人工智能基础

教学安排

周次	日期	内容	作业	编程	项目	内容
1	9月15日	绪论				绪论。课程内容。Python 简介
2	9月22日	无信息搜索		1		状态空间表示，宽度优先，一致代价，深度优先
3	9月29日	有信息搜索	1		1	算法复杂度分析。A*算法。问题求解项目
4	10月 6日	约束满足		2		启发函数的设计，约束满足问题，回溯搜索
5	10月13日	对抗搜索	2			局部搜索，极小极大搜索，蒙特卡洛树搜索
6	10月20日	命题逻辑	3			命题逻辑，演绎定理，归结原理
7	10月27日	谓词逻辑	4			谓词逻辑
8	11月 3日	线性回归	5			线性回归
9	11月10日	Logistic 回归		3	2	Logistic 回归，Softmax 回归
10	11月17日	前馈神经网络	6			前馈神经网络，机器学习项目
11	11月24日	卷积神经网络		4		卷积神经网络，深度学习框架
12	12月 1日	马尔可夫决策过程	7			强化学习的数学基础
13	12月 8日	策略迭代与价值迭代	8			状态转移概率已知时的预测与控制
14	12月15日	蒙特卡洛与时序差分		5		状态转移概率未知时的预测与控制
15	12月22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

人工智能基础

Fundamentals of Artificial Intelligence

江瑞

自动化系

命题逻辑 — 事实是否成立

- 假设：***The world consists of facts that either hold or do not hold***
- 优点
  - 陈述性语言：语义从语句及模型得来
  - 有表达能力：甚至能处理不完全信息
  - 上下文无关：语法不依赖上下文语境
  - 语义合成性：语义来自于语句各部分
  - 没有歧义性：不存在似是而非的语义
- 缺点
  - 缺乏足够的表达能力，难以简洁地描述多个事实
    - “清朝的皇帝比明朝的皇帝善于骑马” — 自然语言
    - 努尔哈赤比朱元璋善于骑马
    - 皇太极比朱元璋善于骑马
    - 福临比朱元璋善于骑马
    - …

自然语言

- 具有强大的表达能力
- 缺点
  - 上下文相关：语义取决于语句的上下文
  - 存在歧义性：同样的语句有不同的语义

海水朝朝朝朝朝朝朝落
浮云长长长长长长长消

海水潮，朝朝潮，朝潮朝落
浮云涨，长长涨，长涨长消

海水朝潮，朝朝潮，朝朝落
浮云长涨，长长涨，长长消

海水朝潮，朝潮朝潮，朝落
浮云长涨，长涨长涨，长消

I think **that that that that that** student wrote on the blackboard was wrong.

如何结合命题逻辑与自然语言的优势

- 表达能力强
- 上下文无关
- 语义合成的
- 没有歧义的

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

谓词逻辑 — 关系是否成立

- 假设：***The world consists of objects with certain relations among them that do or do not hold***
  - 命题逻辑：事实是否成立
  - 谓词逻辑：关系是否成立

