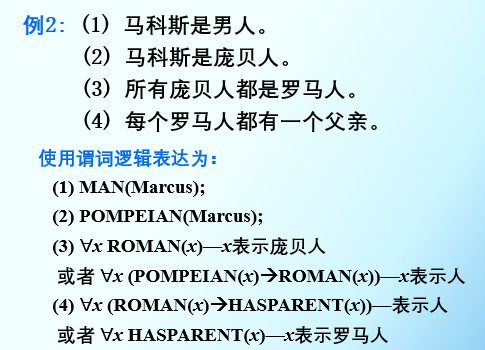
人工智能复习

# 概述

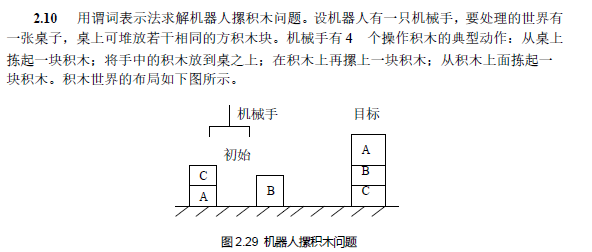
* 如何衡量机器是否具有智能？
  + 图灵测试：
    - 让一个询问者分别与一个人类和一台机器进行对话（询问），如果他不能准确区分出回答者是人类还是机器，则认为该机器通过了图灵测试。
  + 中文房间：
    - 让一个只说英语的人呆在一个密闭房间，房间中有完整的中文规则书，输入一个中文问题，这个人能通过这本中文书对这个中文问题进行解答，并输出答案，即使他完全不知道这个问题的含义是什么，也能让房间外的人认为房间内的人熟悉中文。
    - 该实验说明计算机并不能真正理解接收到的信息，只是通过运行一个程序处理信息，根据某种规则给出答案，给人一种智能的印象。
* 什么是人工智能？
  + **从能力的角度**：人工智能是用人工的方法在机器（计算机）上实现的智能。是智能机器所执行的通常与人类智能有关的功能，如判断、推理、证明、识别、感知、理解等思维活动。
  + **从学科的角度**：人工智能是一门研究如何构造智能机器或智能系统，去模拟、延伸和扩展人类智能的学科。
* 人工智能的发展简史
  + 第一代人工智能（1956-2010）：知识驱动（符号+知识）
  + 第二代人工智能（2010-2020）：数据驱动（大数据+深度学习）
  + 第三代人工智能（2020-）：数据和知识驱动结合
  + 人工智能四大要素：知识+数据+算法+算力
* **人工智能三大学派**
  + **符号主义**
    - 智能的基础是知识，核心是知识表示和知识推理；知识可以用符号表示，也可用符号进行推理。
    - 功能模拟，构造能够模拟大脑功能的智能系统。
  + **联结主义**
    - **智能的基元是神经元，思维过程是神经元的联结活动过程。**
    - 结构模拟，构造模拟大脑结构的神经网络系统。
  + **行为主义**
    - **智能取决于感知和行动，**采取“感知-动作”模型，智能不需要知识、不需要表示、不需要推理。人工智能可以像人类智能那样逐步**进化**。
    - 行为模拟，构造具有进化能力的智能系统。

