

# Python 相关的代码

使用keras作为前端，tensorflow作为后端来训练CNN模型

## model

---

存放模型参数 / tensorflow运行的中间断点数据

## scripts

---

存放模型配置文件 <网络描述文件.txt>

存放各类脚本（批处理）

0. 创建运行环境文件夹.bat
1. 运行CNN训练.bat
2. 保存CNN模型的参数到csv文件.bat
3. 根据csv文件产生NPU指令CNN参数.bat
4. 测试CNN & 生成仿真样本.bat
5. 删除仿真样本.bat
6. 删除CNN模型参数.bat
7. 运行CNN推理.bat
8. 运行tensorboard可视化.bat
9. 评估模型硬件化后的开销.bat
10. 生成训练样本展示文件.bat

### 使用说明：

- 1 首先双击运行《0. 创建运行环境文件夹.bat》创建好运行环境
- 2 如果修改《网络描述文件.txt》，就要重新训练CNN，那么先运行《6. 删除CNN模型参数.bat》删除旧的模型，然后运行《1. 运行CNN训练.bat》
- 3 如果要继续训练CNN，那么运行《1. 运行CNN训练.bat》
- 4 如果仅仅是想看看CNN的推理效果，可以运行《7. 运行CNN推理.bat》
- 5 如果想看看硬件化以后的时间/资源开销，可以运行《9. 评估模型硬件化后的开销.bat》，在../isa-npu/time\_consuming.txt中可以查看
- 6 如果要生成FPGA硬件化的配置文件，首先运行《2. 保存CNN模型的参数到csv文件.bat》，然后运行《3. 根据csv文件--产生NPU指令 & CNN参数.bat》，就能生成NPU指令和CNN参数，并生成FPGA需要的mif文件
- 7 如果要生成modelsim能够使用的仿真样本，运行《4. 测试CNN & 生成仿真样本.bat》；如果要删除，就运行《5. 删除仿真样本.bat》

### 关于网描述文件：

- 1 用“I” / “Lk” 来区分输入层和内部第k层
- 2 对于输入层，格式为“I, H, W, Ch”，表示输入层是H高，W宽，Ch通道的输入
- 3 对于第k层，如果是卷积层，格式就是“Lk, C, Hk, Wk, Chk, func”，其中，C表示卷积，Hk是卷积核的高，Wk是卷积核的宽，Chk是卷积层输出通道数，func是激活函数

- 4 对于第k层，如果是池化层，格式就是“Lk, S, Hk, Wk”，其中，S表示下采样/池化，Hk是池化核的高，Wk是池化核的宽
  - 5 对于第k层，如果是压平层，格式就是“Lk, STRIP, D”，其中，STRIP表示压平，D是压平后的维度（这里需要*自行计算*，也可以写“-1”来表示自动推导）
  - 6 对于第k层，如果是全连接层，格式就是“FC, D, O, func”，其中，FC表示全连接，D是输入维度【也可以写“-1”来表示自动推导】，O是输出维度，func是激活函数
  7. 在每一层的结尾加上"dropout"可以开启训练时候的dropout
- 指令数量上限为**1023**条，参数数量上限为**64K**个

## source

---

存放源代码

### · train\_my\_cnn\_model.py

训练CNN模型参数

### · save\_parameters.py

保存CNN模型参数到csv文件（在para文件夹中）

### · generate\_npu\_inst.py

生成NPU的指令

### · test\_npu\_inst.py

生成测试NPU的样本

### · generate\_cnn\_layers.py

解析网络描述文件，并且生成CNN模型

### · generate\_cnn\_layers.py

解析网络描述文件，并且生成CNN模型

### · load\_our\_samples.py

加载CNN训练/测试用的样本

### · estimate\_time\_consuming.py

根据模型配置文件，评估运算时间/内存的开销

## · test\_npu\_on\_FPGA.py

通过串口发送NPU指令、配置CNN参数、传输测试图案到SRAM  
并回读CNN分类结果，与python结果进行比较，验证CNN硬件化的正确性

## test

---

测试用，可能tensorflow的某个功能不确定怎么用

## · test\_keras\_conv.py

测试卷积函数的实现过程

## para

---

里面存放了训练好的CNN模型参数（csv文件格式表格存储）

## samples

---

里面存放了语音识别的样本的展示