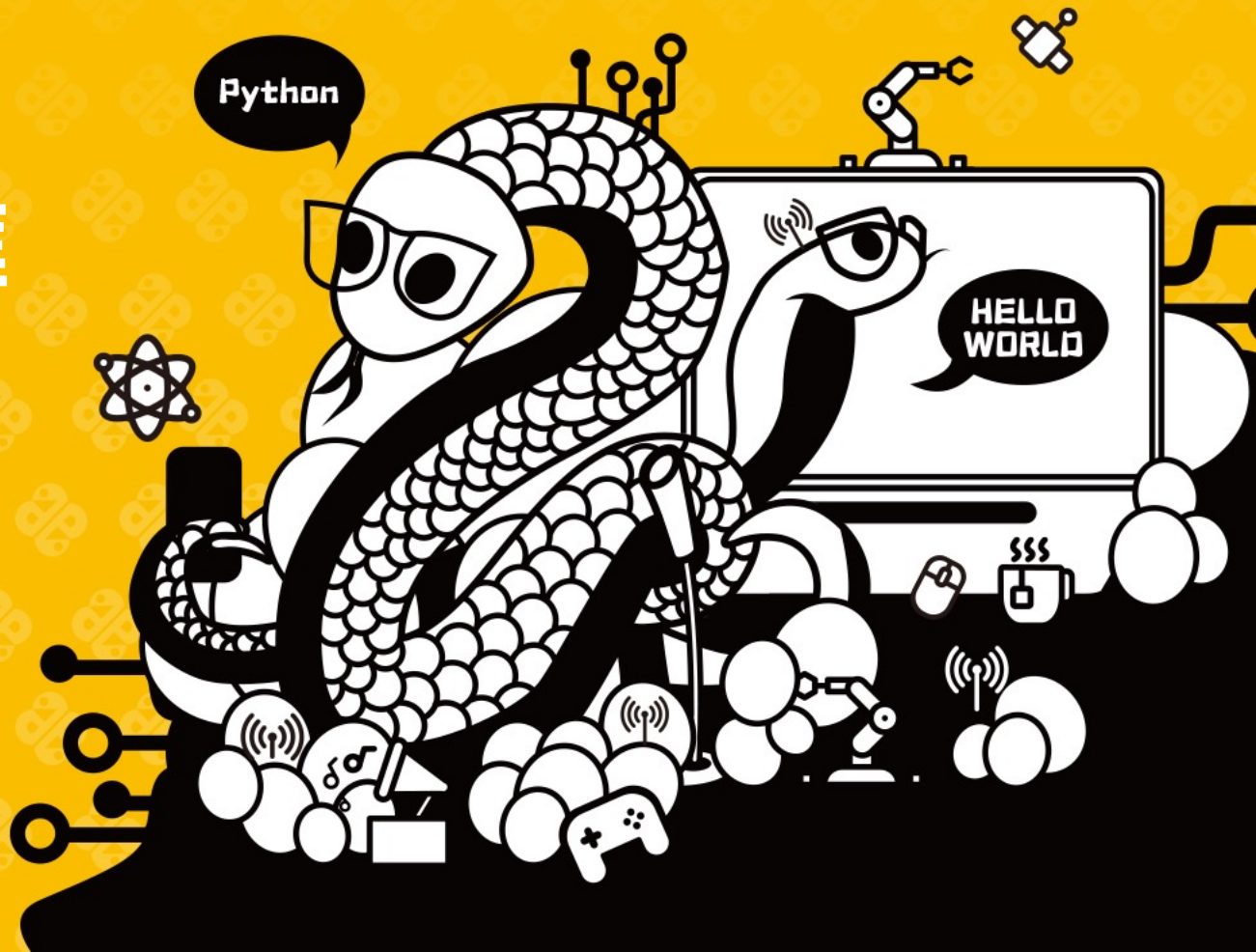
python × PYCHINA Python 中国社区

Python for Good

PyCon China 2022

基于Python的深度学习框架设计与实现

主讲人： 刘凡平



一、背景

二、原理：深度学习框架的一般性结构

三、设计

四、应用案例

五、思考

深度学习框架是包含深度学习模型设计、训练和验证的一套标准接口、特性库和工具包，集成深度学习的算法封装、数据调用以及计算资源的使用，同时面向开发者提供了开发界面和高效的执行平台，是算法工程师的必备工具之一。

