教学安排（星期四，08:00-09:35，六教6A116）

周次	日期	内容	作业	编程	项目	内容
1	2月29日	绪论				绪论，课程内容
2	3月 7日	无信息搜索		1		状态空间表示，宽度优先，深度优先
3	3月14日	有信息搜索（1）			1	算法复杂度分析，一致代价，贪婪最佳优先搜索
4	3月21日	有信息搜索（2）	1			A* 算法
5	3月28日	约束满足	2			约束满足问题，回溯法，局部搜索
6	4月 4日	清明节假期				
7	4月11日	对抗搜索	3			博弈树搜索，蒙特卡洛树搜索
8	**4月18日**	**逻辑推理**	**4**			**归结原理**
9	4月25日	线性回归			2	线性回归
11	5月 9日	Logistic 回归	5			Logistic 回归
11	5月11日	前馈神经网络		2		前馈神经网络，机器学习项目
12	5月16日	卷积神经网络	6			卷积神经网络
13	5月23日	马尔可夫决策过程	7			强化学习的数学基础
14	5月30日	策略迭代与价值迭代		3		状态转移概率已知时的预测与控制
15	6月 6日	蒙特卡洛与时序差分	8			状态转移概率未知时的预测与控制
16	6月13日	深度强化学习				价值函数的近似，深度强化学习，考试解读

问题求解

逻辑推理

机器学习

人工智能原理

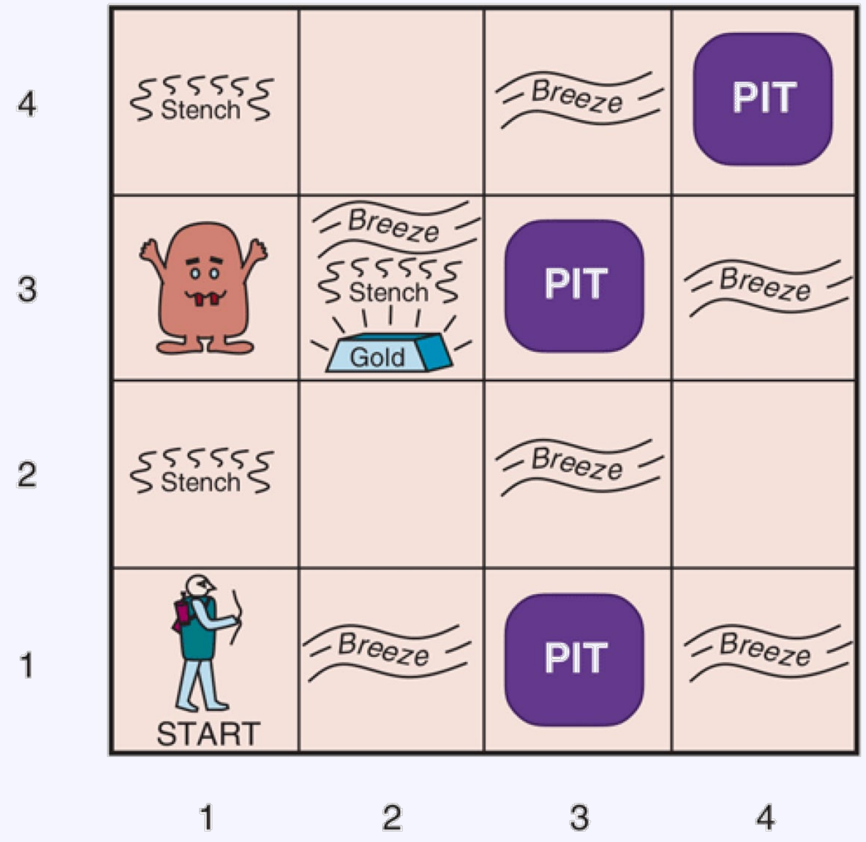
Principles of Artificial Intelligence

乔晖

自动化系

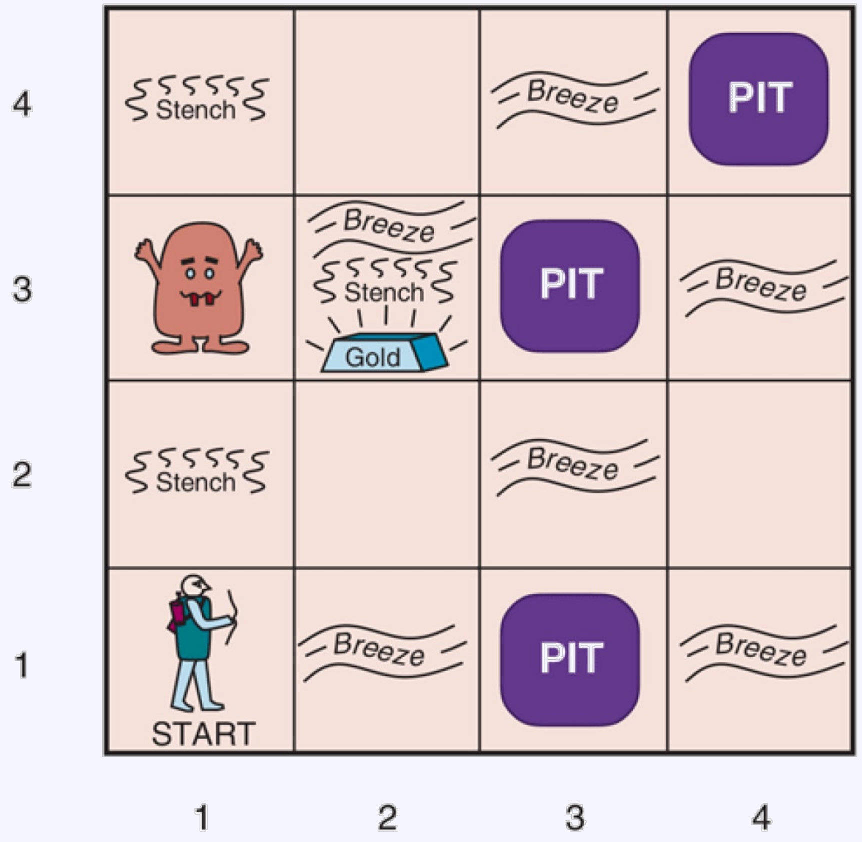
怪兽游戏 (Wumpus)

- 探险者进入一个山洞，目的是找到**金子**（获得1000分）
- 山洞分割为4x4的区域，有如下设置：
  - **金子**：所在区域会有闪光，找到金子获得1000分
  - **陷阱**：周围区域会有凉风，掉进陷阱扣除1000分
  - **怪兽**：周围区域会有臭味，被怪兽吃掉扣1000分
    可以有一次机会射杀怪兽，获得1000分
- 每次可以横向或纵向走一步（消耗1000分）
- 怪兽、陷阱和金子的位置事先并不知道
- 可以感知一些周围的情况，包括：
  - 怪兽附近区域的臭味：$x_1 = \{\textbf{Stench}, \quad \textbf{None}\}$
  - 陷阱相邻区域的凉风：$x_2 = \{\textbf{Breeze}, \quad \textbf{None}\}$
  - 金子所在区域的闪光：$x_3 = \{\textbf{Golden}, \quad \textbf{None}\}$
  - 撞墙时发出的撞击声：$x_4 = \{\textbf{Wall}, \quad \textbf{None}\}$
  - 怪兽被射杀时的嚎叫：$x_5 = \{\textbf{Howl}, \quad \textbf{None}\}$
- **寻找一条找到金子的路径**



搜索求解

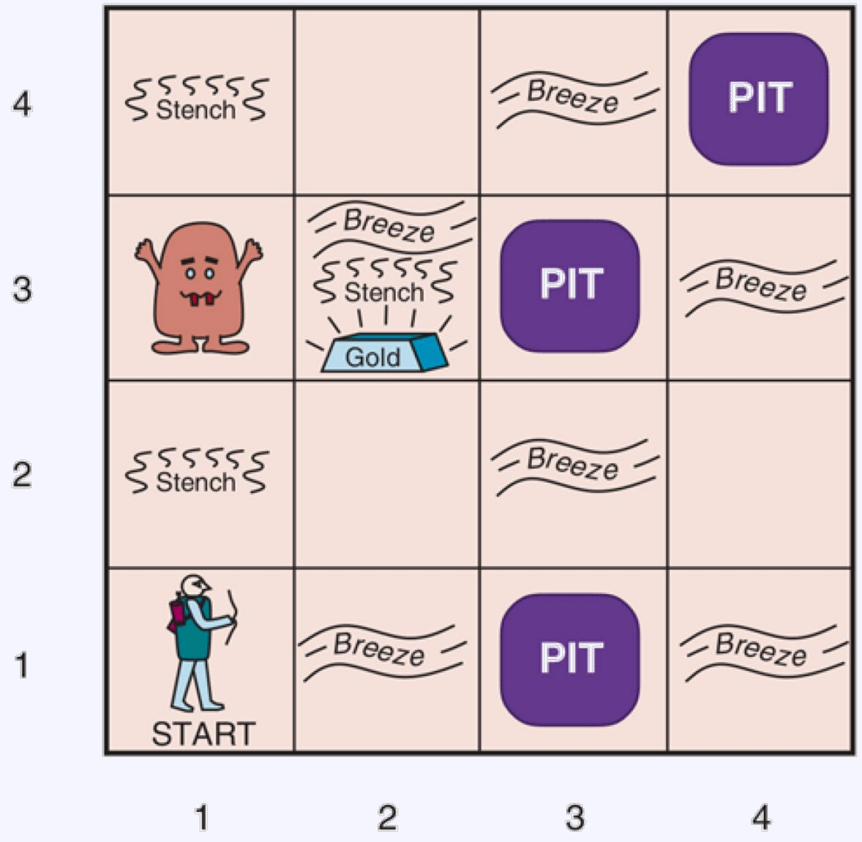
- **完全可观测：怪兽、陷阱和金子的位置事先知道**
- 与迷宫问题并无多少不同
- 状态空间表示
  - 状态：所处位置（原子表示法）
  - 行动：横向或纵向移动一格
  - 代价：移动一格：1000　　通向金子：0
  - 　　　　通向陷阱：2000　　通向怪兽：2000
- 无信息或有信息搜索经典算法求解
- 可能存在的问题
  - 难以利用环境信息（原子表示法）



