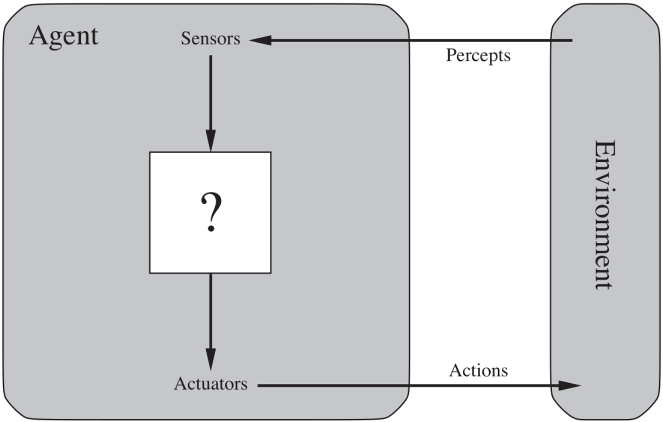
人工智能：经典逻辑推理

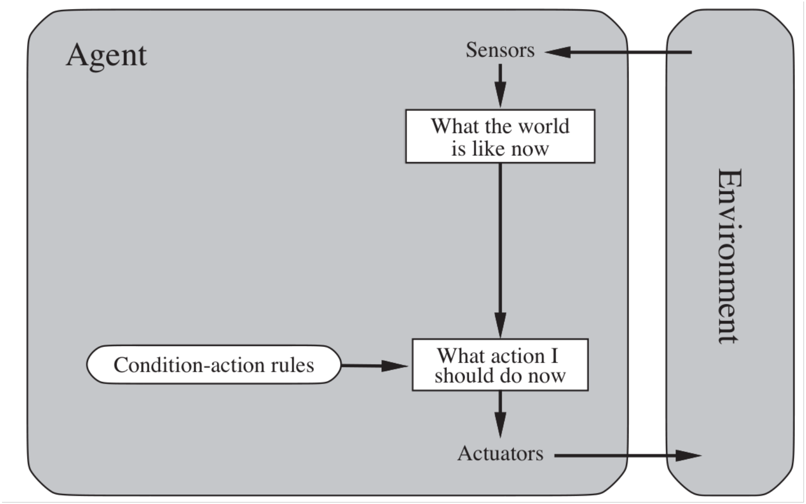
陈冠毅

g.chen@ccnu.edu.cn
计算机学院
华中师范大学

智能体



反射智能体



Agent
Sensors
What the world is like now
Condition-action rules
What action I should do now
Actuators
Environment

How?

1. What the world is like now?
    - 知识及知识表示(第二章)
2. What action I should do now?
    - 经典逻辑推理(第三章) $\leftarrow$
    - 不确定性推理(第四章)

Overview

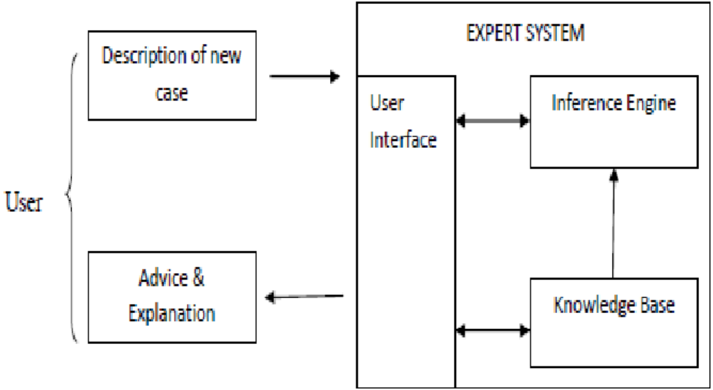
什么是推理

推理就是按某种策略由从某些已知前提导出结论。

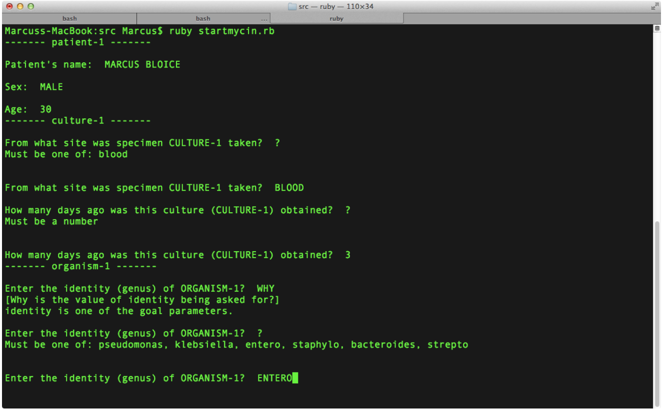
- 已知前提：包括已掌握的与求解问题有关的知识及关于问题的已知事实。
- 结论：新的事实、新的知识...

推理由程序程序实现，称为推理机。

MYCIN


User
Description of new case
Advice & Explanation
EXPERT SYSTEM
User Interface
Inference Engine
Knowledge Base

MYCIN


src — ruby — 110×34
bash
bash
ruby
Marcuss-MacBook:src Marcus$ ruby startmycin.rb
------- patient-1 -------
Patient's name: MARCUS BLOICE
Sex: MALE
Age: 30
------- culture-1 -------
From what site was specimen CULTURE-1 taken? ?
Must be one of: blood
From what site was specimen CULTURE-1 taken? BLOOD
How many days ago was this culture (CULTURE-1) obtained? ?
Must be a number
How many days ago was this culture (CULTURE-1) obtained? 3
------- organism-1 -------
Enter the identity (genus) of ORGANISM-1? WHY
[Why is the value of identity being asked for?]
identity is one of the goal parameters.
Enter the identity (genus) of ORGANISM-1? ?
Must be one of: pseudomonas, klebsiella, entero, staphylo, bacteroides, strepto
Enter the identity (genus) of ORGANISM-1? ENTERO

