
作业二：深度神经网络

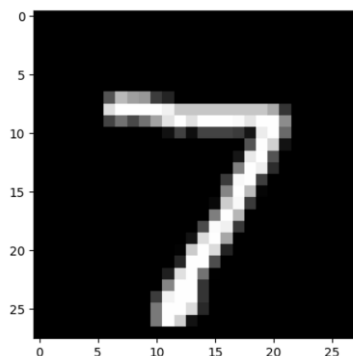
一、作业目标

搭建深度神经网络，并在给定数据集上进行训练与测试。具体来说：

- (1) 使用深度学习框架（TensorFlow、PyTorch、Caffe 等）完成网络搭建；
- (2) 不限制编程语言，推荐使用 Python 语言；
- (3) 数据集可以使用提供的 MINIST 数据集（mnist.npz），也可自行调用神经网络框架相应接口下载并加载 MINIST 数据集；
- (4) 可使用提供的代码模板（tf_template.py），在其基础上进行修改完善，也可以重新编写代码。代码模板提供了框架，包括数据 IO 与模型训练以及测试，主要任务在于对模型进行定义与参数调试。

二、数据集介绍

输入数据为取值 0~9 的手写数字集。其中训练集包含 60000 张数字图像，测试集包含 10000 张数字图像，每张图像均为 28*28 像素大小。如下图所示的手写数字“7”图片样例。



提供的数据集包含 4 个 .gz 文件：

- t10k-images-idx3-ubyte.gz: 训练集图片数组，使用 tensorflow 的 input_data.read_data_sets 函数读取后，其 shape 为(60000, 28, 28)，共三个维度，分别表示图片的数量、存储图片像素的数据。
- t10k-labels-idx1-ubyte.gz: 训练集标签数组，使用 tensorflow 的 input_data.read_data_sets 函数读取后，其 shape 为(60000,)，只有一个维度，表示训练集中相对应的图片的正确标签。
- train-images-idx3-ubyte.gz: 测试集图片数组，使用 tensorflow 的 input_data.read_data_sets 函数读取后，其 shape 为(10000, 28, 28)
- train-labels-idx1-ubyte.gz: 测试集标签数组，使用 tensorflow 的 input_data.read_data_sets 函数读取后，其 shape 为(10000,)

三、网络要求

1. 网络输入: (batchsz, 28, 28, 1)的数组, 其中 batchsz 为每批次中包含图片的数量, 这个数值可以根据自己硬件条件进行确定; 28 * 28 为给定的图片尺寸。
2. 网络输出: 10 个输出节点, 分别代表 0~9 这 10 个数字。本次作业不对精度做特别的要求, 只需在合理范围内即可。
3. 网络模型: 可以采用 LeNet, 也可以采用其他自己定义的网络模型, 调参并对比效果。

四、作业提交

1. 保存模型和权重数据, 让结果可以复现。
2. 需撰写结果报告, 包括但不限于网络结构介绍、关键代码介绍、测试精度结果截图等。更多细节参考“实验报告撰写格式”。
3. 将代码、权重数据、报告打包成压缩文件, 命名格式: 学号+姓名, 如 XXXXXXXXX+张三+作业二。

五、注意细节

1. 本次作业使用的 MNIST 数据集中包含训练集 60000 张图片, 如果在 CPU 上训练时间过长, 可以考虑采用 GPU, 或者减少训练集的大小 (但会降低训练后的模型精度)。
2. 代码文件说明:
 - tf_minist.py: 主文件, 包含训练、测试流程以及模型保存的实现
 - data_object.py: 数据类文件, 包含数据处理的实现
 - tf_network.py: 网络结构文件, 包含网络结构实现
 - tf_test_model.py: 测试参数固化后保存的模型准确率

六、附录: 使用 tensorboard 对数据进行可视化

TensorBoard 提供机器学习实验所需的可视化功能和工具, 建议学习和使用该工具:

https://www.tensorflow.org/tensorboard/get_started?hl=zh-cn

模板代码中已经写入了 tensorboard 相关语句, 程序运行结束后工程目录下将会生成 logs 文件夹, 接下来执行以下两步打开 tensorboard:

- 1) 在终端中输入: `tensorboard --logdir=./logs`
- 2) 按照提示在浏览器中输入获得的网址打开 tensorboard

如下图所示即是打开后的示例, 能够清晰地看到训练过程 loss 以及 acc 变化曲线:

