

人工智能程序设计 课程回顾

黄书剑



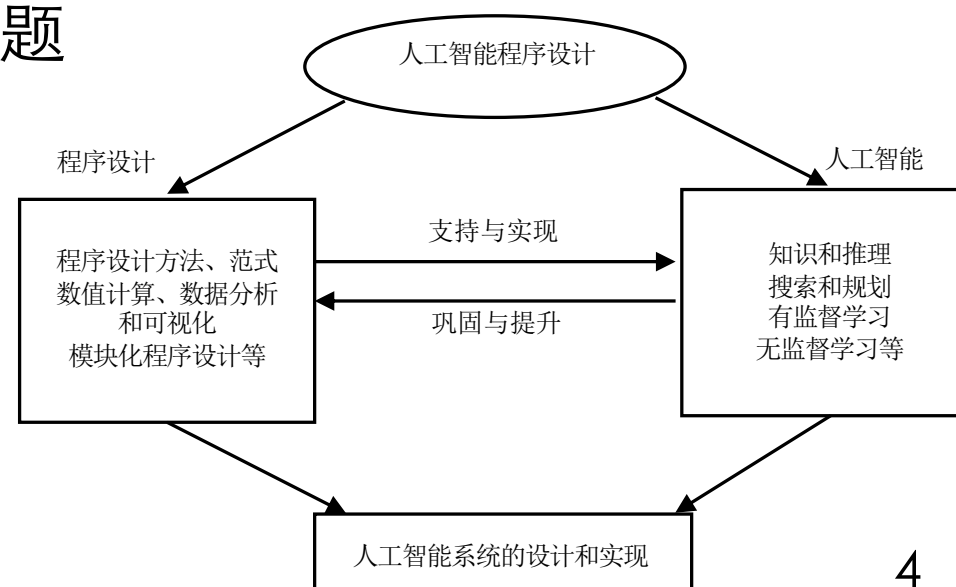
- Search 搜索
- Planning 规划
- Knowledge 知识表示
- Reasoning 推理
- Uncertainty 不确定性
- Learning 机器学习
 - Supervised Learning 有监督学习
 - Unsupervised Learning 无监督学习
 - Reinforcement Learning 强化学习
 - ...
- ...

Where are we?

- Application Software
- Operation/Computational System
- Architecture
- Hardware
- What we are learning:
 - Programming Language
 - Learning Algorithms
 - Applications

课程目标

- 强化利用程序设计语言进行实现的能力
 - 程序设计方法、范式（Python描述），数据分析处理，模块化程序设计等
- 建立面对多种人工智能问题的思维能力
 - 知识和推理、搜索和规划、机器学习等
- 掌握设计实现人工智能系统的基础能力
 - 解决各种类型的智能问题





第一部分：程序设计

- 初识python
- 流程控制、数据结构、模块
- 函数执行环境和状态
- 函数用于行为抽象
- 高阶函数
- 函数式程序设计
- 面向对象
- 继承和多态
- 动态类型

如何学习一门语言？

- 运行环境
 - 编译/解释运行
- 程序基本风格
- 基本符号
 - 变量、类型、表达式、语句
- 流程控制
- 函数
- 类和对象
- 其他（文件处理、异常处理等）



第二部分：数据分析处理

- 数据的抽象和表示 Numpy
- 科学计算
 - scipy、sympy
- 复杂数据分析和可视化
 - pandas、matplotlib

NumPy ndarray v.s. Python List

- 目标：更高的计算效率
- 更高效的数据组织
 - 同类型元素 + 固定长度
- 更高效的计算方式
 - 向量化编程 v.s. 循环
- 实现和底层支持
 - 基于C的实现
 - 利用Basic Linear Algebra Subprograms, BLAS
 - Intel MTK、Open BLAS、CUDA等

第三部分：人工智能问题及求解

- 搜索
- 有监督学习
 - 回归、分类
- 无监督学习
 - 降维
 - 聚类
 - 关联挖掘
 - 异常检测
- 逻辑和推理



课程考核

- 平时作业 15%
- 上机测验 25%
- 课程设计 20%
- 期末笔试 40%

- 解决问题是本质驱动力！
 - 方法和技术只是手段
 - 掌握现有方法和技术可以更好的解决问题
 - 不能被方法技术埋没了解决问题的初心！
- 自主探索、自主学习！
 - 以课堂为切入点
 - 不要被课本、课件、课堂束缚
- 实践是自我提升的最好途径！