

Lab2 深度神经网络

北京航空航天大学计算机学院

1 实验目标

搭建深度神经网络，并在给定数据集上进行训练与测试。具体要求：

1. 使用深度学习框架(TensorFlow、PyTorch、Caffe 等)完成网络搭建,推荐使用 tensorflow-1.15
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 MNIST 数据集,也可自行调用神经网络框架相应接口下载并加载 MNIST 数据集
4. 可使用提供的模板项目(TF-Mnist-Template),在其基础上进行修改完善,也可以重新编写代码。模板提供了框架,包括数据 IO 与模型训练以及测试,同学们的主要任务为模型定义与参数调整

2 数据集获取与使用

本次实验采用 Mnist 官方数据集¹, 模板项目的 dataset 目录下亦有相关文件。首先观察 mnist 的文件结构, 以 train-images.idx3-ubyte 文件为例:

表 1: 数据集文件内容

TRAINING SET IMAGE FILE (train-images-idx3-ubyte):			
[offset]	[type]	[value]	[description]
0000	32 bit integer	0x00000803(2051)	magic number
0004	32 bit integer	60000	number of images
0008	32 bit integer	28	number of rows
0012	32 bit integer	28	number of columns
0016	unsigned byte	??	pixel
0017	unsigned byte	??	pixel
...			
xxxx	unsigned byte	??	pixel

前面若干字节为文件参数信息, 包括了图片数量, 图片行列大小等。从第 16 个字节起是图片的像素信息。

数据集文件总览 (以 tensorflow 为例):

- 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)

¹<http://yann.lecun.com/exdb/mnist/>

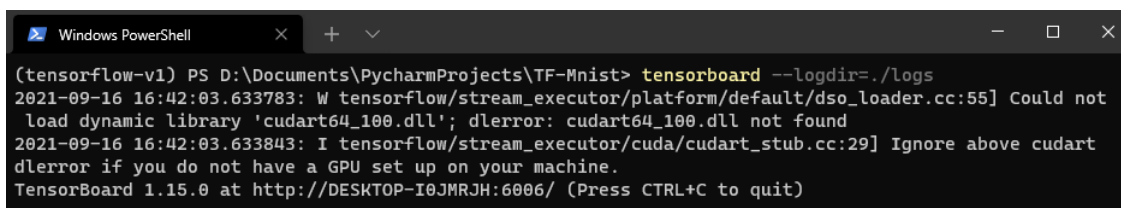
- train-labels-idx1-ubyte.gz: 测试集标签数组, 使用 tensorflow 的 `input_data.read_data_sets` 函数读取后, 其 `shape` 为 (10000,)

3 网络要求

1. 网络输入: (batch_size, 28, 28, 1) 的数组, 其中 batch_size 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: 建议采用 LeNet, 也可以采用其他自己定义的网络模型, 调参并对比效果。

4 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 XXXXXXXXXX + 张三 + 作业二。



```
(tensorflow-v1) PS D:\Documents\PycharmProjects\TF-Mnist> tensorboard --logdir=./logs
2021-09-16 16:42:03.633783: W tensorflow/stream_executor/platform/default/dso_loader.cc:55] Could not
load dynamic library 'cudart64_100.dll'; dlderror: cudart64_100.dll not found
2021-09-16 16:42:03.633843: I tensorflow/stream_executor/cuda/cudart_stub.cc:29] Ignore above cudart
dlderror if you do not have a GPU set up on your machine.
TensorBoard 1.15.0 at http://DESKTOP-I0JMRJH:6006/ (Press CTRL+C to quit)
```

图 1: 打开 Tensorboard 命令

5 其他

5.1 模板项目使用说明

- 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)。
- 模板项目基于 tensorflow-1.15.0 编写
- 代码文件说明:
 - input_data.py: 读取数据集文件, 用于加载 mnist 数据集
 - 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

