

# 人工智能基础：面向考试的复习

## 一、绪论

三大学派：

- 符号主义学派→谓词逻辑，归结原理
- 行为主义学派→Agents
- 统计主义学派→机器学习

## 二、知识表示与知识图谱

### 一阶谓词逻辑

谓词的一般形式  $P(x_1, x_2, \dots, x_n)$

个体：  $x_1, x_2, \dots, x_n$

谓词名：  $P$

逻辑关系：  $\vee \quad \wedge \quad \neg \quad \rightarrow \quad \leftrightarrow$

量词：  $\forall \quad \exists$

$$p \rightarrow q \iff \neg p \vee q \quad p \leftrightarrow q \iff (p \wedge q) \vee (\neg p \wedge \neg q)$$

所有的机器人都是灰色的：  $(\forall x)(Robot(x) \rightarrow Color(x, grey))$

每一个雇员都有一个经理：  $(\forall x)(\exists y)(Employee(x) \rightarrow Manager(y, x))$

想要出国留学就必须通过考试：  $(\forall x)(\neg pass(x, exam) \rightarrow (\neg study(x, abroad)))$

## 三、确定性推理方法

### 归结原理

反证法：  $P \rightarrow Q \iff P \wedge \neg Q = F$

原子谓词公式：不能再分解的命题

文字：原子谓词公式及其否定

子句：任何文字的析取式

Skolem化：用Skolem函数代替每个存在量词量化的变量的过程。  $(\forall x)(\exists y)(P(x, y) \wedge Q(x, y))$  化为  $P(x, f(x)) \wedge Q(x, g(x))$

Skolem标准型：不含存在量词的标准前束析取范式

前束型：前缀(全称量词串)+母式(不含量词的谓词逻辑)

谓词公式化为子句集：以  $(\forall x)((\forall y)P(x, y) \rightarrow \neg(\forall y)(Q(x, y) \rightarrow R(x, y)))$  为例

1. 消去蕴含和等价：  $(\forall x)(\neg(\forall y)P(x, y) \vee \neg(\forall y)(\neg Q(x, y) \vee R(x, y)))$
2. 否定紧贴谓词：  $(\forall x)((\exists y)\neg P(x, y) \vee (\exists y)(Q(x, y) \wedge \neg R(x, y)))$
3. 变量标准化：  $(\forall x)((\exists y)\neg P(x, y) \vee (\exists z)(Q(x, z) \wedge \neg R(x, z)))$
4. 消去存在量词：  $(\forall x)(\neg P(x, f(x)) \vee (Q(x, g(x)) \wedge \neg R(x, g(x))))$
5. 化为前束型：  $(\forall x)(\neg P(x, f(x)) \vee (Q(x, g(x)) \wedge \neg R(x, g(x))))$
6. 化为Skolem标准型：  $(\forall x)((\neg P(x, f(x)) \vee Q(x, g(x))) \wedge (\neg P(x, f(x)) \vee \neg R(x, g(x))))$
7. 略去全称量词：  $(\neg P(x, f(x)) \vee Q(x, g(x))) \wedge (\neg P(x, f(x)) \vee \neg R(x, g(x)))$
8. 消去合取词：  $\{\neg P(x, f(x)) \vee Q(x, g(x)), \neg P(x, f(x)) \vee \neg R(x, g(x))\}$

9. 子句变量标准化:  $\{\neg P(x, f(x)) \vee Q(x, g(x)), \neg P(y, f(y)) \vee \neg R(y, g(y))\}$

归结原理:

$$\neg P \vee Q, R \vee \neg Q \Rightarrow \neg P \vee R$$

例:  $A \vee B, B \vee C, C \vee D, \neg A \vee \neg C, \neg B \vee \neg D$

$$A \vee B, \neg A \vee \neg C \Rightarrow B \vee \neg C$$

$$B \vee \neg C, B \vee C \Rightarrow B$$

$$B, \neg B \vee \neg D \Rightarrow \neg D$$

$$\neg D, C \vee D \Rightarrow C$$

$$C, \neg A \vee \neg C \Rightarrow \neg A$$

综上,  $A = D = F, B = C = T$

## 五、搜索求解策略

### 基础概念

状态空间的表示:  $(S, O, S_0, G)$  (状态集, 操作集, 初始状态, 最终状态或者满足某些性质的路径信息描述)

