NANJING UNIVERSITY

人工智能导论

知识表示

郭兰哲
南京大学 智能科学与技术学院
Homepage: www.lamda.nju.edu.cn/guolz
Email: guolz@nju.edu.cn

大纲

- 知识与知识表示
- 经典逻辑
- 产生式表示法
- 框架式表示法
- 状态空间表示法

大纲

- 知识与知识表示
- 经典逻辑
- 产生式表示法
- 框架式表示法
- 状态空间表示法

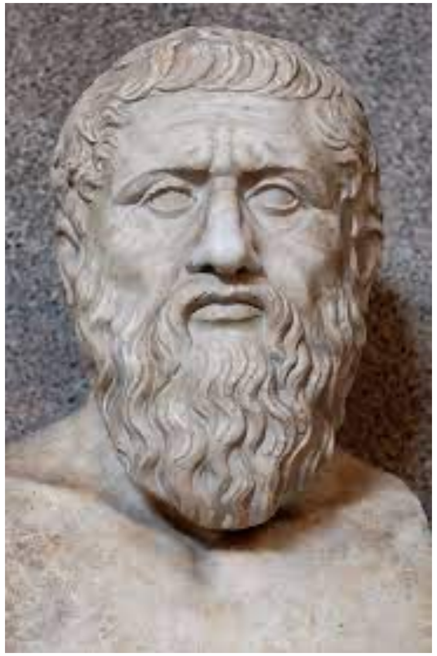
知识

人类的智能活动主要是获得并运用知识，

知识是智能的基础

✓ 知识：被证实的、真的、被相信的陈述

--柏拉图《泰阿泰德篇》



柏拉图
古希腊哲学家

知识

为了使计算机具有智能，能模拟人类的智能行为，必须使它具有知识

人工智能是关于知识的科学

- ✓ 如何获取知识
- ✓ 如何表示知识
- ✓ 如何应用知识

知识的概念

- 知识：人们在长期生活及社会实践中、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
- 知识：把有关信息关联在一起所形成的信息结构
- 知识反映了客观世界中事物之间的关系，不同事物或者相同事物间的不同关系形成了不同的知识

例如：

“如果头痛且流鼻涕，则很有可能患了感冒” 规则

“雪是白色的” 事实

知识的特性

➢ 相对正确性

任何知识都是在一定的条件及环境下产生的，在这种条件及环境下才是正确的

1+1=2　（十进制）
1+1=10　（二进制）

人工智能系统中，通常会将知识限定在所求解任务的范围内，因此知识的相对正确性更加突出

构建老鹰和狗分类的动物识别系统，知识“如果该动物会飞，则该动物是老鹰”，是正确的

知识的特性

➢ 不确定性

由于现实世界的复杂性，知识并不总是只有“真”和“假”两种状态，**在真与假之间存在着许多中间状态**

- **随机性引起的不确定性** —— “如果头痛且流涕，则有可能患了感冒”
- **模糊性引起的不确定性** —— “张三跑得很快”
- **经验性引起的不确定性** —— “老马识途”
- **不完全性引起的不确定性** —— “宇宙中没有外星人”

知识的特性

- ## 可表示性

知识可以用适当形式表示出来，如用语言、文字、图形等

- ## 可利用性

知识可以被利用

知识的表示

- **知识表示（knowledge representation）：将人类知识形式化或者模型化**

知识表示是对知识的一种描述，或者说是一组约定，一种计算机可以接受的用于描述知识的数据结构

常见的知识表示方法

✓ **经典逻辑**：采用命题逻辑、一阶逻辑等表示知识

✓ **产生式表示法**：用于表示事实、规则以及它们的不确定性度量

✓ **框架表示法**：结构化的知识表示方法，知识以框架的结构存储在记忆中

✓ **状态空间表示法**：利用状态向量和操作符号表示系统或问题的有关知识

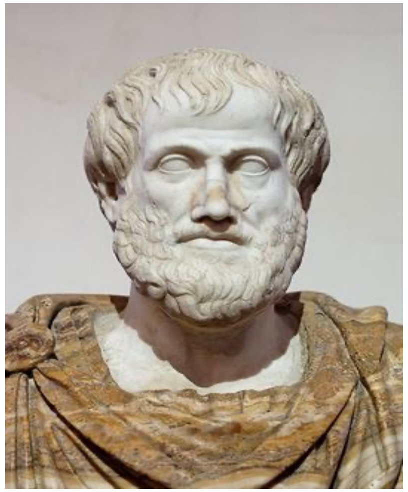
✓ **知识图谱**：以结构化的形式描述客观世界中概念、实体之间的复杂关系

大纲

- 知识与知识表示
- 经典逻辑
- 产生式表示法
- 框架式表示法
- 状态空间表示法

逻辑

- 逻辑本身诞生于古希腊哲学和数学
- 提出了演绎推理中“三段论”方法的古希腊学者亚里士多德被誉为“逻辑学之父”
- 一般而言，逻辑是用数学方法来研究关于推理和证明等问题的研究



亚里士多德
Aristotle
公元前384-前322
古希腊人

逻辑

- 逻辑是最方便的知识表示语言，符号主义人工智能（Symbolic AI)是人工智能研究的一种主流学派，在符号主义中，所有概念均可以通过人类可理解的符号及符号之间的关系来表示