# 知识表示

* 知识表示方法
  + 谓词逻辑表示
    - 经典例题：
      * 利用谓词表示法表示命题



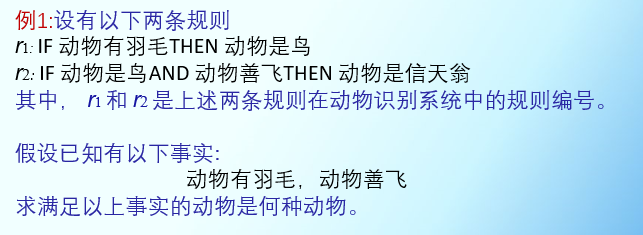
* + - * 利用谓词表示法求解特定问题



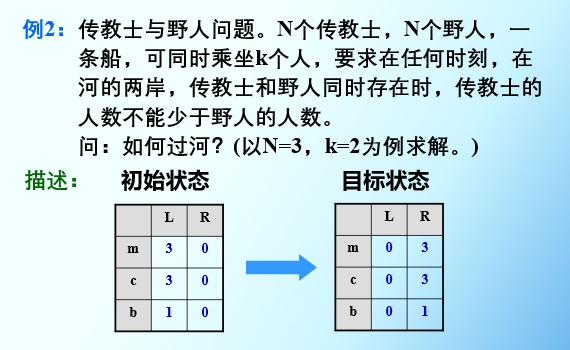
* + - * + 首先定义描述状态的谓词
        + 写出问题的初始状态
        + 写出问题的目标状态
        + 再定义描述操作的谓词

每个操作可以被定义为条件和动作两部分

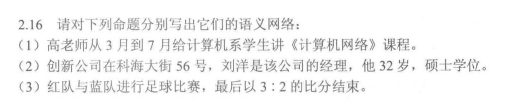
* + - * + 利用上述状态和操作写出从初始状态到目标状态的求解过程
    - 优缺点：
      * 优点：严密、自然、通用、易于实现
      * 缺点：表示能力差、组合爆炸、效率低
  + 产生式表示
    - 事实是断言一个语言变量的值或断言多个语言变量之间的陈述句。
    - 事实的表示：
      * （对象， 属性， 值)：（雪， 颜色， 白）
      * （关系， 对象1， 对象2）：（热爱， 王峰， 祖国）
      * （对象， 属性， 值， 可信度因子）：（雪， 颜色， 白， 1）
    - 使用形如IF P THEN Q（P→Q）的产生式联系前件P和后件Q。
    - 使用综合数据库、规则库和控制系统进行推理：
      * 综合数据库（DB）：实际上是各种状态集合，存储初始状态、输入事实、目标状态和中间结论等信息。
      * 规则库（RB）：用于存放推理所需要的规则
      * 控制系统：用于将综合数据库中的事实与规则库中的规则前件进行匹配，推出新的事实，并把它加入到综合数据库中，直到综合数据库包含目标状态。
    - 例题：
      * 例1



