谓词逻辑中的蕴含式的区别：

- (1) 除逻辑蕴含外，产生式还包括各种操作、规则、变换、算子、函数等。例如，“如果炉温超过上限，则立即关闭风门”是一个产生式，但不是蕴含式。
- (2) 蕴含式只能表示精确知识，而产生式不仅可以表示精确的知识，还可以表示不精确知识。

蕴含式的匹配总要求是精确的。产生式匹配可以是精确的，也可以是不精确的，只要按某种算法求出的相似度落在预先指定的范围内就认为是可匹配的。

3. 产生式表示法

□ 产生式的形式描述及语义——巴科斯范式BNF (Backus Normal Form)

$\langle \text{产生式} \rangle ::= \langle \text{前提} \rangle \rightarrow \langle \text{结论} \rangle$

$\langle \text{前提} \rangle ::= \langle \text{简单条件} \rangle | \langle \text{复合条件} \rangle$

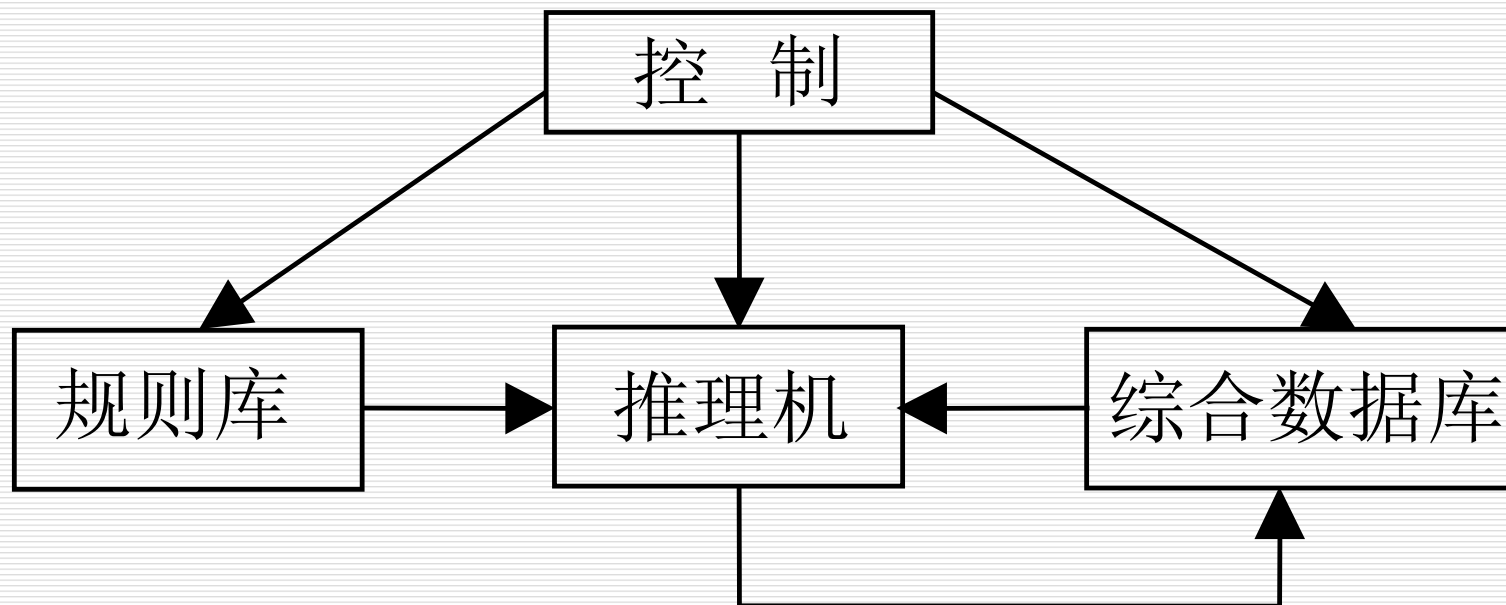
$\langle \text{结论} \rangle ::= \langle \text{事实} \rangle | \langle \text{操作} \rangle$

$\langle \text{复合条件} \rangle ::= \langle \text{简单条件} \rangle \text{AND} \langle \text{简单条件} \rangle [\text{AND} \langle \text{简单条件} \rangle \dots$
 $\quad | \langle \text{简单条件} \rangle \text{OR} \langle \text{简单条件} \rangle [\text{OR} \langle \text{简单条件} \rangle \dots$

$\langle \text{操作} \rangle ::= \langle \text{操作名} \rangle [(\langle \text{变元} \rangle, \dots)]$

符号“ $::=$ ”表示“定义为”；符号“ $|$ ”表示“或者是”；符号“ $[]$ ”表示“可缺省”。

3. 产生式表示法



产生式系统的基本结构

3. 产生式表示法

1. 规则库

- 规则库：用于描述相应领域内知识的产生式集合。

2. 综合数据库（动态改变的）

- 综合数据库（事实库、上下文、黑板等）：一个用于存放问题求解过程中各种当前信息的数据结构。

3. 控制系统

- 控制系统（推理机构）：由一组程序组成，负责整个产生式系统的运行，实现对问题的求解。

3. 产生式表示法

3. 控制系统（续）

控制系统主要完成以下几项工作：

- （1）从规则库中选择与综合数据库中的已知事实进行匹配。
- （2）匹配成功的规则可能不止一条，进行冲突消解。
- （3）执行某一规则时，如果其右部是一个或多个结论，则把这些结论加入到综合数据库中：如果其右部是一个或多个操作，则执行这些操作。
- （4）对于不确定性知识，在执行每一条规则时还要按一定的算法计算结论的不确定性。
- （5）检查综合数据库中是否包含了最终结论，决定是否停止系统的运行。

3. 产生式表示法 - 动物识别系统

■ 例如：动物识别系统——识别虎、金钱豹、斑马、长颈鹿、鸵鸟、企鹅、信天翁等七种动物的产生式系统。



3. 产生式表示法 - 动物识别系统

□ 规则库:

- r_1 : IF 该动物有毛发 THEN 该动物是哺乳动物
- r_2 : IF 该动物有奶 THEN 该动物是哺乳动物
- r_3 : IF 该动物有羽毛 THEN 该动物是鸟
- r_4 : IF 该动物会飞 AND 会下蛋 THEN 该动物是鸟
- r_5 : IF 该动物吃肉 THEN 该动物是食肉动物
- r_6 : IF 该动物有犬齿 AND 有爪 AND 眼盯前方
THEN 该动物是食肉动物
- r_7 : IF 该动物是哺乳动物 AND 有蹄
THEN 该动物是有蹄类动物
- r_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

3. 产生式表示法 - 动物识别系统

r_9 : 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

r_{10} : 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 该动物是虎

r_{11} : 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

r_{12} : IF 该动物有蹄类动物 AND 身上有黑色条纹
THEN 该动物是斑马

r_{13} : IF 该动物是鸟 AND 有长脖子 AND 有长腿 AND 不会飞
AND 有黑白二色 THEN 该动物是鸵鸟

r_{14} : 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

r_{15} : IF 该动物是鸟 AND 善飞 THEN 该动物是信天翁

3. 产生式表示法 - 动物识别系统

□ 设已知初始事实存放在综合数据库中：

该动物身上有：暗斑点，长脖子，长腿，奶，蹄

□ 推理机构的工作过程：

(1) 从规则库中取出 r_1 ，检查其前提是否可与综合数据库中的已知事实匹配。匹配失败则 r_1 不能被用于推理。然后取 r_2 进行同样的工作。匹配成功则 r_2 被执行。