- 如果使用符号A来表示对象概念、IsCar()表示某个对象是否为“汽车”，那么IsCar(A)表示“A是一辆汽车”这样的概念

- 符号主义人工智能假设可以通过逻辑方法对符号及其关系进行计算，实现逻辑推理，判断符号所描述内容是否正确

命题逻辑（Propositional Logic）

- 命题逻辑（propositional logic）是应用一套形式化规则对以符号表示的描述性陈述进行推理的系统

- 命题：命题是一个能够确定为真或为假的陈述句

- 命题通常用字母符号来表示，如p或者q，命题具有一个值，称为真值，真值有真或假两种，分别用符号T(True)和F(False)表示

命题逻辑（Propositional Logic）

例子：下列句子是否为命题：

1. 北京是中国的首都
2. 13能被6整除
3. $x + y \geq 18$
4. 存在最大的素数
5. $m^2 \geq 0$
6. 这座山真高啊

1和5为真命题，2和4为假命题，3和6不是命题

命题逻辑（Propositional Logic）

- **原子命题：** 原子命题指不包含其他命题作为其组成部分的命题，又称简单命题

可以通过命题联结词（connectives）对已有命题进行组合，得到新命题，称之为复合命题

- **复合命题：** 复合命题指包含其他命题作为其组成部分的命题，由简单命题通过逻辑联结词构造而成

例：

✓ 如果机房停电，那么今天的实验无法进行

✓ 图灵不仅是一名数学家，而且是一名逻辑学家

逻辑联结词

逻辑连接词	表示形式	意义
与（and）	$p \wedge q$	命题合取(conjunction)，即“p且q”
或（or）	$p \vee q$	命题析取(disjunction)，即“p或q”
非（not）	$\neg p$	命题否定(negation)，即“非p”
蕴含（implication）	$p \rightarrow q$	命题蕴含(implication)，即“如果p则q”
当且仅当（bi-implication)	$p \leftrightarrow q$	命题双向蕴含(bi-implication)，即“p当且仅当q”

运算规则

$\boldsymbol{p}$	**q**	$\neg p$	$p \wedge q$	$p \vee q$	$p \Rightarrow q$	$p \Leftrightarrow q$
False	**False**	True	False	False	True	True
False	**True**	True	False	True	True	False
True	**False**	False	False	True	False	False
True	**True**	False	True	True	True	True

运算规则

蕴含逻辑联结词中，不要求p和q之间存在相关性或因果关系

例子：5是奇数蕴含东京是日本的首都

当前提p为假时，无论q是真是假，命题都为真

例子：5是偶数蕴含大阪是日本的首都

逻辑等价

- **逻辑等价：**给定命题p和命题q，如果p和q在所有情况下都具有相同的真假结果，那么p和q在逻辑上等价，一般用≡来表示，即$p \equiv q$

例如，容易证明，$p \land q$和$q \land p$是等价的

逻辑等价的另一种定义方式：

任一两个命题是逻辑等价的，仅当它们互相蕴含时，

$$p \equiv q \text{ 当且仅当 } p \rightarrow q \text{ 且 } q \rightarrow p$$

逻辑等价

逻辑等价为命题进行形式转换带来了可能

$\alpha \wedge \beta \equiv \beta \wedge \alpha$ ($\wedge$的交换律)	$(\alpha \Rightarrow \beta) \equiv \neg\alpha \vee \beta$ **(蕴涵消除)**
$\alpha \vee \beta \equiv \beta \vee \alpha$ ($\vee$的交换律)	$(\alpha \Leftrightarrow \beta) \equiv (\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$(双向消除)
$(\alpha \wedge \beta) \wedge \gamma \equiv \alpha \wedge (\beta \wedge \gamma)$ ($\wedge$的结合律)	$\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta)$ (De Morgan)
$(\alpha \vee \beta) \vee \gamma \equiv \alpha \vee (\beta \vee \gamma)$ ($\vee$的结合律)	$\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta)$ (De Morgan)
$\neg(\neg\alpha) \equiv \alpha$ (双重否定)	$(\alpha \wedge (\beta \vee \gamma)) \equiv (\alpha \wedge \beta) \vee (\alpha \wedge \gamma)$ ($\wedge$ 对 $\vee$的分配律)
$(\alpha \Rightarrow \beta) \equiv \neg\beta \Rightarrow \neg\alpha$(逆否命题)	$(\alpha \vee (\beta \wedge \gamma)) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$ ($\vee$ 对 $\wedge$的分配律)

推理规则

假言推理 (Modus Ponens)	$\dfrac{\alpha \Rightarrow \beta,\ \ \alpha}{\beta}$
与消解 (And-Elimination)	$\dfrac{\alpha_1 \wedge \alpha_2 \wedge \cdots \wedge \alpha_n}{\alpha_i (1 \le i \le n)}$
与导入 (And-Introduction)	$\dfrac{\alpha_1, \alpha_2, \ldots, \alpha_n}{\alpha_1 \wedge \alpha_2 \wedge \cdots \wedge \alpha_n}$

推理规则

双重否定 (Double-Negation Elimination)	$\dfrac{\neg\neg\alpha}{\alpha}$
单项消解或单项归结 (Unit Resolution)	$\dfrac{\alpha \vee \beta, \neg\beta}{\alpha}$
消解或归结 (Resolution)	$\dfrac{\alpha \vee \beta, \neg\beta \vee \gamma}{\alpha \vee \gamma}$

应用归结法进行证明

例1：已知如下命题成立，

请证明命题γ是成立的

（1）$\alpha \vee \beta$

（2）$\alpha \rightarrow \gamma$

（3）$\beta \rightarrow \gamma$

- 已知 $\alpha \vee \beta$
- 对（2）进行蕴含消除： $\neg\alpha \vee \gamma$
- 对（3）进行蕴含消除： $\neg\beta \vee \gamma$
- （$\neg\alpha \vee \gamma$） $\wedge$ （$\neg\beta \vee \gamma$）
- 使用分配率： （$\neg\alpha \wedge \neg\beta$） $\vee \gamma$
- 使用De Morgan定律： $\neg(\alpha \vee \beta) \vee \gamma$
- γ