PS(path states)表: 保存当前搜索路径上的状态

NPS(new path states)表: 新的路径状态表, 包含了等待搜索的状态,

NSS(no solvable states)表: 不可解状态集, 用户判定某个状态是否有必要继续扩展

open表: NPS表

closed表: PS表+NSS表

A算法: 使用估价函数f的最佳优先搜索

A\*算法: 当A搜索算法的启发函数 $h(n) \leq h^*(n)$ 时, 称为A\*算法

$h(n)$ 单调  $\iff h(n_1) - h(n_2) \leq cost(n_1, n_2)$ , 后者为从 $n_1$ 到 $n_2$ 的真实代价且目的启发函数值为0

### A\*求解八数码

(1) 证明: 对于任何一个特定的状态 $S = a_1a_2a_3a_4a_5a_6a_7a_8$ 来说, 无论采取何种操作(上下左右)得到 $S'$ ,  $S$ 和 $S'$ 的逆序数相同。

左右移动: 原序列不变, 显然成立。

上下移动: 假设移动的元素为 $a_i$ , 原序列为 $\cdots a_{i-2}a_{i-1}a_i a_{i+1}a_{i+2} \cdots$ ,

向上移动得到:  $\cdots a_i a_{i-2}a_{i-1}a_{i+1}a_{i+2} \cdots$ , 进行了两次对换, 不会改变原有排列的奇偶性。也可以分情况讨论。

- 若 $a_i > a_{i-2} > a_{i-1}$ , 则逆序对变化量为+2, 奇偶性不变
- 若 $a_i < a_{i-1} < a_{i-2}$ , 则逆序对变化量为-2, 奇偶性不变
- 若 $a_{i-1} < a_i < a_{i-2}$ , 则逆序对变化量为0, 奇偶性不变
- 若 $a_{i-2} < a_i < a_{i-1}$ , 则逆序对变化量为0, 奇偶性不变

综上, 无论采取何种操作, 不会改变原序列的奇偶性。

(2) 给出求解过程并证明是A\*算法

估价函数:  $f(n)$ 从初始节点经过  $n$ 节点到达目标节点 的路径的最小代价估计值,  $f(n) = g(n) + h(n)$

$g(n)$ : 从初始节点到节点 $n$ 的真实代价

$h(n)$ : 从节点 $n$ 到目标节点的最优路径的估计代价, 称为启发函数。

四种定义方式:

- 与目标棋局不相符的棋子数
- 将棋子移到目标位置的步数的总和
- 对每一对逆转数码乘以倍数
- 1和3的总和

证明方法:

- 如果某一问题有解, 那么利用A\*搜索算法对该问题进行搜索则一定能搜索到解, 并且一定能搜索到最优的解而结束。
- $h(n)$ 为节点 $n$ 与目标棋局不相同的位数,  $h^*(n)$ 为状态 $n$ 到目的状态的最优路径代价, 显然对于任意 $n$ 有  $h(n) \leq h^*(n)$
- 上例中的八数码A搜索树中满足上述条件, 因此也是A\*搜索树, 所得的解路为最优解路。

## 剪枝

对于2players的解空间树, 顶层为Max。

Max-Min的路径: 或路径, 标注圆弧

Min-Max的路径: 与路径

$\alpha$ 剪枝: 对min方剪枝

$\beta$ 剪枝: 对max方剪枝

通过回溯确定每个状态的01解, 通过路径与非关系向上扩展, 得到问题是否可解。

## 六、智能计算及其应用

### 遗传算法求解问题

求解函数 $f(x, y)$ 在 $-5 < x, y < 5$ 的情况下的最小值。

染色体: 设为两个32位的单精度浮点数表示的二元组 $(x, y)$

交叉: 对两个染色体的两个元素互换。即 $(x_1, y_1), (x_2, y_2)$ 变为 $(x_1, y_2), (x_2, y_1)$

变异: 对两个元素各随机挑选一个位, 将其位翻转。

适应度函数:  $\frac{1}{f(x,y)}$

算法:

随机产生10个浮点数二元组作出初始种群

进行100次迭代:

计算出每个染色体的适应度并计算出被挑选到的概率

根据概率使用轮盘赌随机挑选两个进行交叉, 将得到的新染色体加入种群

根据概率使用轮盘赌随机挑选一个染色体进行变异, 将变异得到的新染色体加入种群

从群体中随机挑选出一些个体保留至下一代, 将其余个体从种群中删除

进入下一轮迭代

## 七、专家系统与机器学习

### 动物识别专家系统

根据规则库一步一步推理即可。

### 机器学习的分类

#### (1) 按反馈类型分

- 有监督学习：通过已标记的数据来训练模型，模型根据输入数据和对应的输出进行学习。目标是使模型能够预测新数据的输出。
- 无监督学习：使用未标记的数据进行训练，模型从中学习数据的隐藏结构或模式，通常用于数据探索和聚类分析。
- 再励学习：在这种类型中，模型学习通过与环境进行交互来做出决策。它通过尝试最大化预期的累积奖励来学习适当的行为。
- 半监督学习：结合有标记数据和未标记数据的技术，旨在提高学习算法的性能，尤其在标记数据少但未标记数据丰富的情况下。