搜索求解

- **部分可观测：怪兽、陷阱和金子的位置并不知道**

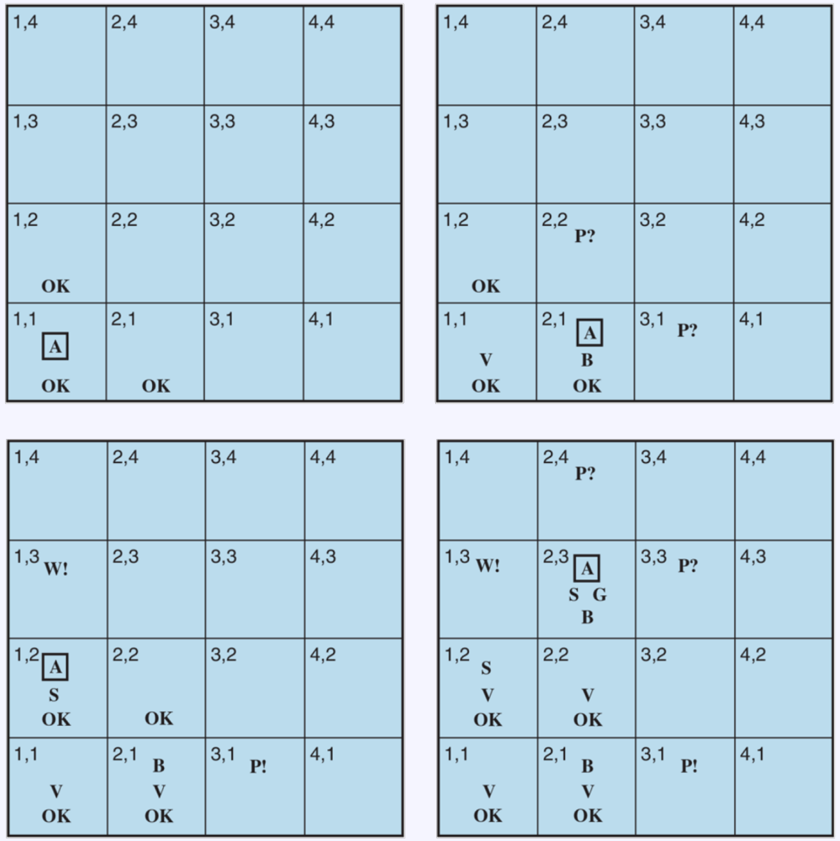
- 状态空间表示
  - 状态：所处位置（原子表示法）
  - 行动：横向或纵向移动一格
  - 代价：移动一格：1000　　通向金子：0
  - 　　　通向陷阱：2000　　通向怪兽：2000
- 无信息或有信息搜索经典算法求解
- 可能存在的问题
  - 难以利用环境信息（原子表示法）
  - 难以避免落入陷阱（位置不知道）
  - 难以设计启发函数（位置不知道）

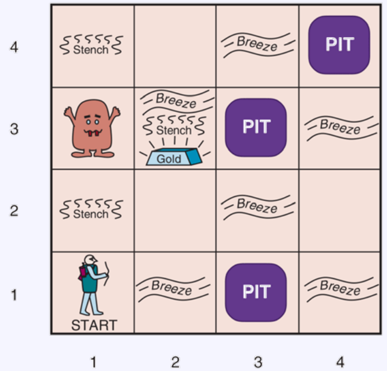


推理求解

- (1,1): [N, N, N, N, N] ⇒ (1,1) OK, (2,1) OK, (1,2) OK
- (2,1): [N, **B**, N, N, N] ⇒ (3,1) P?, (2,2) P?
- (1,2): [**S**, N, N, N, N] ⇒ (2,2) OK, (1,3) W!
- (2,2): [N, N, N, N, N] ⇒ (3,2) OK, (2,3) OK
- (3,2): [N, **B**, N, N, N] ⇒ (4,2) P?, (3,3) P?, (3,1) P?
- (2,3): [N, N, **G**, N, N] ⇒ GOAL

- 可以感知一些周围的情况，包括：
  - 怪兽附近区域的臭味：$x_1 = \{\textbf{Stench}, \quad \textbf{None}\}$
  - 陷阱相邻区域的凉风：$x_2 = \{\textbf{Breeze}, \quad \textbf{None}\}$
  - 金子所在区域的闪光：$x_3 = \{\textbf{Golden}, \quad \textbf{None}\}$
  - 撞墙时发出的撞击声：$x_4 = \{\textbf{Wall}, \quad \textbf{None}\}$
  - 怪兽被射杀时的嚎叫：$x_5 = \{\textbf{Howl}, \quad \textbf{None}\}$





逻辑推理

命题逻辑

- 描述事实是否成立

谓词逻辑

- 描述关系是否成立

逻辑

演绎推理

- 假言推理
- 前向链接
- 后向链接

归结原理

- 合一置换
- 归结原理

命题逻辑

Propositional Logic

语法 Syntax

- 关于事实的陈述称为**命题**
- 命题必须是**陈述句**
- **特殊命题**：True, False
- **原子命题**：对单个事实的陈述
  - 北京是中国的首都
  - 煤是白的
- **复合命题**：用**逻辑连接符**相连的多个原子命题
  - $\neg$ (not) **否定**
  - $\wedge$ (and) **合取**
  - $\vee$ (or) **析取**
  - $\Rightarrow$ (implies) **隐含**
  - $\Leftrightarrow$ (if and only if) **等价**

5种连接符

- 否定式 (negation): $\neg P$
  - P: Positive literal
  - $\neg P$: Negative literal
  - P, $\neg P$: Literal
- 合取式 (conjunction): $P \wedge Q$
  - P, Q: conjuncts
- 析取式 (disjunction): $P \vee Q$
  - P, Q: disjuncts
- 隐含式 (implication): $P \Rightarrow Q$
  - P: premise, antecedent
  - Q: conclusion, consequent
  - Implication: rule, if-then
- 等价式 (bidirectional): $P \Leftrightarrow Q$

语义 Semantics

- 命题有 **真 (TRUE)** 和 **假 (FALSE)** 两种可能取值，称为其 **真值 (Truth)**
- 特殊命题：真值由定义给出
  - True：　在任何模型中都是 TRUE（模型：对原子语句真值的指定）
  - False ：　在任何模型中都是 FALSE
- 原子命题：真值由模型给出
  - 模型 m:　$P = \text{TRUE},\ Q = \text{TRUE}$
- 复合语句：真值通过计算获得
  - 语句 α:　$P \wedge Q$
  - m 满足 α: 在模型 m 下，α 的真值为 TRUE

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

隐含式

- **真推不出假**
- 前项和后项之间**不要求存在因果关系**

5是奇数 $\Rightarrow$ 东京是日本的首都

逻辑蕴涵

- **如果在所有使语句 α 为真的模型中，语句 β 也为真，则称 α 蕴涵 β (α entails β)，记为 $\alpha \vDash \beta$**
  - 所有使语句 α 为真的模型记为 $M(\alpha)$
  - 所有使语句 β 为真的模型记为 $M(\beta)$，则

$$\alpha \vDash \beta \text{ if and only if } M(\alpha) \subseteq M(\beta)$$

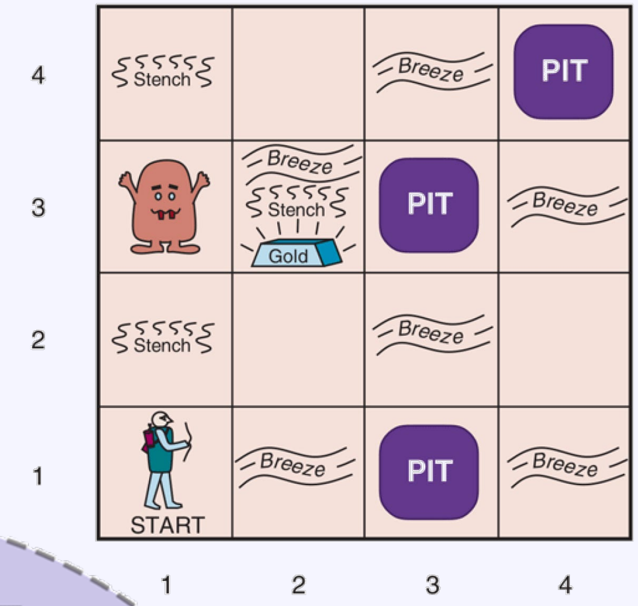
- 如果 α 就是知识库 KB，那么

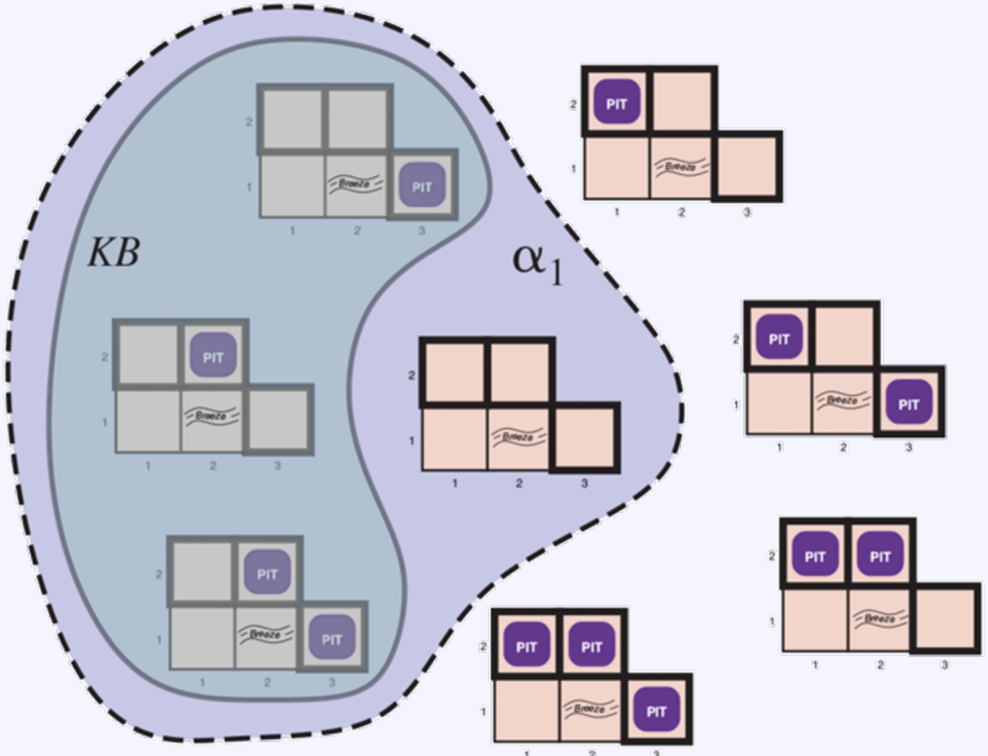
$$KB \vDash \beta \text{ if and only if } M(KB) \subseteq M(\beta)$$