应用归结法进行证明

例2：请证明如下命题集是

不可满足的

（1）$\alpha \vee \beta$

（2）$\neg\alpha \vee \beta$

（3）$\alpha \vee \neg\beta$

（4）$\neg\alpha \vee \neg\beta$

- 已知 $\alpha \vee \beta$
- 已知$\neg\alpha \vee \beta$
- 推出：β
- 已知$\alpha \vee \neg\beta$
- 已知$\neg\alpha \vee \neg\beta$
- 推出：$\neg\beta$

应用归结法进行证明

例3：请证明如下命题集是不可满足的

（1）$\alpha \vee \gamma$

（2）$\neg\beta \vee \gamma$

（3）$\neg\gamma \vee \alpha$

（4）$\neg\alpha \vee \beta$

（5）$\neg\alpha \vee \neg\gamma$

- 已知 $\alpha \vee \gamma$
- 已知$\neg\gamma \vee \alpha$
- 推出：α
- 已知$\neg\alpha \vee \beta$
- 已知α
- 推出：β
- 已知$\neg\beta \vee \gamma$
- 已知β
- 推出：γ
- 已知$\neg\alpha \vee \neg\gamma$
- 推出：$\neg\alpha$

命题范式

范式（normal form）把命题公式化归为一种标准的形式

- ✓ 有限个简单合取式构成的析取式称为析取范式
- ✓ 由有限个简单析取式构成的合取式称为合取范式
- ✓ 析取范式与合取范式统称为范式

◆ 假设$\alpha_i (i = 1,2,\ldots,k)$为简单的合取式，则$\alpha = \alpha_1 \vee \alpha_2 \vee \cdots \vee \alpha_k$为析取范式

例如：$(\neg\, \alpha_1 \wedge \alpha_2) \vee \alpha_3, \neg\, \alpha_1 \vee \alpha_3 \vee \alpha_2$ 等

◆ 假设$\alpha_i (i = 1,2,\ldots,k)$为简单的析取式，则$\alpha = \alpha_1 \wedge \alpha_2 \wedge \cdots \wedge \alpha_k$为合取范式

例如：$(\alpha_1 \vee \alpha_2) \wedge \neg\alpha_3, \neg\, \alpha_1 \wedge \alpha_3 \wedge (\neg\alpha_2 \vee \alpha_4)$等

命题范式

- 一个析取范式是不成立的，当且仅当它的每个简单合取式都不成立
- 一个合取范式是成立的，当且仅当它的每个简单析取式都是成立的
- 任一命题公式都存在着与之等值的析取范式与合取范式(注意：命题公式的析取范式与合取范式都不是唯一的)

例：求$\neg(\alpha \to \beta) \vee \neg\gamma$的析取范式与合取范式

$$\neg(\alpha \to \beta) \vee \neg\gamma$$

$$\Leftrightarrow \neg(\neg\alpha \vee \beta) \vee \neg\gamma$$

$$\Leftrightarrow (\alpha \wedge \neg\beta) \vee \neg\gamma \text{ (析取范式)}$$

$$\Leftrightarrow (\alpha \vee \neg\gamma) \wedge (\neg\beta \vee \neg\gamma) \text{ (合取范式)}$$

从命题逻辑到谓词逻辑

- 命题逻辑的局限性：在命题逻辑中，每个陈述句是最基本的单位(即原子命题)，无法对原子命题进行分解，因此在命题逻辑中，不能表达局部与整体、一般与个别的关系

- 例如，对于苏格拉底论断

✓ α：所有的人总是要死的

✓ β：苏格拉底是人

✓ γ：所以苏格拉底是要死的

无法在命题逻辑基础上推导出：$\alpha \land \beta \rightarrow \gamma$

虽然是正确的，但是无法通过命题逻辑来进行推理判断

从命题逻辑到谓词逻辑

- α：大象<u>是</u>哺乳动物
- β：大象<u>是</u>一种<u>最大的</u>哺乳动物

不同原子命题蕴含个体、群体和关系等内在丰富语义，命题逻辑无法表现内在丰富语义。

因此，需要分析原子命题，分离其主语（个体或群体）和谓语（关系）

需要引入更加强大的逻辑表示方法，谓词逻辑

谓词逻辑

在谓词逻辑中，将原子命题进一步细化，分解出个体、谓词和量词，来表达个体与总体的内在联系和数量关系

谓词逻辑中三个核心概念：

- 个体、谓词（predicate）和量词（quantifier）

谓词逻辑

- 个体：个体是指所研究领域中可以独立存在的具体或抽象的概念

具体或特定的个体：“张三”“李四”“1”“2”“3”

抽象或泛指的个体：x，y，z

谓词逻辑

- 谓词：谓词是用来**刻画个体属性**或者**描述个体之间关系存在性**的元素，其值为真或为假

  - 包含一个参数的谓词称为一元谓词，表示一元关系，通常是用来刻画个体是否包含特定的属性：如$p(x)$: x是质数，表示某个数字是否是质数
  - 包含多个参数的谓词称为多元谓词，表示个体间的多元关系，通常用于描述个体之间是否存在特定的关联，如$\mathrm{Father}(x, y)$: 表示x是y的父亲