命题逻辑

自然语言

  - 结合命题逻辑与自然语言各自的优势

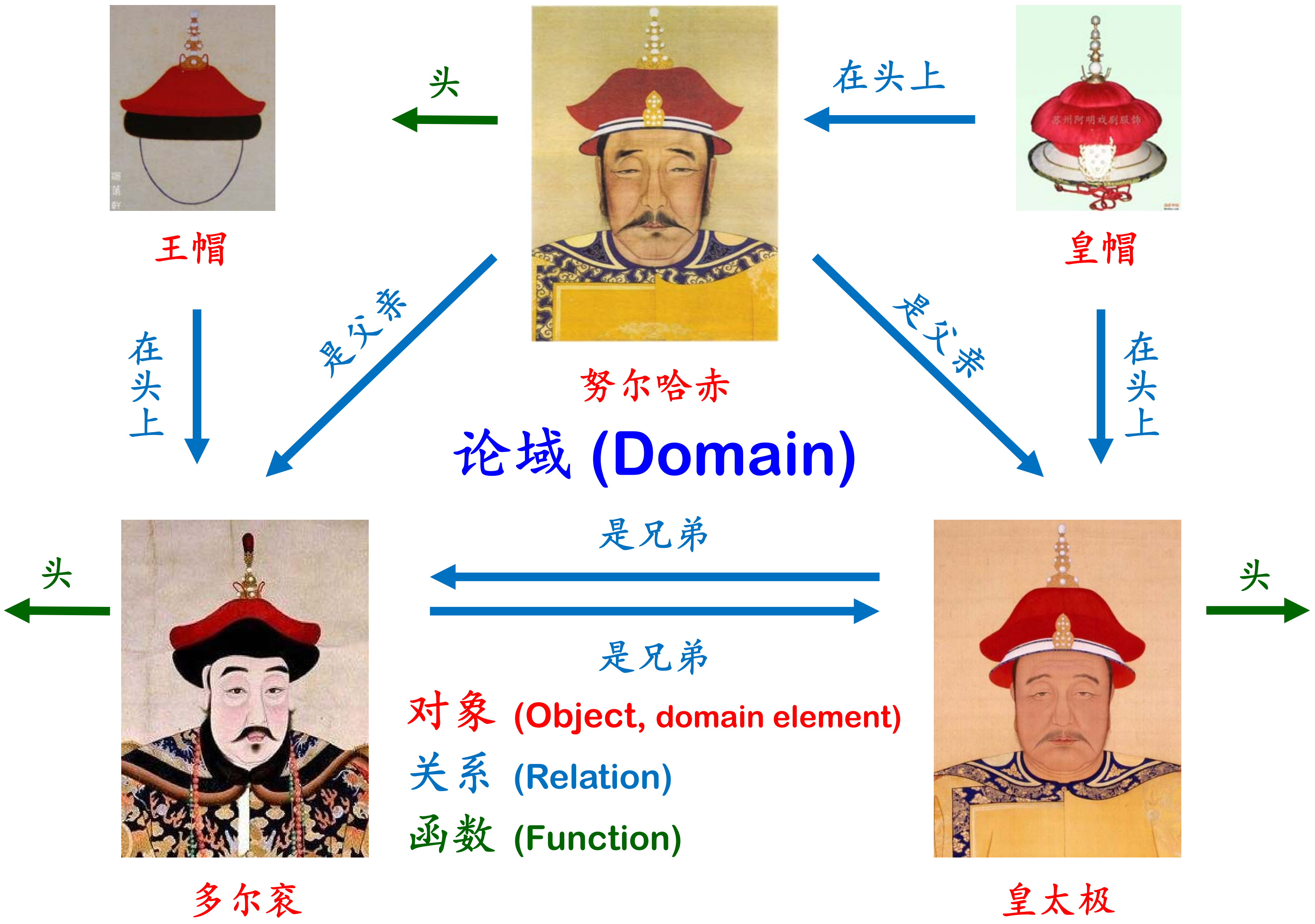
- 一阶谓词逻辑：关系和函数本身是固定的
- 高阶谓词逻辑：关系和函数本身也是对象

自然语言的关键元素

- 主谓宾
  - 皇太极骑战马
  - 特朗普吃牛排
- 主系表
  - 煤是黑色　　炭比煤更黑　　这是我的头
  - 我个子高　　我比你更胖　　这是我的手
- 名词：指代事物（对象，Object）
  - 皇太极、特朗普、战马、牛排，人物、动物、数字、定理 ...
- 动词：表示对象之间的关系（Relation）
  - 一元关系：颜色、高矮、胖瘦 ...　　属性（Property）
  - 二元关系：高于、长于、部分 ...　　关系（Relation）
  - 结果唯一：父亲、母亲、爷爷 ...　　函数（Function）

- 数学语言
  - $1 + 2 = 3$
  - 对象：1，2，3，1+2
  - 关系：=
  - 函数：+

关系



王帽
头
在头上
皇帽
在头上
是父亲
努尔哈赤
是父亲
在头上
论域 (Domain)
头
是兄弟
头
是兄弟
对象 (Object, domain element)
关系 (Relation)
函数 (Function)
多尔衮
皇太极

语法

Grammar

符号

- 表示论域中对象、关系、函数的字母或单词

- 常量符号：首字母大写的名词短语，努尔哈赤、皇太极、多尔衮
- 谓词符号：首字母大写的动词短语，在头上、是父亲、是兄弟
- 函数符号：首字母大写的名词短语，头、腿、手

- Constant symbols: *Nurhachi*, *Taiji*, *Dorgon* …
- Predicate symbols: *OnHead*, *Father*, *Brother* …
- Function symbols: *Head*, *Leg*, *Hand* …

解释

- 进行符号和对象、关系、函数之间映射的规则
  - 将常量符号映射到对象
  - 将谓词符号映射到关系
  - 将函数符号映射到函数

- *Nurhachi*: 指代努尔哈赤
- *Taiji*: 指代皇太极
- *Dorgon*: 指代多尔衮
- *OnHead*: 指代在头上关系
- *Father*: 指代是父亲关系
- *Brother*: 指代是兄弟关系
- *Head*: 指代头函数
- *Hand*: 指代手函数
- *Leg*: 指代腿函数

项　　Term

- **指代对象的表达式称为项 (Term)**
- 语法
  - 常量符号：*Nurhachi*, *Taiji*, *Dorgon*
  - 函数符号作用的结果：*Head*(*Nurhachi*) — 努尔哈赤的头

    函数的结果是唯一，因此不用再为 *Head of Nurhachi* 命名

    可以认为 ***Head*(*Nurhachi*) 就是一个对象**的名称
- 语义
  - 函数作用到对象，得到另一个对象

    LeftLeg(*Taiji*) — 皇太极的左腿

    RightHand(*Gorgon*) — 多尔衮的右手

原子语句 Atomic sentence

- 谓词符号作用于项的结果称为原子语句
- 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
  - *Married*(*Father*(*Taiji*), *Mother*(*Gorgon*)) — 皇太极的父亲和多尔衮的母亲结婚了
- 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 Complex sentence