推理的分类：演绎推理、归纳推理和默认推理

演绎推理:

- 又叫正向推理：$P \Rightarrow Q$
- 亚里士多德："结论，可从叫做前提的已知事实，**必然地**得出的推理"
- 如果前提为真，则结论必然为真。
- 演绎推理无法使知识扩增，因为结论自包含于前提之内。
- 三段论是最常见的演绎推理：
  1. 大前提：已知的一般性知识或假设
  2. 小前提：关于所研究的具体情况或个别事实的判断
  3. 结论：由大前提推出的适合于小前提所示情况的新判断
- 例如：
  1. 人皆有一死；苏格拉底是人；所以，苏格拉底会死。
  2. 任何三角形只可能是锐角三角形、直角三角形和钝角三角形；这个三角形既不是锐角三角形，也不是钝角三角形；所以，它是一个直角三角形。

推理的分类：演绎推理、归纳推理和默认推理

演绎推理:

- 又叫正向推理：$P \Rightarrow Q$
- 亚里士多德：“结论，可从叫做前提的已知事实，**必然地**得出的推理”
- 如果前提为真，则结论必然为真。
- 演绎推理无法使知识扩增，因为结论自包含于前提之内。
- 三段论是最常见的演绎推理：
  1. 大前提：已知的一般性知识或假设
  2. 小前提：关于所研究的具体情况或个别事实的判断
  3. 结论：由大前提推出的适合于小前提所示情况的新判断
- 例如：
  1. 人皆有一死；苏格拉底是人；所以，苏格拉底会死。
  2. 任何三角形只可能是锐角三角形、直角三角形和钝角三角形；这个三角形既不是锐角三角形，也不是钝角三角形；所以，它是一个直角三角形。

推理的分类：演绎推理、归纳推理和默认推理

归纳推理：

- 归纳推理是从足够多的事例中归纳出一般性结论的推理过程，是一种从个别到一般的推理；
- 完全归纳推理和不完全归纳推理
  1. 完全归纳推理是在进行归纳时考察了相应事物的全部对象，并根据这些对象是否都具有某种属性，从而推出这个事物是否具有这个属性；
  2. 不完全归纳推理是指只考察了相应事物的部分对象就得出了结论。
- 当前提为真时，可推出某种几率性的结论。
- 归纳推理可以扩展知识，因为结论比前提包含更多的信息。
- 大卫休谟：在我记得的过去每一天中，太阳都会升起；所以，太阳明天将会升起。

默认推理：又称缺省推理，它是在知识不完全的情况下假设某些条件已经具备所进行的推理。例如，在条件A已成立的情况下，如果没有足够的证据能证明条件B不成立，则默认B是成立的，并在此默认的前提下进行推理，推导出某个结论。

推理的分类：演绎推理、归纳推理和默认推理

归纳推理:

- 归纳推理是从足够多的事例中归纳出一般性结论的推理过程，是一种从个别到一般的推理;
- 完全归纳推理和不完全归纳推理
  1. 完全归纳推理是在进行归纳时考察了相应事物的全部对象，并根据这些对象是否都具有某种属性，从而推出这个事物是否具有这个属性;
  2. 不完全归纳推理是指只考察了相应事物的部分对象就得出了结论。
- 当前提为真时，可推出某种几率性的结论。
- 归纳推理可以扩展知识，因为结论比前提包含更多的信息。
- 大卫休谟：在我记得的过去每一天中，太阳都会升起；所以，太阳明天将会升起。

默认推理：又称缺省推理，它是在知识不完全的情况下假设某些条件已经具备所进行的推理。例如，在条件A已成立的情况下，如果没有足够的证据能证明条件B不成立，则默认B是成立的，并在此默认的前提下进行推理，推导出某个结论。

推理的分类：确定性推理和不确定性推理

- 确定性推理：推理时所用的知识都是精确的，推出的结论也是确定的，其真值或者为真，或为假，没有第三种情况出现。
- 不确定性推理：推理时所用的知识不都是精确的，推出的结论也不完全是肯定的，真值位于真与假之间，命题的外延模糊不清。

推理的分类：单调推理和非单调推理

- 单调推理：在推理过程中随着推理的向前及新知识的加入，推出的结论是呈单调增加的趋势，并且越来越接近最终目标，在推理过程中不出现反复的情况。
- 非单调推理：在推理过程中由于新知识的加入，不仅没有加强已推出的结论，反而要否定它，使得推理退回到前面的某一步，重新开始。

Overview

基本概念

推理的逻辑基础

产生式系统推理

自然演绎推理

归结演绎推理

谓词公式的解释

定义：设D为谓词公式P的个体域。若对P中的个体常量、函数和谓词按如下规定赋值:

1. 为每个个体常量指派D中的一个元素。
2. 为每个n元函数指派一个从D^n到D的映射，其中:

$$D^n = \{(x_1, x_2, ..., x_n) | x_1, x_2, ..., x_n \in D\}$$

3. 为每个n元谓词指派一个从D^n到$\{F, T\}$的映射

则称这些指派为公式P在D上的一个解释

例子

设个体域为$D = \{1, 2\}$，求公式$A = (\forall x)(\exists y)P(x, y)$在$D$上的解释，并指出在每一种解释下的真值。

几个定义

- 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**。
- 对于谓词公式P，如果至少存在一个解释使得公式P在此解释下的真值为T，则称公式P是**可满足**的。
- 如果谓词公式P对于个体域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**。
- 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，记作$P \Leftrightarrow Q$。
- 对于谓词公式P和Q，如果$P \rightarrow Q$永真，则称P永真蕴涵Q，且称Q为P的逻辑结论，称P为Q的前提，记作$P \Rightarrow Q$。

模式匹配

模式匹配是指对两个知识模式的比较与耦合，即检查这两个知识是否完全一致或近似一致。若两个知识完全一致或者虽不一致但其相似程度落在指定的限度内，则称它们是**可以匹配的**，否则不可匹配。