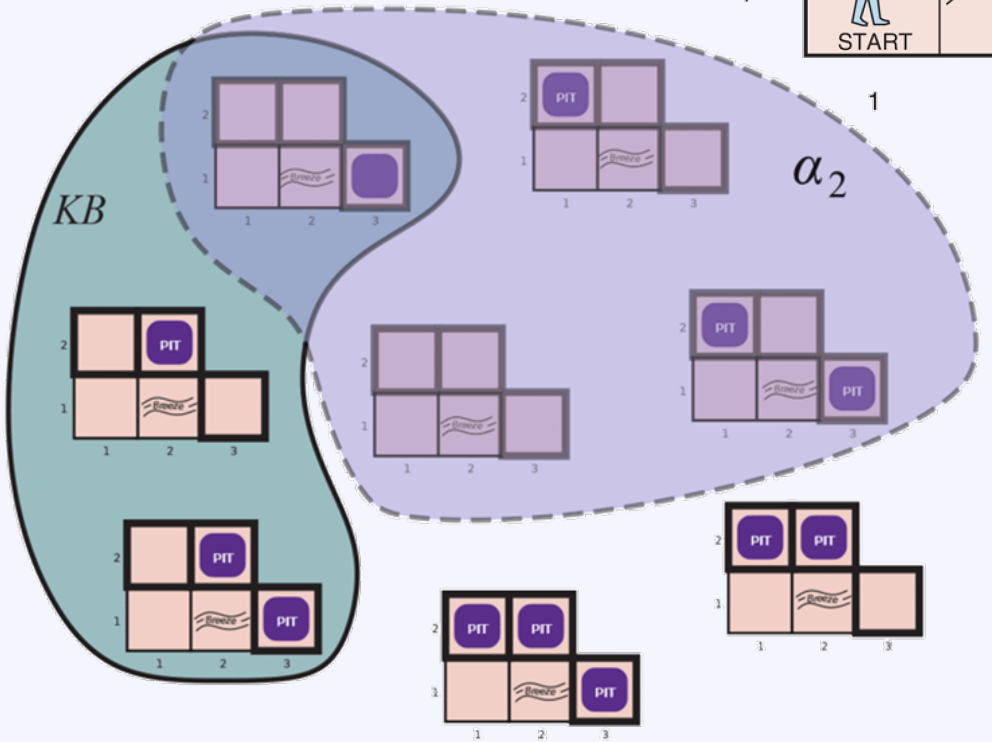
- 如果知识库为真的模型使得语句 β 也为真，则可以从知识库推断出语句 β

怪兽问题的逻辑蕴涵

- KB = (1,1) 中什么也没感知到 **并且** (2,1) 中有微风
- α_1 = (1,2) 中没有陷阱
- α_2 = (2,2) 中没有陷阱
- 所有KB 为真的模型，α_1 亦为真，因此 KB $\vDash \alpha_1$







等值演算

- **逻辑等值：两个语句在同样的模型集合中为真，则称它们等值 ($\equiv$)**

$$\alpha \equiv \beta \text{ if and only if } \alpha \vDash \beta \text{ and } \beta \vDash \alpha$$

- 双重否定：$\neg(\neg A) \equiv A$
- 等幂律：$A \wedge A \equiv A$；$A \vee A \equiv A$
- 排中律：$A \vee \neg A \equiv \mathrm{TRUE}$
- 矛盾律：$A \wedge \neg A \equiv \mathrm{FALSE}$
- 同一律：$A \wedge \mathrm{TRUE} \equiv A$；$A \vee \mathrm{FALSE} \equiv A$
- 零律：$A \vee \mathrm{TRUE} \equiv \mathrm{TRUE}$；$A \wedge \mathrm{FALSE} \equiv \mathrm{FALSE}$
- 摩根律：$\neg(A \wedge B) \equiv \neg A \vee \neg B$；$\neg(A \vee B) \equiv \neg A \wedge \neg B$
- 交换律：$A \wedge B \equiv B \wedge A$；$A \vee B \equiv B \vee A$
- 结合律：$A \wedge (B \wedge \mathrm{C}) \equiv (A \wedge B) \wedge \mathrm{C}$；$A \vee (B \vee \mathrm{C}) \equiv (A \vee B) \vee \mathrm{C}$
- 分配律：$A \wedge (B \vee \mathrm{C}) \equiv (A \wedge B) \vee (A \wedge \mathrm{C})$；$A \vee (B \wedge \mathrm{C}) \equiv (A \vee B) \wedge (A \vee \mathrm{C})$
- **隐含等值：$A \Rightarrow B \equiv \neg A \vee B$**
- 假言移位：$A \Rightarrow B \equiv \neg B \Rightarrow \neg A$
- **等价等值：$A \Leftrightarrow B \equiv (A \Rightarrow B) \wedge (B \Rightarrow A)$**
- 等价否定：$A \Leftrightarrow B \equiv \neg A \Leftrightarrow \neg B$
- 归谬论：$(A \Rightarrow B) \wedge (A \Rightarrow \neg B) \equiv \neg A$

一阶谓词逻辑

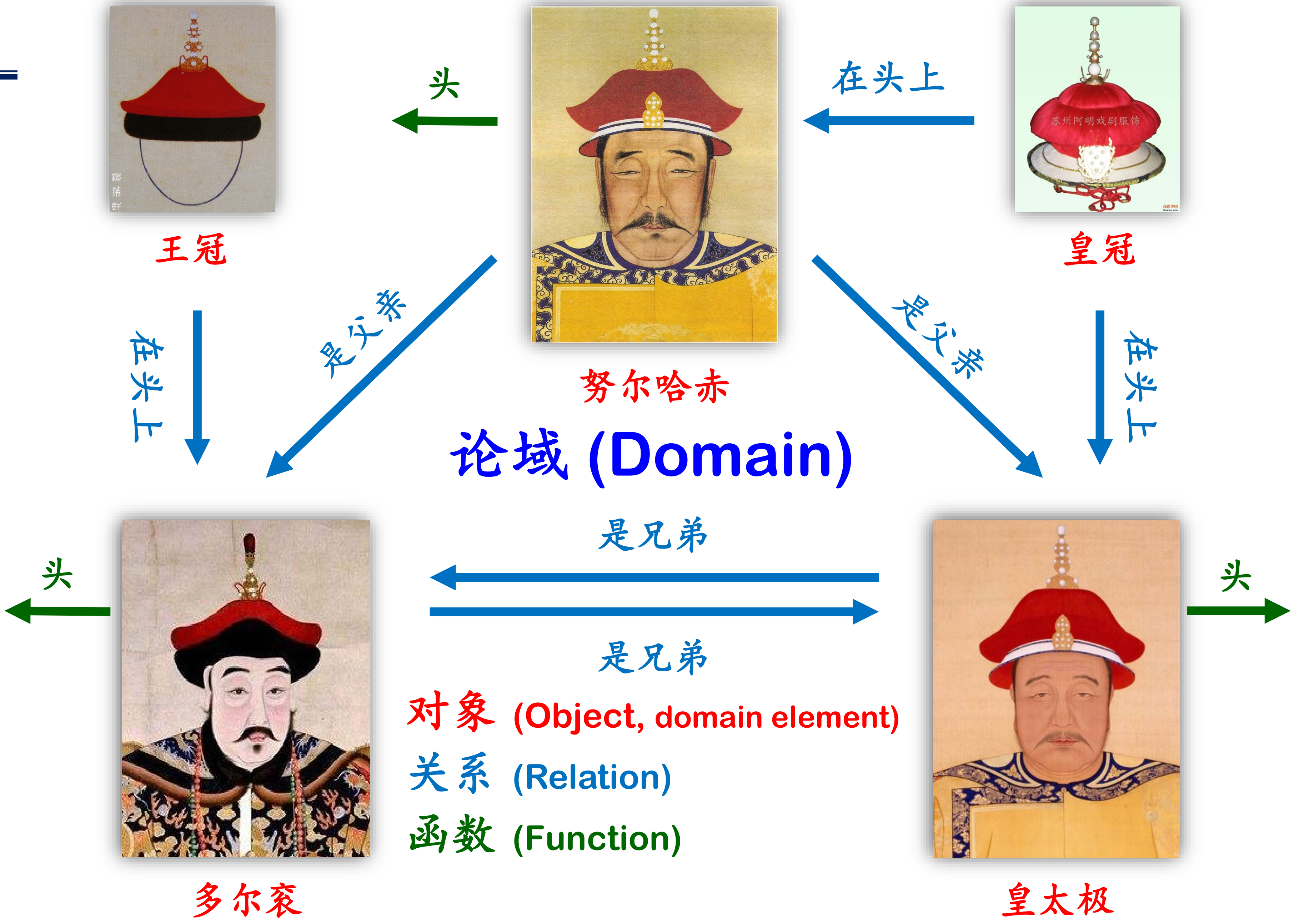
谓词逻辑 / Predicate logic

一阶逻辑 / First-order logic

一阶谓词演算 / First-order predicate calculus

论域

- 对象
  - *Nurhachi*
  - *Taiji*
  - *Gorgon*
  - *Crown*
- 关系
  - *Emperor(Nurhachi)*
  - *Prince(Gorgon)*
  - *Father(Nurhachi, Taiji)*
  - *Brother(Taiji, Gorgon)*
- 函数
  - *Head(Nurhachi)*



语句

- **原子语句**：谓词作用于项的结果称为原子语句（项：指代对象的表达式）
  - 语法
    - $Emperor(Taiji)$ — 皇太极是皇帝
    - $Prince(Gorgon)$ — 多尔衮是王爷
    - $Brother(Taiji, Gorgon)$ — 多尔衮是皇太极的兄弟
  - 语义：原子语句是逻辑表达式，**如果谓词所指代的关系在参数所指代的对象中成立，那么原子语句在给定模型、给定解释下为真**
- **复合语句**：由逻辑连接符相连的语句
  - 语法：同命题逻辑
    - $\neg Emperor(\mathrm{Gorgon})$ — 多尔衮不是皇帝
    - $Father(Taiji) \wedge Mother(Gorgon)$ — 皇太极的父亲与多尔衮的母亲
    - $Head(Taiji) \vee Head(\mathrm{Gorgon})$ — 皇太极的头或多尔衮的头
    - $Emperor(Taiji) \Rightarrow Prince(Gorgon)$ — 皇太极是皇帝隐含多尔衮是王爷
    - $Father(Taiji) \Leftrightarrow Father(Gorgon)$ — 皇太极的和多尔衮的父亲等价
  - 语义：同命题逻辑