- **由逻辑连接符相连的语句**
- 语法：同命题逻辑
  - $\neg Emperor(\text{Gorgon})$ — 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
  - $Head(Taiji) \vee Head(\text{Gorgon})$ — 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

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

全称量词 (∀) Universal quantifier

- 谓词逻辑通过**全称量词指代论域中的所有对象**

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

  - 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**

全称量词隐含了合取关系

- $\forall x$, 论域中的任意对象
  - 努尔哈赤、皇太极、多尔衮
  - 皇帽、王帽、努尔哈赤的头、皇太极的头、多尔衮的头 …
- 扩展解释 $\forall x \quad Emperor(x) \Rightarrow Person(x)$
  - $Emperor(Nurhachi) \Rightarrow Person(Nurhachi)$
  - $Emperor(Taiji) \Rightarrow Person(Taiji)$
  - $Emperor(Dorgon) \Rightarrow Person(Dorgon)$
  - $Emperor(\text{皇帽}) \Rightarrow Person(\text{皇帽})$
  - $Emperor(Head(Nurhachi)) \Rightarrow Person(Head(Nurhachi))$
  - … … 所有这些语句是合取的关系
- 隐含关系什么时候为真? 什么时候为假? 用处是什么?
- $\forall x$，只用考虑前项为真的隐含关系
- 使用全称量词 $\forall$ 时，隐含关系 $\Rightarrow$ 是完美的选择

存在量词 (Existential quantifier, ∃)

- 谓词逻辑通过**存在量词指代论域中的某个或某些对象**

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

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

存在量词隐含了析取关系

- $\exists x$, 论域中的某个或某些对象（至少一个，at least）
  - 努尔哈赤、皇太极、多尔衮、皇帽、王帽、努尔哈赤的头、皇太极的头、多尔衮的头 …
  - 其中有至少一个（一个或者多个）
- 扩展解释 $\exists x \quad Crown(x) \wedge OnHead(x, Taiji)$
  - $Crown(Nurhachi) \wedge OnHead(Nurhachi, Taiji)$
  - $Crown(Taiji) \wedge OnHead(Taiji, Taiji)$
  - $Crown(Dorgon) \wedge OnHead(Dorgon, Taiji)$
  - $Crown(\text{皇帽}) \wedge OnHead(\text{皇帽}, Taiji)$
  - $Crown(Head(Nurhachi)) \wedge OnHead(Head(Nurhachi), Taiji)$
  - … … 所有这些语句是析取的关系
- 只要这些语句有一个是真就行
- 使用存在量词 $\exists$ 时，合取关系 $\wedge$ 是完美的选择

$\forall \Rightarrow$ 在一起，$\exists \land$ 在一起

- 考虑 $\forall x \quad Emperor(x) \land Person(x)$
  - $Emperor(Nurhachi) \land Person(Nurhachi)$
  - $Emperor(Taiji) \land Person(Taiji)$
  - $Emperor(Dorgon) \land Person(Dorgon)$
  - $Emperor(\text{皇帽}) \land Person(\text{皇帽})$
  - $Emperor(Head(Nurhachi)) \land Person(Head(Nurhachi))$
- 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！

- $\forall$ 作为限定词的情况下，使用 $\land$ 会使陈述过强

$\forall \Rightarrow$ 在一起，$\exists \wedge$ 在一起

- 考虑 $\exists x \;\; Crown(x) \Rightarrow OnHead(x, \; Taiji)$
  - $Crown(Nurhachi) \Rightarrow OnHead(Nurhachi, \; Taiji)$
  - $Crown(Taiji) \Rightarrow OnHead(Taiji, \; Taiji)$
  - $Crown(Dogon) \Rightarrow OnHead(Dogon, \; Taiji)$
  - $Crown(\text{皇帽}) \Rightarrow OnHead(\text{皇帽}, \; Taiji)$
  - $Crown(Head(Nurhachi)) \Rightarrow OnHead(Head(Nurhachi), \; Taiji)$
- 前项为假时，隐含式为真，结果是？

上面的子句全都是真！

- $\exists$ 作为限定词的情况下，使用 $\Rightarrow$ 会使陈述过弱

量词嵌套

- 多个量词可以嵌套起来，表示更复杂的语句

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

- 同一种类型的多个量词可以写到一起

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

- 混合量词

$\forall x\, \exists y\ Loves(x, y)$	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

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

- 嵌套时使用不同的变量

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

$\forall$ 和 $\exists$ 的关系

- 互反性

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

每个人都喜欢吃冰淇淋 $\equiv$ 没有人不喜欢吃冰淇淋

$\forall x\ Likes(x,\ IceCream)$ 每个人都喜欢吃冰淇淋

$\equiv Likes(a,\ IceCream) \wedge Likes(b,\ IceCream) \wedge Likes(c,\ IceCream) \wedge \ldots$

$\neg(\forall x\ Likes(x,\ IceCream))$ 不是每个人都喜欢吃冰淇淋

$\equiv \neg(Likes(a,\ IceCream) \wedge Likes(b,\ IceCream) \wedge Likes(c,\ IceCream) \wedge \ldots)$