- 确定性匹配: 两个知识模式完全一致，或者经过变量代换后变得完全一致
- 不确定性匹配: 两个知识模式不完全一致，但从整体上看，它们的相似程度落在规定的范围内

代换

定义：代换是形如

$$\{t_1/x_1, t_2/x_2, ..., t_n/x_n\}$$

的有限集合。其中，t_i是项，x_i是变元；t_i/x_i表示用t_i替换x_i，不允许t_i与x_i相同，也不允许变元x_i循环出现在另一个t_i中。

代换

定义：代换是形如

$$\{t_1/x_1, t_2/x_2, ..., t_n/x_n\}$$

的有限集合。其中，t_i是项，x_i是变元；t_i/x_i表示用t_i替换x_i，不允许t_i与x_i相同，也不允许变元x_i循环出现在另一个t_i中。

例：判断以下是不是代换：

1. $\{a/x, c/y, f(b)/z\}$
2. $\{g(a)/x, f(x)/y\}$
3. $\{f(y)/x, f(x)/y\}$

代换

定义：代换是形如

$$\{t_1/x_1, t_2/x_2, ..., t_n/x_n\}$$

的有限集合。其中，t_i是项，x_i是变元；t_i/x_i表示用t_i替换x_i，不允许t_i与x_i相同，也不允许变元x_i循环出现在另一个t_i中。

例：判断以下是不是代换：

1. $\{a/x, c/y, f(b)/z\}$
2. $\{g(a)/x, f(x)/y\}$
3. $\{f(y)/x, f(x)/y\}$

令代换$\theta = \{t_1/x_1, t_2/x_2, ..., t_n/x_n\}$，$P$是一个谓词公式。$P\theta$表示将$P$中的变量$x_i$用$t_i$取代。

代换

定义：代换是形如

$$\{t_1/x_1, t_2/x_2, ..., t_n/x_n\}$$

的有限集合。其中，t_i是项，x_i是变元；t_i/x_i表示用t_i替换x_i，不允许t_i与x_i相同，也不允许变元x_i循环出现在另一个t_i中。

例：判断以下是不是代换：

1. $\{a/x, c/y, f(b)/z\}$
2. $\{g(a)/x, f(x)/y\}$
3. $\{f(y)/x, f(x)/y\}$

令代换$\theta = \{t_1/x_1, t_2/x_2, ..., t_n/x_n\}$，$P$是一个谓词公式。$P\theta$表示将$P$中的变量$x_i$用$t_i$取代。

例：已知$\theta = \{g(a)/x, f(x)/y\}$，$P(x, y)$，求$P\theta$

复合代换

定义：设$\theta = \{t_1/x_1, t_2/x_2, ..., t_n/x_n\}$和$\lambda = \{u_1/y_1, u_2/y_2, ..., u_n/y_n\}$
则这两个代换的复合$\theta \circ \lambda$也是一个代换。

复合代换的求解步骤:

1. 生成代换：$\{t_1\lambda/x_1, t_2\lambda/x_2, ..., t_n\lambda/x_n, u_1/y_1, u_2/y_2, ..., u_n/y_n\}$
2. 当$y_i \in \{x_1, x_2, ..., x_n\}$时，删去$u_i/y_i$。
3. 当$t_i\lambda = \ x_i$时，删去$t_i\lambda/x_i$。

复合代换

定义：设$\theta = \{t_1/x_1, t_2/x_2, ..., t_n/x_n\}$和$\lambda = \{u_1/y_1, u_2/y_2, ..., u_n/y_n\}$
则这两个代换的复合$\theta \circ \lambda$也是一个代换。

复合代换的求解步骤:

1. 生成代换：$\{t_1\lambda/x_1, t_2\lambda/x_2, ..., t_n\lambda/x_n, u_1/y_1, u_2/y_2, ..., u_n/y_n\}$
2. 当$y_i \in \{x_1, x_2, ..., x_n\}$时，删去$u_i/y_i$。
3. 当$t_i\lambda = x_i$时，删去$t_i\lambda/x_i$。

例：设有代换$\theta = \{f(y)/x, z/y\}, \lambda = \{a/x, b/y, y/z\}$，求$\theta \circ \lambda$。

合一

定义：设有公式集$F = \{F_1, F_2, ..., F_n\}$，若存在一个代换$\lambda$使得

$$F_1\lambda = F_2\lambda = ... = F_n\lambda$$

则称λ为公式集F的一个合一，且称$F_1, F_2, ..., F_n$是**可合一的**。

合一

定义：设有公式集$F=\{F_1,F_2,...,F_n\}$，若存在一个代换λ使得

$$F_1\lambda=F_2\lambda=...=F_n\lambda$$

则称λ为公式集F的一个合一，且称$F_1,F_2,...,F_n$是**可合一的**。

例：公式集$F_1,F_2,F1=P(a,b),F_2=P(x,y)$时可合一的。

一个公式集的合一一般来说是不唯一的。

最一般合一

定义：设σ是公式集F的一个合一，如果对任一合一θ都存在一个代换λ，使得:

$$\theta = \sigma \circ \lambda$$

则称σ是一个最一般的合一。

最一般合一是唯一的。

最一般合一算法

1. 令$k=0, F_k=F, \sigma_k=\epsilon$其中$\epsilon$代表空代换
2. 若F_k只含有一个表达式，则算法停止。
3. 找出F_k的差异集D_k。$\leftarrow$ 对应符号不全相同的第一个符号
4. 若D_k中存在元素x_k和t_k，其中x_k是变元，t_k是项，且x_k不在t_k中出现，则做5，否则不可合一.
5. 令：

$$\sigma_{k+1} = \sigma_k \circ \{t_k/x_k\}$$
$$F_{k+1} = F_k\{t_k/x_k\}$$
$$k = k+1$$

