

MSBD6000B Project 2
Name: Chan Ngae Chau
Student ID: 20411891

Data Analysis

This is a dataset containing flower images. It has the categories of daisy, dandelion, roses, sunflowers, tulips. There is a total of 3670 images. The training set contains 2569 images (70% of data), the validation set contains 550 images (15% of data), the testing set contains 551 images (15% of data). The size of each images is different. This is a classification problem.

Model

ResNet on Tensorflow framework is used. The model (Appendix #1) is trained from scratch with the training set.

Data Preprocessing

The images data are converted into a folder structure of:

```
data_dir/daisy/aaa.jpg
data_dir/daisy/bbb.jpg
...
data_dir/sunflowers/xxx.jpg
data_dir/sunflowers/yyy.jpg
...
```

Then run the build_image_data.py (Appendix #2) to convert everything into TFRecord format. The TFRecord format allows easy loading of the images into Tensorflow. TFRecord for Training set and validation set are generated. A standalone testing set is also generated which is being fed into the model and get predictions

Training

Due to hardware limitation, some parameters are modified to suit the constraints. The original parameter for image size is 224 and is changed to 196. The number of classes is 6 (there are 5 classes of flowers and zero is being the background class.) The images are resized to standard size of 196 x 196, normalized and are shuffled on each training step. Training are done based on ResNet 34 architecture. (Appendix #3) (ResNet with any higher levels is causing crash on my hardware available.)

Validation

After 8100 steps, the validation accuracy is 87.45%.

Screen capture:

```
tensorflow:Saving dict for global step 8100: accuracy = 0.874545, global_step = 8100, loss = 1.02811
```

Prediction

The TFRecord of testing set is loaded into the model and a list of predictions are generated. The predictions are stored in the file Project2_20411891.csv.

Source code:

- 1) Deeplearning-assignment2.ipynb: for image pre-processing
- 2) Build-image-data.py: convert to TFRecord
- 3) Imagenet_main_project2.py: for training
- 4) Imagenet_predict.py: for predictions

Appendix:

#1 ResNet model: <https://github.com/tensorflow/models/tree/master/official/resnet>

#2 Convert to TFRecord:

https://github.com/tensorflow/models/blob/master/research/inception/inception/data/build_image_data.py

#3 Training:

https://github.com/tensorflow/models/blob/master/official/resnet/imagenet_main.py