$\equiv \neg Likes(a,\ IceCream) \vee \neg Likes(b,\ IceCream) \vee \neg Likes(c,\ IceCream) \wedge \ldots)$

$\equiv \exists x\ \neg Likes(x,\ IceCream)$ 有的人不喜欢吃冰淇淋

$\forall x\ Likes(x,\ IceCream)$ 每个人都喜欢吃冰淇淋

$\equiv \neg(\neg(\forall x\ Likes(x,\ IceCream)))$ 不是不是每个人都喜欢吃冰淇淋

$\equiv \neg(\exists x\ \neg Likes(x,\ IceCream))$ 不是有的人不喜欢吃冰淇淋

$\equiv \neg\exists x\ \neg Likes(x,\ IceCream)$ 没有人不喜欢吃冰淇淋

$\forall$ 和 $\exists$ 的互反性与摩根律

- 互反性

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

- 摩根律

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

- $\forall x\ \neg P(x) \equiv \neg P(a) \wedge \neg P(b) \wedge \neg P(c) \ldots \ldots$
- $\neg(\forall x\ \neg P(x)) \equiv \neg(\neg P(a) \wedge \neg P(b) \wedge \neg P(c) \ldots \ldots)$
- $\equiv \neg(\neg P(a)) \vee \neg(\neg P(b)) \vee \neg(\neg P(c)) \ldots \ldots)$
- $\exists x\ P(x) \equiv P(a) \vee P(b) \vee P(c) \ldots \ldots$
- $\forall x\ \neg P(x) \equiv \neg(\neg(\forall x\ \neg P(x))) \equiv \neg(\exists x\ P(x)) \equiv \neg\exists x\ P(x)$

等词

- 声明两个项指代同一个对象

$$Father(Taiji) = Nurhachi$$

- 皇太极至少有两个兄弟
  - x 和 y 可以是同一个人

$$\exists x,y \; Brother(Taiji,x) \land Brother(Taiji,y)$$

  - x 和 y 不能是同一个人

$$\exists x,y \; Brother(Taiji,x) \land Brother(Taiji,y) \land \neg(x=y)$$

$$\exists x,y \; Brother(Taiji,x) \land Brother(Taiji,y) \land x \neq y$$

自然语言的形式化

- 谓词逻辑能够较好地描述自然语言
- 没有共产党就没有新中国

  $\forall x\ \neg CommunistParty(x) \Rightarrow \neg BuildNewChina(x)$

  $\forall x\ BuildNewChina(x) \Rightarrow CommunistParty(x)$
- 只有共产党才能救中国

  $\forall x\ \neg CommunistParty(x) \Rightarrow \neg SaveChina(x)$

  $\forall x\ SaveChina(x) \Rightarrow CommunistParty(x)$
- 四项基本原则是立国之本

  $\forall x\ FourCardinalPrinciples(x) \Rightarrow FoundationOfCountry(x)$

  $\forall x\ \neg FoundationOfCountry(x) \Rightarrow \neg FourCardinalPrinciples(x)$
- 我们要坚持思政教育

  $\forall x\ PoliticalEducation(x) \Rightarrow WeInsist(x)$

自然语言的形式化

- 大家要好好学习天天向上

  $\forall x\ StudyHard(x) \wedge (\forall y\ MakeProgress(x,y))$

- 要玩得好也要学得好

  $\forall x\ PlayWell(x) \wedge StudyHard(x)$

- 双十一的东西又好又便宜

  $\forall x\ SaledInDoubleEleven(x) \Rightarrow (Good(x) \wedge Cheap(x))$

- 双十一的东西也有质量不好还挺贵的

  $\exists x\ SaledInDoubleEleven(x) \wedge LowQuality(x) \wedge Expensive(x)$

- 我们班有不用怎么学习就考高分的人

  $\exists x\ InOurClass(x) \wedge NotStudyMuch(x) \wedge ReceiveHighScore(x)$

自然语言的形式化

- 谓词逻辑还能描述其他语言
- 所有有理数都是实数
  - $\forall x\ RatNum(x) \Rightarrow RealNum(x)$
  - $RatNum$: 谓词，对有理数为真
  - $RealNum$: 谓词，对实数为真
- 有的实数是有理数
  - $\exists x\ RealNum(x) \wedge RatNum(x)$
- 没有无理数是有理数
  - $\forall x\ IrrNum(x) \Rightarrow \neg RatNum(x)$
  - $\forall x\ RatNum(x) \Rightarrow \neg IrrNum(x)$
  - $\neg \exists x\ RatNum(x) \wedge IrrNum(x)$
  - $IrrNum$: 谓词，无理数为真

定义自然数

- *NatNum*: a predicate that will be true of natural numbers
- 0: a constant symbol
- *S*: a function symbol, meaning successor

