

10 理解面向对象编程（OOP）

AI领域中的Python开发 — by 丁宁

SIGAI课程录制

AI学习与实践平台



www.sigai.cn

本节课的OKR

三节课之间的关系：层层递进

- 10：抛开Python语言，理解面向对象编程
- 11：用Python进行面向对象编程
- 12：用Special Method实现Pythonic面向对象编程

本节课的Object

- 抛开代码，在认知层面理解面向对象的思维方式

本节课的Key Result

1. 能用面向过程的编程思维去解决问题或完成任务
2. 能用面向对象的编程思维去解决问题或完成任务
3. 能阐述清楚面向对象与面向过程的区别
4. 能清楚的对问题背景进行分析，在面向过程与面向对象直接作出合适的选择
5. 当决定使用面向对象设计时，能遵守面向对象设计的通用原则

什么是面向过程，什么是面向对象

面向过程式编程（符合人类思考方式的一种编程模式）

1. 来了一个任务
2. 对任务进行流程分解，得到任务的不同阶段
3. 对不同阶段内的子任务分解，直到可以用分支语句和循环语句分解到几个表达式为止
4. 将不同子任务连接在一起，共同完成总任务
5. 识别重复性工作，抽象出基于任务或功能的函数，对整个程序进行重构
6. 实现功能函数（func）与执行过程（main）分离
7. 基础功能不变的情况下，下一个任务中可以复用的是函数

面向对象式编程

1. 来个一个任务
2. 找到任务中所有的利益相关方，并对利益相关方进行归类
3. 找到每个利益相关方需要履行的职责和希望被满足的需求，并进行分类
4. 将每个利益相关方类别的属性量化，并定义明确的行为，抽象出一个类别

5. 将所有同一类的利益相关方用共同的类模板进行实例化
6. 实现类别（class）与执行任务（main）的分离
7. 识别不同类别之间的关系，梳理衍生关系
8. 基于各种不同类别之间的衍生关系，抽象出基础类别
9. 将所有类别重构至每个类别都基于各自的基类层层继承而来
10. 利益相关方不变的情况下，下一个任务中可以复用的是抽象基类和各种派生类

举例说明

任务：训练一个基于深度卷积神经网络的人脸识别模型，并将其封装成可调用API部署上线

基于过程分解出子任务：

1. 找到合适的用于训练模型的数据库
2. 定义模型的结构，loss函数，优化方法
3. 根据模型的输入输出，将数据库制作为train, validation, test三个数据集
4. 多次训练模型，寻找合适的超参数
5. 找到训练出的最优方案，封装成RESTful API部署上线

基于对象分解出利益相关方：

1. 数据：收集，验证，转换，加载，切分，增强
2. 模型：结构，loss，优化器，超参数，训练流程，评估，使用
3. 基础设施：模型定义环境，模型训练环境，模型部署环境

面向对象与面向过程的区别？

- 面向过程思考的出发点是事情本身
- 面向对象思考的出发点是假设所有事物都有生命，他们之间会怎么分工协作
- 两种编程思路无明显优劣之分，一切只看适不适合
- 评估方法：预测未来，找到复用概率最高的几个点
- 将这几个点用如下原则进行初步评估：**高内聚，低耦合**
- 按照以上方法评估之后，心中往往已经有了答案

一般而言：

- 注重**复用和可维护性**时，OOP多数要胜出的
- 注重**短期开发速度**，而且是一次性的，面向过程肯定是首选

举个例子，以下两种场景下，你的选择肯定是有区别的：

1. 你未来一两年内都要做人脸识别研究；
2. 你在一家创业公司，做人脸识别模型仅仅是老板或者产品经理拍脑袋想出来的.....

比设计模式更重要的是设计原则

面向对象设计的目标

1. 可扩展：新特性很容易添加到现有系统中，基本不影响系统原有功能
2. 可修改：当修改某一部分代码时，不会影响到其他不相关的部分
3. 可替代：用具有相同接口的代码去替换系统中某一部分的代码时，系统不受影响

以上三点就是用来检测软件设计是否合理的要素

面向对象设计的SOLID原则

1. 单一职责原则：设计出来的每一个类，只有一个引起这个类变化的原因
2. 开闭原则：对扩展开放，对修改封闭
3. 替换原则：父类能用的功能，换成子类一样可以用
4. 接口隔离原则：接口设计要按需供给（类似微服务设计）
5. 依赖倒置原则：抽象不依赖于细节，细节应该依赖于抽象（针对接口编程）

遵循以上原则去设计面向对象程序，效果一般不会很差

AI（场景）+ Python（语言）+ OOP（编程模式）：

- AI：业务导向不明显，需求变动频率较低，实现和复现频率较高
- Python：虽然是一门面向对象语言，但与传统的面向对象并不相同
- OOP：使用Python时，并不需要深入学习OOP或者OOD那些理论

用Python这门面向对象语言去做AI开发，只需要懂两句话：

高内聚，低耦合

抽象不变的接口，封装变化的细节

AI学习与实践平台



www.sigai.cn