注意：项可以是变量、常数或函数，但是变元一定是变量

例子

1. 设公式集为$\{P(x,y,z), P(x,f(a),h(b))\}$, 求差异集。
2. 求$\{P(f(x),y), P(y,f(b))\}$
3. 判断下列公式是否可以合一：
   - $\{P(a,b,c), P(d,b,c)\}$ 注意：a,b,c都是常数。
   - $\{P(x,b), P(f(x),y)\}$

范式

前束范式: 一个公式，如果量词均在全式的开头，它们的作用域延伸到整个公式的末尾，则该公式叫做前束范式。前束范式的形式如下：

$$(\square x_1)(\square x_2)\ldots(\square x_n)A$$

其中$\square$代表全称量词$\forall$或是存在量词$\exists$，x_i是个体变元，A是不含量词的谓词公式。

Skolem范式：如果前束范式中的所有存在量词都在全称量词之前，则称这种形式的谓词公式为Skolem范式。

任何一个含有量词的谓词公式均与一个前束范式等价和一个Skolem范式等价。

Overview

基本概念

推理的逻辑基础

产生式系统推理

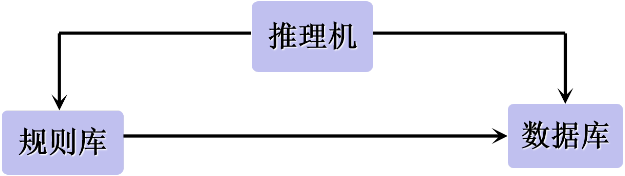
自然演绎推理

归结演绎推理

Recall: 产生式系统

把一组产生式放在一起，让它们互相配合，协同工作，一个产生式的结论可供另一个产生式作为已知事实使用，以求问题的解决，这样的系统称为产生式系统。

产生式系统通常由规则库、数据库和推理机这三个基本部分组成:



Recall: 规则库

用于描述领域内知识的产生式集合称为规则库

1. 有效表达领域内的过程性知识
2. 对知识进行合理的组织与管理

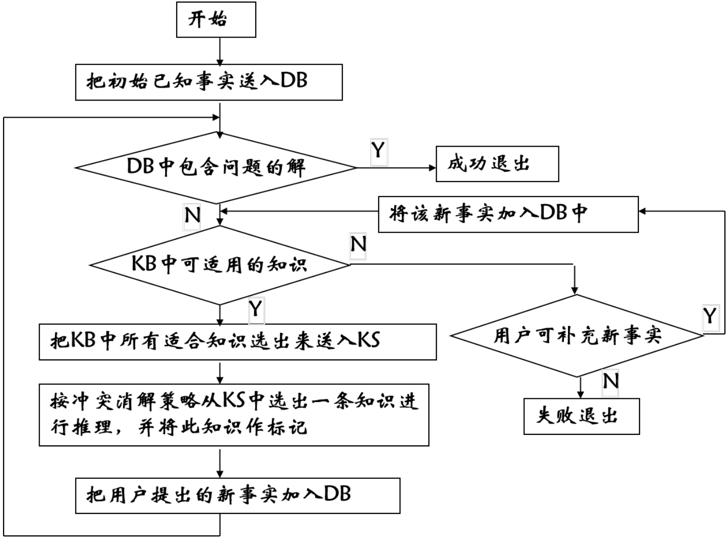
Recall: 产生式系统

综合数据库

- 综合数据库又称事实库、上下文、黑板等。
- 它是一个用于存放问题求解过程中各种当前信息的数据结构。
- 综合数据库中的内容是在不断变化的，动态的。

控制系统

- 控制系统又称推理机构，由一组程序构成，负责产生式系统的运行，实现对问题的求解
- 匹配
- 冲突消解
- 操作
- 对于不确定知识，需按一定算法计算不确定性
- 掌握结束产生式系统的运行时机


开始
把初始已知事实送入DB
DB中包含问题的解
Y
成功退出
N
将该新事实加入DB中
KB中可适用的知识
N
Y
用户可补充新事实
Y
把KB中所有适合知识选出来送入KS
N
按冲突消解策略从KS中选出一条知识进行推理，并将此知识作标记
失败退出
把用户提出的新事实加入DB

控制策略：推理的分类

1. **正向推理**：以已知事实作为出发点的一种推理，又称为数据驱动推理、前向链推理、模式制导推理及前件推理。
2. **逆向推理**：以某个假设目标为出发点的一种推理，又称为目标驱动推理、逆向链推理、目标制导推理及后件推理。
3. **混合推理**
   - 已知的事实不充分。通过正向推理先把其运用条件不能完全匹配的知识都找出来，并把这些知识可导出的结论作为假设，然后分别对这些假设进行逆向推理。
   - 由正向推理推出的结论可信度不高。
   - 希望得到更多的结论。
   - 推理的形式：
     1. 先正向再逆向，通过正向推理，即从已知事实演绎出部分结果，然后再用逆向推理证实该目标或提高其可信度。
     2. 先逆向再正向，先假设一个目标进行逆向推理，然后再利用逆向推理中得到的信息进行正向推理，以推出更多的结论。

控制策略：推理的分类 Cont.