量词 (∀ 和 ∃)

- **全称量词指代论域中的所有对象** (Universal quantifier)

$$\forall x \;\; Emperor(x) \Rightarrow Person(x)$$

  - For all x, if x is an emperor, then x is a person
  - 对于所有 x，如果 x 是皇帝，那么 x 是人（所有皇帝都是人）
  - x: **变量 (variable)，指代对象，是一种项**

- **存在量词指代论域中的某个或某些对象** (Existential quantifier)

$$\exists x \;\; Crown(x) \wedge OnHead(x, \; Taiji)$$

  - There exists an x such that x is a crown and x is on the head of Taiji
  - 存在某个 x，它是皇冠并且戴在皇太极的头上（皇太极戴着皇冠）

全称量词隐含合取关系

- $\forall x$, 论域中的任意对象
  - 每一个：努尔哈赤、皇太极、多尔衮，皇冠、王冠、努尔哈赤的头、皇太极的手 …
- $\forall x$, 隐含扩展解释之间的合取关系
  - $\forall x\ Emperor(x) \Rightarrow Person(x)$
    - $Emperor(Nurhachi) \Rightarrow Person(Nurhachi)$
    - $Emperor(Taiji) \Rightarrow Person(Taiji)$
    - $Emperor(Dorgon) \Rightarrow Person(Dorgon)$
    - $Emperor(\text{皇冠}) \Rightarrow Person(\text{皇冠})$
    - $Emperor(Head(Nurhachi)) \Rightarrow Person(Head(Nurhachi))$
    - … … **所有这些语句是合取的关系**
- $\forall x$, 只用考虑前项为真的隐含关系
  - 隐含关系什么时候为真?
  - **使用全称量词 $\forall$ 时，隐含关系 $\Rightarrow$ 是完美的选择**

$\forall$ 与 $\Rightarrow$ 在一起

$\forall$ 作为限定词时，使用 $\wedge$ 会使陈述过强
- $\forall x\ Emperor(x) \wedge Person(x)$
  - $Emperor(Nurhachi) \wedge Person(Nurhachi)$
  - $Emperor(Taiji) \wedge Person(Taiji)$
  - $Emperor(Dorgon) \wedge Person(Dorgon)$
  - $Emperor(\text{皇帽}) \wedge Person(\text{皇帽})$
  - $Emperor(Head(Nurhachi)) \wedge Person(Head(Nurhachi))$

合取项有一个为假，语句为假，结果是?

上面的子句几乎必然会出现为假的！

存在量词隐含析取关系

- $\exists x$, 论域中的某个或某些对象（至少一个，at least）
  - 一个或多个：努尔哈赤、皇太极、多尔衮，皇冠、王冠、努尔哈赤的头、皇太极的手 …
- $\exists x$, 隐含扩展解释之间的析取关系
  - $\exists x\ Crown(x) \land OnHead(x, Taiji)$
    - $Crown(Nurhachi) \land OnHead(Nurhachi, Taiji)$
    - $Crown(Taiji) \land OnHead(Taiji, Taiji)$
    - $Crown(Dorgon) \land OnHead(Dorgon, Taiji)$
    - $Crown(\text{皇帽}) \land OnHead(\text{皇帽}, Taiji)$
    - $Crown(Head(Nurhachi)) \land OnHead(Head(Nurhachi), Taiji)$
    - … … **所有这些语句是析取的关系**
- $\exists x$, 只用考虑语义为真的合取关系
  - 只要这些语句有一个是真就行
  - **使用存在量词 ∃ 时，合取关系 ∧ 是完美的选择**

∃ 与 ∧ 在一起

∃ 作为限定词时，使用⇒会使陈述过弱

- $\exists x\ Crown(x) \Rightarrow OnHead(x, Taiji)$
  - $Crown(Nurhachi) \Rightarrow OnHead(Nurhachi, Taiji)$
  - $Crown(Taiji) \Rightarrow OnHead(Taiji, Taiji)$
  - $Crown(Dogon) \Rightarrow OnHead(Dogon, Taiji)$
  - $Crown(\text{皇冠}) \Rightarrow OnHead(\text{皇冠}, Taiji)$
  - $Crown(Head(Nurhachi)) \Rightarrow OnHead(Head(Nurhachi), Taiji)$

隐含式前项为假时，语句为真，结果是?

上面的子句全都是真！

$\forall$ 和 $\exists$ 具有互反性

- 互反性

$$\forall x\ Likes(x,\ IceCream) \equiv \neg\exists x\ \neg Likes(x,\ IceCream)$$

$$\begin{aligned}
\forall x\ P(x) &\equiv \neg\exists x\ \neg P(x)\\
\exists x\ P(x) &\equiv \neg\forall x\ \neg P(x)\\
\forall x\ \neg P(x) &\equiv \neg\exists x\ P(x)\\
\neg\forall x\ P(x) &\equiv \exists x\ \neg P(x)
\end{aligned}$$

$\forall x\ P(x)$	$\equiv \neg(\neg(\forall x\ P(x)))$
正用定义	$\equiv \neg(\neg(P(a) \wedge P(b) \wedge P(c)\ \ldots\ ))$
用摩根律	$\equiv \neg(\neg P(a) \vee \neg P(b) \vee \neg P(c)\ \ldots)$
逆用定义	$\equiv \neg(\exists x\ \neg P(x))$
简写	$\equiv \neg\exists x\ \neg P(x)$

量词辖域的收缩与扩张

Q 里不出现 x

- 全称量词

$$\forall x\,(P(x)\wedge Q) \equiv (\forall x\, P(x)) \wedge Q \qquad \forall x\,(Q \wedge P(x)) \equiv Q \wedge (\forall x\, P(x))$$

$$\forall x\,(P(x)\vee Q) \equiv (\forall x\, P(x)) \vee Q \qquad \forall x\,(Q \vee P(x)) \equiv Q \vee (\forall x\, P(x))$$

$$\boldsymbol{\forall x\,(P(x)\Rightarrow Q) \equiv (\exists x\, P(x)) \Rightarrow Q} \qquad \forall x\,(Q \Rightarrow P(x)) \equiv Q \Rightarrow (\forall x\, P(x))$$

- 存在量词

$$\exists x\,(P(x)\wedge Q) \equiv (\exists x\, P(x)) \wedge Q \qquad \exists x\,(Q \wedge P(x)) \equiv Q \wedge (\exists x\, P(x))$$

$$\exists x\,(P(x)\vee Q) \equiv (\exists x\, P(x)) \vee Q \qquad \exists x\,(Q \vee P(x)) \equiv Q \vee (\exists x\, P(x))$$

$$\boldsymbol{\exists x\,(P(x)\Rightarrow Q) \equiv (\forall x\, P(x)) \Rightarrow Q} \qquad \exists x\,(Q \Rightarrow P(x)) \equiv Q \Rightarrow (\exists x\, P(x))$$

$$\boldsymbol{\forall x\,(P(x)\vee Q)} \equiv (P(a)\vee Q) \wedge (P(b)\vee Q) \wedge (P(c)\vee Q) \wedge \ldots$$

逆用分配律 $\equiv ((P(a)\wedge P(b)) \vee Q) \wedge (P(c)\vee Q) \wedge \ldots$

$$\equiv ((P(a)\wedge P(b)\wedge P(c)) \vee Q) \wedge \ldots$$

$$\equiv (P(a)\wedge P(b)\wedge P(c) \wedge \ldots) \vee Q$$

逆用定义 $\equiv (\forall x\, P(x)) \vee Q$

$$\boldsymbol{\forall x\,(P(x)\Rightarrow Q)} \equiv \forall x\,(\neg P(x) \vee Q)$$

$$\equiv (\forall x\,(\neg P(x)) \vee Q$$

$$\equiv \neg(\exists x\, P(x)) \vee Q$$

$$\equiv \exists x P(x) \Rightarrow Q$$

量词分配

Q 里出现 x

- 全称量词可以对合取分配，不能对析取分配
- 存在量词可以对析取分配，不能对合取分配

$$\forall x\ (P(x) \land\ Q(x))\ \equiv \forall x\ P(x) \land \forall x\ \ Q(x)$$
$$\exists x\ (P(x) \lor\ Q(x))\ \equiv \exists x\ P(x) \lor \exists x\ \ Q(x)$$
$$\forall x\ (P(x) \lor\ Q(x))\ \equiv \forall x\ P(x) \lor \forall x\ \ Q(x)$$ 错！
$$\exists x\ (P(x) \land\ Q(x))\ \equiv \exists x\ P(x) \land \exists x\ \ Q(x)$$ 错！

- 反例
  - $\forall x\ (P(x) \lor\ Q(x))$
    每一名自动化系的同学，要么学习成绩好，要么社会工作好
  - $\forall x\ P(x) \lor \forall x\ \ Q(x)$
    每一名自动化系的同学都学习成绩好，或者 每一名自动化系的同学都社会工作好

量词嵌套

- 多个量词可以嵌套起来，表示更复杂的语句

$$\forall x\, \forall y\ Brother(x,\ y) \Rightarrow Sibling(x,\ y)$$

- 同一类型的多个量词可以写到一起

$$\forall x{,}y\ Brother(x,\ y) \Rightarrow Sibling(x,\ y)$$

- 不同类型的多个量词可以混合使用

$\forall x\, \exists y\ Loves(x,\ y)$	每个人都会爱上某人
$\exists x\, \forall y\ Loves(x,\ y)$	存在某个人爱所有人
$\exists y\, \forall x\ Loves(x,\ y)$	存在某人被所有人爱
$\forall x\ (\exists y\ Loves(x,\ y))$	每个人都会爱上某人