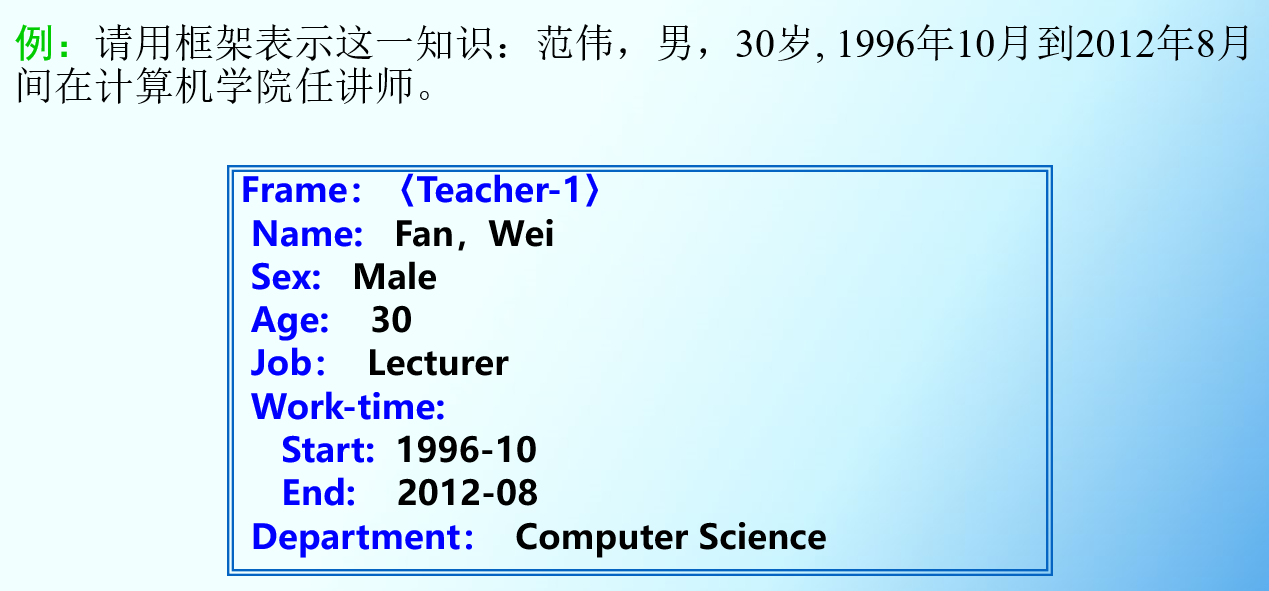
* + - * 例2

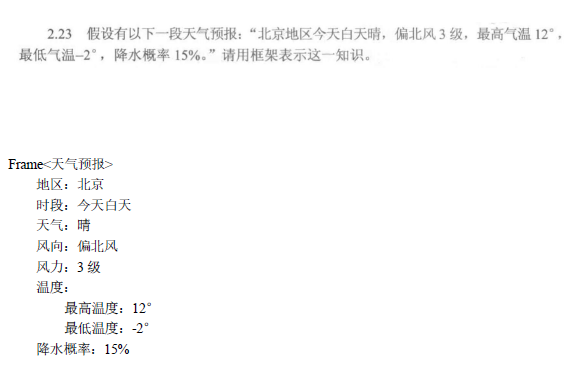


* + - 优缺点：
      * 优点：自然、模块（规则之间只能通过综合数据库联系，而不能相互调用）、有效（不仅可以表示确定性知识，还能表示不确定性知识）
      * 缺点：效率低，不便于表示结构性知识
  + 语义网络表示
    - 用节点表示各种概念，用有向弧表示节点间的联系或关系
    - 用三元组（节点1， 弧， 节点2）来表示语义基元，可以用图来表示
    - 经典关系：
      * ISA：“是一个”，一个事物是另一个事物的实例
      * AKO：“是一种”，一个事物是另一个事务的类型
      * A-Member-of：“是一员”，一个事物是另一个事物的一员，有继承性
      * Part-of：“是一部分”，聚类关系，表示一个事物是另一个事物的一部分，无继承性
      * 属性关系：例如Can和Have关系
      * 时间关系：Before，After
      * 位置关系：Located-on（at、inside、outside、under）
      * 相近关系：Similar-to，Near-to
    - 使用中间节点进行拓展：最好用的是事件节点和动作节点，可以方便的对“事件“和”动作“本身进行拓展。
    - 经典例题：



* + - 优缺点：
      * 优点：结构性、联想性、自索引性
      * 缺点：非严格性，复杂性
  + 框架表示
    - 与面向对象的思想类似，利用框架表示类，将其实例化得到具体的对象。槽对应类的属性，侧面是槽的某一方面特征。

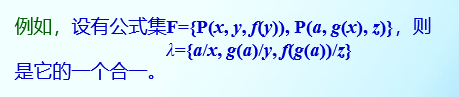




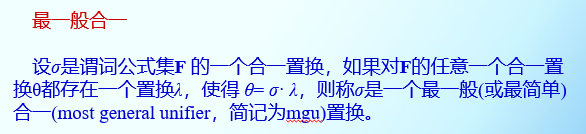
* + - 将ISA，AKO当作属性可以实现继承。可以将一个框架当作另一个框架的属性类型。可以为某个属性定义IF-NEEDED和IF-ADDED侧面，前者用于查询，后者用于修改。
    - 优缺点：
      * 优点：结构性、深层性、继承性、自然性
      * 缺点：缺乏框架的形式理论，缺乏过程性知识表示（领域无关的推理规则），清晰性难以保证

