ELEC5307 Project 1

Weichen Zhang
(wzha8158@uni.sydney.edu.au)
Dongzhan Zhou
(dzho8854@uni.sydney.edu.au)

Project 1 Overview

Part 1: Go through the basic steps

- split the dataset
- Draw the loss to assist

Part 2: Select the best hyper-parameters

- learning rate
- batch size
- epoch number

Part 3: Explore more options

- Build a new baseline: 3conv+3pool+3fc+1out, using anything you believe necessary
- Control variables and explore the influence of three components (related to the last three digits of your SID)
- Combine in one .py file

Submission

1. Modified notebook file project1.ipynb

2. Python file project1.py (for part3 only)

3. Trained models: baseline.pth and modified.pth

Hint 1

- **Split dataset into training set and validation set.**

Training set: Contains samples to train the model.

Validation set: Usually split from original dataset. Used to evaluate the trained model. Avoid overfitting

Test Set: Usually not available (in competition).

Hint 1

Sampler: one attribute of DataLoader.

```
valloader = torch.utils.data.DataLoader(trainset, batch_size=4,
                                        sampler=val_sampler)
```

- The samplers include different ways of sampling data from the original dataset.
- Use SubsetRandomSampler to generate subset, the input should be the indices of images. The first several data are used for training, and then followed by validation.

Hint 1

SubsetRandomSampler

- You need to shuffle the dataset so that for each trial the training/val samples are not the same.

```
train_val_size = 50000 # this is the original size
train_size = 45000 # the number can be changed
val_size = 5000 # the number can be changed
all_index = np.random.permutation(train_val_size) # do shuffle
train_index = all_index[:train_index].tolist()
val_index = all_index[train_size:train_size+val_size].tolist()
val_sampler = torch.utils.data.SubsetRandomSampler(val_index)
```

- This can also used for fast training (change sizes for training and validation)

Hint 2

- **Store the loss for training and validation.**

– The type of the loss computed by your criterion is Tensor, you need change it to number by:

• computed_loss.item()

– For each iteration, the training loss is computed. However, the training loss used in curves should be the average loss in one epoch (or several iterations, considering the small epoch number).

– The training loss and validation loss should be stored in two lists. Draw them by using matplotlib.

Hint 3

- **Draw the loss curve:**

1. Remember to use %matplotlib inline in .ipynb, otherwise, you cannot see the output.

2. Draw multiple lines in one graph with different colors for each curve.

```
plt.plot(epochs, train_loss, color='red', linestyle='--')
plt.plot(epochs, val_loss, color='blue', linestyle='-.')
```

3. Add labels

```
plt.ylabel('loss') # y-axis: loss values
plt.xlabel('epoch') # x-axis: epoch numbers
```

Hint 4

- **Hyper-Parameter Selection**

1. Use the idea of variable control. Change one hyper-parameter and fix other hyper-parameters unchanged

2. Print/Draw the loss of each changes.

3. The results might fluctuate for using the same hyper-parameter. Try to find the optimal one.

Project 2

Task: Classify real world images collected by your own hands.

- Collect data with camera, or mobile phone. Zip, Upload them and submit the link before the next Lab (11/10).
- Choose One Fruit category and One Office stationery category from the Google Drive Link for each group. 10+ images/category are required.
- Try to build some hard examples, e.g. parts, occlusions.
- No Blur images. No Photoshop. No Multi-label. No downloaded images.

What about these examples?

If you are not sure…

Check ImageNet:

- http://www.image-net.org/

Check the paper of ImageNet and/or Pascal VOC:

- The PASCAL Visual Object Classes (VOC) Challenge, IJCV 2010
- ImageNet: A Large-Scale Hierarchical Image Database, CVPR 2009

Project 2 Link

Please upload the compressed image folder (.zip) to Google Drive and put the link of the file in the Google Doc below:

https://docs.google.com/spreadsheets/d/1yLcrQAJjOZKhnaHPcFyTre-5ld8RTm8gNnn_1M-07p4/edit?usp=sharing