- **变量属于引用它的最内层量词，且不再属于其他任何量词**

$$\forall x\ (Crown(x) \lor (\exists x\ Brother(Taiji,\ x)))$$

- 嵌套时应使用不同的变量以避免歧义

$$\forall x\ (Crown(x) \lor (\exists y\ Brother(Taiji,\ y)))$$

综合使用

$$(\forall x\ P(x, y) \Rightarrow \exists y\ Q(y)) \Rightarrow \forall x\ R(x, y)$$

$$\equiv (\forall x\ P(x, y') \Rightarrow \exists y\ Q(y)) \Rightarrow \forall x'\ R(x', y')$$ 变量换名

$$\equiv \exists x\ (P(x, y') \Rightarrow \exists y\ Q(y)) \Rightarrow \forall x'\ R(x', y')$$ 量词辖域扩张 $\exists x\ (P(x) \Rightarrow Q) \equiv (\forall x\ P(x)) \Rightarrow Q$

$$\equiv \forall x\ ((P(x, y') \Rightarrow \exists y\ Q(y)) \Rightarrow \forall x'\ R(x', y'))$$ 量词辖域扩张 $\forall x\ (P(x) \Rightarrow Q) \equiv (\exists x\ P(x)) \Rightarrow Q$

$$\equiv \forall x\ (\exists y\ (P(x, y') \Rightarrow Q(y)) \Rightarrow \forall x'\ R(x', y'))$$ 量词辖域扩张 $\exists x\ (Q \Rightarrow P(x)) \equiv Q \Rightarrow (\exists x\ P(x))$

$$\equiv \forall x\ (\forall y\ ((P(x, y') \Rightarrow Q(y)) \Rightarrow \forall x'\ R(x', y')))$$ 量词辖域扩张 $\forall x\ (P(x) \Rightarrow Q) \equiv (\exists x\ P(x)) \Rightarrow Q$

$$\equiv \forall x\ (\forall y\ (\forall x'\ ((P(x, y') \Rightarrow Q(y)) \Rightarrow R(x', y'))))$$ 量词辖域扩张 $\forall x\ (Q \Rightarrow P(x)) \equiv Q \Rightarrow (\forall x\ P(x))$

$$\equiv \forall x\ \forall y\ \forall z\ \underbrace{((P(x, w) \Rightarrow Q(y)) \Rightarrow R(z, w))}_{\text{按命题逻辑处理}}$$ 变量换名

前束范式

- **语句中所有量词（不含否定词）都在最左边，且这些量词的辖域都到公式最末端**
- 一阶谓词逻辑中任何语句均可转换为与之等价的前束范式
- 前束范式不唯一

$$(\forall x\ P(x,\ y)\ \Rightarrow \exists y\ Q(y))\Rightarrow \forall x\ R(x,\ y)\ \equiv$$

$$\forall x\, \forall y\, \forall z\ ((P(x,\ w)\ \Rightarrow\ Q(y))\Rightarrow\ R(z,\ w))$$

前束范式

前面的全称量词存在量词怎么处理？

全称量词的实例化

- 全称量词

$$\forall x \ Emperor(x) \Rightarrow CanFight(x)$$

- 全称量词指代论域中的任意对象，因此
- 将变量替换成论域中的任意对象，逻辑表达式都是成立的
- 这种变量的替换称为**置换**
  - 置换：$\boldsymbol{\theta = \{x/Taiji\}}$ 把 x 换成 $Taiji$，有的书写为 $\theta = \{Taiji/x\}$
  - 形式：$\boldsymbol{\theta = \{v/g\}}$ $\boldsymbol{v}$**: variable,** $\boldsymbol{g}$**: ground term**
- 既然置换成任意对象都可以，
- **直接消去全称量词**，待需要替换时再考虑置换为哪个对象

存在量词的实例化

- 存在量词

$$\exists y\ Crown(y) \wedge OnHead(y, Nurhachi)$$

- 存在量词指代论域中的某个或某些对象，因此
- 将变量替换成论域中的某个具体的对象，逻辑表达式会成立
- 给这个特定对象起名字 C_N，这种起名也是一种**置换**
  - 置换：$\boldsymbol{\theta = \{y/C_N\}}$ 给 x 起名 C_N， C_N 不能在其他语句中出现
  - 形式：$\boldsymbol{\theta = \{v/c\}}$ **v: variable, c: Skolem constant**
- 但是，如果该特定**对象被全称量词所管辖**

$$\forall \boldsymbol{x}\ \exists y\ Crown(y) \wedge OnHead(y, x)$$

- 则该对象的起名依赖于管辖它的变量，此时**应置换为函数**
  - 置换：$\boldsymbol{\theta = \{y/F(x)\}}$ 给 x 起名 $\boldsymbol{F(x)}$， F不能在其他语句中出现
  - 形式：$\boldsymbol{\theta = \{v/c\}}$ **v: variable, $F(x)$: Skolem function**

演绎推理

Deductive Reasoning

通过模型检查进行逻辑推理

- 如果今天下雨，则带雨伞或雨衣
- 如果走路上班，则不带雨衣
- 今天下雨，走路上班，所以带雨伞

- P: 今天下雨
- Q: 带雨伞
- R: 带雨衣
- S: 走路上班
- KB: $(P \Rightarrow (Q \vee R)) \wedge (S \Rightarrow \neg R) \wedge P \wedge S$
- β: Q

$$KB \vDash Q\,?$$

P	Q	R	S	$\neg R$	$Q \vee R$	$P \Rightarrow (Q \vee R)$	$S \Rightarrow \neg R$	KB
F	F	F	F	T	F	T	T	F
F	F	F	T	T	F	T	T	F
F	F	T	F	F	T	T	T	F
F	F	T	T	F	T	T	F	F
F	T	F	F	T	T	T	T	F
F	T	F	T	T	T	T	T	F
F	T	T	F	F	T	T	T	F
F	T	T	T	F	T	T	F	F
T	F	F	F	T	F	F	T	F
T	F	F	T	T	F	F	T	F
T	F	T	F	F	T	T	T	F
T	F	T	T	F	T	T	F	F
T	T	F	F	T	T	T	T	F
T	**T**	**F**	**T**	**T**	**T**	**T**	**T**	**T**
T	T	T	F	F	T	T	T	F
T	T	T	T	F	T	T	F	F

演绎定理 Deduction

- **演绎定理：For any sentences α and β,**
  $\alpha \vDash \beta$ if and only if the sentence $\alpha \Rightarrow \beta$ is valid

- 有效性：如果一个语句对所有模型都为真，则称该语句是有效的
- 等价性：有效的语句与 TRUE 等价
- 重言式：有效的语句也称为重言式
  - $A \vee$ TRUE 是有效的
  - $P \vee \neg P$ 是有效的

- 充分性：如果 $\alpha \Rightarrow \beta$ 有效，则根据演算 α 为真时 β 亦为真，所以 $\alpha \vDash \beta$
- 必要性：如果 $\alpha \vDash \beta$ 成立，则根据定义 α 为真时 β 亦为真，所以 $\alpha \Rightarrow \beta$ 是有效的
- α 为假时 β 不论真假，$\alpha \Rightarrow \beta$ 都是有效的

- **逻辑蕴含等价于隐含式是有效的**
- 要证明逻辑蕴含关系 $\alpha \vDash \beta$ 成立，只需证明隐含式 $\alpha \Rightarrow \beta$ 是有效的，即 $\alpha \Rightarrow \beta$ 在任何模型中均为 TRUE，即**不会出现 α 为 TRUE，β 为 FALSE 的情况**

推理规则

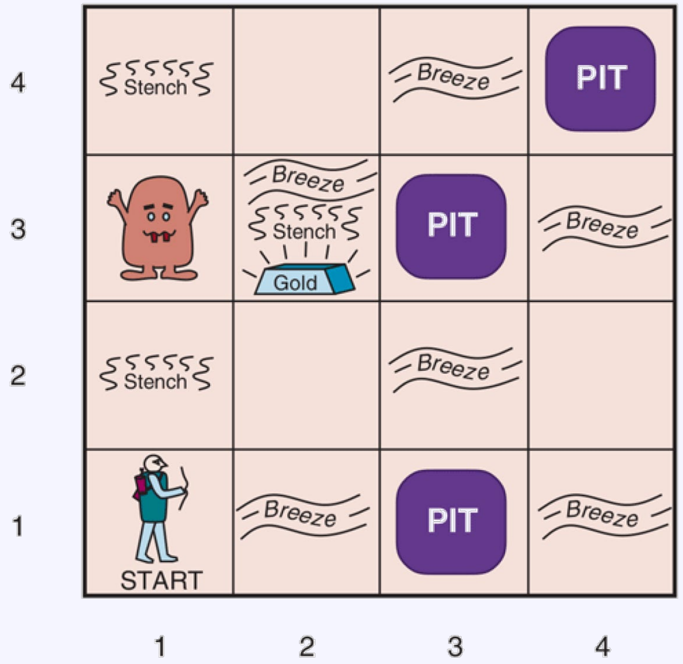
- **演绎定理：For any sentences α and β,**
  $\alpha \vDash \beta$ if and only if the sentence $\alpha \Rightarrow \beta$ is valid

- 附加： $A \Rightarrow (A \vee B)$
- 简化： $(A \wedge B) \Rightarrow A$
- 拒取式： $((A \Rightarrow B) \wedge \neg B) \Rightarrow \neg A$
- **假言推理：** $\boldsymbol{((A \Rightarrow B) \wedge A) \Rightarrow B}$
- 析取三段论： $((A \vee B) \wedge \neg A) \Rightarrow B$
- 假言三段论： $((A \Rightarrow B) \wedge (B \Rightarrow C)) \Rightarrow (A \Rightarrow C)$
- 等价三段论： $((A \Leftrightarrow B) \wedge (B \Leftrightarrow C)) \Rightarrow (A \Leftrightarrow C)$
- 构造性二难： $(A \Rightarrow B) \wedge (C \Rightarrow D) \wedge (A \vee C) \Rightarrow (B \vee D)$

逻辑推理

- R_2: $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$ 知识
- R_6: $(B_{1,1} \Rightarrow (P_{1,2} \vee P_{2,1})) \wedge ((P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1})$
- R_7: $(P_{1,2} \vee P_{2,1}) \Rightarrow B_{1,1}$ 简化
- R_8: $\neg B_{1,1} \Rightarrow \neg(P_{1,2} \vee P_{2,1})$ 假言移位
- R_4: $\neg B_{1,1}$ 知识
- R_9: $\neg(P_{1,2} \vee P_{2,1})$ 假言推理
- R_{10}: $\neg P_{1,2} \wedge \neg P_{2,1}$ 摩根律
- R_{11}: $\neg P_{1,2}$ 简化

要用到好多推理规则，不是行家搞不定



- R_1: $\neg P_{1,1}$
- R_2: $B_{1,1} \Leftrightarrow (P_{1,2} \vee P_{2,1})$
- R_3: $B_{2,1} \Leftrightarrow (P_{1,1} \vee P_{2,2} \vee P_{3,1})$
- R_4: $\neg B_{1,1}$
- R_5: $B_{2,1}$

$\neg P_{1,2}$?

