人工智能：

* 研究如何用计算机来表示和执行人类的智能活动，以模拟人脑所从事的推理、学习、思考和规划等思维活动，并解决需要人类的智力才能处理的复杂问题等。
* 定义
  + 智能机器：能够在各类环境中自主地或交互地执行各种拟人任务的机器。
  + 人工智能（学科）：人工智能(学科)是计算机科学中涉及研究、设计和应用智能机器的一个分支。它的近期主要目标在于研究用机器来模仿和执行人脑的某些智力功能，并开发相关理论和技术。
  + 人工智能（能力）：人工智能(能力)是智能机器所执行的通常与人类智能有关的智能行为，如判断、推理、 证明、识别、感知、理解、 通信、 设计、 思考、 规划、 学习和问题求解等思维活动。
* 【注】
  + 固定了算法的，基本上不算AI。（目前世界上90%的计算都不属于AI范畴）
  + 自动控制类（飞机自动驾驶），科学计算类（石油勘探，天气预报，办公室自动化等）。
* 历史
  + 1950年，由艾伦图灵提出了计算机器与智能的概念（图灵测试，图灵机）
  + 1956年，达特茅斯会议，正式提出人工智能（Artificial Intelligence）的概念
* 分类
  + 弱人工智能，擅长于单个方面的人工智能。
  + 强人工智能，类似人类级别的人工智能，在各个方面都能和人类比肩的人工智能
  + 超人工智能，在几乎所有领域都比最聪明的人类大脑都聪明很多的人工智能。
* 三大流派：
  + 符号主义：把现实的物，映射到代表它的符号，在符号上完成所有的推理，计算等
  + 连接主义：人脑就是神经元之间的连接（神经网络模型）
  + 行为主义：只能来自更低级的感知和行动，表现的好就行
* 发展阶段
  + 20世纪50年代末-70年代初
    - 人工智能诞生
  + 20世纪80年代初-90年代初
    - 数学模型重大突破，诞生了很多神经网络、反向传播算法
  + 21世纪至今
    - 大数据、算法模型重大突破，深度神经网络提出
  + 新一代人工智能与前两轮浪潮的本质区别：
    - 新一代人工智能技术已经可以大规模应用于经济社会时间，给企业带来盈利，给社会管理带来实质性改善
    - 从学术驱动、研究驱动转变为需求驱动和产业驱动
  + 传统人工智能：自动化、电子、计算机之间的
  + 新人工智能：计算机与软件之间的
* 相关领域
  + 机器定理证明
  + 博弈
  + 模式识别
  + 自然语言处理
  + 数据挖掘和知识发现
  + 专家系统

图灵机：

* 七元组：
  + 纸带，符号，读写头，规则，状态，起始，结束（存储，符号，读写，程序，数据，开始，结束）
* 停机问题：
  + 判断任意一个[程序](https://baike.baidu.com/item/%E7%A8%8B%E5%BA%8F/13831935" \t "_blank)是否能在有限的时间之内结束运行的问题。该问题等价于如下的判定问题：是否存在一个程序P，对于任意输入的程序w，能够判断w会在有限时间内结束或者死循环。
  + 可计算性：并不是每一个函数都可以被描述为一个算法
  + 哥德尔定理：
    - 任何强壮的数学系统必定包含一个真的陈述句，然而在该系统中，这个陈述句却是不可证明的。（任何一个公理系统，都存在着一些命题无法被证明是真是假
    - 第一定理：任意一个包含一阶谓词逻辑与初等数论的[形式系统](https://baike.baidu.com/item/%E5%BD%A2%E5%BC%8F%E7%B3%BB%E7%BB%9F" \t "_blank)，都存在一个命题，它在这个系统中既不能被证明为真，也不能被证明为否。
    - 第二定理：如果系统S含有初等数论，当S无矛盾时，它的无矛盾性不可能在S内证明。
  + 康托尔对角线论证法：用于说明实数集合是不可数集的证明。
* 图灵测试：
  + 屋子里有人、计算机，外边的人提问，交互用打印机、电话等方式，如果提问者无法分辨是人回答的还是计算机回答的，就是通过了图灵测试
* 摩尔定理
  + 当价格不变时，集成电路上可容纳的元器件的数目，约每个18-24个月便会增加一倍，性能也将提升一倍

大数据：

* 指无法在一定时间范围内用常规软件工具进行捕捉、管理和处理的数据集合，是需要新处理模式才能具有更强的决策力、洞察发现力和流程优化能力的海量、高增长率和多样化的信息资产。（巨大量的数据）（相关性，非因果性）
* 特点
  + 大量（volume）
  + 高速（velocity）
  + 多样（variety）
  + 低价值密度（value）
  + 真实性（veracity）
  + 【注】：大数据时代大量使用相关性而不是因果性
* 分类
  + 局部大数据：部分领域内的数据
  + 打通的大数据：