美国互联网对AI底层技术战略性投入力度较大，但中国的AI产业主要受需求拉动，大多数人工智能公司布局应用层。

一、背景：深度学习框架的演进

深度学习框架的演进

萌芽阶段

- 时间范围：2000年-2012年左右
- 使用特点
  - 1、API复杂；
  - 2、手写神经网络结构；
  - 3、无GPU算力；
- 应用情况：受限于算力不足，神经网络技术影响力相对有限，更多以机器学习工具的形式产生。
- 示例框架：OpenNN、Torch

发展阶段

- 时间范围：2012年左右-2016年左右
- 使用特点
  - 1、生态友好；
  - 2、支持GPU以及分布式计算；
  - 3、性能优化、支持扩展等；
- 应用情况：使用效果高于之前传统机器学习水平，融合CNN、RNN等经典模型结构
- 示例框架：Caffe、Chainer、Theano等

深化阶段

- 时间范围：2020年-至今
- 使用特点
  - 1、全场景支持；
  - 2、超大规模AI；
  - 3、安全可信等；
- 应用情况：随着人工智能的进一步发展，新的趋势不断涌现。要求 AI 框架最大化的实现编译优化，更好地利用算力、调动算力，充分发挥硬件资源的潜力。
- 示例：华为推出昇思 MindSpore、旷视推出天元 MegEngine等

稳定阶段

- 时间范围：2016年左右-2020年
- 使用特点
  - 1、编译层优化；
  - 2、多场景、多任务支持；
  - 3、丰富的开发套件；
- 应用情况：在计算机视觉和自然语言处理的应用效果得到极大的提升，经过激烈的竞争后，最终形成了TensorFlow 和PyTorch两大阵营
- 示例框架：PyTorch、TensorFlow、MXNet、PaddlePaddle等