假言推理 Modus Ponens, *mode that affirms*

- **如果语句 α 为真，且 $\alpha \Rightarrow \beta$，则 β 为真**

$$(\alpha \wedge (\alpha \Rightarrow \beta)) \vDash \beta$$

IF α 成立
THEN β 成立

- 等价于证明 $(\alpha \wedge (\alpha \Rightarrow \beta)) \Rightarrow \beta$ 是重言式

$$
\begin{aligned}
(\alpha \wedge (\alpha \Rightarrow \beta)) \Rightarrow \beta &\equiv \neg(\alpha \wedge (\alpha \Rightarrow \beta)) \vee \beta \\
&\equiv \neg(\alpha \wedge (\neg\alpha \vee \beta)) \vee \beta \\
&\equiv \neg((\alpha \wedge \neg\alpha) \vee (\alpha \wedge \beta)) \vee \beta \\
&\equiv \neg(\text{FALSE} \vee (\alpha \wedge \beta)) \vee \beta \\
&\equiv \neg(\alpha \wedge \beta) \vee \beta \\
&\equiv (\neg\alpha \vee \neg\beta) \vee \beta \\
&\equiv \neg\alpha \vee (\neg\beta \vee \beta) \\
&\equiv \neg\alpha \vee \text{TRUE} \\
&\equiv \text{TRUE}
\end{aligned}
$$

假言推理（推广）

- **如果 $P_1, \ldots, P_n$ 为真，且 $P_1 \wedge \ldots \wedge P_n \Rightarrow Q$，则 Q 为真**

P_1: 能见度差
P_2: 空气刺鼻
P_3: 咽喉干涩

$P_1 \wedge P_2 \wedge P_3 \Rightarrow Q$ Q: 是雾霾天

P_1: 一张大嘴
P_2: 两只眼睛
P_3: 四条长腿
P_4: 绿色身体
P_5: 会呱呱叫

$P_1 \wedge P_2 \wedge P_3 \wedge P_4 \wedge P_5 \Rightarrow Q$ Q: 是只青蛙

$$P_1 \wedge \ldots \wedge P_n \Rightarrow Q \equiv \neg P_1 \vee \ldots \vee \neg P_n \vee \boxed{Q}$$

仅包含限定子句的知识库

- **限定子句：有且仅有一个正文字的析取式**

$$(\neg A \vee \neg B \vee C) \equiv (A \wedge B) \Rightarrow C$$

  和隐含式等值

- 可解释性强
  - **事实：仅包含一个正文字的限定子句**

    P_1（能见度差） P_2（空气刺鼻） P_3（咽喉干涩）
  - **规则：包含若干反文字和一个正文字的限定子句**

    $P_1 \wedge P_2 \wedge P_3 \Rightarrow Q$ （能见度差 $\wedge$ 空气刺鼻 $\wedge$ 咽喉干涩 $\Rightarrow$ 是雾霾天）

    IF (能见度差 并且 空气刺鼻 并且 咽喉干涩) THEN 是雾霾天
- 推理方法好
  - 自然、简单，复合人的思维习惯
- 推理效率高
  - 从事实出发，在线性时间遍历整个知识库

前向链接 (Forward chaining)

正向推理
正向演绎
Forward reasoning

- **在限定子句上使用假言推理，从事实出发，匹配规则，产生事实，直到结论**

事实

能见度差(A)　空气刺鼻(B)　咽喉干涩(C)　处于室内(D)

规则

处于室外(E)　是雾霾天(F)　进行防护(G)　开净化器 (H)　带上口罩(I)

如果 **能见度差** 并且 **空气刺鼻** 并且 **咽喉干涩** 那么 **是雾霾天**　$A \wedge B \wedge C \Rightarrow F$

如果 **是雾霾天** 那么 **进行防护**　$F \Rightarrow G$

如果 **进行防护** 并且 **处于室内** 那么 **开净化器**　$G \wedge D \Rightarrow H$

如果 **进行防护** 并且 **处于室外** 那么 **带上口罩**　$G \wedge E \Rightarrow I$

推理（反复应用假言推理）

$A \wedge B \wedge C \Rightarrow F$

$F \Rightarrow G$

$G \wedge D \Rightarrow H$ **应该开净化器**

后向链接 (Backward chaining)

反向推理
逆向演绎
Backward reasoning

- **在限定子句上使用假言推理，从查询出发，匹配规则，倒推事实，直到满足**

事实

能见度差(A) **空气刺鼻**(B) **咽喉干涩**(C) **处于室内**(D)

查询

是否应该 开净化器 (H)

规则

处于室外(E) 是雾霾天(F) 进行防护(G) 开净化器 (H) 带上口罩(I)

如果 **进行防护** 并且 **处于室内** 那么 **开净化器**	$G \wedge D \Rightarrow H$
如果 **是雾霾天** 那么 **进行防护**	$F \Rightarrow G$
如果 **能见度差** 并且 **空气刺鼻** 并且 **咽喉干涩** 那么 **是雾霾天**	$A \wedge B \wedge C \Rightarrow F$
如果 进行防护 并且 处于室外 那么 带上口罩	$G \wedge E \Rightarrow I$

推理（反复应用假言推理）

$G \wedge D \Rightarrow H$ **应该开净化器**

$F \Rightarrow G$

$A \wedge B \wedge C \Rightarrow F$

谓词逻辑的演绎推理

- 清朝皇帝能打仗，皇太极是清朝皇帝，所以皇太极能打仗

$$\forall x\ Emperor(x) \Rightarrow CanFight(x) \qquad Emperor(Taiji) \qquad CanFight(Taiji)$$

$$((\forall x\ Emperor(x) \Rightarrow CanFight(x)) \wedge Emperor(Taiji)) \quad \Rightarrow \quad CanFight(Taiji)$$

- 要是命题逻辑的话，可以直接用假言推理

$$((Emperor(Taiji) \Rightarrow CanFight(Taiji)) \wedge Emperor(Taiji)) \Rightarrow CanFight(Taiji)$$

- 谓词逻辑该怎么办，如何转化为命题逻辑?
  - 置换：$\theta = \{x/Taiji\}$ 把 x 换成 $Taiji$
  - 形式：$\theta = \{v/g\}$ v: variable, g: ground term
- **在置换 $\theta = \{x/Taiji\}$ 下，谓词逻辑转化为命题逻辑**

$$((Emperor(x) \Rightarrow CanFight(x)) \wedge Emperor(Taiji)) \quad \Rightarrow \quad CanFight(Taiji)$$

$$((Emperor(Taiji) \Rightarrow CanFight(Taiji)) \wedge Emperor(Taiji)) \Rightarrow CanFight(Taiji)$$

- 于是可以使用假言推理了

合一

- 贪婪的皇帝都是暴君，人都是贪婪的，努尔哈赤是皇帝

$$\forall x\ Emperor(x) \wedge Greedy(x) \Rightarrow Tyrant(x)$$

$$\forall y\ Greedy(y)$$

$$Emperor(Nurhachi)$$

- 置换

$$\boldsymbol{\theta = \{x/Nurhachi,\ y/Nurhachi\}}$$

$$\left.\begin{array}{l} Emperor(Nurhachi) \wedge Greedy(Nurhachi) \Rightarrow Tyrant(Nurhachi) \\ Greedy(Nurhachi) \\ Emperor(Nurhachi) \end{array}\right\} \quad Tyrant(Nurhachi)$$

- 合一：**θ 这样使多个语句变得相同的置换称为合一**
  - 变量分离：不同语句中的变量实际上是不同的，应对变量重新命名

    $Knows(John,\ \boldsymbol{x})$ 和 $Knows(\boldsymbol{x},\ Elizabeth)$ 看上去不能合一

    $Knows(John,\ \boldsymbol{x})$ 和 $Knows(\boldsymbol{y},\ Elizabeth)$ 在 $\{y/John,\ x/Elizabeth\}$ 下合一

广义假言推理 Generalized *Modus Ponens*

- **假言推理：如果语句 α 为真，且 $\alpha \Rightarrow \beta$，则 β 为真，即 $(\alpha \wedge (\alpha \Rightarrow \beta)) \vDash \beta$**
- **命题逻辑：如果 $P_1, \ldots, P_n$ 为真，且 $P_1 \wedge \ldots \wedge P_n \Rightarrow Q$，则 Q 为真**
- **谓词逻辑：如果 $P_1, \ldots, P_n$ 为真，$P'_1 \wedge \ldots \wedge P'_n \Rightarrow Q$，**
  且存在合一置换 θ，使得 $\textsc{Subst}(\theta, P_i) = \textsc{Subst}(\theta, P'_i)$
  则 $\textsc{Subst}(\theta, Q)$ 为真

$$Emperor(Nurhachi)$$
$$\forall y\ Greedy(y)$$
$$\forall x\ Emperor(x) \wedge Greedy(x) \Rightarrow Tyrant(x)$$

$$\xrightarrow[\text{置换}]{\theta = \{x/Nurhachi,\ y/Nurhachi\}}$$

$$Emperor(Nurhachi)$$
$$Greedy(Nurhachi)$$
$$Emperor(Nurhachi) \wedge Greedy(Nurhachi) \Rightarrow Tyrant(Nurhachi)$$

基于广义假言推理即可进行限定性知识库的演绎推理

改变中国历史的神器

红衣大炮，16世纪初欧洲制造的前装重型滑膛炮，明代后期传入中国，也称为红夷大炮（荷兰人称为红夷）

天启六年（1626年），在宁远之战中发挥极大威力，八旗官兵血肉横飞，尸积如山，努尔哈赤本人也被击伤，后去世

崇祯四年（1631年），后金制成红衣大炮“天佑助威大将军”，还改进了铸跑工艺，领先于明朝

崇祯十二年（1639年），清军拥有六十门自制的红衣大炮，连破明军据守的塔山、杏山二城

顺治元年（1645年）十二月，在入关战争中出击潼关，重创李自成的大顺军，李自成流窜至湖北通山县被害

为什么清军的红衣大炮反而强于明军？

有明朝叛徒相助

- 皇太极深知红衣大炮的厉害，在沈阳利用从明朝俘虏过来的工匠刘汉，成功仿制了红衣大炮 “天佑助威大将军”

刘汉有罪吗？

- 明朝：为敌对国制造武器的人有罪
- 清朝：皇太极时代已拥有红衣大炮
- 刘汉：为清朝制造大炮的明朝工匠
- 大炮：红衣大炮是威力极大的武器
- 敌对：处于交战中的国家是敌对国
- 交战：明朝和清朝当时正在交战中

形式化

- 明朝：为敌对国制造武器的人有罪

$$Ming(x) \wedge Weapon(y) \wedge Makes(x, y, z) \wedge Hostile(z) \Rightarrow Criminal(x)$$