■ 综合数据库：

该动物身上有：暗斑点，长脖子，长腿，奶，蹄，哺乳动物

3. 产生式表示法 - 动物识别系统

■ 推理机构的工作过程：

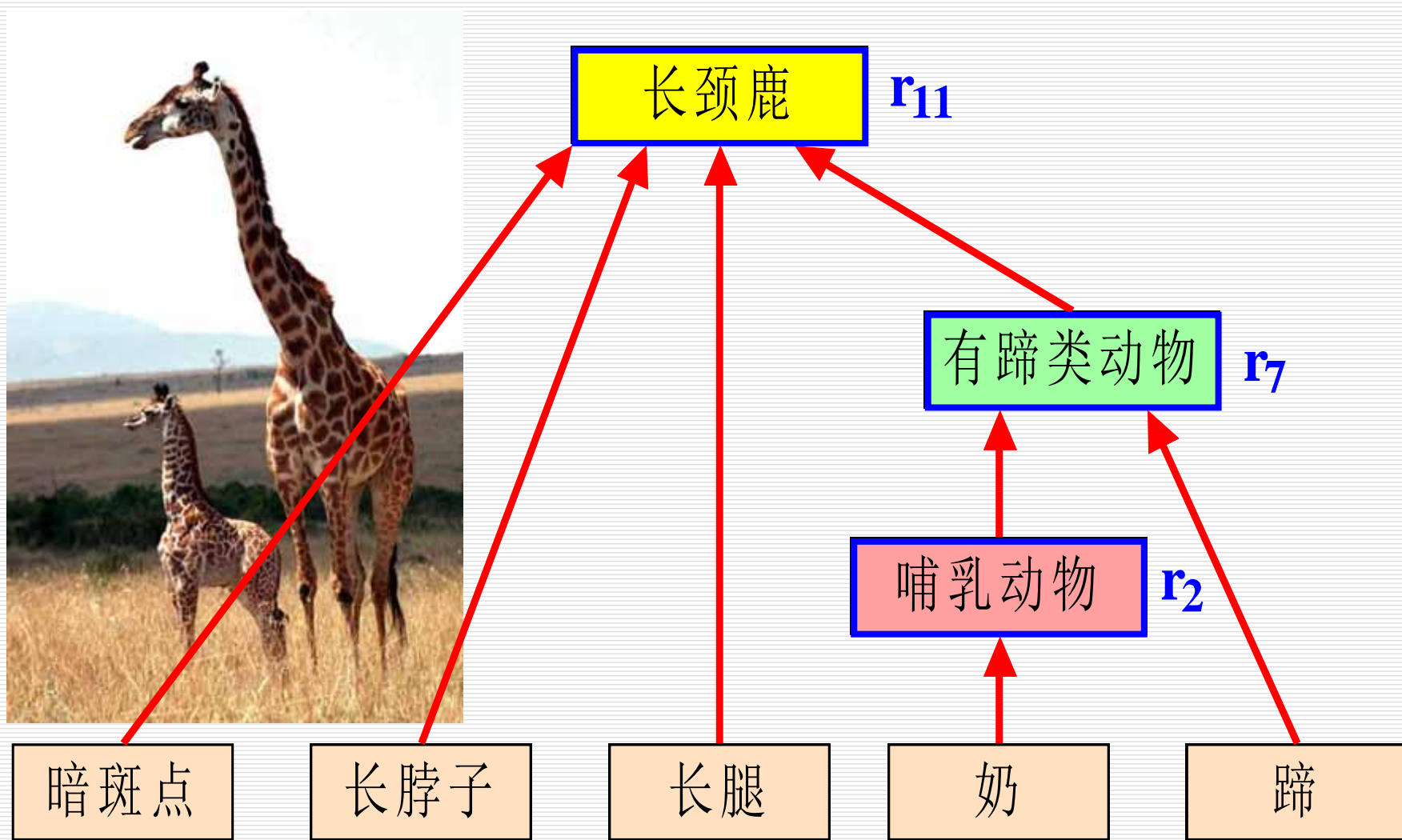
(2) 分别用 r_3 , r_4 , r_5 , r_6 综合数据库中的已知事实进行匹配, 均不成功。 r_7 匹配成功, 执行 r_7 。

■ 综合数据库:

该动物身上有：暗斑点，长脖子，长腿，奶，蹄，哺乳动物，有蹄类动物

(3) r_{11} 匹配成功, 并推出“该动物是长颈鹿”。

3. 产生式表示法 - 动物识别系统



3. 产生式表示法 – 优缺点

1. 产生式表示法的优点

- (1) 自然性
- (2) 模块性
- (3) 有效性
- (4) 清晰性

2. 产生式表示法的缺点

- (1) 效率不高（若规则多更不高）
- (2) 不能表达结构性知识

3. 适合产生式表示的知识

- (1) 领域知识间关系不密切，不存在结构关系。
- (2) 经验性及不确定性的知识，且相关领域中对这些知识没有严格、统一的理论。
- (3) 领域问题的求解过程可被表示为一系列相对独立的操作，且每个操作可被表示为一条或多条产生式规则。