# 确定性推理

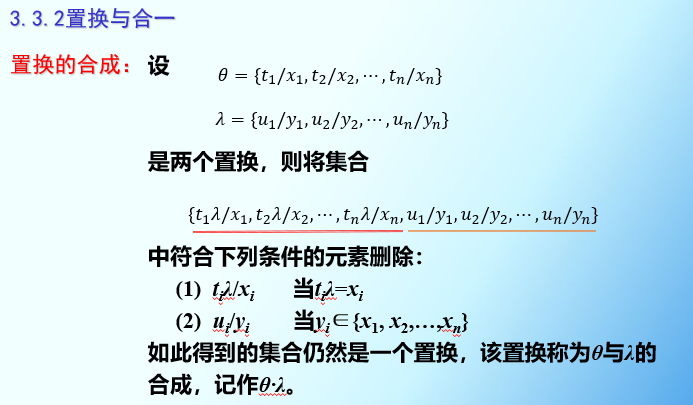
* 置换与合一
  + 置换可简单地理解为在一个谓词公式中用**项**去替换**变元**。
    1. 要求项ti与变元xi不能相同，xi不能循环地出现在另一个ti中
  + 合一可简单地理解为是寻找对应变量地置换，使两个或多个谓词公式一致。



* + 最一般合一



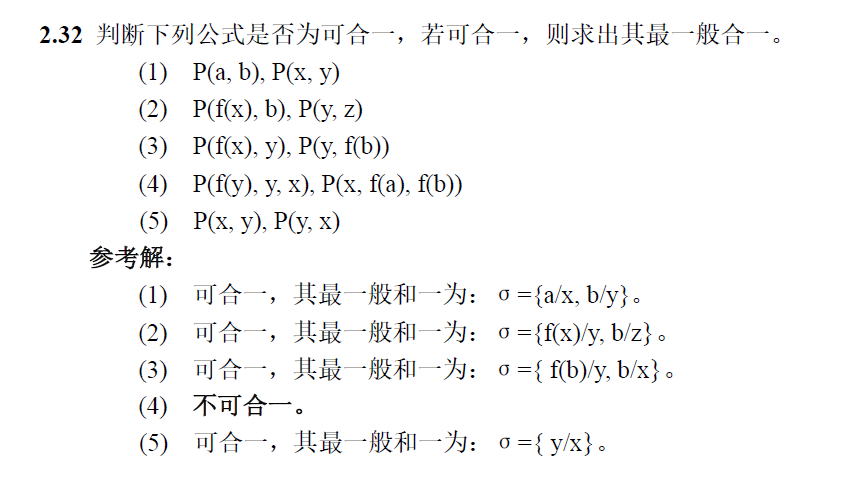
* + 置换合成



* + 最一般合一求取算法：

求公式集S的最一般合一σ（σ初值为ϵ）：

* + 1. 若S只有一个谓词公式，则σ即为所求，算法终止，否则转2)
    2. 求S的差异集差异集Dk，转3)
    3. 若Dk中存在变元Xk和项tk，且Xk不在tk中出现过，置S=S{tk/Xk}，σ=σ{tk/Xk}，转1) 。若不存在这样的Xk和tk（注意Dk每次都变化），则说明S的最一般合一不存在，算法终止



* 归结演绎推理
  + 鲁滨逊归结原理
    - 命题逻辑的归结：通常不含变元
    - 谓词逻辑的归结：通常含有变元，需要对变元进行置换合一
      * 设C1和C2是两个没有公共变元的子句，L1和L2分别是C1和C2中的文字，若L1和﹁L2存在最一般合一σ，则可将其归结为二元归结式C12。
        + 通常对C1和C2做变名处理以确保C1和C2没有公共变元
        + 归结式不能同时消去两个互补对
        + 若某子句内部有可合一的文字，则在进行归结之前先合一，合一后的子句称为合一前子句的因子
      * 设F为已知前提，G为要证明的结论，归结原理要证**F∧﹁G**不可满足
        + 否定公式目标G，得到**﹁G**
        + 把**﹁G**并入到公式集F中，得到**{F**，﹁**G}**
        + 把**{F**，﹁**G}**化为子句集S