4. **双向推理**:
   - 双向推理是指正向推理与逆向推理同时进行，且在推理过程中的某一步骤上“碰头”的一种推理。
   - 正向推理所得的中间结论恰好是逆向推理此时要求的证据。

控制策略：推理过程中需要使用的策略

1. 求解策略：推理是只求一个解还是求所有解以及最优解等。
2. 限制策略：对推理的深度、宽度、时间、空间等进行限制。
3. 冲突消解策略。

冲突消解策略

在推理过程中，匹配会出现三种情况：

1. 已知事实不能与知识库中的任何知识匹配成功;
2. 已知事实恰好只与知识库中的一个知识匹配成功;
3. 已知事实可与知识库中的多个知识匹配成功；或者有多个（组）已知事实都可与知识库中某一知识匹配成功；或者有多个（组）已知事实可与知识库中的多个知识匹配成功;

- 对正向推理而言，如果有多条产生式规则的前件都和已知的事实匹配成功；或者有多组不同的已知事实都与同一条产生式规则的前件匹配成功；或者两种情况同时出现。
- 对逆向推理而言，如果有多条产生式的后件都和同一假设匹配成功，或者有多条产生式后件可与多个假设匹配成功。

冲突消解策略

1. **就近原则排序**：把最近使用过的规则赋予较高的优先级。
2. **按已知事实的新鲜性排序**: 数据库中后生成的事实称为新鲜事实，即后生成的事实比先生成的事实具有较大的新鲜性。设规则r_1可与事实组A匹配成功，规则r_2可与事实组B匹配成功，则A与B中哪一组新鲜，与它匹配的产生式规则就先被应用：

- 把A与B中的事实逐个地比较其新鲜性，若A中包含的更新鲜事实比B多，就认为A比B新鲜。
- 以A中最新鲜的事实与B中最新鲜事实相比较，哪一个更新鲜，就认为相应的事实组更新鲜。
- 以A中最不新鲜的事实与B中最不新鲜事实相比较，哪一个更不新鲜，就认为相应的事实组有较小的新鲜性。

3、**按匹配度排序**：在进行不确定推理时，若产生式规则r1与r2都可匹配成功，则可根据它们的匹配度来决定哪一个产生式规则可优先被应用。

冲突消解策略

4. **根据领域问题的特点进行排序**
   - 当领域问题有固定的解题次序时，可按该次序排列相应的知识，排在前面的知识优先被应用。
   - 当已知某些产生式规则被应用后会明显地有利于问题的求解时，就使这些产生式规则优先被应用。
5. **按上下文限制排序**: 把产生式规则按它们所描述的上下文分成若干组，在不同的条件下，只能从相应的组中选取有关的产生式规则。
6. **按条件个数排序**: 如果有多条产生式规则生成的结论相同，则要求条件少的产生式规则被优先应用，因为要求条件少的规则匹配时花费的时间较少。
7. **按规则的次序排序**: 以知识库中预先存入规则的排列顺序作为知识排序的依据，排在前面的规则具有较高的优先级。

Overview

推理规则

- P规则：推导过程中，可以随时引入前提。
- T规则：推导过程中已得到的结论可作为后续推导的前提。
- E规则：如果是利用等价式而得来的，推理规则就相应的写为E。
- I规则：某个公式如果是利用蕴涵式推理规则而得来的，其推理规则相应的简写为I。
- 反证法：设$H_1, H_2, ..., H_m$，B是公式，B是$H_1, H_2, ..., H_m$的有效结论，等价于$H_1, H_2, ..., H_m$与$\neg B$不相容。
- CP规则：设A, B, C是公式，则$A \Rightarrow (B \rightarrow C)$等价于$A, B \Rightarrow C$。

常用推理

- 假言推理：

  $$P, P \to Q \Rightarrow Q$$

  如果某个数能被2整除则该数为整数。
  6能被2整除。
  $\Rightarrow$ 6是偶数。

- 拒取式：

  $$P \to Q, \neg Q \Rightarrow \neg P$$

  任何人违反了交通规则，都要受到罚款。
  我没有受到罚款。
  $\Rightarrow$ 我没有违反交通规则。

例子

1. 已知：$A, B, A \to C, B \wedge C \to D, D \to Q$，求证：Q为真。
2. 设已知事实：（1）所有勤学苦练的人，都会成为技术能手。（2）学习积极分子都是勤学苦练的人。（3）李明是学习积极分子。求证：李明成为技术能手。
3. 例：一个热力站有5个阀门控制对外送蒸汽，使用这些阀门必须遵守以下操作规则：（1）如果开启1号阀，那么必须同时打开2号阀并且关闭5号阀。（2）如果开启2号阀或者5号阀，则要关闭4号阀。（3）不同同时关闭3号阀和4号阀。求证：如果1号阀打开，同时要打开2号阀和3号阀。

自然演绎推理的特点

- 优点：定理证明过程自然，容易理解，而且它拥有丰富的推理规则，推理过程灵活，便于在它的推理规则中嵌入领域启发式知识。
- 缺点：容易产生组合爆炸，推理过程中得到的中间结论一般呈指数形式递增。

Overview

子句

- 原子谓词公式及其否定统称为文字。
  例：$P(x), Q(x), \neg P(x), \neg Q(x)$
- 任何文字的析取式称为子句。
  例：$P(x) \vee Q(x), P(x, f(x)) \vee Q(x, g(x))$
- 不包含任何文字的子句称为空子句NIL。空子句中不包含任何文字，不能被任何解释满足，所以空子句是永假的，不可满足的。
- 由子句或空子句构成的集合称为子句集。
- 任何谓词公式都可以变化成为相应的子句集。

转化子句集的步骤

1. 利用等价谓词关系消去谓词公式中的$\rightarrow$和$\leftrightarrow$。
2. 利用等价关系把$\neg$移到紧靠谓词的位置上。
3. 换名，使不同量词约束的变元有不同的名字。
4. 消去存在量词:
   - 存在量词不出现在全称量词的辖域内:

     $$(\exists y)(\forall x_1)(\forall x_2)...(\forall x_n)P(x_1, x_2, ..., x_n, y)$$

     此时只要用一个新的个体常量替换该存在量词的约束变元可消去存在量词:

     $$(\forall x_1)(\forall x_2)...(\forall x_n)P(x_1, x_2, ..., x_n, a)$$

   - 存在量词位于一个或多个全称量词的辖域内:

     $$(\forall x_1)(\forall x_2)...(\forall x_n)(\exists y)P(x_1, x_2, ..., x_n, y)$$

     使用Skolen函数法$f(x_1, x_2, ..., x_n)$替换:

     $$(\forall x_1)(\forall x_2)...(\forall x_n)P(x_1, x_2, ..., x_n, f(x_1, x_2, ..., x_n))$$

转化子句集的步骤 Cont.