谓词逻辑：谓词与函数的区别

- 函数中个体变元用个体常量（来自定义域）代入后结果仍然是个体（值域），如定义函数$f(x)=x+10$，则$f(2)=12$

- 谓词中个体变元用个体常量带入后就变成了命题，　如$p(x)\colon x$是质数，带入$x=3$，$p(3)$是命题

- 函数是从定义域到值域的映射；谓词是从定义域到$\{\mathrm{Ture},\ \mathrm{False}\}$的映射

谓词逻辑：事实符号化

例：将下列客观事实符号化

1. Richard是国王。
2. Lucy和Lily是姐妹。
3. 北京是中国的首都。

(1) $King$(Richard)。其中，Richard是一个个体常量，$King$是一个描述“国王”这个一元关系的谓词。

(2) $Sister$(Lucy, Lily)。其中，Lucy和Lily是两个个体常量，$Sister$是一个描述“姐妹”这个二元关系的谓词。

(3) $Capital$(北京, 中国)。其中，北京和中国是两个个体常量，$Capital$是一个描述“首都”这个二元关系的谓词。

谓词逻辑：量词

全称量词 (universal quantifier, ∀)

全称量词用符号∀表示，表示一切的、凡是的、所有的、每一个等。

$\forall x$表示定义域中的所有个体， $\forall xP(x)$表示定义域中的所有个体具有性质P

存在量词(existential quantifier, ∃)

存在量词用符号∃表示，表示存在、有一个、某些等。

$\exists x$表示定义域中存在一个或若干个个体，

$\exists xP(x)$表示定义域中存在一个个体或若干个体具有性质P

全称量词和存在量词统称为量词

谓词逻辑

全称量词与存在量词之间的关系

- $\forall x P(x) \equiv \neg \exists x \neg P(x)$
- $\forall x \neg P(x) \equiv \neg \exists x P(x)$
- $\neg \forall x P(x) \equiv \exists x \neg P(x)$
- $\exists x P(x) \equiv \neg \forall x \neg P(x)$

谓词逻辑

- **约束变元：**在全称量词或存在量词约束条件下的变量称为约束变元
- **自由变元：**不受全称量词或存在量词约束的变量称为自由变元

自由变元即可以存在于量词的约束范围之内，也可以存在于量词的约束范围之外

$$(\forall x)(A(x) \vee B) \equiv (\forall x)A(x) \vee B$$

$$(\forall x)(A(x) \wedge B) \equiv (\forall x)A(x) \wedge B$$

$$(\exists x)(A(x) \vee B) \equiv (\exists x)A(x) \vee B$$

$$(\exists x)(A(x) \wedge B) \equiv (\exists x)A(x) \wedge B$$

谓词逻辑

- 在约束变元相同的情况下，全称量词对合取满足分配律，存在量词对析取满足分配律

$$(\forall x)(A(x) \wedge B(x)) \equiv (\forall x)A(x) \wedge (\forall x)B(x)$$

$$(\exists x)(A(x) \vee B(x)) \equiv (\exists x)A(x) \vee (\exists x)B(x)$$

$$(\forall x)(A(x) \vee B(x)) \equiv (\forall x)A(x) \vee (\forall x)B(x)$$ （不成立）

$$(\exists x)(A(x) \wedge B(x)) \equiv (\exists x)A(x) \wedge (\exists x)B(x)$$ （不成立）

谓词逻辑

例：使用全称量词或存在量词描述下列事实。

（1）所有的国王都是人。

（2）所有的国王都头戴皇冠。

（1）“所有的国王都是人”表示的含义为“**对于所有的x，如果x是国王，那么x是人**”，其符号化表示为$(\forall x)(King(x) \rightarrow Person(x))$。其中$x$是变量符号，由于$x$受到全称量词的约束，因此$x$是约束变元；$King(x)$是一个一元谓词，表示$x$是国王，$Person(x)$是一个一元谓词，表示$x$是人。

（2）“所有的国王都头戴皇冠”表示的含义为“**对于所有的x，如果x是国王，那么x头戴皇冠**”，符号化表示为$(\forall x)(King(x) \rightarrow Head_On(Crown, x))$。其中$x$是变量符号，由于$x$受到全称量词的约束，因此$x$是约束变元；Crown是一个常量符号，表示皇冠；$King(x)$是一个一元谓词，表示$x$是国王，$Head_On(\mathrm{Crown}, x)$是一个二元谓词，表示$x$头戴皇冠。

谓词逻辑

- 当公式中存在多个量词时：
  - ✓ 若多个量词都是全称量词或者都是存在量词，则量词的位置可以互换
  - ✓ 若多个量词中既有全称量词又有存在量词，则量词的位置不可以互换

设$A(x,y)$是包含变元x,y的谓词公式，则如下关系成立：

$$(\forall x)(\forall y)A(x,y) \Leftrightarrow (\forall y)(\forall x)A(x,y)$$

$$(\exists x)(\exists y)A(x,y) \Leftrightarrow (\exists y)(\exists x)A(x,y)$$

$$(\forall x)(\forall y)A(x,y) \Rightarrow (\exists y)(\forall x)A(x,y)$$

$$(\forall x)(\forall y)A(x,y) \Rightarrow (\exists x)(\forall y)A(x,y)$$

$$(\exists y)(\forall x)A(x,y) \Rightarrow (\forall x)(\exists y)A(x,y)$$

$$(\exists x)(\forall y)A(x,y) \Rightarrow (\forall y)(\exists x)A(x,y)$$

$$(\forall x)(\exists y)A(x,y) \Rightarrow (\exists y)(\exists x)A(x,y)$$

$$(\forall y)(\exists x)A(x,y) \Rightarrow (\exists x)(\exists y)A(x,y)$$

谓词逻辑：项与原子谓词公式

- **项：** 项是描述对象的逻辑表达式，被递归地定义为：

（1）常量符号和变量符号是项

（2）若$f(x_1, x_2, \cdots, x_n)$是n元函数符号， $t_1, t_2, \cdots, t_n$是项，则$f(t_1, t_2, \cdots, t_n)$是项

