Task

1. Please develop a simple neural network training pipeline (with frameworks such as Tensorflow, Keras, Pytorch or others, with Native Tensorflow preferred) for MNIST handwritten digits classification task (you can find the background and requirements below)

2. Please also implement an inference server with trained (frozen) model above, accepting raw image as input and predicted number with probability as output

Background

MNIST is one of basic computer vision datasets, which consists of images of handwritten digits as below:

It also includes labels for each image, telling us which digit it is. For example, the labels for the above images are 5, 0, 4, and 1.

Requirements

For MNIST datasets, you can use one of below resources:

- Download directly from original host website

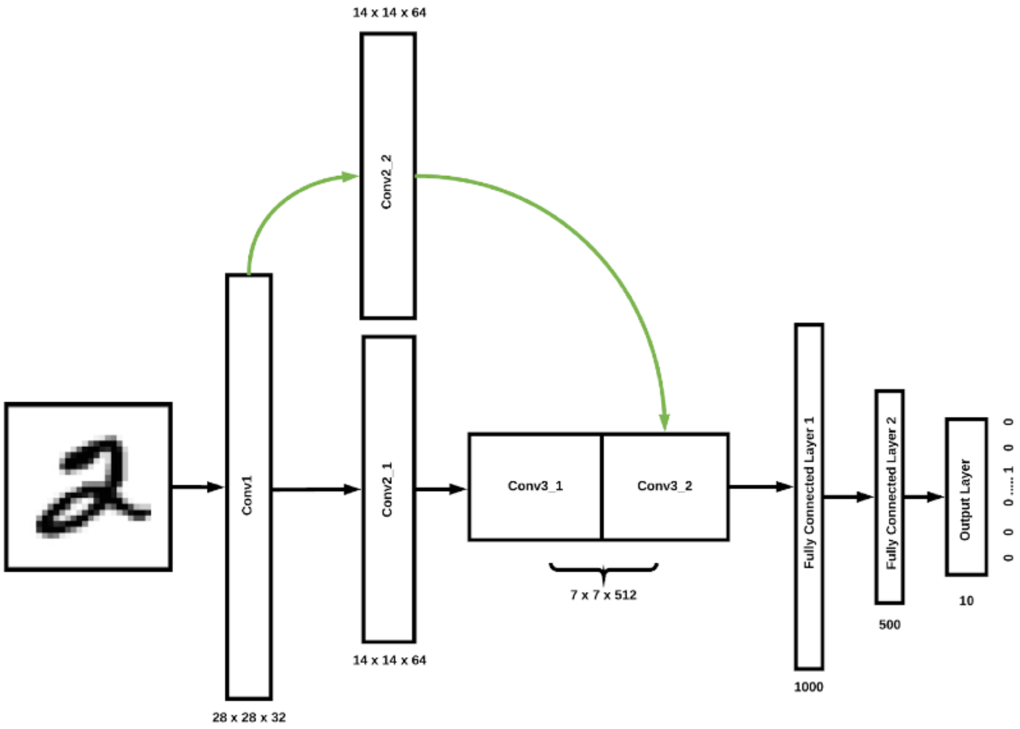
- Tensorflow API

- Tensorflow Keras API

- Pytorch MNIST API

- Others

The architecture of the neural network you shall implement are as follows:



Some explanations for this neural network:

- It contains three convolutional neural network layers (shown above as conv1, conv2_1, conv2_2, conv3_1 and conv3_2), two fully connected layers (named as fc1and fc2) and one output layer

(named as output)

- All non-linear activations shall be implemented using Rectifier Linear Unit (ReLu) and for downsampling, max-pooling will be used

- For each layer, the shape can be found above

- Note that conv3 are combined layer from conv3_1 and conv3_2

- You can define the default value for hyper-parameters that are not listed above

Expectations

1. **Production quality**

    1. For training task, we shall be able to use it do **large scale experiments** tuning hyper-parameters on diverse datasets **easily**

    2. For inferencing task (if you implemented), we shall directly use it for production with downtime as low as possible and speed as fast as possible

    3. Tests covered will be bonus

2. **Documentations**

1. As an Engineer from other team without depth background, by reading the documentation, I shall be able to understand how to do a black box experiments/tuning hyper-parameters on the training task and how to use the server

2. It shall be concise