1. 利用等价谓词关系消去谓词公式中的$\rightarrow$和$\leftrightarrow$。
2. 利用等价关系把$\neg$移到紧靠谓词的位置上。
3. 换名，使不同量词约束的变元有不同的名字。
4. 消去存在量词。
5. 把全称量词移到公式左边。
6. 利用等价关系把公式化为Skolem标准形（合取范式）。
7. 消去全称量词。
8. 对变元更名，使不同子句中的变元不同名。
9. 消去合取词。

例子

将公式$(\forall x)((\forall y)P(x,y) \to \neg(\forall y)(Q(x,y) \to R(x,y)))$转化为子句集。

$P \to Q$永真?

$P \wedge \neg Q$不可满足?

Herbrand（海伯伦）理论：H域

设S为子句集，定义在个体域D上，按下述方法构造的域H_∞称为海明伦域，简称为H域:

1. 令H_0是S中所出现的常量的集合。所S中没有常量出现，则令$H_0 = \{a\}$，其中$a \in D$。
2. 令
$$H_i = H_{i-1} \cup \{S\text{中所有形如}f(t_1, ..., t_n)\text{的函数}\}$$
其中$t_1, ..., t_n$是H_{i-1}中的元素。

Herbrand（海伯伦）理论：H域

设S为子句集，定义在个体域D上，按下述方法构造的域H_∞称为海明伦域，简称为H域:

1. 令H_0是S中所出现的常量的集合。所S中没有常量出现，则令$H_0 = \{a\}$，其中$a \in D$。
2. 令
$$H_i = H_{i-1} \cup \{S\text{中所有形如}f(t_1, ..., t_n)\text{的函数}\}$$
其中$t_1, ..., t_n$是H_{i-1}中的元素。

例：求下列子句集的H域:

1. $S = \{P(x), Q(f(x)) \vee \neg R(x)\}$
2. $S = \{\neg P(x), P(y) \vee Q(y)\}$
3. $S = \{P(f(x)) \vee Q(a), Q(g(y)) \vee R(b)\}$

海伯伦理论：原子集

令：

$$A = \{\text{所有形如}P(t_1, ..., t_n)\}$$

则称A为子句集S的原子集，其中$t_1, ..., t_n$是H中的元素。如果子句集S的原子集为A，则对A中各元素的真假值的一个具体设定都是S的一个H解释。

海伯伦理论：原子集

令：

$$A = \{所有形如P(t_1, ..., t_n)\}$$

则称A为子句集S的原子集，其中$t_1, ..., t_n$是H中的元素。如果子句集S的原子集为A，则对A中各元素的真假值的一个具体设定都是S的一个H解释。

例：求子句集$S = \{P(x), Q(f(x))\}$的原子集。

海伯伦理论

- 设I是子句集S在域D上的一个解释，则存在对应于I的H域解释I^*，使得若有$S|I=T$，就必有$S|I^*=T$。
- 子句集S不可满足的充要条件是S对H域上的一切解释都为假。
  1. 充分性：若S在一般域D上是不可满足的，必然在H域上是不可满足的，从而对H域上的一切解释都为假。
  2. 必要性：若S在任一H解释下均为假，必然会使S在D域上的每一个解释为假。否则，如果存在一个解释使S为真，那么依据定理2可知，一定可以在H域找到相对应的一个解释使S为真。这与S在所有H解释下均为假矛盾。
- 子句集S不可满足的充要条件是存在一个有限的不可满足的基例集S'。

鲁宾逊归结原理

- 鲁宾逊归结原理基本思想是：检查子句集S中是否包含空子句，若包含，则S不可满足；若不包含，就在子句集中选择合适的子句进行归结，一旦通过归结能推出空子句集，就说明子句集S是不可满足的。
- **命题逻辑**中的归结原理：设C_1与C_2是子句集中的任意两个子句，如果C_1中的文字L_1与C_2中的文字L_2互补，那么从C_1和C_2中分别消去L_1和L_2，并将二个子句中余下的部分析取，构成一个新子句C_{12}，则称这一过程为归结，称C_{12}为C_1和C_2的归结式，称C_1和C_2为C_{12}的亲本子句。

鲁宾逊归结原理

- 鲁宾逊归结原理基本思想是：检查子句集S中是否包含空子句，若包含，则S不可满足；若不包含，就在子句集中选择合适的子句进行归结，一旦通过归结能推出空子句集，就说明子句集S是不可满足的。
- **命题逻辑**中的归结原理：设C_1与C_2是子句集中的任意两个子句，如果C_1中的文字L_1与C_2中的文字L_2互补，那么从C_1和C_2中分别消去L_1和L_2，并将二个子句中余下的部分析取，构成一个新子句C_{12}，则称这一过程为归结，称C_{12}为C_1和C_2的归结式，称C_1和C_2为C_{12}的亲本子句。

例：求下列归结式：

1. $C_1 = P \vee R, C_2 = \neg P \vee Q$
2. $C_1 = P, C_2 = \neg P \vee Q$
3. $C_1 = P, C_2 = \neg P$
4. $C_1 = \neg P \vee Q, C_2 = P \vee Q, C_3 = \neg Q \vee R$

鲁宾逊归结原理 Cont.