（3）有限次数地使用上述规则产生的符号串是项

- **原子谓词公式：** 若$P(x_1, x_2, \cdots, x_n)$是n元谓词， $t_1, t_2, \cdots, t_n$是项，则称$P(t_1, t_2, \cdots, t_n)$是原子谓词公式，简称原子公式。

谓词逻辑：合式公式

- **合式公式：** 合式公式是由逻辑联结词和原子公式构成的用于陈述事实的复杂语句，又称谓词公式

例： 使用合式公式描述下列事实。

（1）Tom不仅喜欢踢足球，还喜欢打篮球

（2）Tom的所有同学都喜欢他

（3）不是所有的男生都喜欢打篮球

（4）一个人是华侨，当且仅当他在国外定居并且具有中国国籍

谓词逻辑：合式公式

(1) $Like(\mathrm{Tom}, \mathrm{Football}) \land Like(\mathrm{Tom}, \mathrm{Basketball})$。其中，$Like(\mathrm{Tom}, x)$表示Tom喜欢$x$，Football和Basketball分别表示踢足球和打篮球。

(2) $(\forall x)(Classmate(\mathrm{Tom}, x) \to Like(x, \mathrm{Tom}))$。其中，$Classmate(\mathrm{Tom}, x)$表示$x$是Tom的同学，$Like(x, \mathrm{Tom})$表示$x$喜欢Tom。

(3) $\neg(\forall x)(Boy(x) \to Like(x, \mathrm{Basketball}))$。其中，$Boy(x)$表示$x$是男生，$Like(x, \mathrm{Basketball})$表示$x$喜欢打篮球。

(4) $(\forall x)(Overseas\_Chinese(x) \leftrightarrow (Overseas(x) \land Chinese(x)))$。其中，$Overseas_Chinese(x)$表示$x$是华侨，$Overseas(x)$表示$x$在国外定居，$Chinese(x)$表示$x$具有中国国籍。

谓词逻辑：推理规则

设$A(x)$是谓词公式， x和y是变元， a是常量符号，则存在如下谓词逻辑中的推理规则：

- 全称量词消去（Universal Instantiation, UI）： $(\forall x)A(x) \rightarrow A(y)$
- 全称量词引入（Universal Generalization, UG）： $A(y) \rightarrow (\forall x)A(x)$
- 存在量词消去（Existential Instantiation, EI）： $(\exists x)A(x) \rightarrow A(c)$
- 存在量词引入（Existential Generalization, EG）：$A(a) \rightarrow (\exists x)A(x)$

谓词逻辑：推理规则

例：已知

- $(\forall x)(P(x) \rightarrow Q(x))$
- $(\forall x)(Q(x) \rightarrow R(x))$

证明：$(\forall x)(P(x) \rightarrow R(x))$

证明过程：

- $(\forall x)(P(x) \rightarrow Q(x))$
- $P(x) \rightarrow Q(x)$（消去全称量词）
- $(\forall x)(Q(x) \rightarrow R(x))$
- $Q(x) \rightarrow R(x)$（消去全称量词）
- $P(x) \rightarrow R(x)$
- $(\forall x)(P(x) \rightarrow R(x))$（引入$x$）

谓词逻辑：推理规则

例：已知

- $(\forall x)(F(x) \to (G(x) \land H(x))$
- $(\exists x)(F(x) \land P(x))$

试证明：$(\exists x)\ (P(x) \land H(x))$

证明过程：

1. $(\forall x)(F(x) \to (G(x) \land H(x))$（已知）
2. $(\exists x)(F(x) \land P(x))$（已知）
3. $F(a) \land P(a)$（2的EI）
4. $F(a) \to (G(a) \land H(a))$（1的UI）
5. $F(a)$（由3知）
6. $G(a) \land H(a)$（4和5的假言推理）
7. $P(a)$（由3知）
8. $H(a)$（由6知）
9. $P(a) \land H(a)$（7和8的合取）
10. $(\exists x)\ (P(x) \land H(x))$（9的EG）

自然语言的形式化

为了对自然语言所描述的内容进行精确表达，可通过**逻辑语言**

对自然语言进行形式化

例：每一个奇数均存在一个大于它的奇数：

- $\mathrm{odd}(x)$： x是奇数
- $\mathrm{Great(x, y)}$： x 大于y
- $(\forall x)\left(odd(x) \rightarrow (\exists y)\left(odd(y) \wedge Great(y, x)\right)\right)$

自然语言的形式化

例：证明苏格拉底三段论“所有人都是要死的，苏格拉底是人，所以苏格拉底是要死的”。

设$F(x)$：x是人，$G(x)$：x是要死的，a是苏格拉底

前提：$(\forall x)(F(x) \rightarrow G(x))$，$F(a)$，结论：$G(a)$

证明：

（1）$(\forall x)(F(x) \rightarrow G(x))$

（2）$F(a) \rightarrow G(a)$（1的全称量词消去）

（3）$F(a)$

（4）$G(a)$（2和3的假言推理）

自然语言的形式化

例：前提：1）每驾飞机或者停在地面或者飞在天空；2）并非每驾飞机都飞在天空

结论：有些飞机停在地面

形式化：$plane(x)$：x是飞机；$in\_ground(x)$：x停在地面；$on\_fly(x)$：x飞在天空

已知：$(\forall x)(plane(x)\rightarrow in\_ground(x) \vee on\_fly(x))$，$(\neg\forall x)(plane(x)\rightarrow on\_fly(x))$