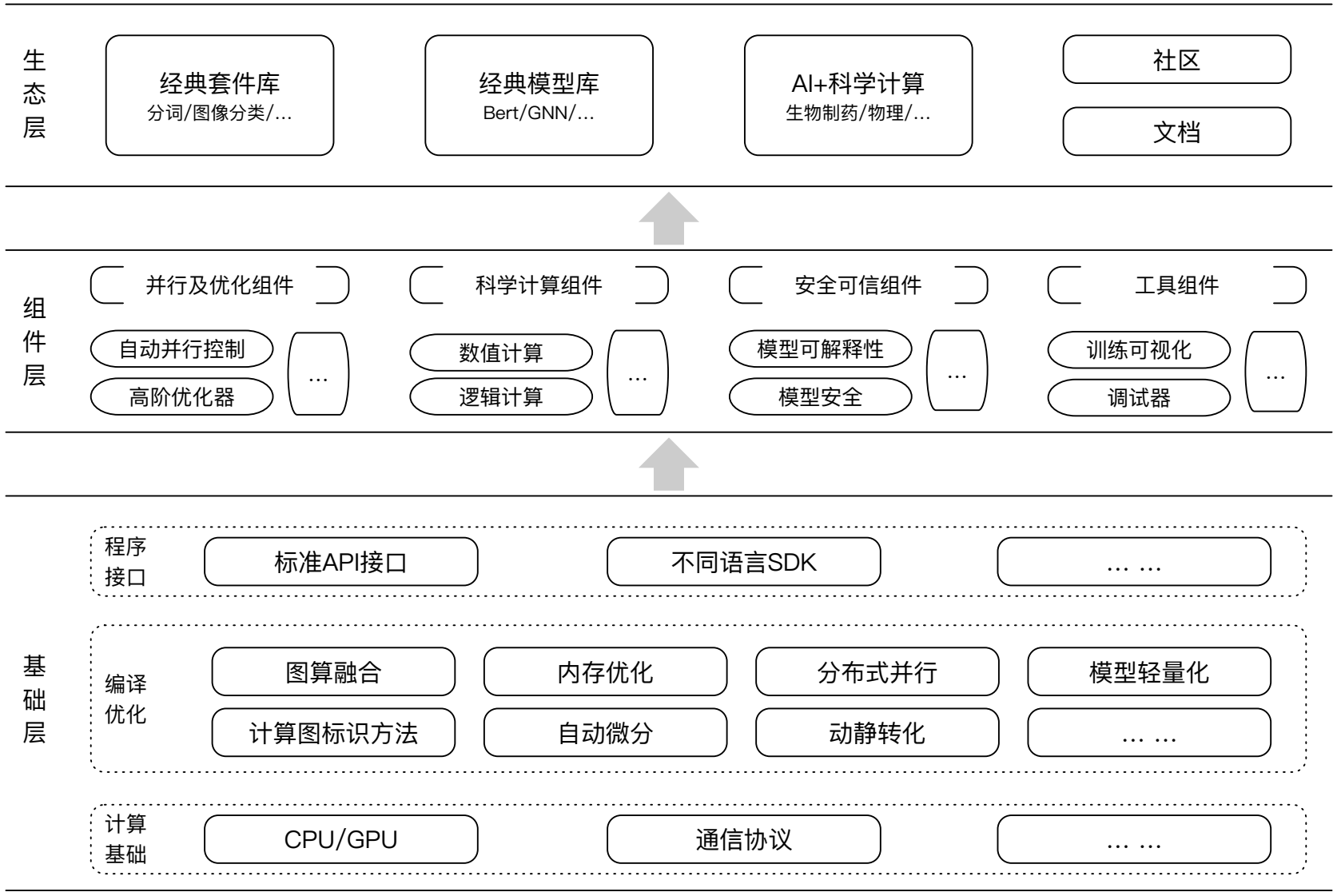
一、背景：深度学习框架的重要性

二、原理：深度学习框架结构

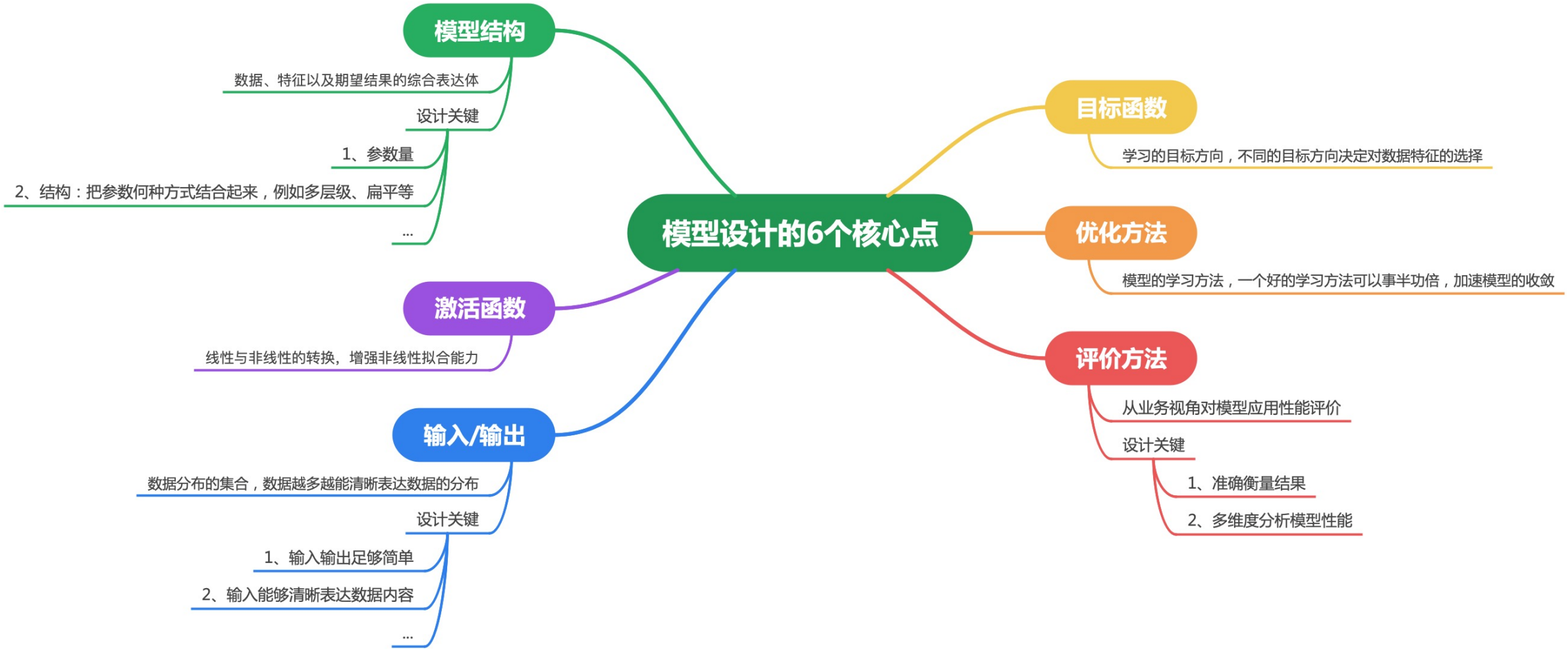
面向应用服务提供解决方案或一般性方法。

提供模型生命周期中科配置的各类功能组件。

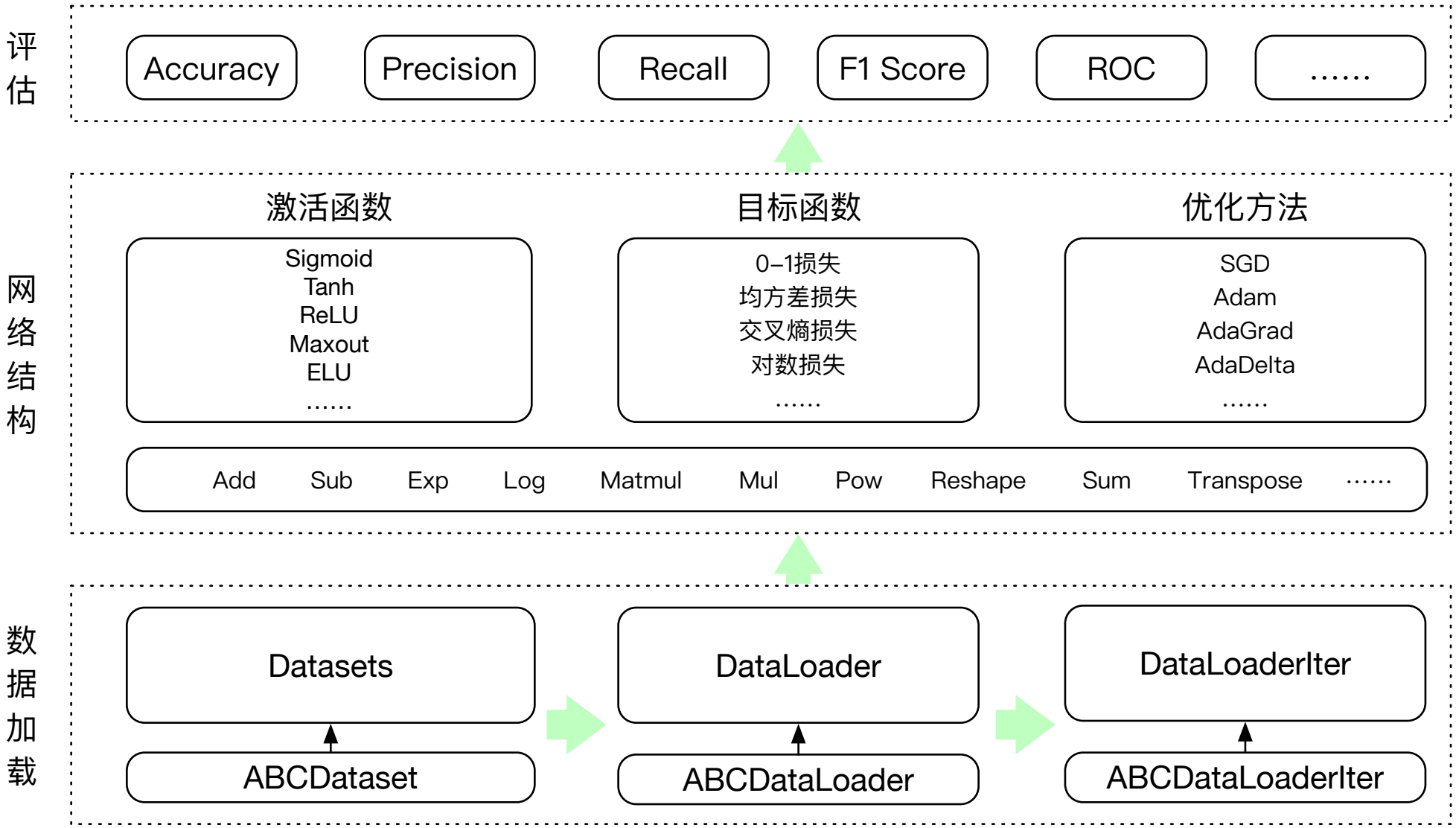
实现框架最基础、最核心的功能，帮助开发者屏蔽底层硬件技术细节。