- 清朝：皇太极时代已拥有红衣大炮

$$\exists x \; Hongyipo(x) \wedge Owns(Qing, x)$$

- 刘汉：为清朝制造大炮的明朝工匠

$$Hongyipo(x) \wedge Owns(Qing, x) \Rightarrow Makes(Liuhan, x, Qing)$$

$$Ming(Liuhan)$$

- 大炮：红衣大炮是威力极大的武器

$$Hongyipo(x) \Rightarrow Weapon(x)$$

- 敌对：处于交战中的国家是敌对国

$$Enemy(x, Ming) \Rightarrow Hostile(x)$$

- 交战：明朝和清朝当时正在交战中

$$Enemy(Qing, Ming)$$

前向链接

$\exists x\ Hongyipo(x) \wedge Owns(Qing, x)$

$\boldsymbol{x/Tianyou}$

$\boldsymbol{Hongyipo(Tianyou)}$

$\boldsymbol{Owns(Qing, Tianyou)}$

$Hongyipo(Tianyou)$

$Owns(Qing, Tianyou)$

$Hongyipo(x) \wedge Owns(Qing, x) \Rightarrow Makes(Liuhan, x, Qing)$

$\boldsymbol{x/Tianyou}$

$\boldsymbol{Makes(Liuhan, Tianyou, Qing)}$

$Enemy(Qing, Ming)$

$Enemy(x, Ming) \Rightarrow Hostile(x)$

$\boldsymbol{x/Qing}$

$\boldsymbol{Hostile(Qing)}$

$Hongyipo(Tianyou)$

$Hongyipo(x) \Rightarrow Weapon(x)$

$\boldsymbol{x/Tianyou}$

$\boldsymbol{Weapon(Tianyou)}$

$Ming(Liuhan)$

$Weapon(Tianyou)$

$Makes(Liuhan, Tianyou, Qing)$

$Hostile(Qing)$

$Ming(x) \wedge Weapon(y) \wedge Makes(x, y, z) \wedge Hostile(z) \Rightarrow Criminal(x)$

$\boldsymbol{x/Liuhan,\ y/Tianyou,\ z/Qing}$

$\boldsymbol{Criminal(Liuhan)}$

归结原理

Resolution Rule

可满足性 Satisfiability

- 一个语句可满足， 就是有一些模型能使该语句为真
- 一个语句不可满足，就是在所有模型下该语句均为假
- 有效性和可满足性的关系
  - 语句 α 有效，则 α 总为真，$\neg\alpha$ 总为假，所以 $\neg\alpha$ 不可满足
  - **语句 α 有效，当且仅当 $\neg\alpha$ 不可满足**

  - 语句 α 可满足，则 α 不会总为假，$\neg\alpha$ 不会总为真，所以 $\neg\alpha$ 不是有效的
  - **语句 α 可满足，当且仅当 $\neg\alpha$ 不是有效的**

归谬法 *reductio ad absurdum*

- 演绎定理：For any sentences α and β, $\alpha \vDash \beta$
  if and only if the sentence $\alpha \Rightarrow \beta$ is valid

$$
\begin{aligned}
\alpha \Rightarrow \beta \text{ is valid } &\equiv \neg(\alpha \Rightarrow \beta) \text{ is unsatisfiable} \\
&\equiv \neg(\neg\alpha \vee \beta) \text{ is unsatisfiable} \\
&\equiv \alpha \wedge \neg\beta \text{ is unsatisfiable}
\end{aligned}
$$

- **归谬法：For any sentences α and β,**
  $\alpha \vDash \beta$ if and only if the sentence $\alpha \wedge \neg\beta$ is unsatisfiable

Reduction to an absurd thing

归结原理 (命题逻辑)

- **如果 $\alpha \vee \beta$ 和 $\neg\alpha \vee \gamma$ 均为真，则 $\beta \vee \gamma$ 亦为真**

$$\underbrace{(\alpha \vee \beta) \wedge (\neg\alpha}_{\text{消除了}} \vee \gamma) \vDash (\beta \vee \gamma)$$

α	β	$\neg\alpha$	γ	$\alpha\vee\beta$	$\neg\alpha\vee\gamma$	$(\alpha\vee\beta)\wedge(\neg\alpha\vee\gamma)$	$\beta \vee \gamma$
F	F	T	F	F	T	F	F
F	F	T	T	F	T	F	T
F	T	T	F	T	T	T	T
F	T	T	T	T	T	T	T
T	F	F	F	T	F	F	F
T	F	F	T	T	T	T	T
T	T	F	F	T	F	F	T
T	T	F	T	T	T	T	T

归结原理 (谓词逻辑)

- **如果 $\alpha \vee \beta$ 和 $\neg\alpha \vee \gamma$ 均为真，则 $\beta \vee \gamma$ 亦为真**

$$\underbrace{(\alpha \vee \beta) \wedge (\neg\alpha \vee \gamma)}_{\text{消除了}} \models (\beta \vee \gamma)$$

如果存在合一使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结

$$\begin{array}{ccc} \neg Loves(y,\ Taiji) & \xrightarrow[\text{置换}]{\theta = \{y/G(Taiji)\}} & \neg Loves(G(Taiji),\ Taiji) \\ Loves(G(Taiji),\ Taiji) & & Loves(G(Taiji),\ Taiji) \end{array}$$

合取范式

- 合取范式：**析取子句的合取形式**
- 获取合取范式的步骤：

1. 等价消去　($A \Leftrightarrow B \equiv (A \Rightarrow B) \wedge (B \Rightarrow A)$)
2. 隐含消去　($A \Rightarrow B \equiv \neg A \vee B$)
3. 否定深入　($\neg(A \wedge B) \equiv \neg A \vee \neg B, \neg(A \vee B) \equiv \neg A \wedge \neg B$)
4. **变量换名**
5. **量词前移**　(量词辖域扩张)
6. **量词消去**　(先消去存在量词，再消去全称量词)
7. 分配律　($A \vee (B \wedge \mathrm{C}) \equiv (A \vee B) \wedge (A \vee \mathrm{C})$)

归结算法

- 给定知识库 KB 和语句 α，欲证明 $KB \vDash \alpha$。
- 采用归谬法，只需证明 $(KB \wedge \neg\alpha)$ 不可满足。
- 根据归结原理，算法如下：

1. 将 $KB \wedge \neg\alpha$ 转换为合取范式。
2. 抽取合取范式中所有子句，构成子句集。
3. 重复从子句集中选取含有互补文字的子句，应用归结原理产生新子句，若其尚未出现过，则将其加入子句集。直到发生下列两种情况：
    1. 没有可以添加的新子句，意味着不能从 KB 得出 α。
    2. 两个子句归结出空子句，意味着 $KB \vDash \alpha$。

$$\left.\begin{array}{l}\text{仁厚之人为民所爱}\\ \text{残暴之人为民所弃}\\ \text{皇太极仁厚之人也}\\ \text{褚英为其父弟所害}\end{array}\right\} KB$$

诛英者努尔哈赤也 ➡ α

仁厚之人为民所爱 $\wedge$
残暴之人为民所弃 $\wedge$
皇太极仁厚之人也 $\wedge$
褚英为其父弟所害 $\wedge$
$\neg$ 诛英者努尔哈赤也

- 给定知识库 KB 和语句 α，欲证明 $KB \vDash \alpha$。
- 采用归谬法，只需证明 $(KB \wedge \neg\alpha)$ 不可满足。

褚英是清太祖努尔哈赤的长子。努尔哈赤的胞弟舒尔哈齐死后，努尔哈赤开始逐渐让褚英带兵并主持一部分军政事务。但后来褚英被努尔哈赤所杀。

自然语言转换为一阶逻辑

- 仁厚之人为民所爱（爱所有人的人，存在爱他的人）

$$\forall x\ [(\forall y\ Person(y) \Rightarrow Loves(x, y)) \Rightarrow (\exists y\ Loves(y, x))]$$

- 残暴之人为民所弃（杀人的人，没人爱他）

$$\forall x\ [(\exists y\ Person(x) \wedge Kills(x, y)) \Rightarrow (\forall y\ \neg Loves(y, x))]$$

- 皇太极仁厚之人也

$$\forall x\ Person(x) \Rightarrow Loves(Taiji, x)$$

- 褚英为其父弟所害

$$Prince(Cuyen)$$

$$\forall x\ Prince(x) \Rightarrow Person(x)$$

$$Kills(Nurhachi, Cuyen) \vee Kills(Taiji, Cuyen)$$

- 诛英者努尔哈赤也

$$Kills(Nurhachi, Cuyen)$$

- 根据归结原理，算法如下：
  1. 将 $KB \wedge \neg\alpha$ 转换为合取范式。
  2. 抽取合取范式中所有子句，构成子句集。
  3. 重复从子句集中选取含有互补文字的子句，应用归结原理产生新子句，若其尚未出现过，则将其加入子句集。直到发生下列两种情况：
     1. 没有可以添加的新子句，意味着不能从 KB 得出 α。
     2. 两个子句归结出空子句，意味着 $KB \vDash \alpha$。

转换为合取范式，获得子句集

- 仁厚之人为民所爱 $\forall x\ [(\forall y\ Person(y) \Rightarrow Loves(x, y)) \Rightarrow (\exists y\ Loves(y, x))]$

  $Person(F(x)) \vee Loves(G(x), x)$

  $\neg Loves(x, F(x)) \vee Loves(G(x), x)$

- 残暴之人为民所弃 $\forall x\ [(\exists y\ Person(x) \wedge Kills(x, y)) \Rightarrow (\forall y\ \neg Loves(y, x))]$

  $\neg Loves(y, x) \vee \neg Person(z) \vee \neg Kills(x, z)$

- 皇太极仁厚之人也 $\forall x\ Person(x) \Rightarrow Loves(Taiji, x)$

  $\neg Person(x) \vee Loves(Taiji, x)$

- 褚英为其父弟所害 $\forall x\ Prince(x) \Rightarrow Person(x)$

  $Prince(Cuyen)$

  $\neg Prince(x) \vee Person(x)$

  $Kills(Nurhachi, Cuyen) \vee Kills(Taiji, Cuyen)$

- 诛英者努尔哈赤也 $Kills(Nurhachi, Cuyen)$

  $\neg Kills(Nurhachi, Cuyen)$

- 获取合取范式的步骤：
  1. 等价消去 （ $A \Leftrightarrow B \equiv (A \Rightarrow B) \wedge (B \Rightarrow A)$ ）
  2. 隐含消去 （ $A \Rightarrow B \equiv \neg A \vee B$ ）
  3. 否定深入 （ $\neg(A \wedge B) \equiv \neg A \vee \neg B,\ \neg(A \vee B) \equiv \neg A \wedge \neg B$ ）
  4. **变量换名**
  5. **量词前移** （ 量词辖域扩张 ）
  6. **量词消去** （ 先消去存在量词，再消去全称量词 ）
  7. 分配律 （ $A \vee (B \wedge C) \equiv (A \vee B) \wedge (A \vee C)$ ）