请证明：$(\exists x)(plane(x)\wedge in\_ground(x))$

证明：

1. $(\neg\forall x)(plane(x) \rightarrow on\_fly(x))$（已知）
2. $(\exists x)\neg(plane(x) \rightarrow on\_fly(x)$
3. $(\exists x)\neg(\neg plane(x) \vee on\_fly(x))$
4. $(\exists x)(plane(x) \wedge \neg on\_fly(x))$

自然语言的形式化

1. $(\neg\forall x)(plane(x) \rightarrow on\_fly(x))$
2. $(\exists x)\neg(plane(x) \rightarrow on\_fly(x))$
3. $(\exists x)\neg(\neg plane(x) \vee on\_fly(x))$
4. $(\exists x)(plane(x) \wedge \neg on\_fly(x))$
5. $plane(a) \wedge \neg on\_fly(a)$
6. $plane(a)$
7. $\neg on\_fly(a)$
8. $(\forall x)(plane(x) \rightarrow in\_ground(x) \vee \mathrm{on\_fly}(x))$
9. $plane(a) \rightarrow in\_ground(a) \vee \mathrm{on\_fly}(a)$
10. $in\_ground(a) \vee \mathrm{on\_fly}(a)$
11. $in\_ground(a)$
12. $plane(a) \wedge in\_ground(a)$
13. $(\exists x)(plane(x) \wedge in\_ground(x))$

大纲

- 知识与知识表示
- 逻辑知识表示
- 产生式表示法
- 框架式表示法
- 状态空间表示法

产生式表示法

又称为产生式规则（production rule）表示法

通常用于表示事实、规则及它们的不确定性，适合于表示事实的知识和规则性知识

- 确定性规则的产生式表示

IF P THEN Q

或者 P -> Q

例如：IF $\underbrace{\text{动物会飞 AND 会下蛋}}_{P}$ THEN $\underbrace{\text{该动物是鸟}}_{Q}$

产生式表示法

- 不确定性规则的产生式表示

IF **P** THEN **Q** （置信度）

或者 **P** -> **Q** （置信度）

例如：IF $\underbrace{\text{头痛 AND 流鼻涕}}_{P}$ THEN $\underbrace{\text{患了感冒}}_{Q}$ $\underbrace{(0.7)}_{\text{置信度}}$

产生式表示法

- 确定性事实的产生式表示

一般用三元组表示：（对象，属性，值）

或者 （关系，对象1，对象2）

例如：

- ✓ 老李年龄是四十岁：（Li, Age, 40）
- ✓ 老李和老王是朋友：（Friends, Li, Wang）

产生式表示法

➢ 不确定性事实的产生式表示

一般用四元组表示：（对象，属性，值，置信度）

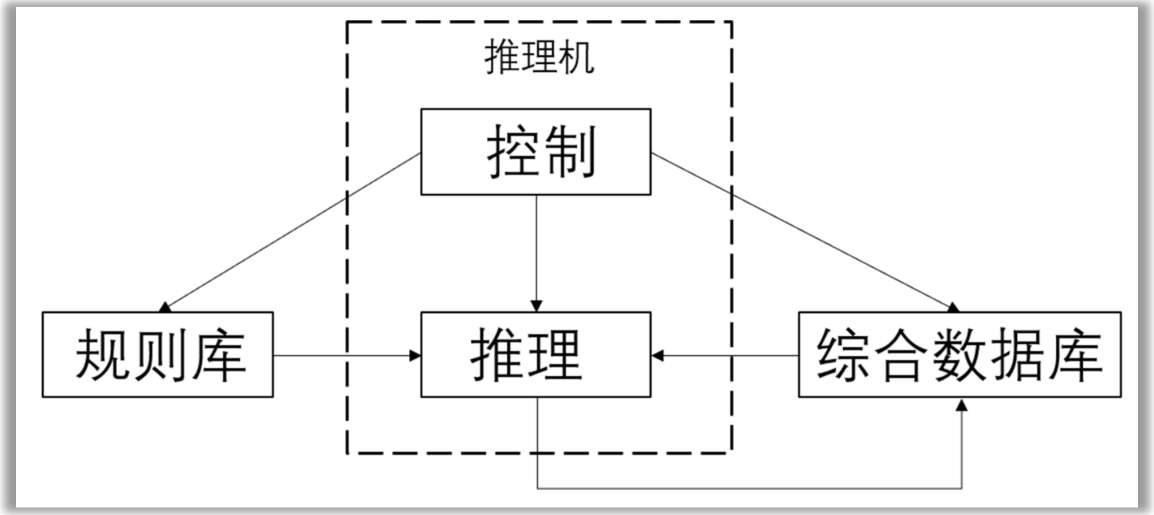
或者（关系，对象1，对象2，置信度）

例如：

✓ 老李年龄很可能是四十岁：（Li, Age, 40, 0.8）

✓ 老李和老王不大可能是朋友：（Friends, Li, Wang, 0.1）

产生式系统



- **规则库**：用于描述某领域内知识的产生式集合，是某领域知识（规则）的存储器，其中的规则是以产生式形式表示的。规则库中包含着将问题从初始状态转换成目标状态（或解状态）的那些变换规则
- **综合数据库**：又称为事实库，用于存放输入的事实、从外部数据库输入的事实以及中间结果（事实）和最后结果的工作区
- **推理机**：是一个或一组程序，用来控制和协调规则库与综合数据库的运行，包含了推理方式和控制策略。控制策略的作用就是确定选用什么规则或如何应用规则。