- 定义
  1. $NatNum(0)$
  2. $\forall n\ NatNum(n) \Rightarrow NatNum(S(n))$
  3. $\forall n\ 0 \neq S(n)$
  4. $\forall m,n\ m \neq n \Rightarrow S(m) \neq S(n)$
- 加法
  1. $\forall m\ NatNum(m) \Rightarrow +(0,m) = m$
  2. $\forall m,n\ NatNum(m) \wedge NatNum(n) \Rightarrow +(S(m),n) = S(+(m,n))$

Prefix:	$+(m,n)$
Infix:	$m+n$
$S(n)$:	$n+1$

语义

Semtantics

等值演算

- 基本原理：**命题逻辑的等值演算 + 对量词的处理**

- 变量换名

$$\forall x\ P(x,\ \ldots) \equiv \forall y\ P(y,\ \ldots)$$
$$\exists x\ P(x,\ \ldots) \equiv \exists y\ P(y,\ \ldots)$$

- 量词消去

$$\begin{aligned}\forall x\ P(x) &\equiv P(a) \wedge P(b) \wedge P(c) \wedge \ldots \\ \exists x\ P(x) &\equiv P(a) \vee P(b) \vee P(c) \vee \ldots\end{aligned}$$

- 量词否定

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

全称量词辖域收缩与扩张

Q 里不能出现 x

- 全称量词

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

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

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

$$\forall x\,(Q\Rightarrow P(x))\equiv Q\Rightarrow(\forall x\,P(x))$$

$$\begin{aligned}\forall x\,(P(x)\wedge Q) &\equiv (P(a)\wedge Q)\wedge(P(b)\wedge Q)\wedge(P(b)\wedge Q)\wedge\ldots\\ &\equiv (P(a)\wedge P(b)\wedge P(b)\wedge\ldots)\wedge Q\\ &\equiv (\forall x\,P(x))\wedge Q\end{aligned}$$

$$\begin{aligned}\forall x\,(P(x)\vee Q) &\equiv (P(a)\vee Q)\wedge(P(b)\vee Q)\wedge(P(c)\vee Q)\wedge\ldots\\ &\equiv ((P(a)\wedge P(b))\vee Q)\wedge(P(c)\vee Q)\wedge\ldots\\ &\equiv ((P(a)\wedge P(b)\wedge P(c))\vee Q)\wedge\ldots\\ &\equiv (P(a)\wedge P(b)\wedge P(c)\wedge\ldots)\vee Q\\ &\equiv (\forall x\,P(x))\vee Q\end{aligned}$$

全称量词辖域收缩与扩张

Q 里不能出现 x

- 全称量词

$$\forall x\,(P(x)\land Q) \equiv (\forall x\,P(x)) \land Q$$

$$\forall x\,(P(x)\lor Q) \equiv (\forall x\,P(x)) \lor Q$$

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

$$\forall x\,(Q\Rightarrow P(x)) \equiv Q \Rightarrow (\forall x\,P(x))$$

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

$$\begin{aligned}
\forall x\,(Q\Rightarrow P(x)) &\equiv \forall x\,(\neg Q \lor P(x)) \\
&\equiv \forall x\,(P(x) \lor \neg Q) \\
&\equiv \forall x\,(P(x)) \lor \neg Q \\
&\equiv \neg Q \lor (\forall x\,P(x)) \\
&\equiv Q \Rightarrow (\forall x\,P(x))
\end{aligned}$$

存在量词辖域收缩与扩张

Q 里不能出现 x

- 全称量词

$$\forall x\,(P(x)\wedge Q) \equiv (\forall x\, P(x)) \wedge Q$$
$$\forall x\,(P(x)\vee Q) \equiv (\forall x\, P(x)) \vee Q$$
$$\forall x\,(P(x)\Rightarrow Q) \equiv (\exists x\, P(x)) \Rightarrow Q$$
$$\forall x\,(Q\Rightarrow P(x)) \equiv Q \Rightarrow (\forall x\, P(x))$$

- **存在量词**

$$\exists x\,(P(x)\wedge Q) \equiv (\exists x\, P(x)) \wedge Q$$
$$\exists x\,(P(x)\vee Q) \equiv (\exists x\, P(x)) \vee Q$$
$$\exists x\,(P(x)\Rightarrow Q) \equiv (\forall x\, P(x)) \Rightarrow Q$$
$$\exists x\,(Q\Rightarrow P(x)) \equiv Q \Rightarrow (\exists x\, P(x))$$

同理可证

全称量词的分配

- 全称量词可以对合取分配，不能对析取分配

$$\forall x\,(P(x) \land Q(x)) \equiv \forall x\,P(x) \land \forall x\,Q(x)$$

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

$$\begin{aligned}\forall x\,(P(x)\land Q(x)) &\equiv (P(a)\land Q(a)) \land (P(b)\land Q(b)) \land (P(c)\land Q(c)) \land \ldots \\ &\equiv (P(a)\land P(b)\land P(c)\land\ldots) \land (Q(a)\land Q(b)\land Q(c)\land\ldots) \\ &\equiv (\forall x\,P(x)) \land (\forall x\,Q(x))\end{aligned}$$