专家系统（产生式系统）：

* 组成
  + 综合数据库（信息）
  + 产生式规则（知识）
  + 控制系统（规则的解释或执行程序）
* 特点：
  + 数据驱动
  + 知识无序
  + 控制系统与问题无关
  + 数据、知识和控制相互独立
* 类型
  + 正向、逆向、双向产生式系统
  + 可交换的产生式系统
  + 可分解的产生式系统
* 搜索策略：
  + 内容
    - 状态空间的搜索问题
  + 方式
    - 盲目式搜索
    - 启发式搜索
  + 关键
    - 利用知识，尽可能有效的找到问题的解
  + 回溯（递归）策略
    - 深度问题（深度加以限制）
    - 死循环问题（记录通过的路径状态）
  + 图搜索：保留所有已搜索过的路径
    - 基本概念
      * 节点深度
        + 根节点，0
        + 其他节点：父节点+1
      * 路径
        + 节点序列
        + 散耗值：路径中所有节点散耗值的总和
        + 扩展结点：生成该节点的所有后继节点，并给出他们的散耗值
    - 无信息图搜索过程
      * 深度优先：
        + 一般不能保证找到最优解；
        + 当深度限制不合理时，可能找不到解；
        + 最坏情况时，搜索空间等于穷举；
        + 与回溯法的差别：图搜索
        + 通用的与问题无关的方法
      * 宽度优先：
        + 当问题有解时，一定能够找到解；
        + 当问题为单位散耗值，且问题有解时，一定能找到最优解；
        + 方法与问题无关，具有通用性；
        + 效率较低；
        + 属于图搜索方法
      * 渐进式深度优先搜索方法：
        + 目的

解决宽度优先方法的空间问题和回溯方法不能找到最优解的问题

* + - * + 思想

给回溯法一个较小的深度限制，逐步渐增增加深度限制，直到找到解或找遍所有分支。（解决宽度优先方法的空间问题和回溯方法不能找到最优解的问题）

* + - * 启发式图搜索
        + 利用知识引导搜索，减少搜索范围，降低问题复杂度。
        + 启发信息的强度

启发强度强：降低搜索工作量，但可能导致找不到最优解

启发强度弱：工作量极大，极限情况下玮盲目搜索，但可能找到最优解

* + - * + 思想

定义评价函数f，对当前的搜索状态进行评估，找到一个最有希望的结点来拓展

f（启发函数）= g + h（评价函数）

* + - * + 算法：

A\*算法

f(n) = g(n) + h(n)，且h(n)≤h\*(n)