- **定理**: 归结式C_{12}是亲本子句C_1与C_2的逻辑结论。
- **推论**: 设C_1与C_2是子句集S中的两个子句，C_{12}是它们的归结式，若用C_{12}代替C_1与C_2得到新子句集S_1，则由S_1的不可满足性可推出原子句集S的不可满足性，即

$$S_1\text{的不可满足性} \Rightarrow S\text{的不可满足性}$$

- **推论**: 设C_1与C_2是子句集S中的两个子句，C_{12}是它们的归结式，若将C_{12}加入S得到新子句集S_2，则S_2的不可满足性等价于原子句集S的不可满足性，即

$$S_2\text{的不可满足性} \Leftrightarrow S\text{的不可满足性}$$

鲁宾逊归结原理 Cont.

- **定理**: 归结式C_{12}是亲本子句C_1与C_2的逻辑结论。
- **推论**：设C_1与C_2是子句集S中的两个子句，C_{12}是它们的归结式，若用C_{12}代替C_1与C_2得到新子句集S_1，则由S_1的不可满足性可推出原子句集S的不可满足性，即

$$S_1\text{的不可满足性} \Rightarrow S\text{的不可满足性}$$

- **推论**: 设C_1与C_2是子句集S中的两个子句，C_{12}是它们的归结式，若将C_{12}加入S得到新子句集S_2，则S_2的不可满足性等价于原子句集S的不可满足性，即

$$S_2\text{的不可满足性} \Leftrightarrow S\text{的不可满足性}$$

鲁宾逊归结原理 Cont.

- **定理**: 归结式C_{12}是亲本子句C_1与C_2的逻辑结论。
- **推论**: 设C_1与C_2是子句集S中的两个子句，C_{12}是它们的归结式，若用C_{12}代替C_1与C_2得到新子句集S_1，则由S_1的不可满足性可推出原子句集S的不可满足性，即

$$S_1\text{的不可满足性} \Rightarrow S\text{的不可满足性}$$

- **推论**: 设C_1与C_2是子句集S中的两个子句，C_{12}是它们的归结式，若将C_{12}加入S得到新子句集S_2，则S_2的不可满足性等价于原子句集S的不可满足性，即

$$S_2\text{的不可满足性} \Leftrightarrow S\text{的不可满足性}$$

鲁宾逊归结原理 Cont.

- **定理**：归结式C_{12}是亲本子句C_1与C_2的逻辑结论。
- **推论**：设C_1与C_2是子句集S中的两个子句，C_{12}是它们的归结式，若用C_{12}代替C_1与C_2得到新子句集S_1，则由S_1的不可满足性可推出原子句集S的不可满足性，即

$$S_1\text{的不可满足性} \Rightarrow S\text{的不可满足性}$$

- **推论**：设C_1与C_2是子句集S中的两个子句，C_{12}是它们的归结式，若将C_{12}加入S得到新子句集S_2，则S_2的不可满足性等价于原子句集S的不可满足性，即

$$S_2\text{的不可满足性} \Leftrightarrow S\text{的不可满足性}$$

例：设有子句集$S = \{\neg P \vee \neg Q \vee R, P \vee R, Q \vee R, \neg R\}$，求证该子句集是不可满足的。

鲁宾逊归结原理 Cont.

设C_1与C_2是两个没有相同变元的子句，L_1和L_2分别是C_1和C_2中的文字，若σ是L_1和$\neg L_2$的最一般合一，则称:

$$C_{12} = (C_1\sigma - \{L_1\sigma\}) \cup (C_2\sigma - \{L_2\sigma\})$$

为C_1和C_2的二元式，L_1和L_2称为归结式上的文字。

鲁宾逊归结原理 Cont.

设C_1与C_2是两个没有相同变元的子句，L_1和L_2分别是C_1和C_2中的文字，若σ是L_1和$\neg L_2$的最一般合一，则称:

$$C_{12} = (C_1\sigma - \{L_1\sigma\}) \cup (C_2\sigma - \{L_2\sigma\})$$

为C_1和C_2的二元式，L_1和L_2称为归结式上的文字。

例：求$C_1 = P(x) \vee Q(a), C_2 = \neg P(a) \vee R(y)$的二元归结式$C_{12}$。

鲁宾逊归结原理 Cont.

定义：子句C_1和C_2的归结式是下列二元归结式之一：

1. C_1和C_2的二元归结式;
2. C_1和C_2的因子$C_2\sigma_2$的二元归结式;
3. C_1的因子$C_1\sigma_1$与C_2的二元归结式;
4. C_1的因子$C_1\sigma_1$与C_2的因子$C_2\sigma_2$的二元归结式。

鲁宾逊归结原理 Cont.

归结过程需要注意的问题:

1. 确保每个子句有不同的变元名，以免在归结的过程中产生不便或错误;
   例：求$C_1 = P(a) \vee Q(x), C_2 = \neg P(x) \vee R(b)$的归结式$C_{12}$。
2. 如果参加归结的子句内部含有可合一的文字时，在进行归结之前不要对这些文字先进行合一。
3. 在求归结式时，若两个子句中有两对可互补的文字，不能同时消去，否则会产生错误。
   例：设有两个子句$C_1 = P \vee \neg Q$和$C_2 = \neg P \vee Q$，在C_1和C_2中有两对互补的文字P和$\neg P$，以及Q和$\neg Q$。若同时消去，就会得到空子句NIL，由于空子句是一个不可满足的式子，所以会推导出子句集$\{C_1, C_2\}$是不可满足的错误结论。实际上子句集$\{C_1, C_2\}$是可满足的。
   例：求$C_1 = P(x) \vee \neg Q(b), C_2 = \neg P(a) \vee Q(y) \vee R(z)$的归结式$C_{12}$。

应用归结原理证明问题

设F为已知前提的公式集，Q为目标公式（结论），用归结反演证明Q为真的步骤是：

1. 否定Q，得到$\neg Q$;
2. 把$\neg Q$并入到公式集F中，得到$\{F, \neg Q\}$；
3. 把公式集$\{F, \neg Q\}$化为子句集S；
4. 应用归结原理对子句集S中的子句进行归结，并把每次归结得到的阿归结式都并入S中。如果反复进行，若出现了空子句，则停止归结，此时就证明了Q永真。