产生式系统

动物识别系统——识别虎、金钱豹、斑马、长颈鹿、鸵鸟、企鹅、信天翁七种动物的产生式系统







产生式系统

➢规则库：

r_1： IF 该动物有毛发 THEN 该动物是哺乳动物
r_2： IF 该动物胎生 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

产生式系统

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

产生式系统

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，胎生，蹄

➢ 推理机的工作过程

（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，

综合数据库中：

该动物身上有：暗斑点，长脖子，长腿，胎生，蹄，哺乳动物

产生式系统

- 推理机的工作过程

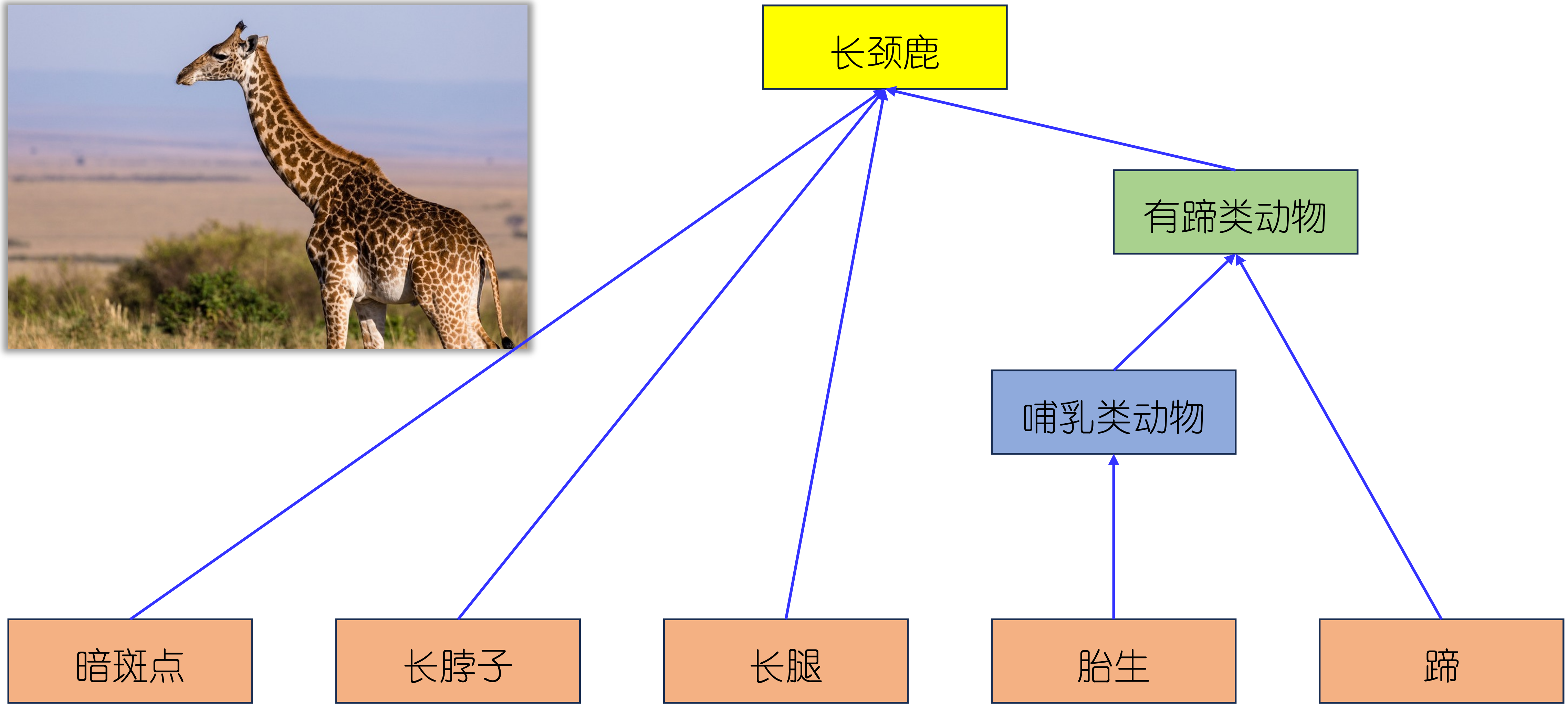
（2）分别用r3，r4，r5，r6综合数据库中的已知事实进行匹配，均不成功。 r7匹配成功，执行r7

综合数据库：

该动物身上有：暗斑点，长脖子，长腿，胎生，蹄，哺乳动物，有蹄类动物

（3）r11匹配成功，并推出 “该动物是长颈鹿”。

产生式系统的例子 － 动物识别系统



大纲

- 知识与知识表示
- 逻辑知识表示
- 产生式表示法
- 框架式表示法
- 状态空间表示法

框架式表示法

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（frame）：是一种描述所论对象（一个事物、事件或概念）属性的数据结构。

在框架中，新的资料可以用经验中得到的概念来分析和解释。因此，框架也是一种结构化表示法

框架的一般结构

一个框架由若干个被称为“槽”（slot）的结构组成，一个槽用于描述对象某一方面的属性

每一个槽可划分为若干个“侧面”（facet），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称为槽值和侧面值。

