Retraining Inception V3 Model Commands with TensorFlow for Poets 2

To run these commands we first had to install TensorFlow version 1.7 and then clone the Git repository of TensorFlow for Poets:

```
pip install --upgrade "tensorflow==1.7.*"
git clone https://github.com/googlecodelabs/tensorflow-for-poets-2
```

Then, we cd into the repository's tensorflow files folder, created a folder for our lung ct scans, and then create two folders inside the aforementioned folder: one which will contain images positive for lung cancer and one which will contain images negative for lung cancer.

```
cd tensorflow-for-poets-2/tf_files/
mkdir lung_photos
cd lung_photos
mkdir positive
mkdir negative
```

Then, the scans were transferred to their respective folders. These images were converted from their original DICOM format to JPEG format and then resized to be 299x299 pixels before being placed in the virtual environment.

The MobileNet, which is a small efficient convolutional neural network, is re-trained with the following configurations: **input image resolution of 224px** and **relative size of the model as a fraction of the largest MobileNet of 1.0**

```
IMAGE_SIZE=224
ARCHITECTURE="mobilenet_1.0_${IMAGE_SIZE}"
```

Tensorboard is turned on prior to training to monitor training progress

```
tensorboard --logdir tf_files/training_summaries &
```

While in the repository, retraining is done with the following command:

```
python -m scripts.retrain   --bottleneck_dir=tf_files/bottlenecks
--how_many_training_steps=500   --model_dir=tf_files/models/
--summaries_dir=tf_files/training_summaries/"${ARCHITECTURE}"
--output_graph=tf_files/retrained_graph.pb
--output_labels=tf_files/retrained_labels.txt   --architecture="${ARCHITECTURE}"
--image_dir=tf_files/lung_photos
```

Note that to create multiple models, one simply changes the names of the output graph and output labels files (in our case, a number was added at the end of these).It is also worth noting that hyperparameters of the command mentioned can be changed to improve the results (classifications) obtained from the model. More of this will be discussed later.

Now that we have a model trained, we can ask our model to classify our images as being positive or negative to lung cancer. To do this we created a directory called TESTING and added 10 images of lungs with cancerous tumors and 10 images of healthy lungs to it.

```
cd
mkdir TESTING
```

Testing is performed using the following command while in the tensorflow for poets git repo.

```
cd tensorflow-for-poets-2/
python -m scripts.label_image     --graph=tf_files/retrained_graph.pb
--image=/home/cc/TESTING/test_pos1.jpg
```

This command then classifies the image using the model corresponding to the graph provided and displays its results to the user. Our experimental results can be found in our GitHub repository.

Changing Hyperparameters

Aside from adding more images for retraining a model, changing hyperparameters in the training command can also improve the results obtained, thus improving the quality of classifications.

The hyperparameters we chose to change for our models are defined as follows:

- --how_many_training_steps: How many training steps to run before ending training.
- --learning_rate: How large a learning rate to use when training.
- --testing_percentage: What percentage of images to use as a test set.
- --validation_percentage: What percentage of images to use as a validation set.

The changes to hyperparameters performed can be observed in the experimental results document in our GitHub repository.