从模型的设计者角度思考，一个模型设计的最小使用内容。



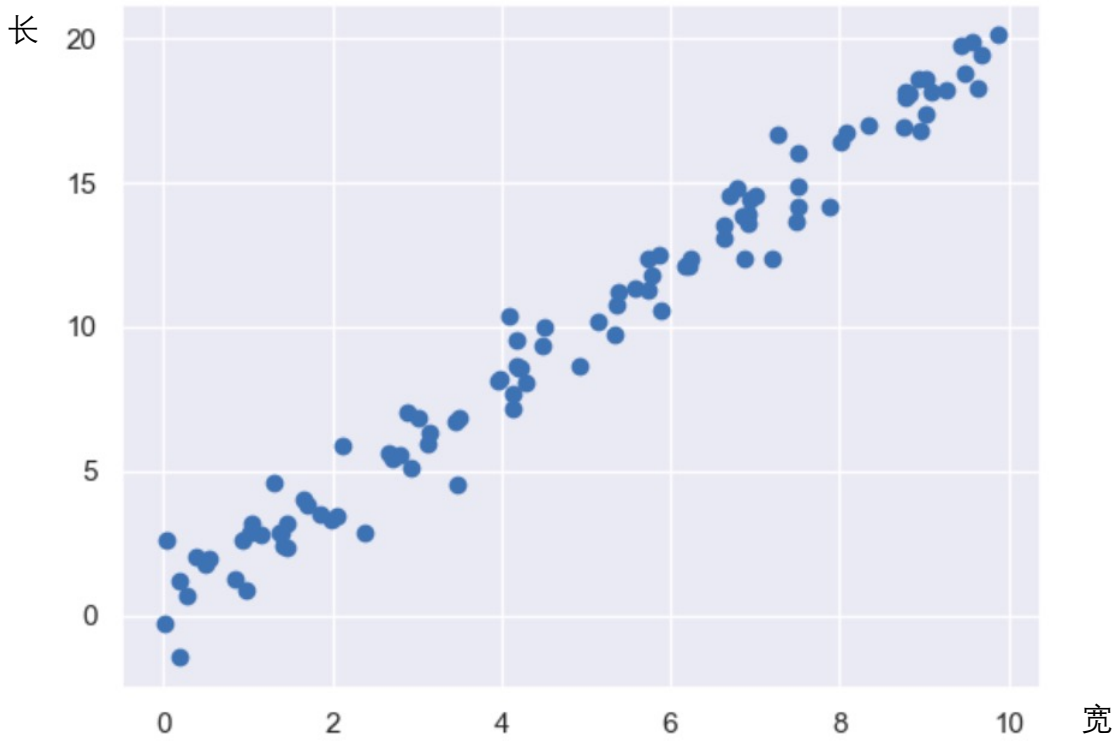
三、设计：最小MVP深度学习框架的层次逻辑



1、定义问题

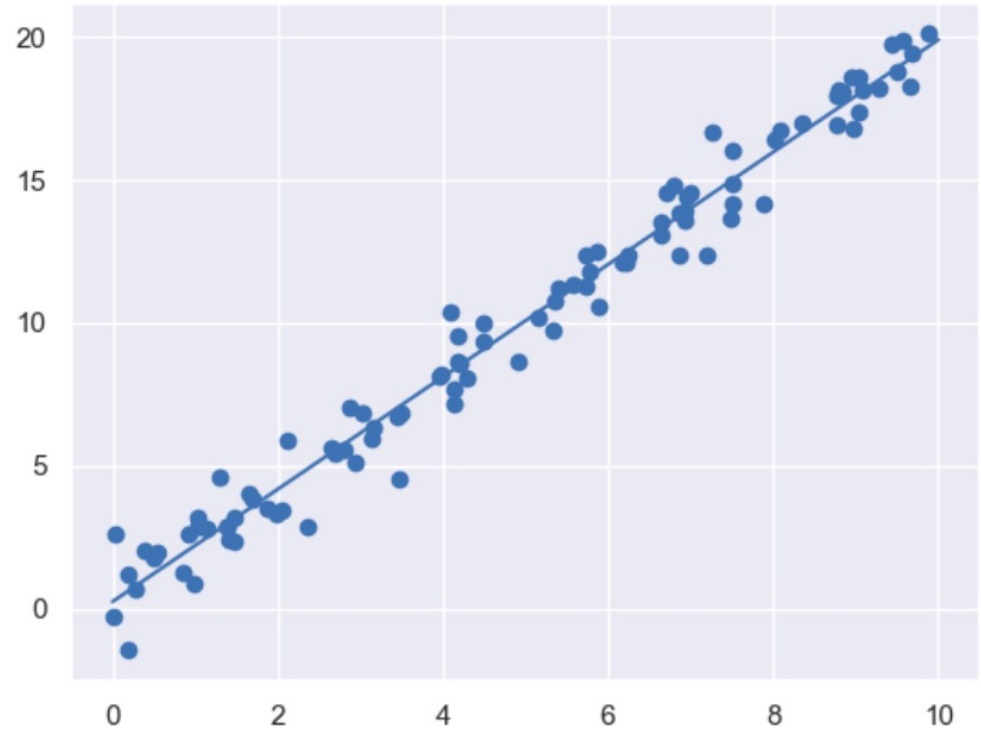
已知某类工业零件不同长与宽的产品数据集，求长与宽的关系。

2、宽与高的数据关系示意图：



模型结构：

```
class SampleModel(nn.Module):

    def __init__(self):
        super(SampleModel, self).__init__()
        self.line_layer = nn.Tensor.line(1, 1)

    def forward(self, x: nn.Tensor) -> nn.Tensor:
        out = self.line_layer(x)
        return out
```