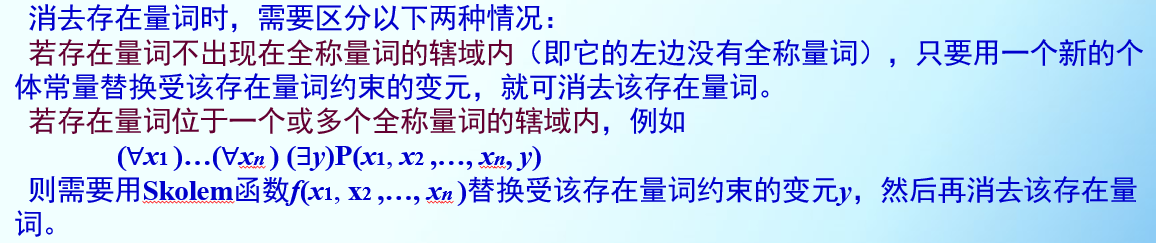
消去连接词**“**→**”**和**“**↔**”**

减少否定符号的辖域：德摩根律

对变元标准化：量词辖域内改名

化为前束范式：将量词提到公式最前方，不能改变量词相对顺序

消去存在量词：



注意：这里判断在不在全称量词辖域内实际上是在化为前束范式前的公式上进行判断。实际上消去存在量词这一步应该先于化为前束范式前执行。

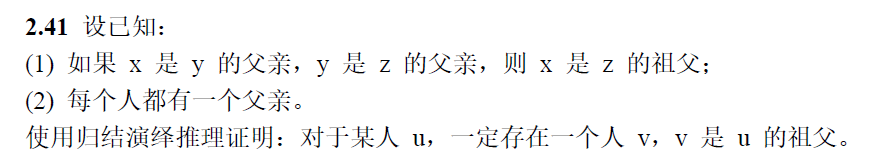
化为Skolem标准形（合取范式）：与或式

消去全称量词：直接消去即可

消去合取词：直接消去与，被与符号分割的或语句都是子句

更换变量名称：将这些子句的变量进行改名

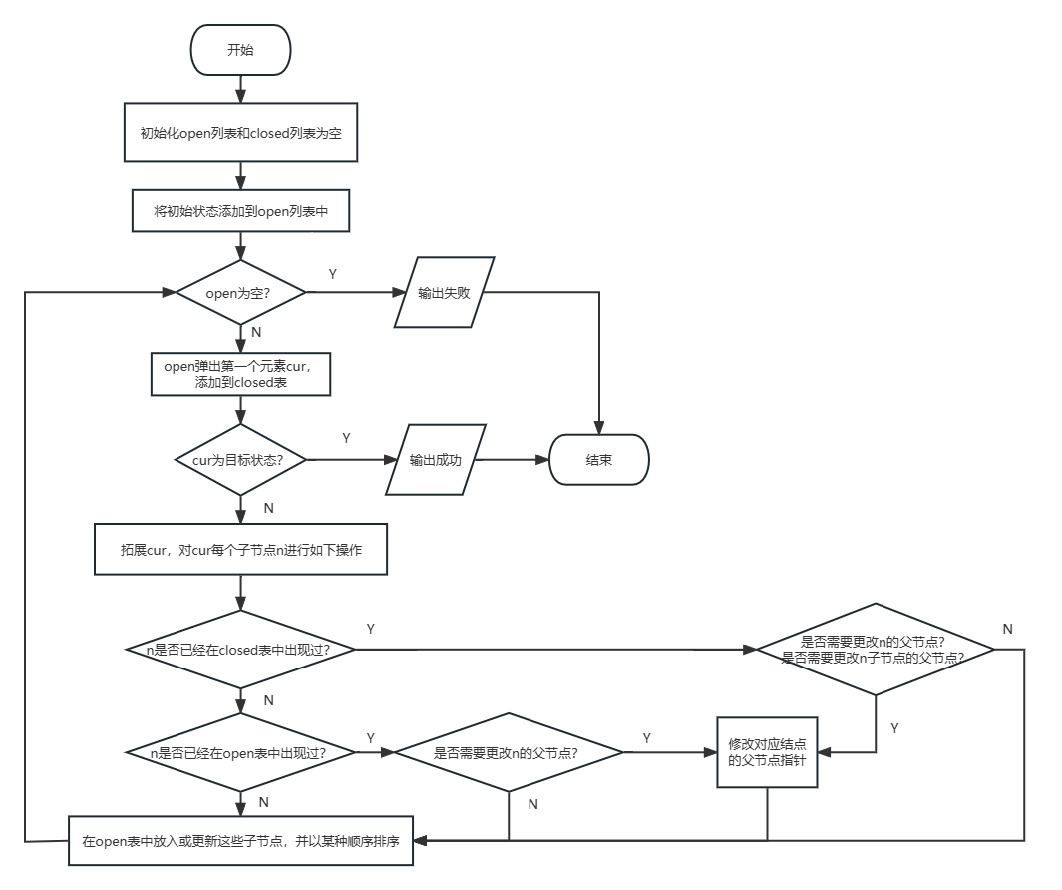
* + - * + 应用归结原理对子句集S中的子句进行归结（注意要考虑合一），把每次得到的归结式并入到S中。如此反复进行，直到出现空子句，证明G为真。



！！！这种普遍性的命题前面都加任意量词！！！

# 搜索

* 一般图搜索过程



* 广度优先搜索
  + 先生成的节点先拓展
  + 注意是pop()出目标节点时，算法才结束，而不是“发现”目标节点。
* 深度优先搜索
  + 后生成的节点先扩展
  + 注意是pop()出目标节点时，算法才结束，而不是“发现“目标节点。
* 代价一致性搜索
  + 对有权图，令g(i)为根节点S0到任一节点i的路径代价。优先拓展g(i)最小的节点。
  + 注意是pop()出目标节点时，算法才结束，而不是“发现“目标节点。
* A\*算法
  + 对有权图，令g(i)为根节点S0到任一节点i的路径代价估计，h(i)为节点i到目标节点Sg的代价估计（启发式函数），选择f(i)=g(i)+h(i)最小的节点优先拓展。其中要求h(i)≤h\*(i)，h\*(i)是节点i到目标节点的真实路径。