- 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

存在称量词的分配

- 存在量词可以对析取分配，不能对合取分配

$$\exists x\,(P(x) \lor Q(x)) \equiv \exists x\, P(x) \lor \exists x\; Q(x)$$

$$\exists x\,(P(x) \land Q(x)) \equiv \exists x\, P(x) \land \exists x\; Q(x)$$ **错！**

$$\begin{aligned}\exists x\,(P(x)\lor Q(x)) &\equiv (P(a)\lor Q(a)) \lor (P(b)\lor Q(b)) \lor (P(c)\lor Q(c)) \lor \ldots \\ &\equiv (P(a)\lor P(b)\lor P(c)\lor\ldots) \lor (Q(a)\lor Q(b)\lor Q(c)\lor\ldots) \\ &\equiv (\exists x\, P(x)) \lor (\exists x\, Q(x))\end{aligned}$$

- 反例
  - $\exists x\,(P(x) \land Q(x))$
    自动化系有一些同学，学习成绩好并且社会工作好
  - $\exists x\, P(x) \land \exists x\; Q(x)$
    自动化系有一些同学学习成绩好，并且
    自动化系有一些同学社会工作好

量词分配

- 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

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

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

$\forall x\ (P(x) \vee Q(x)) \equiv \forall x\ P(x) \vee \forall x\ Q(x)$ **错！**

$\exists x\ (P(x) \wedge Q(x)) \equiv \exists x\ P(x) \wedge \exists x\ Q(x)$ **错！**

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

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

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

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

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

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

有什么特点？

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
- 一阶逻辑中任何语句均可转换为与之等值的前束范式
- 前束范式不唯一

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

$$\boldsymbol{\forall x\ \exists y\ \forall z\ ((P(x,\ w)\ \Rightarrow\ Q(y))\Rightarrow\ R(z,\ w))}$$ **前束范式**

前面的全称量词存在量词怎么办?

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**置换应为函数**
  - 置换：$\boldsymbol{\theta = \{y/F(x)\}}$ 给 x 起名$\boldsymbol{F(x)}$，　$\boldsymbol{F}$不能在其他语句中出现
  - 形式：$\boldsymbol{\theta = \{v/c\}}$ **v: variable, $F(x)$: Skolem function**

合取范式

- 在前束范式的基础上，消去量词，转化为析取子句的合取形式
- 得到的结果称为合取范式，获取合取范式的步骤：

1. 等价消去
2. 隐含消去
3. 否定深入
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

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

- 等价消去：$P \Leftrightarrow Q \equiv (P \Rightarrow Q) \wedge (Q \Rightarrow P)$
- 隐含消去：$P \Rightarrow \mathrm{Q} \equiv \neg P \vee Q$

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

- 否定深入：$\neg(\forall y\ P) \equiv \exists y\ \neg P,\ \neg(\exists y\ P) \equiv \forall y\ \neg P$

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

- **变量换名：**

  $\forall x\ [(\exists y\ Person(y) \wedge \neg Loves(x, y)) \vee (\exists z\ Loves(z, x))]$

- **消去存在量词：斯柯林化**

  $\forall x\ [(Person(F(x)) \wedge \neg Loves(x, F(x))) \vee Loves(G(x), x)]$

- **消去全称量词：直接消去**

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

- 分配律：

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

演绎推理

Deduction

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

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

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

- 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，谓词逻辑转化为了命题逻辑

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

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

- 于是可以使用假言推理了

更一般的情况

- 贪婪的皇帝都是暴君，人都是贪婪的，皇太极是皇帝

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

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

$$Emperor(Nurhachi)$$

- 置换

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

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

$$Greedy(Nurhachi)$$

$$Emperor(Nurhachi)$$

- 结论： $Tyrant(Nurhachi)$
- 特点：置换 θ 使不同的逻辑表达式变得相同

合一

- 使多个语句变得相同的置换称为合一

  $Knows(John, x)$ 和 $Knows(John, Jane)$, 在 $\{x/Jane\}$ 下合一

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

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

- 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

  $Knows(John, x)$ 和 $Knows(x, Elizabeth)$ 看上去不能合一

  $Knows(John, x)$ 和 $Knows(y, Elizabeth)$ 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 $(\alpha \land (\alpha \Rightarrow \beta)) \vDash \beta$
- 命题逻辑：如果 $P_1, \ldots, P_n$ 为真，且 $P_1 \land \ldots \land P_n \Rightarrow Q$，则 Q 为真
- 谓词逻辑：如果 $P_1, \ldots, P_n$ 为真，$P_1' \land \ldots \land P_n' \Rightarrow Q$，
  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