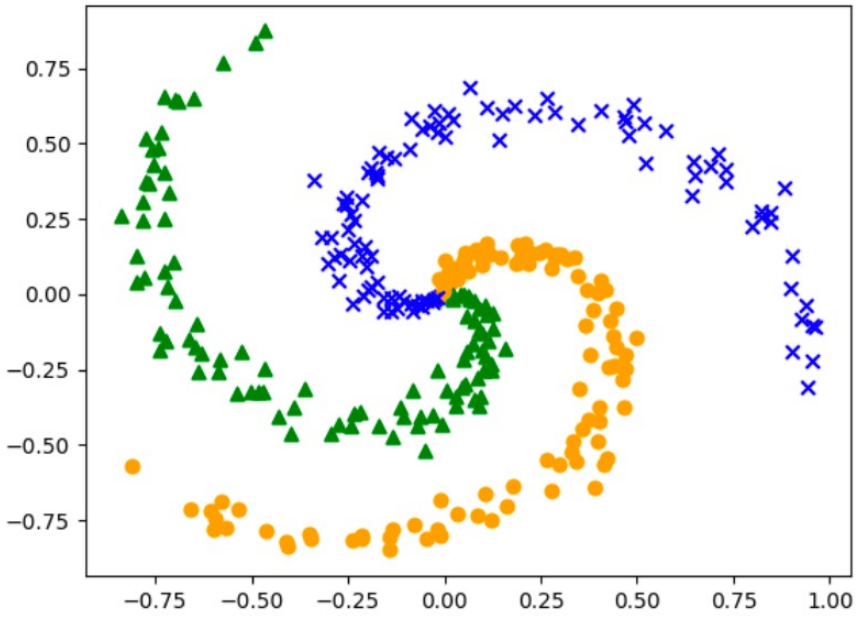


线性回归后的结果

1、定义问题

100个二维数据点，一共分为3个类别，分别识别出他们的分类信息。

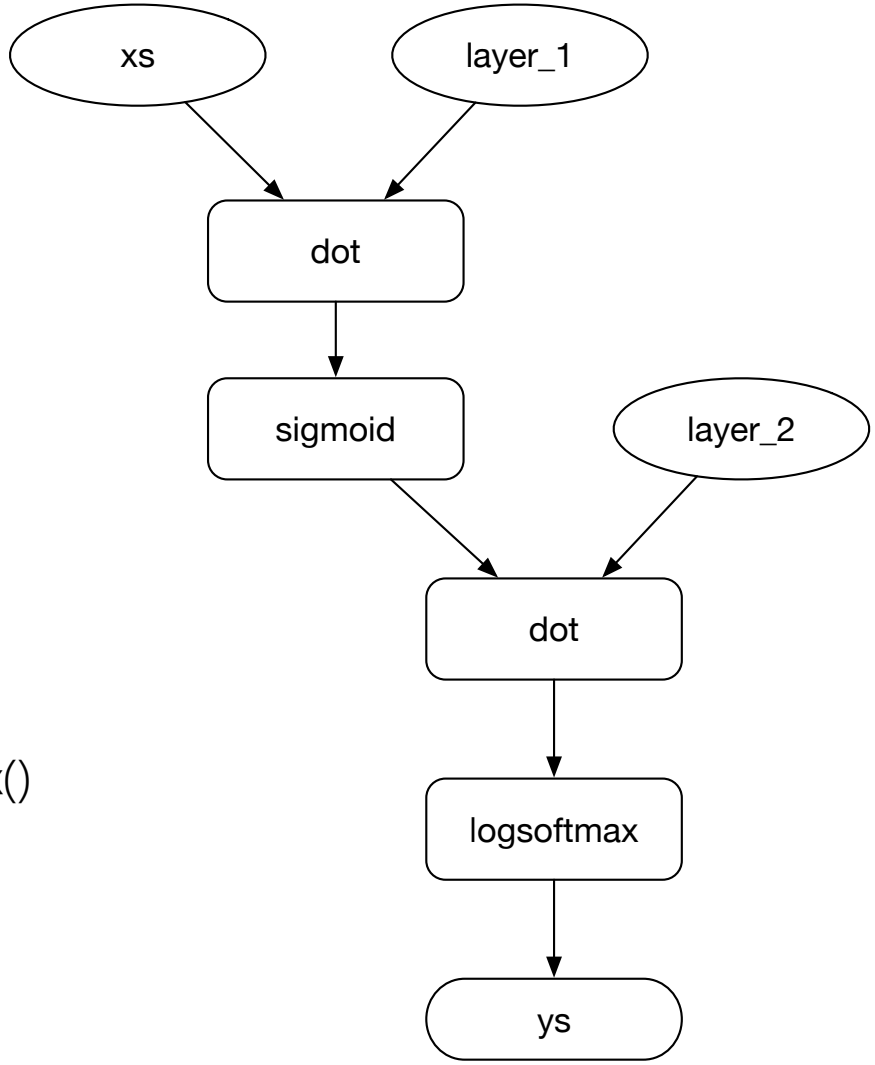
2、示例图



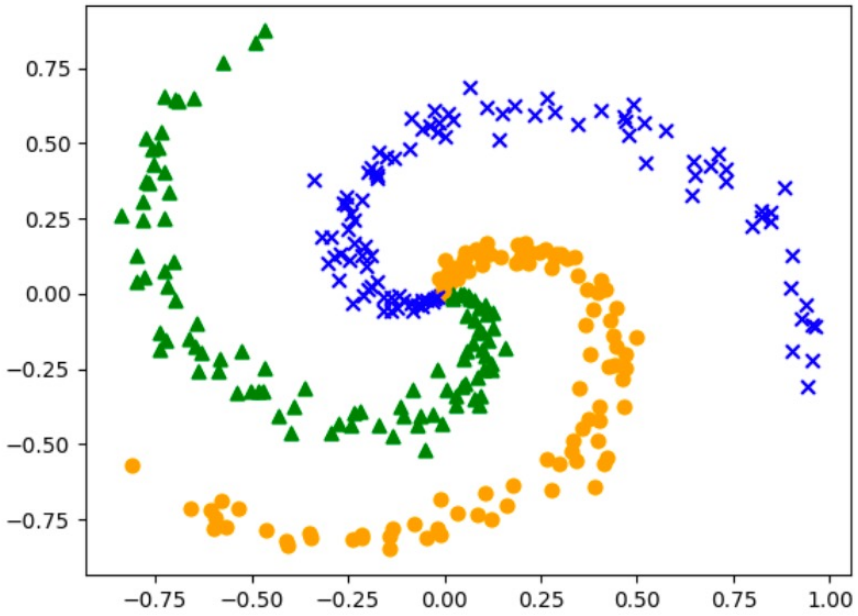
左侧为测试数据的样例（训练数据的分布与此类似），目前已经通过三种数据颜色和标识标注了他们所属的类别，要求通过模型分辨出划分出他们的类别。

模型结构：

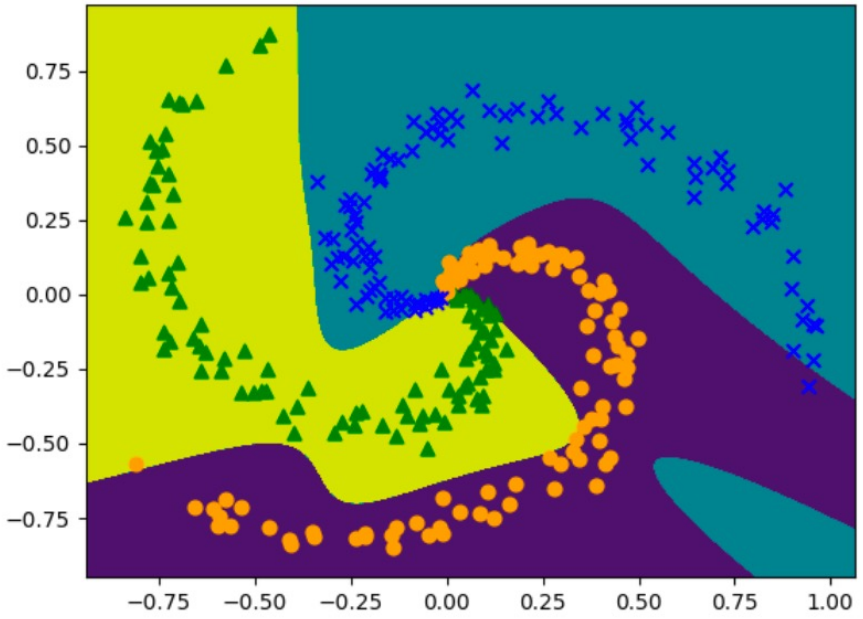
```
class SampleModel(nn.Module):

    def __init__(self, in_size, hidden_size, out_size):
        super().__init__()
        self.layer_1 = nn.Tensor.uniform(in_size, hidden_size)
        self.layer_2 = nn.Tensor.uniform(hidden_size, out_size)

    def forward(self, xs):
        # xs为原始的批量数据集合。
        return xs.dot(self.layer_1).sigmoid().dot(self.layer_2).logsoftmax()
```