对于框架、槽或侧面，可以加上一些约束条件，比如什么样的值才能填入槽和侧面中去。

框架的一般结构

〈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$_n$

用框架表示知识的例子

➢ 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，助理教授）

　　部门：单位（学校、教研室）

　　住址：〈住址框架〉

　　开始工作时间：单位（年、月）

　　截止时间：单位（年、月），缺省：现在

用框架表示知识的例子

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事例框架

框架名：〈教师-1〉

姓名：夏冰

年龄：36

性别：女

职称：教授

部门：计算机系软件所

住址：江苏省南京市

开始工作时间：2010

截止时间：现在

用框架表示知识的例子

- 例2：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，水氡含量为0.43，地形改变为0.60。”

框架名：〈地震〉

地　　点：某地

日　　期：某年某月某日

震　　级：6.0

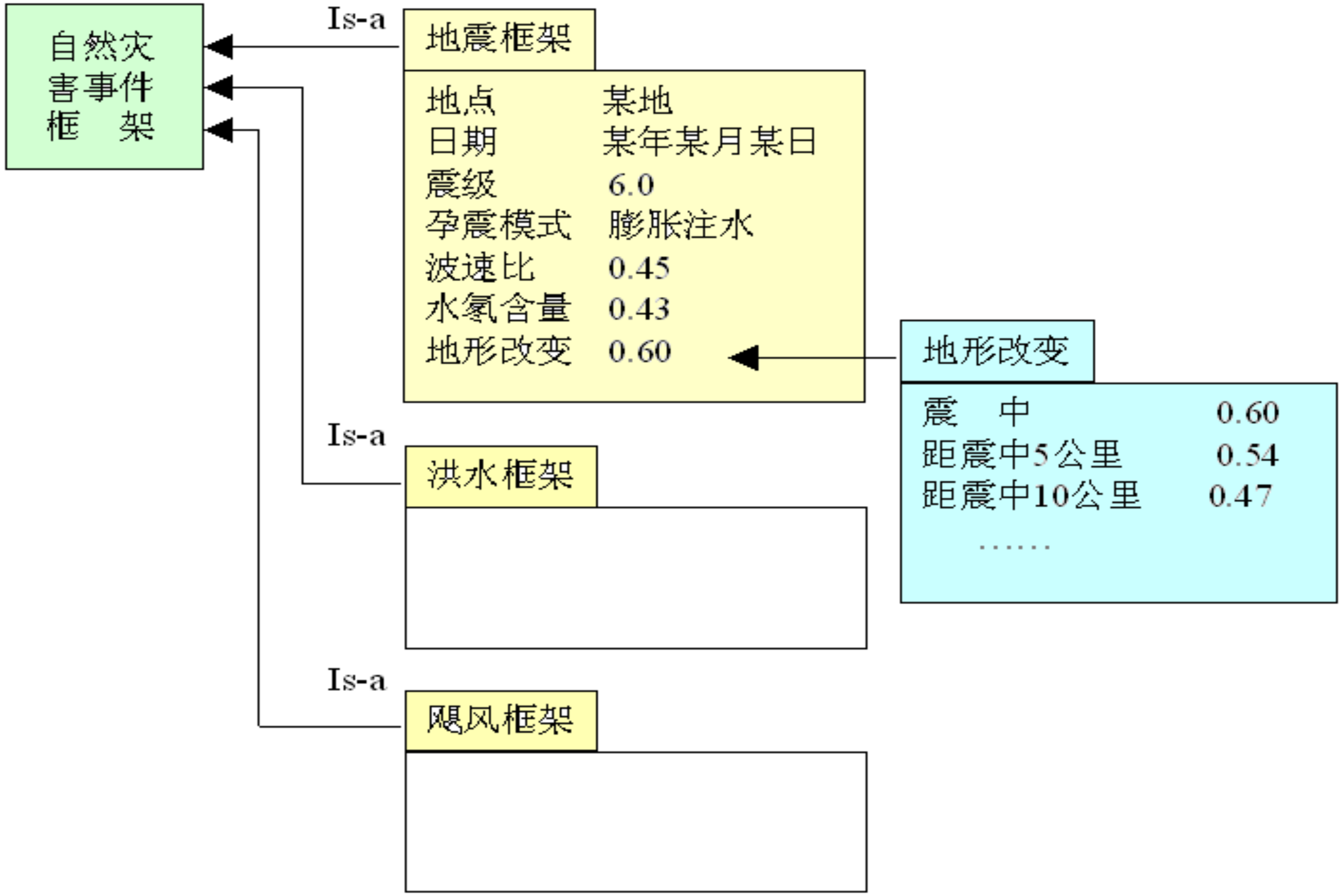
波 速 比：0.45

水氡含量：0.43

地形改变：0.60

用框架表示知识的例子

➢ 例3：自然灾害事件框架，通过框架表示法，可以建立复杂知识的框架网络



大纲

- 知识与知识表示
- 逻辑知识表示
- 产生式表示法
- 框架式表示法
- 状态空间表示法

状态空间表示法

状态空间（state space）是利用状态变量和操作符号表示系统或问题有关知识的符号体系，可以用一个四元组表示

（状态集合 S，操作算子集合 O，初始状态 S_0，目标状态 G）

例：八数码问题

2	8	3
1	6	4
7		5

初始状态

1	2	3
8		4
7	6	5

目标状态

- ✓ 任何一种摆法都是一种状态，所有的摆法构成状态集 S
- ✓ 初始状态记作 S_0
- ✓ 目标状态记作 G
- ✓ 操作算子集合记作 O

状态空间的图描述

状态空间可以用有向图来表示，图的节点表示问题的状态，边表示状态之间的关系。