#### (2) 按映射类型分

- 演绎：从一般规则或事实推导出特定结论。在机器学习中，这可以理解为从已知的普遍规律或模型中推导出具体的预测或结果。
- 归纳：从特定事实或案例中推断出一般规律或模式。在机器学习中，归纳推理是从数据中学习普遍规律或模型的过程，用于生成新的见解或预测。
- 类比：基于相似性比较不同情境或问题之间的关系。在机器学习中，类比推理可以帮助将已学习到的知识或模式应用到新的领域或情境中，以解决类似的问题。
- 转导：从给定的前提或条件中推导出结论。在机器学习中，这可以理解为利用已有的知识或模型来推导出新的结论或预测，通常在逻辑推理或推断问题中使用

### 贝叶斯定理的应用

根据信息论的最小描述长度得到的假设 $h_{MDL}$ 就等于根据最大后验概率得到的假设 $h_{MAP}$

设定 $D$ 为样本集， $H$ 为假设集，则

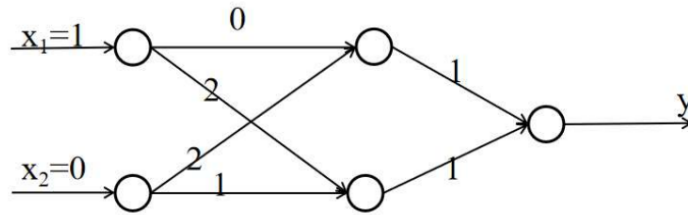
$$h_{MAP} = \underset{h \in H}{\operatorname{argmax}} P(D|h)P(h) = \underset{h \in H}{\operatorname{argmax}} \log_2 P(D|h) + \log_2 P(h) = \underset{h \in H}{\operatorname{argmin}} -\log_2 P(D|h) - \log_2 P(h) = L_{C_H}(H) + L_{C_{D|h}}(D|h) = h_{MDL}$$

其中， $L_{C_H}(h)$ 和 $L_{C_{D|h}}(D|h)$ 分别是 $H$ 和给定 $h$ 后 $D$ 的最优编码，在此编码方案下，任意 $h$ 与给定 $h$ 后 $D$ 的信息长度之和最小，则从信息论的角度来说， $h_{MAP}$ 具有最小描述长度。

## 八、人工神经网络及其应用

### BP神经网络前向传播

给定样本 $X=[1,0]$ ，标签为 $Y=1$ ，选学习率 $\varepsilon=1$ ,BP神经网络的初始权值矩阵选为  $W_1=\begin{bmatrix} 0 & 2 \\ 2 & 1 \end{bmatrix}$ ,  $W_2=\begin{bmatrix} 1 & 1 \end{bmatrix}$  隐层和输出层的神经元的输入与输出的非线性函数取为  $f(x)=\begin{cases} x, & x \geq 1 \\ 1, & x < 1 \end{cases}$ ，试用BP算法计算权值的调整过程。



注意第一层为输入层，不需要激活。上述神经网络需要两次激活过程，答案为3。

## 九、智能体与多智能体系统

### 反应式Agent的设计

问题	解决方案
<ul style="list-style-type: none"> <li>火星探测器在火星上收集岩石样本并把样本运回基地</li> <li>岩石的位置未知</li> <li>探测器可以接受到基地发出的无线电信号</li> </ul>	<ul style="list-style-type: none"> <li>1. 如果发现障碍物，则改变方向</li> <li>2. 如果处于基地并且携带着样本，那么放下样本</li> <li>3. 如果携带着样本且不在基地，则往无线电信号增强的方向移动</li> <li>4. 如果检测到样本，则采集样本</li> <li>5. 随机移动（if true）</li> </ul>

注意：序号为优先级，序号最小的优先级最高

### 博弈论

给定收益矩阵：

优势策略：任意一个分量大于其他所有元组的分量

佩瑞多最优：无法转变为各个分量均有提升的元组

社会福利最大：分量之和最大的元组

纳什平衡：A选s，B最好选t；B选t，A最好选s

## 十、卷积神经网络

### 互相关卷积的计算

对应相乘再相加。