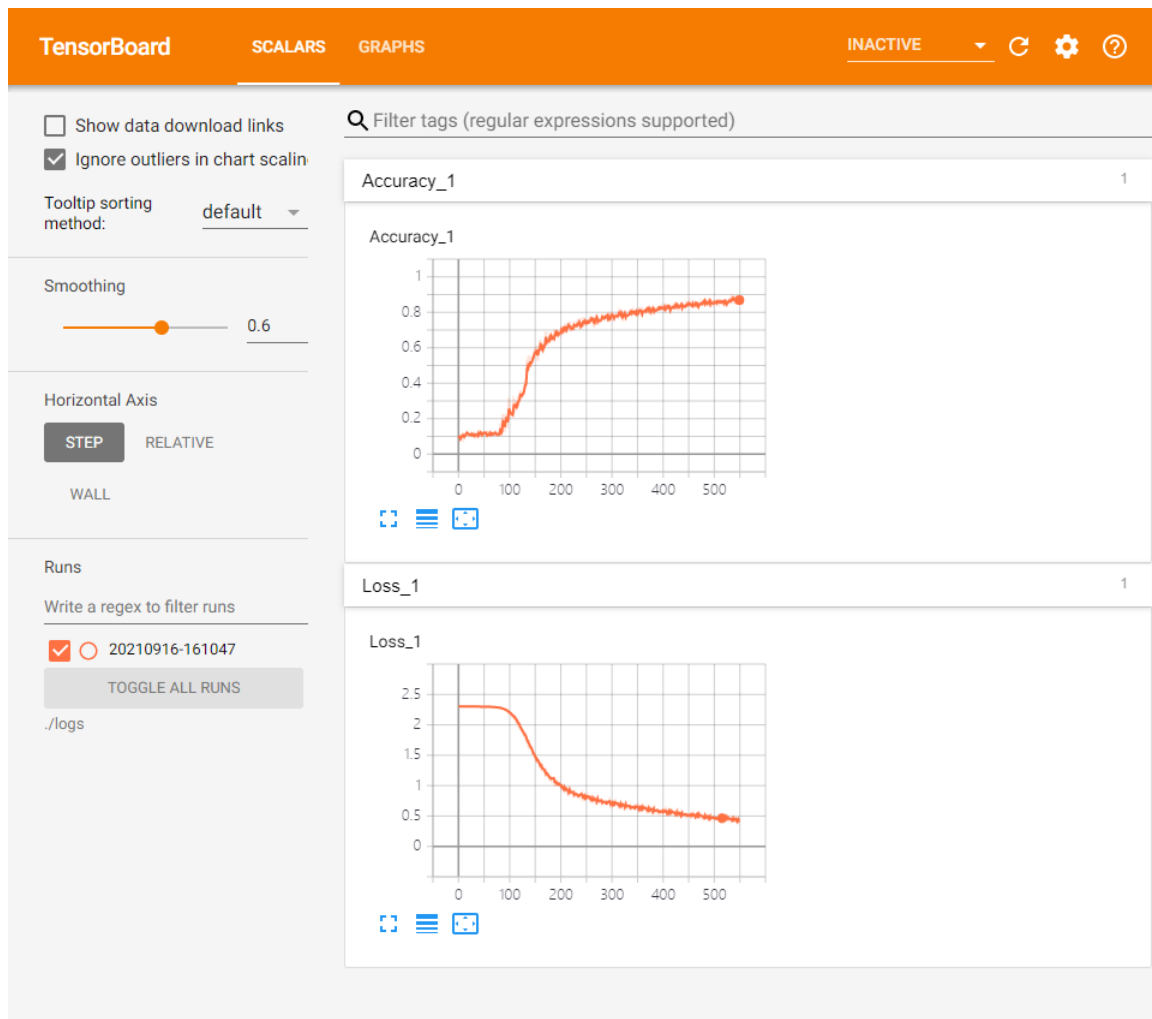


图 2: Tensorboard 页面

5.2 Tensorboard 数据可视化

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²：

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：

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

如图 2所示即是打开后的示例，能够清晰地观察训练过程 loss 以及 acc 变化曲线：

²https://www.tensorflow.org/tensorboard/get_started?hl=zh-cn