训练10个epoch后的效果对比图：



测试数据分布



预测后分类区域图

- 为什么要设计一个深度学习框架？

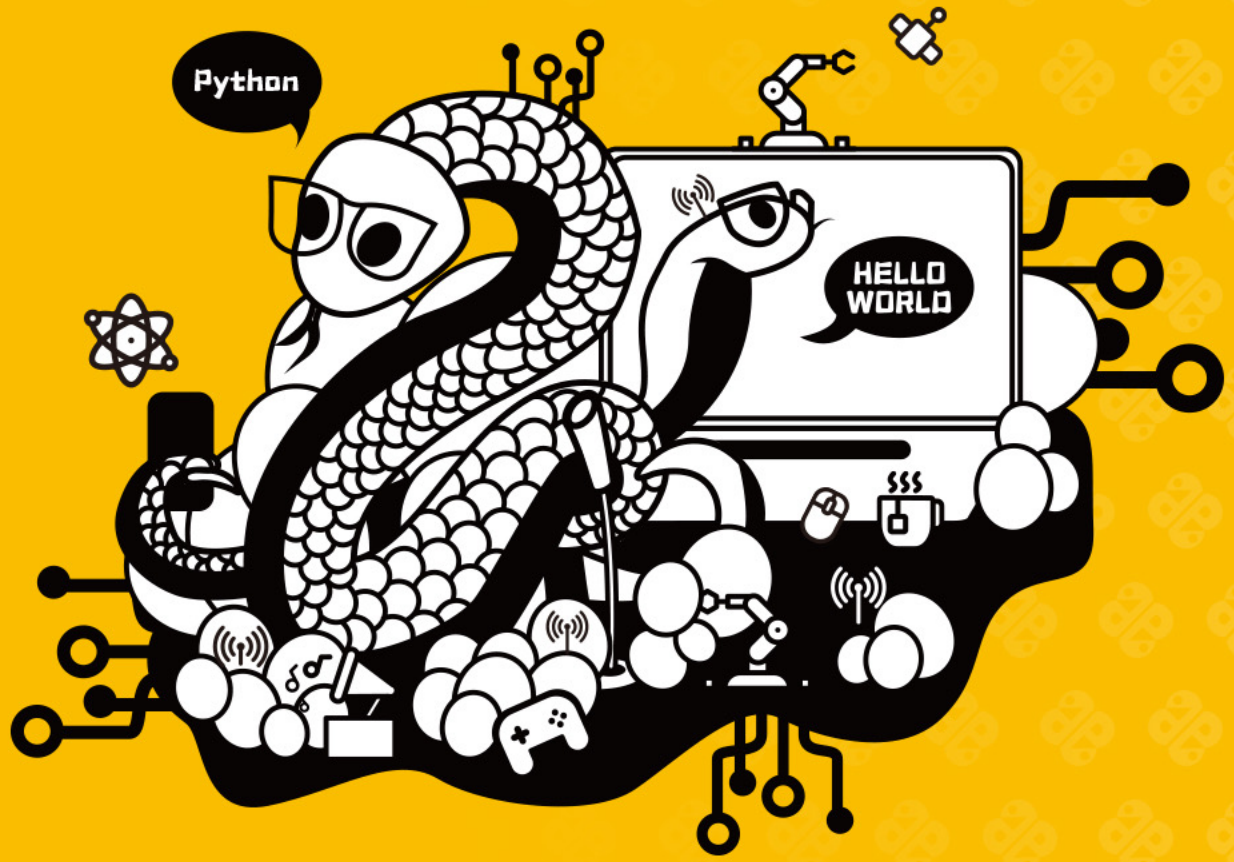
  切勿以造轮子的初衷去设计深度学习框架，一切均需**围绕业务**进行。脱离业务的技术体系价值不大。

- 是否存在完美的深度学习框架？

  一切以**落地场景**为根基，满足业务使用即可，不要过度设计，过度设计将会导致框架越来越复杂、臃肿。

- 实现的深度学习框架与目前主流开源的结果计算结果不一致怎么办？

  一方面深入原理，在掌握**系统理论基础**后去实现；另一方面每一个算法的实现细节都会不同，注意实现**细节**。



感 谢！

微信扫一扫，让我们成为朋友