归结

$Person(F(x)) \vee Loves(G(x), x)$

$\neg Loves(x, F(x)) \vee Loves(G(x), x)$

$\neg Loves(y, x) \vee \neg Person(z) \vee \neg Kills(x, z)$

$\neg Person(x) \vee Loves(Taiji, x)$

如果存在合一置换使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结

$\boldsymbol{Prince(Cuyen)}$

$\boldsymbol{\neg Prince(x) \vee Person(x)}$

$\rightarrow Person(Cuyen)$

$\boldsymbol{Kills(Nurhachi, Cuyen) \vee Kills(Taiji, Cuyen)}$

$\boldsymbol{\neg Kills(Nurhachi, Cuyen)}$

$\rightarrow Kills(Taiji, Cuyen)$

归结

$Person(F(x)) \vee Loves(G(x), x)$

$\neg Loves(x, F(x)) \vee Loves(G(x), x)$

$\boxed{\neg \boldsymbol{Loves(y, x)} \vee \neg \boldsymbol{Person(z)} \vee \neg \boldsymbol{Kills(x, z)}}$

$\neg Person(x) \vee Loves(Taiji, x)$

$Person(Cuyen)$

$\boxed{\boldsymbol{Kills(Taiji, Cuyen)}}$

$\Rightarrow$ $\neg \boldsymbol{Loves(y, Taiji)} \vee \neg \boldsymbol{Person(Cuyen)}$

归结

$Person(F(x)) \lor Loves(G(x), x)$

$\neg Loves(x, F(x)) \lor Loves(G(x), x)$

$\neg Person(x) \lor Loves(Taiji, x)$

$\boldsymbol{Person(Cuyen)}$

$\boldsymbol{\neg Loves(y, Taiji) \lor \neg Person(Cuyen)}$

$\Rightarrow$ $\boldsymbol{\neg Loves(y, Taiji)}$

如果存在合一置换使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结

归结

$\boldsymbol{Person(F(x)) \vee Loves(G(x), x)}$

$\neg Loves(x, F(x)) \vee Loves(G(x), x)$

$\boldsymbol{\neg Person(x) \vee Loves(Taiji, x)}$

$\neg Loves(y, Taiji)$

$Loves(G(x), x) \vee Loves(Taiji, F(x))$

如果存在合一置换使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结

归结

$\neg Loves(x, F(x)) \vee Loves(G(x), x)$

$Loves(Taiji, F(x)) \vee Loves(G(x), x)$

$\Rightarrow Loves(G(Taiji), Taiji)$

$\neg Loves(y, Taiji)$

归结

$\neg Loves(y, Taiji)$

$Loves(G(Taiji), Taiji)$

$\Rightarrow \square$

如果存在合一置换使得两个子句包含互补的原子语句，则可通过消去该原子语句对这两个子句进行归结

从怪兽开始，以怪兽结束

$(\forall\ x,y\ B(x,y) \Leftrightarrow P(x\text{-}1,y) \vee P(x+1,y) \vee P(x,y\text{-}1) \vee P(x,y+1)) \wedge \neg B(1,1) \wedge P(1,2)$

- 等价消去

  $(\forall\ x,y\ (B(x,y) \Rightarrow P(x-1,y) \vee P(x+1,y) \vee P(x,y\text{-}1) \vee P(x,y+1))$

  $\wedge\ (P(x-1,y) \vee P(x+1,y) \vee P(x,y-1) \vee P(x,y+1) \Rightarrow B(x,y))$

  $\wedge \neg B(1,1) \wedge P(1,2)$

- 隐含消去

  $(\forall\ x,y\ (\neg B(x,y) \vee P(x-1,y) \vee P(x+1,y) \vee P(x,y-1) \vee P(x,y+1))$

  $\wedge \neg((P(x-1,y) \vee P(x+1,y) \vee P(x,y-1) \vee P(x,y+1)) \vee B(x,y))$

  $\wedge \neg B(1,1) \wedge P(1,2)$

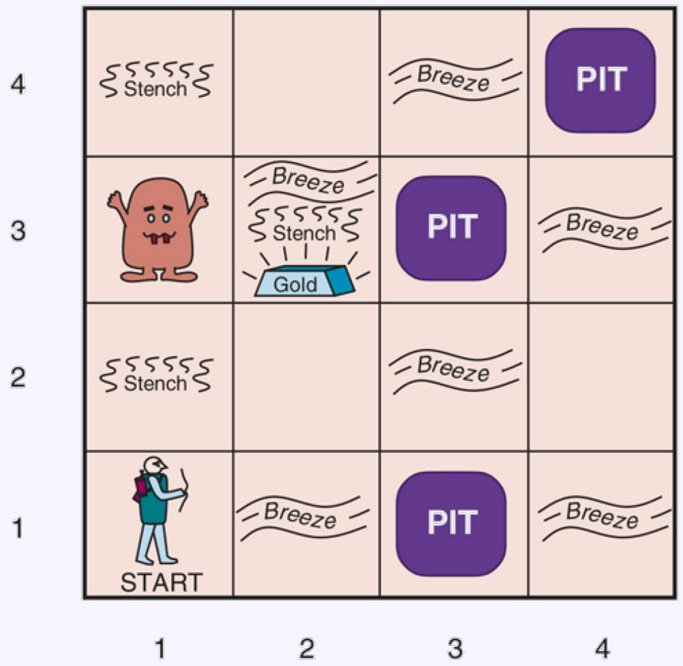
- 否定深入

  $(\forall\ x,y\ (\neg B(x,y) \vee P(x\text{-}1,y) \vee P(x+1,y) \vee P(x,y\text{-}1) \vee P(x,y+1))$

  $\wedge\ \ (\neg P(x-1,y) \wedge \neg P(x+1,y) \wedge \neg P(x,y-1) \wedge \neg P(x,y+1)) \vee B(x,y))$

  $\wedge \neg B(1,1) \wedge P(1,2)$

- 变量换名
- 存在量词消去（斯柯林化）



- (x, y) 有微风：$B(x, y)$
- (x, y) 有陷阱：$P(x, y)$
- $\neg B(1, 1)$
- $\forall\ x, y\ B(x, y) \Leftrightarrow P(x-1, y) \vee P(x+1, y) \vee P(x, y-1) \vee P(x, y+1)$

$\neg P(1,2)$?

从怪兽开始，以怪兽结束

$(\forall\ x,y\ B(x,y) \Leftrightarrow P(x\text{-}1,y) \vee P(x+1,y) \vee P(x,y\text{-}1) \vee P(x,y+1)) \wedge \neg B(1,1) \wedge P(1,2)$

- 全称量词消去

$$(\neg B(x,y) \vee P(x-1,y) \vee P(x+1,y) \vee P(x,y-1) \vee P(x,y+1))$$
$$\wedge((\neg P(x-1,y) \wedge \neg P(x+1,y) \wedge \neg P(x,y-1) \wedge \neg P(x,y+1)) \vee B(x,y))$$
$$\wedge\ \neg B(1,1) \wedge P(1,2)$$

- 分配律

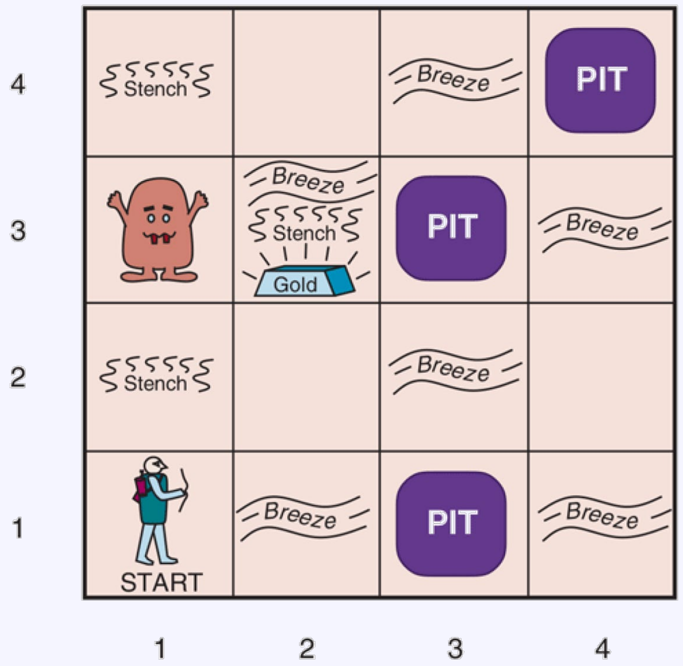
$$(\neg B(x,y) \vee P(x-1,y) \vee P(x+1,y) \vee P(x,y-1) \vee P(x,y+1))$$
$$\wedge (\neg P(x-1,y) \vee B(x,y)) \wedge (\neg P(x+1,y) \vee B(x,y)) \wedge (\neg P(x,y-1) \vee B(x,y))$$
$$\wedge (\neg P(x,y+1) \vee B(x,y))$$
$$\wedge\ \neg B(1,1) \wedge P(1,2)$$

- 归结

$\neg P(x,y+1) \vee B(x,y)$

$\neg B(1,1)$

$\theta = \{x/1, y/1\}$ $\Rightarrow$ $\neg P(1,2)$, $P(1,2)$ $\Rightarrow$ $\square$



- (x, y) 有微风：$B(x, y)$
- (x, y) 有陷阱：$P(x, y)$
- $\neg B(1, 1)$
- $\forall\ x, y\ B(x, y) \Leftrightarrow P(x\text{-}1, y) \vee P(x+1, y) \vee P(x, y\text{-}1) \vee P(x, y+1)$

$\neg P(1,2)$?

演绎推理与归结原理

演绎推理

- 基于假言推理
- 对知识库有限定性要求
- 不能处理非限定知识库
- 推理方法很直观
- 推理效率非常高（线性复杂度）
- 推理可解释性强

归结原理

- 基于归谬法
- 对知识库无限定性要求
- 可以处理非限定知识库
- 推理方法不直观
- 推理效率不太高
- 推理可解释性弱

等值变换可能丢失原语句的含义

$$\neg A \Rightarrow (B \vee C) \qquad (\neg A \wedge \neg B) \Rightarrow C$$

$$\neg B \Rightarrow (C \vee A) \qquad (\neg B \wedge \neg C) \Rightarrow A$$

$$\neg C \Rightarrow (A \vee B) \qquad (\neg C \wedge \neg A) \Rightarrow B$$

$$A \vee B \vee C$$

Thank you very much!