4. 框架表示法

- 1975年，美国明斯基提出了框架理论：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。
- **框架表示法：**一种结构化的知识表示方法，已在多种系统中得到应用。

4. 框架表示法

- ❑ 框架（frame）：一种描述所论对象（一个事物、事件或概念）属性的数据结构。
- ❑ 一个框架由若干个被称为“槽”（slot）的结构组成，每一个槽又可根据实际情况划分为若干个“侧面”（facet）。
- ❑ 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。
- ❑ 槽和侧面所具有的属性值分别被称为槽值和侧面值。

4. 框架表示法

<框架名>

槽名1:	侧面名 ₁₁	侧面值 ₁₁₁ , ... , 侧面值 _{11P1}
	⋮	
	侧面名 _{1m}	侧面值 _{1m1} , ... , 侧面值 _{1mPm}
槽名n:	侧面名 _{n1}	侧面值 _{n11} , ... , 侧面值 _{n1P1}
	⋮	
	侧面名 _{nm}	侧面值 _{nm1} , ... , 侧面值 _{nmPm}
约束:	约束条件 ₁	
	⋮	
	约束条件 _n	

4. 框架表示法 - 实例

■ 例1 教师框架

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）

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

4. 框架表示法 - 实例

■ 例2 教师框架

当把具体的信息填入槽或侧面后，就得到了相应框架的一个实例框架。

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：〈adr-1〉

工资：〈sal-1〉

开始工作时间：1988， 9

截止时间：1996， 7

4. 框架表示法 - 实例

■ 例3 教室框架

框架名：〈教室〉

墙数：

窗数：

门数：

座位数：

前墙：〈墙框架〉

后墙：〈墙框架〉

左墙：〈墙框架〉

右墙：〈墙框架〉

门：〈门框架〉

窗：〈窗框架〉

黑板：〈黑板框架〉

天花板：〈天花板框架〉

讲台：〈讲台框架〉

4. 框架表示法 - 实例

- 例4 将下列一则地震消息用框架表示：“某年某月某日，某地发生6.0级地震，若以膨胀注水孕震模式为标准，则三项地震前兆中的波速比为0.45，水氦含量为0.43，地形改变为0.60。”
- 解：地震消息用框架如下图所示。