例子

1. 已知条件：$(\exists x)(R(x) \wedge (\forall y)(D(y) \rightarrow L(x,y)))$和$(\forall x)(R(x) \rightarrow (\forall y)(S(y) \rightarrow \neg L(x,y)))$试证明可推导结论$(\forall x)(D(x) \rightarrow \neg S(x))$。
2. 一个数据库中现有A、B 、C 、D 、E 、F 六个语句，但目前这个数据库是不协调的，必须删除某些语句才能恢复数据库的协调性。已知：（1）如果保留语句A，那么必须保留语句B和语句C 。（2）如果保留语句E ，则必须同时删除语句D 和语句C 。（3）如果保留语句F ，那么必须保留语句E 。（4）语句A 是重要的信息不能删除。求证：如果上述各项为真，则需同时删除语句E和F。
3. 每个使用Internet网络的人都想从网络中获取信息。求证：如果网络没有信息就不会有人使用Internet。

应用归结原理求取问题的答案

1. 把已知前提用谓词公式表示出来，并且化为相应的子句集，设该子句集的名字为S。
2. 把待求解的问题也用谓词公式表示出来，然后把它否定并与谓词ANSWER构成析取式，ANSWER是一个为了求解问题而专设的谓词，其变元必须与谓词公式的变元完全一致。
3. 把此析取式化为子句集，并且把该子句集并入到子句集S中，得到子句集S'。
4. 对S'应用归结原理进行归结。
5. 若得到归结式ANSWER，则答案在ANSWER中。

例子

1. 已知：小张和小李是同班同学。如果x和y是同班同学，则x上课的教室也是y上课的教室。现在小张在301教室上课，请问小李在哪个教室上课?
2. 某旅游团去西藏旅游，除拉萨市之外，还有6个城市或景区可供选择：E市、F市、G湖、H山、I峰、J湖。考虑时间、经费、高原环境和人员身体状况等因素，决定如下：（1）G湖和J湖中至少去一个地方。（2）如果不去E市或者不去F市，则不能去G湖游览。（3）如果不去E市，也不能H山游览。（4）如果去J湖，则去I峰。如果由于气候的原因，这个团队不去I峰，那么该团队一定去哪些景点游览?
3. 已知下列事实：小唐、小闵和小网都是挑战Club的会员。每个会员或者是一个滑雪爱好者，或者是一个登山爱好者，或者都是。没有一个登山爱好者喜欢下雨，所有的滑雪爱好者都喜欢雪。小唐喜欢的所有东西小闵都不喜欢，小唐不喜欢的所有东西小闵都喜欢，小唐喜欢雨和雪。试求解谁是该俱乐部的会员，他是一个登山爱好者，但不是滑雪爱好者。

如何选取子句进行归结?

归结的一般过程

$$S = \{C_1, C_2, C_3, C_4\}$$

1. 从子句C_1开始，逐个与C_2，C_3，C_4进行比较，看哪两个子句可进行归结。若能找到就求出归结式。然后用C_2与C_3，C_4进行比较，凡可归结的都进行归结，最后用C_3与C_4比较，若能进行归结也可对它们进行归结。经过这一轮的比较及归结后，就会得到一组归结式，称为第一级归结式。
2. 再从C_1开始，用S中的子句分别与第一级归结式中的子句逐个地进行比较、归结，这样又会得到一组归结式，称为第二级归结式。
3. 仍然从C_1开始，用S中地子句及第一级归结式中地子句逐个地与第二级归结式中的子句进行比较，得到第三级归结式。

如此继续，直到出现了空子句或者不能再继续归结时为止。只要子句集是不可满足的，上述归结过程一定会归结出空子句而终止。

归结策略

1. 删除策略：通过删除某些无用的子句来缩小归结的范围。
2. 限制策略：通过对参加归结的子句进行种种限制，尽可能地减小归结的盲目性，使其尽快地归结出空子句。

删除策略

1. **纯文字删除法**: 如果某文字L在子句集中不存在可与之互补的文字$\neg L$，则称该文字为纯文字。在归结时纯文字不可能被消去，因而用包含它的子句归结时不可能得到空子句，即这样的子句对归结是无意义的，所以可把它所在的子句从子句集中删去。
2. **重言式删除法**: 如果一个子句中同时包含互补文字时，则称该子句为重言式。重言式是真值为真的子句。对于一个子句集来说，不管是增加或者删除一个真值为真的子句都不会影响它的不可满足性，因而可从子句集中删除重言式
3. **包孕删除法**: 设有子句C_1和C_2，如果存在一个代换σ，使得$C_1\sigma \subseteq C_2$，则称C_1包孕于C_2。删除C_1不会影响子句集的不可满足性。

限制策略

1. 支持集策略：每一次归结时，亲本子句中至少应有一个是由目标公式的否定得到的子句，或者是它们的后裔。支持集策略是完备的，即若子句集是不可满足的，则由支持集策略一定可以归结出空子句。
2. 线性输入策略：参加归结的两个子句中必须至少有一个是初始子句集中的子句。线性输入策略是不完备的。
3. 单文字子句策略：如果一个子句只包含一个文字，则称它是单文字子句。单文字子句策略要求参加归结的两个子句中必须至少有一个是单文字子句。单文字子句归结策略是不完备的。
4. 祖先过滤形策略：当对两个子句C_1和C_2进行归结时，只要它们满足下述两个条件中的任意一个就可进行归结：
   - C_1和C_2中至少有一个时初始子句集中的子句。
   - 如果两个子句都不是初始子句集中的子句，则一个应是另一个的祖先。一个子句(C_1)是另一个子句(C_2)的祖先是指C_2是由C_1与别的子句归结后得到的归结式。

Questions?