$Emperor(Nurhachi)$

$\forall y\ Greedy(y)$

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

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

置换

$Emperor(Nurhachi)$

$Greedy(Nurhachi)$

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

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

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

$\boldsymbol{x/Tianyou}$

$\boldsymbol{Hongyipo(Tianyou)}$

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

$Hongyipo(Tianyou)$

$Owns(Qing,\ Tianyou)$

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

$\boldsymbol{x/Tianyou}$

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

$Enemy(Qing,\ Ming)$

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

$\boldsymbol{x/Qing}$

$\boldsymbol{Hostile(Qing)}$

$Hongyipo(Tianyou)$

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

$\boldsymbol{x/Tianyou}$

$\boldsymbol{Weapon(Tianyou)}$

$Ming(Liuhan)$

$Weapon(Tianyou)$

$Makes(Liuhan,\ Tianyou,\ Qing)$

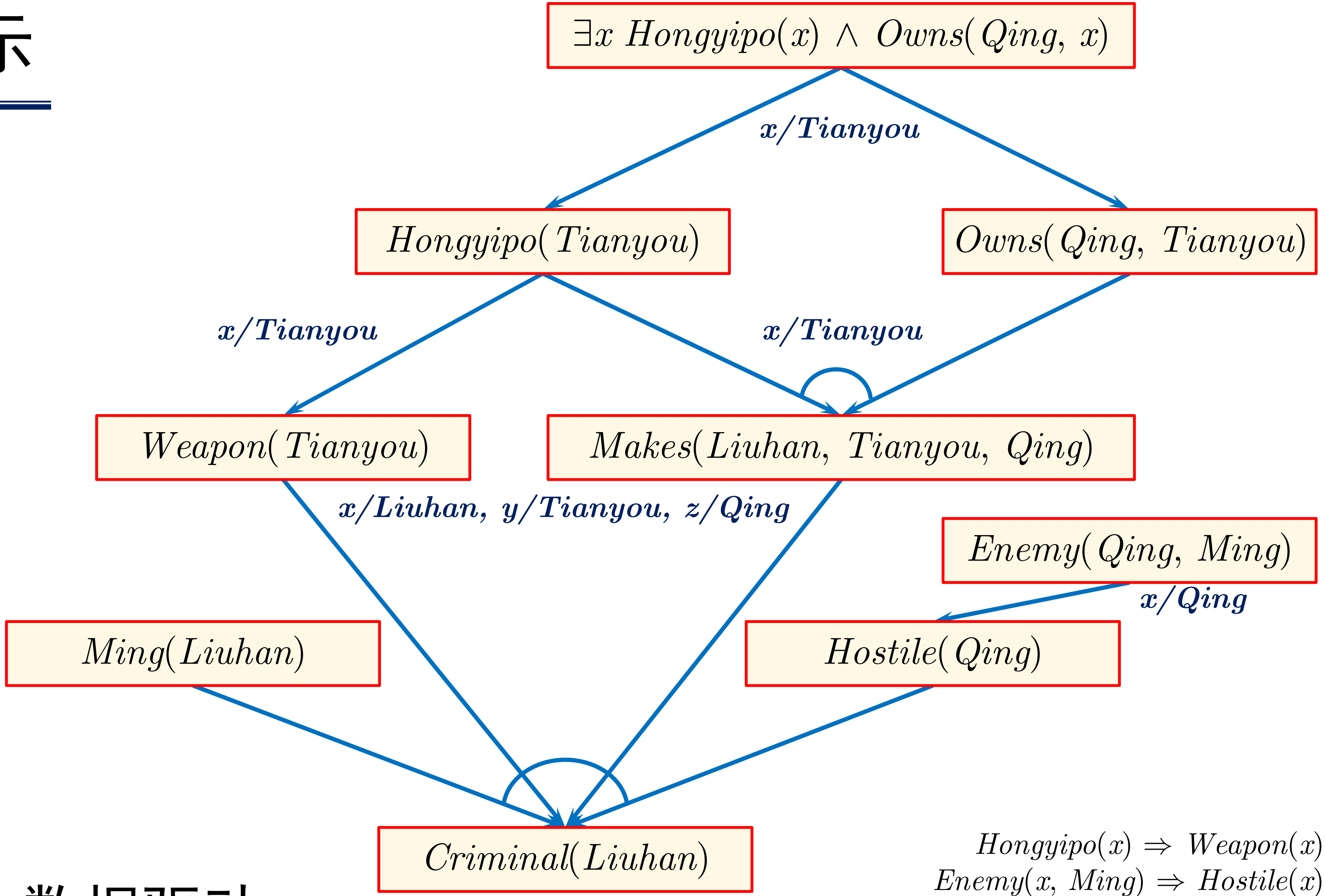
$Hostile(Qing)$

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

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

$\boldsymbol{Criminal(Liuhan)}$

与或图表示



数据驱动

$$Hongyipo(x) \Rightarrow Weapon(x)$$
$$Enemy(x, Ming) \Rightarrow Hostile(x)$$
$$Hongyipo(x) \wedge Owns(Qing, x) \Rightarrow Makes(Liuhan, x, Qing)$$
$$Ming(x) \wedge Weapon(y) \wedge Makes(x, y, z) \wedge Hostile(z) \Rightarrow Criminal(x)$$

归结原理

Resolution rule

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**

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

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

$\neg Loves(y, Taiji)$

$Loves(G(Taiji), Taiji)$

$$\xrightarrow[\text{置换}]{\theta = \{y/G(Taiji)\}}$$

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

$Loves(G(Taiji), Taiji)$

是谁杀了褚英

仁厚之人为民所爱
残暴之人为民所弃
皇太极仁厚之人也
褚英为其父弟所害
诛英者其父耶弟耶

爱新觉罗·褚英是清太祖爱新觉罗·努尔哈赤的长子。努尔哈赤的胞弟爱新觉罗·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。

归结证明

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

爱新觉罗·褚英是清太祖爱新觉罗·努尔哈赤的长子。努尔哈赤的胞弟爱新觉罗·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。

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

$$Kills(Nurhachi,\ Cuyen)$$

转换为合取范式，获得子句集

- 仁厚之人为民所爱

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

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

- 残暴之人为民所弃

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

- 皇太极仁厚之人也

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

- 褚英为其父弟所害

  $Prince(Cuyen)$

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

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

- 诛英者努尔哈赤也

  $\neg Kills(Nurhachi, Cuyen)$

归结

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

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

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

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

$\boldsymbol{Prince(Cuyen)}$

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

→ $\boldsymbol{Person(Cuyen)}$

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

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

→ $\boldsymbol{Kills(Taiji, Cuyen)}$

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

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

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

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

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

$Person(Cuyen)$

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

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

归结

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

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

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

$\boxed{Person(Cuyen)}$

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

$\Rightarrow \neg Loves(y, Taiji)$

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

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

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

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

$$\neg Loves(y,\ Taiji)$$

$\Rightarrow$ $\boldsymbol{Loves(G(x),\ x) \lor Loves(Taiji,\ F(x))}$

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

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

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

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

$$\neg Loves(y, Taiji)$$

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

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

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

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

- 隐含消去

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

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

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

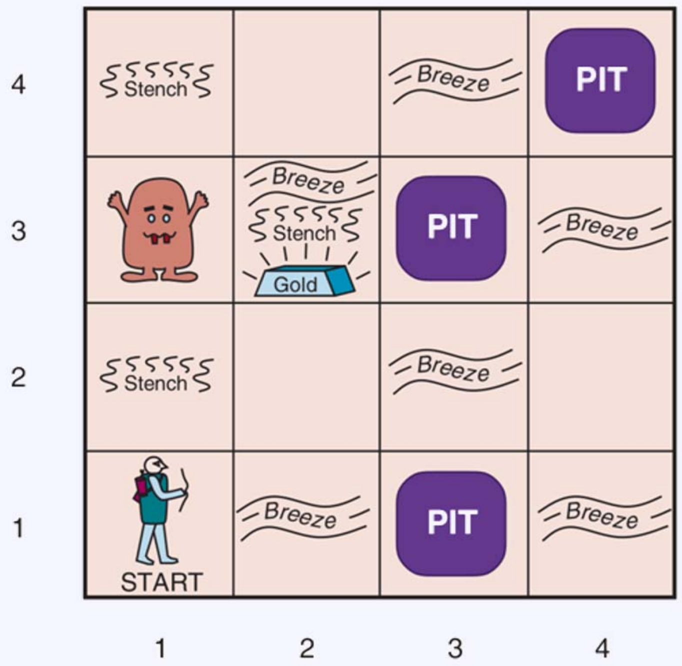
- 否定深入

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

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

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

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

$\neg P(1,2)$?

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

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

- 分配律

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

- 归结

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

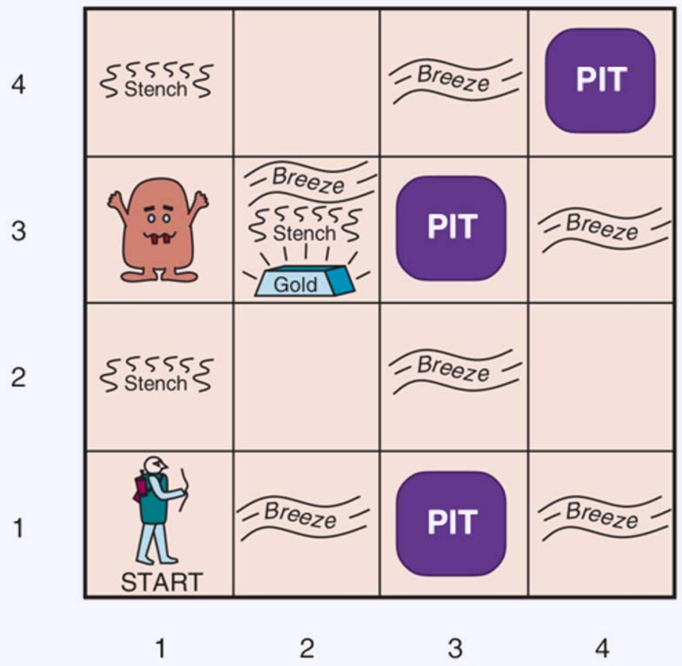
$\neg B(1,1)$

$\theta = \{x/1, y/1\} \Rightarrow$

$\neg P(1,2)$

$P(1,2)$

$\Rightarrow \square$



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

$\neg P(1,2)$?

Thank you very much!