框架名：〈地震〉

地 点：某地

日 期：某年某月某日

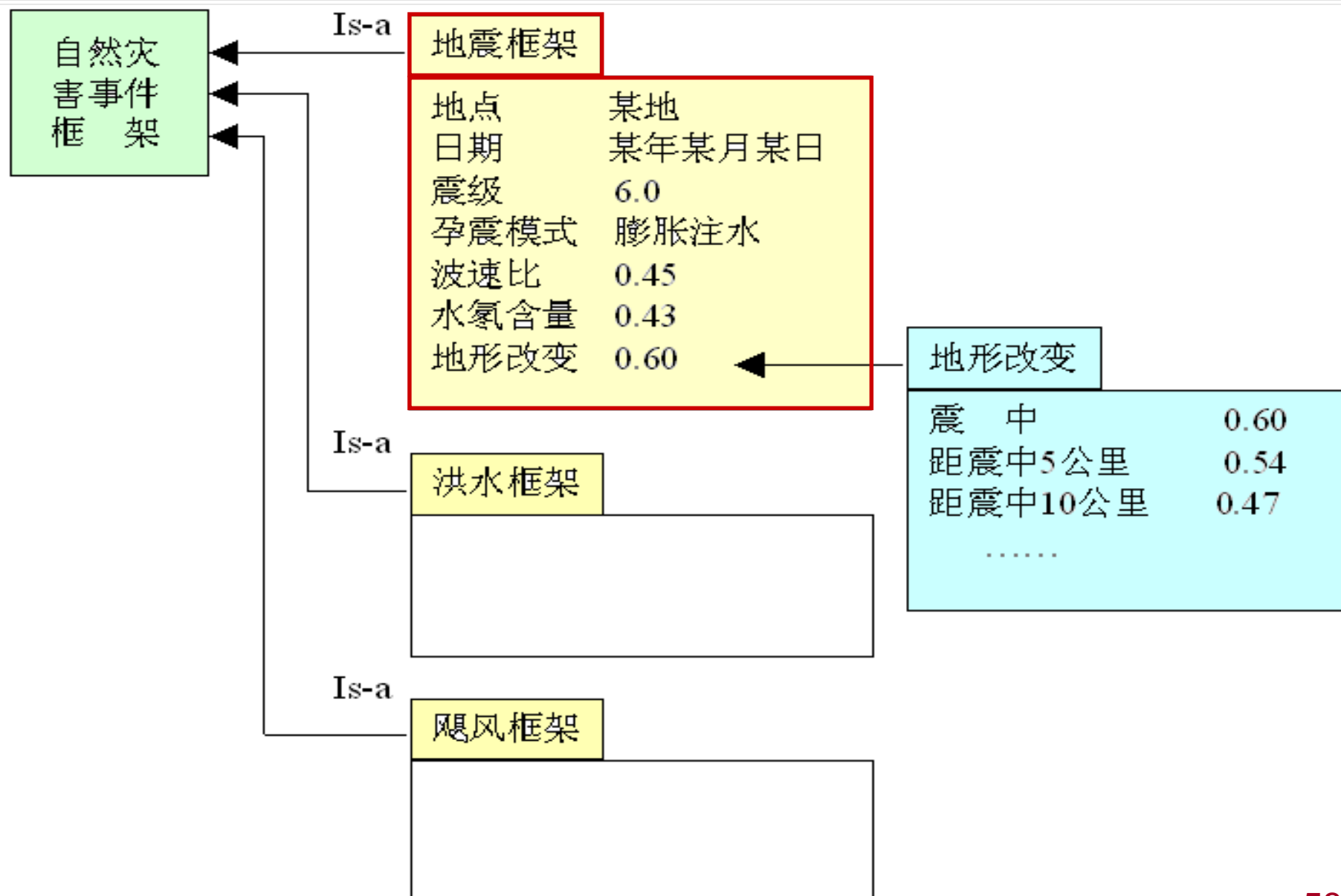
震 级：6.0

波 速 比：0.45

水氦含量：0.43

地形改变：0.60

4. 框架表示法 - 实例



4. 框架表示法 - 实例

(1) 结构性

便于表达结构性知识，能够将知识的内部结构关系及知识间的联系表示出来。

(2) 继承性

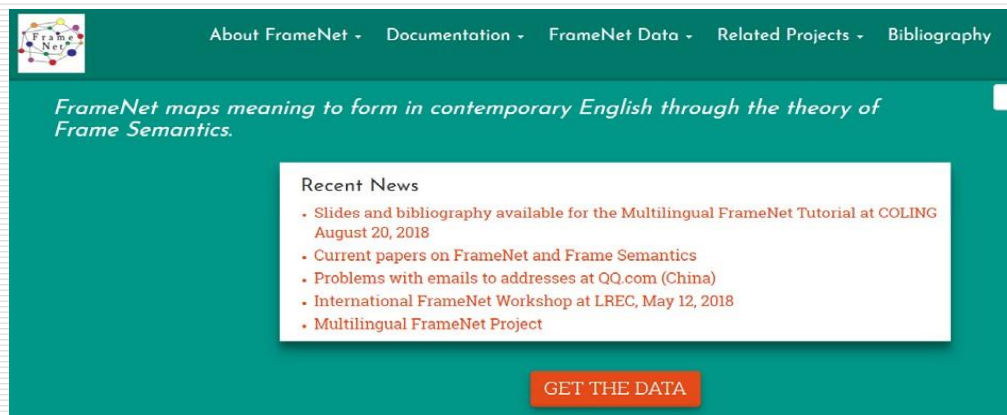
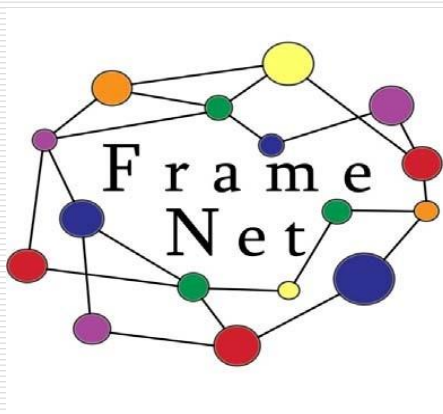
框架网络中，下层框架可以继承上层框架的槽值，也可以进行补充和修改。