  + 可以证明，h(i)越逼近h\*(i)，就能越快找到目标节点。
  + 若h(i)满足：1) h(Sg)=0; 2) 0≤h(i)-h(i+1)≤c(ni,ni+1)，则称该启发式函数满足单调限制。（实际上，2) 可写为h(i)≤c(ni,ni+1)+h(i+1)，左边0≤h(i)-h(i+1)不是必要的，也称为启发式函数的一致性）。满足单调限制的启发式函数可以证明：当A\*算法拓展节点n时，就已经找到了通往节点n的最佳路径（否则，可能需要进行父节点更改）；A\*算法拓展的节点序列满足f(ni)≤f(ni+1)。
* 与或树的启发式搜索
  + 将初始节点放在open表
  + 对目前已经拓展的树，计算希望树T
  + 依次在open表中取出T的节点，放入closed表，记为n
  + 若n为终止节点，标记n为可解节点，依次向上按照与或规则标记其祖先为可解节点，如果能使初始节点为可解状态，则此时的T即为最优树。否则转2)
  + 若n不是终止节点，且不可拓展，标记n为不可解节点，依次向上按照与或规则标记其祖先为不可解节点，如果使初始节点为不可解状态，则问题无解。否则转2)
  + 若n不是终止节点，但能够拓展，拓展n的所有子节点加入到open表，重新计算这些子节点和它们祖先的h值，转2)。
* 博弈树的极大极小过程和α-β剪枝
  + 一次拓展两层，每一层与或节点交替出现。或节点的值为其子节点的最大值，称为α值。与节点的值为其子节点的最小值，称为β值。通常是估计叶子结点的值，倒推回根节点的值。
  + α-β剪枝：对与节点，如果它的β值小于或等于其父节点（一定是或节点）的α值，该与节点的后续子节点停止搜索（因为父节点现在要找比当前α值更大的子节点，而目前这个与节点只能选到β≤α的值，后续节点被剪枝）。对或节点同理，如果它的α值大于或等于其父节点的β值，后续子节点停止搜索。（剪枝时注意检测每个节点和及其父节点的值！）

# 不确定性推理

* 主观贝叶斯
  + 全概率公式：



* + 贝叶斯公式：



* 贝叶斯网络



利用4式分别解出和（因为往往比好解），再利用1、2、3式解出α，最后带入1式得到（或通过归一化直接得到）。特别注意4式需要拓展到全联合概率分布表示，如还有变量Y,Z，则

# 机器学习和深度学习

* ID3决策树
  + 设(S,X)为根节点，其中S={s1,s2,…,sn}为整个样本集，X={x1,x2,…,xm}为全体属性，计算节点(S,X)的信息熵E(S)，再计算(S,X)关于每个属性xt的加权信息熵E(S|xt)，利用Gain(S,xt)=E(S)-E(S|xt)得到信息增益，选择使得Gain最大的xt进行分类，得到新的子节点(S1’,X’)和(S2’,X‘)（可能还有S3’、S4’等），X’=X-xt，S1’和S2‘是S依据xt划分成的两个子样本集。若S1’或S2’中的样本对应的都是同一类Y1，则令对应节点为叶子节点，其类别即为Y1；若X’为空，令这两个节点为叶子节点，按照少数服从多数的原则分配其类别；否则，递归处理这两个节点。
* 填充、卷积、池化的概念
  + Padding：向输入数据周围填充对应圈数数据，通常是填充0
  + 卷积：按照指定步长在输入数据上用指定的卷积核做卷积操作。（是否需要将所给的卷积核旋转180度？）
  + 池化：按照指定步长在输入数据上用指定大小的池化窗口做指定池化操作。
  + 注意步长在所有轴上都适用，也就是说向右移动窗口时要考虑步长，向下移动窗口时也要移动步长。

# 强化学习

* 术语部分：
  + Agent
  + 状态（State）
  + 动作（Action）
  + 策略（Policy）：表现为概率π(a|s)，意为在状态s下采取动作a的概率
  + 奖励（Reward）
  + 状态转移：表现为概率p(s’|s,a)，以为在状态s下，采取动作a后状态转移到s‘的概率
  + Return（未来的奖励和）：表现为Ut=Rt+γRt+1+γ^2Rt+2+…，其中γ是(0,1)之间的常数，表示未来的奖励没有现在重要。
    1. 由于Ut依赖于Rt，而Rt实际上依赖于St和At，所以Ut依赖于St和At
* 动作价值函数Qπ(s, a)：有两个参数s和a，表示为在状态s下，执行动作a时能获得的期望Return，其中π指定了策略。Qπ(s, a)=E[Ut|St=s,At=a]
* 状态价值函数Vπ(st)：有一个参数st，表示在状态s下，无论执行什么动作a能获得的期望Return，Vπ(st)=EA[Qπ(st, A)]=Σa π(a|st)\*Q(st,a)