f\*（s）=f\*（t）=h\*（s）=g\*（t）=f\*（n）

定理

对有限图，如果从初始节点s到目标节点t有路径存在，则算法A一定成功结束

对无限图，若有从初始节点s到目标节点t的路径，则A\*不结束时，在OPEN表中即使最小的一个f值也将增到任意大，或有f(n)>f\*(s)。

A\*结束前，OPEN表中必存在f(n)≤f\*(s)。

对无限图，若从初始节点s到目标节点t有路径存在，则A\*一定成功结束

OPEN表上任一具有f(n)<f\*(s)的节点n，最终都将被A\*选作扩展的节点。

若存在从初始节点s到目标节点t有路径，则A\*必能找到最佳解结束。

A\*选作扩展的任一节点n，有f(n)≤f\*(s)。

如果h2(n) > h1(n) (目标节点除外)，则A1扩展的节点数≥A2扩展的节点数

* + - * + 队友线图

**重点**

归结原理

* 定理证明方法，从理论上解决了定理证明的问题
* 归结原理的题出对机器定理证明问题起到了推动作用
* 若S是合取公式F的子句集，则F永假的充要条件是S不可满足（nil∈S）（一种定理证明方法，从理论上解决了定理证明问题）
* 使用归结原理证明定理的思路
  + 目标的否定连同已知条件一起，化为子句集，并给出一种变换方法，使得S→S1→S2→... →Sn同时保证当Sn不可满足时，有S不可满足。
* 子句集
  + 无量词约束
  + 否定符只作用于单个文字
  + 元素默认为合区
  + 化子句集的方法
    - 消除蕴含符
      * a🡪b = ~a V b
    - 移动否定符号
      * ~(a ∨b) => ~a ∧~b
      * ~(a ∧b) => ~a ∨~b
      * ~(∃x)P(x)=>(∀x)~P(x)
    - 变量标准化
      * 对不同的约束，对应于不同的变量
      * (∃x)A(x) ∨(∃x)B(x) => (∃x)A(x) ∨(∃y)B(y)
    - 量词左移
      * (∃x)A(x) ∨(∃y)B(y) => (∃x) (∃y) {A(x) ∨B(y)}
    - 消除存在量词
      * 对于一个受存在量词约束的变量，如果他不受全程量词约束，则该变量用一个常量代替，如果他受全程量词约束，则该变量用一个函数代替。
      * (∃z) (∀x)(∃y){[(~P(x) ∧~Q(x)) ∨R(y)] ∨U(z)}=> (∀x) {[(~P(x) ∧~Q(x)) ∨R(f(x))] ∨U(a)}
    - 化为合取范式
    - 隐去全称量词
    - 表示为子句集
    - 变量标准化（变量换名）
* 谓词逻辑的归结原理
  + 关键：找到归结对
  + 基本概念
    - 置换
      * s={t1/v1, t2/v2, ..., tn/vn}对公式E实施置换s后得到的公式称为E的例，记作Es。
      * s1={z/x, A/y}, 则：P[x, f(y), B]s=P[z, f(A), B]
    - 合一
      * 如果存在一个S置换，使得{Ei}中E1s=E2s=E3s=...=Ens，则称{Ei}是可合一的。S为{Ei}的合一者。
      * {P(x, f(y), B), P(z, f(B), B)}置换s={A/x, B/y, A/z}是一个合一者, 因为：P(x, f(y), B)s= P(A, f(B), B)P(z, f(B), B)s= P(A, f(B), B)
      * 【注】：合一者不唯一
      * 最一般合一者（mgu）：置换最少，限制最少，产生的例子最具一般性
      * 合一算法
  + 谓词逻辑的归结方法
    - 对于子句C1∨L1和C2∨L2，如果L1与~L2可合一，且s是其合一者，则(C1∨C2)s是其归结式。
  + 提取回答的过程
    - 进行归结，证明结论的正确性
    - 用重言式代替结论求反得到子句
    - 按照证明过程，进行归结
    - 在原来为空的地方得到的就是提取的回答
    - 修改后的证明树被称为修改证明树

机器学习：寻找一个映射函数（最优的）

分类：监督学习（给定目标），非监督学习

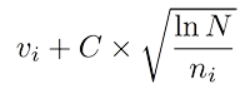
深度学习：深度🡪许多隐藏层

步骤：定义函数集，评价函数，选择最优的函数

神经网络结构：全连接神经网络，卷积神经网络，循环神经网络

博弈论

* 定义
  + 关于制定决策，具体而言，理性决策者之间冲突和合作的数学模型
* 应用
  + 经济，政治，物理，逻辑学，计算机科学，生物学
  + 行为关系，决策科学
* 特点
  + 至少两个玩家
  + 轮流 / 同步
  + 信息完全可知 / 信息不完全可知
  + 确定的 / 随机的
  + 合作的 / 竞争的
  + 零和 / 非零和
    - 零和
      * 目标对立
      * 纯竞争关系
      * 结果的总和为0
    - 非零和
      * 又相互独立的目标
      * 关系可以为合作，竞争，冷漠
      * 结果总和不为0
* 博弈树：
  + 确定的，信息完全可知的，轮流的，两个玩家，零和
  + 两个玩家分别为：Max，Min
  + 结点为局面，边为动作
* 搜索与博弈
  + 搜索
    - 没有对手
    - 解决方案是采用启发式方法去寻找目标
    - 启发式和CSP技术能够找到最佳方案
    - 评估函数：从开始到结束的总耗费
    - 路径规划，活动调度
  + 博弈
    - 解决方案是策略（特指每一步得到的对手的反馈）
    - 因为时间要求，强制得到一个近似解
    - 评估函数：评估决策的好坏
    - 下棋
* 极小极大搜索
* αβ剪枝：
  + 将min层选择的值作为β，将max层选择的值作为α
  + min层向下穿β，max层向下穿α
  + α剪枝：min结点的β值小于等于父节点的α值
  + β剪枝：max结点的α值大于等于父节点的β值
* 蒙特卡罗树搜索
  + 选择：从根节点 R 开始，递归选择最优的子节点（后面会解释）直到达到叶子节点 L。



* + 扩展：如果 L 不是一个终止节点（也就是，不会导致博弈游戏终止）那么就创建一个或者更多的字子节点，选择其中一个 C。
  + 仿真：从 C 开始运行一个模拟的输出，直到博弈游戏结束。
  + 反向传播：用模拟的结果输出更新当前行动序列

机器学习

* 思想：学习是一个系统通过经验提升能力的过程
* 机器学习是一个关于<P，T，E>的算法（P，表现；T，任务；E，经验）
* 传统程序：数据+程序🡪输出
* 机器学习：数据+输出🡪程序（寻找一个函数）
* 步骤
  + 获得一个函数集
  + 训练数据
  + 挑选一个优函数
* 学习也就是泛化
* 学习的种类
  + 监督学习
    - 训练数据+期望输出
    - 回归
    - 分类
  + 非监督学习
    - 训练数据
    - 聚类
  + 半监督学习
    - 训练数据+很少的期望输出
  + 强化学习
    - 对行为有反馈
    - 玩游戏
    - 平衡球
    - 信用预估
* 搭建学习系统
  + 选择训练领域
  + 选择学习目标（靶向函数）
  + 选择如何表示目标
  + 选择一个算法

深度学习

* 步骤
  + 定义函数集
  + 评估函数
  + 挑选最有的函数