Raymond Muller

Group 6 Project 3 Part 3: Batch Notebook

In [1]:
```
import sys
!{sys.executable} -m pip install tensorflow==2.0.0
!{sys.executable} -m pip install keras
!{sys.executable} -m pip install --upgrade numpy --user --no-warn-script-location
!{sys.executable} -m pip install import_ipynb
```

```
Requirement already satisfied: tensorflow==2.0.0 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (2.0.0)
Requirement already satisfied: absl-py>=0.7.0 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (from tensorfl
ow==2.0.0) (0.8.1)
Requirement already satisfied: keras-preprocessing>=1.0.5 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (f
rom tensorflow==2.0.0) (1.1.0)
Requirement already satisfied: grpcio>=1.8.6 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (from tensorflo
w==2.0.0) (1.24.3)
Requirement already satisfied: astor>=0.6.0 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (from tensorflow
==2.0.0) (0.8.0)
Requirement already satisfied: protobuf>=3.6.1 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (from tensorf
low==2.0.0) (3.10.0)
Requirement already satisfied: tensorflow-estimator<2.1.0,>=2.0.0 in /Users/RayM/opt/anaconda3/lib/python3.7/site-pac
kages (from tensorflow==2.0.0) (2.0.1)
Requirement already satisfied: termcolor>=1.1.0 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (from tensor
flow==2.0.0) (1.1.0)
Requirement already satisfied: wheel>=0.26 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (from tensorflow=
=2.0.0) (0.33.6)
Requirement already satisfied: google-pasta>=0.1.6 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (from ten
sorflow==2.0.0) (0.1.7)
Requirement already satisfied: tensorboard<2.1.0,>=2.0.0 in /Users/RayM/opt/anaconda3/lib/python3.7/site-packages (fr
om tensorflow==2.0.0) (2.0.1)
```

Load Dependencies

```
import tensorflow as tf
import os
import cv2
import numpy as np
import matplotlib.pyplot as plt
import tqdm
import datetime
import json
from sklearn.preprocessing import  LabelBinarizer
#import pickle
#import import_ipynb
#import LSTM_XX_Training
#import FeatureExtrationNotebook
```

Import Statements

```
# code to load model saved in SavedModel format
BASE_DIRECTORY = '/Users/RayM/Documents/School/CompVision/Project3'
#Load from h5
model_file = os.path.join(BASE_DIRECTORY, 'TF1-14_LSTM_Model_V1.h5')
model = tf.keras.models.load_model(model_file)

#SAVED_MODEL_DIR = os.path.join(BASE_DIRECTORY, 'SavedModelDir/LSTM/V1/')
#model = tf.keras.experimental.load_from_saved_model(SAVED_MODEL_DIR)
```

Load the model for use in prediction

```
# show summary of LSTM model
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
masking (Masking)            multiple                  0
_________________________________________________________________
lstm (LSTM)                  multiple                  3672064
_________________________________________________________________
dense (Dense)                multiple                  131328
_________________________________________________________________
dropout (Dropout)            multiple                  0
_________________________________________________________________
dense_1 (Dense)              multiple                  9509
=================================================================
Total params: 3,812,901
Trainable params: 3,812,901
Non-trainable params: 0
_________________________________________________________________
```

The Summary of the Model

```
# load the data for all video files generated in a list
gen_list_path = '/Users/RayM/Documents/School/CompVision/Project3/new_paths.txt'
with open(gen_list_path) as f:
    file_list = [row.strip() for row in list(f)]

#data = np.load('/Users/RayM/Documents/School/CompVision/Project3/Training_Set/CS663-Swat-Set-2/CellLeader/R_CellLeade
#print(data.shape)
```

Load in the paths of all of the new test data npys. The paths are generated by the following bit of code, separately:

```
import os
import glob

BASE_PATH = '/Users/RayM/Documents/School/CompVision/Project3/new_test_data'
VID_PATH = os.path.join(BASE_PATH, '**', '**.npy')

filelist= glob.glob(VID_PATH)


# Write to file
with open('new_paths.txt', 'w') as filehandle:
    for listitem in filelist:
        filehandle.write('%s\n' % listitem)


```

```
# perform prediction using video data on LSTM model
for path in file_list:
    print(path)
    data = np.load(path)
    print(data.shape)
    data = np.expand_dims(data, axis=0)
    print(data.shape)
    results = model.predict(data,batch_size=1)
    #results = model.predict(data)
    print(results)
```

```
/Users/RayM/Documents/School/CompVision/Project3/new_test_data/Eight/R_81.npy
(18, 1280)
(1, 18, 1280)
[[1.58509348e-08 4.19904979e-11 3.49969923e-07 4.61558921e-06
  7.64886181e-08 1.67986034e-08 1.98167847e-08 9.56235890e-09
  1.96227070e-08 1.15395684e-04 2.89580306e-08 1.07175486e-08
  1.61429838e-08 1.39969396e-08 1.81292172e-08 2.15401812e-07
  1.01665822e-08 2.83136679e-08 3.93175839e-08 3.48644242e-08
  2.00128607e-08 3.95053590e-08 2.93351277e-05 2.68962162e-06
  1.14115216e-01 8.84198427e-01 2.21902763e-08 1.22532322e-08
  1.64570018e-13 2.98853524e-06 1.42753323e-10 4.71205759e-08
  1.53016881e-03 1.24319510e-09 7.66583312e-08 1.24942856e-08
  3.67130824e-08]]
/Users/RayM/Documents/School/CompVision/Project3/new_test_data/Eight/R_82.npy
(24, 1280)
(1, 24, 1280)
[[2.9553306e-15 1.1786611e-15 3.3564971e-14 3.7789364e-12 6.2087655e-15
  2.7855691e-15 3.4260994e-15 1.1573153e-15 3.3547483e-15 2.9294546e-14
```

This code loops through all of the paths and runs the prediction on them, outputting the results.