(3) 自然性

框架表示法与人在观察事物时的思维活动是一致的。

典型的框架知识库：FrameNet

- FrameNet (<https://framenet.icsi.berkeley.edu/fndrupal>)
- 是目前最著名的语义知识框架库，由加州大学伯克利分校主持构建，旨在建立通用的语义知识库。目前该框架库中包括了1200多个语义框架，设计13000多个语义单元，并提供超过200万个标注框架的例句，广泛应用于信息提取，机器翻译，事件识别，情感分析等应用。



5. 语义网络表示法

- 语义网络是Quillian在1968年研究人类联想记忆时提出的心理学模型，认为记忆是由概念间的联系实现的。
1972年Simmons首先将语义网络表示法用于自然语言理解系统。
- 语义网络是知识的一种图解表示，它由结点和弧线组成。结点用于表示实体、概念和事件等，弧线用于表示结点间的关系。

5. 语义网络表示法

语义网络的组成部分

- 1. 词法部分** 决定该表示方法词汇表中允许有哪些符号，它涉及各个结点和弧线。
- 2. 结构部分** 叙述符号排列的约束条件，指定各弧线连接的结点对。
- 3. 过程部分** 说明访问过程，这些过程能用来建立和修正概念的描述，以及回答相关问题。
- 4. 语义部分** 确定与描述相关的意义的方法，即确定有关结点和对应弧线的排列及其相互关系。

5. 语义网络表示法

语义网络结构

语义网络是对知识的有向图表示方法。一个语义网络是由一些以三元组：**（结点1，弧，结点2）**的图形表示连接而成的有向图。

- 结点表示概念、事物、事件、情况等；
- 弧是有方向和有标注的，方向体现主次关系，结点1为主，结点2为辅。
- 弧上的标注表示结点1的**属性**或结点1和结点2之间的**关系**。

这样一个三元组的图形表示为：



5. 语义网络表示法

二元语义网络的表示

- 二元语义网络可以用来表示一些涉及变元的简单的事实，其实质还是一个三元组：(R, x, y)。
- 例如，表示“所有的燕子都是鸟”这一事实，可建立二个结点：SWALLOW 和 BIRD。两节点以ISA（表示“是一个”）链相连，如下图所示：



5. 语义网络表示法

二元语义网络的表示

对于事实“知更鸟是鸟，所有的鸟都有翅膀”，知更鸟、鸟以及翅膀分别用 **ROBIN**、**BIRD**及 **WINGS**表示。因为知更鸟是鸟的一部分，因此在**ROBIN**和**BIRD**之间用弧线连接，并加标记**AKO**（a kind of的缩写），以表示这种关系；又因为翅膀属于鸟的一个组成部分，所以在**BIRD**和**WINGS**之间也用弧线连接，并加标记**has-part**，其语义网络为：



5. 语义网络表示法

二元语义网络的表示

如果增添新的事实，只需在语义网络中增加新的结点和弧线就可以了。如果在上图所示的语义网络中，要增添事实：

“CLYDE是一个知更鸟，并且有一个叫做NEST-1的巢。”



5. 语义网络表示法

➤ 语义网络的特点

- 语义网络能把实体的结构、属性与实体间的因果关系显式地且简明地表达出来，这样可以联想方式实现对系统的解释。
- 由于与概念相关的属性和联系被组织在一个相应的结点中，因而语义网络使概念易于访问和学习。
- 语义网络表现问题更加直观，更易于理解，适合于知识工程师与领域专家沟通。

5. 语义网络表示法

➤ 语义网络的特点（续）

- 语义网络结构的语义解释依赖于该结构的推理过程而没有固定结构的约定，因而得到的推理不能保证像谓词逻辑法那样有效。
- 语义网络结点间的联系可能是线状、树状或网状的，甚至是递归状的结构，使相应的知识存储和检索需要比较复杂的过程。

Q/A?