DESCRIPTION

Do you have your father’s nose?

Blood relatives often share facial features. Now researchers at Northeastern University want to improve their algorithm for facial image classification to bridge the gap between research and other familial markers like DNA results. That will be challenge in this new Kaggle competition.

An automatic kinship classifier has been in the works at Northeastern since 2010. Yet this technology remains largely unseen in practice for a couple of reasons:

1. Existing image databases for kinship recognition tasks aren't large enough to capture and reflect the true data distributions of the families of the world.
2. Many hidden factors affect familial facial relationships, so a more discriminant model is needed than the computer vision algorithms used most often for higher-level categorizations (e.g. facial recognition or object classification).

In this competition, you’ll help researchers build a more complex model by determining if two people are blood-related based solely on images of their faces. If you think you can get it "on the nose," this competition is for you.

The SMILE Lab at Northeastern focuses on the frontier research of applied machine learning, social media analytics, human-computer interaction, and high-level image and video understanding. Their research is driven by the explosion of diverse multimedia from the Internet, including both personal and publicly-available photos and videos. They start by treating fundamental theory from learning algorithms as the soul of machine intelligence and arm it with visual perception.

EVALUATION

Evaluated on area under the ROC curve between the predicted probability and the observed target. Not all pairs will be scored.

ABOUT DATA

train-faces.zip - the training set is divided in Families (F0123), then individuals (MIDx). Images in the same MIDx folder belong to the same person. Images in the same F0123 folder belong to the same family. train.csv - training labels. Remember, not every individual in a family shares a kinship relationship. For example, a mother and father are kin to their children, but not to each other.

test-faces.zip - the test set contains face images of unknown individuals

In [1]:

```
# importing the tensorflow package
import tensorflow as tf
```

In [2]:

```
tf.__version__
```

Out[2]:

```
'2.0.0'
```

In [3]:

```
tf.test.is_built_with_cuda()
```

Out[3]:

```
True
```

In [4]:

```
print("Num GPUs Available: ", len(tf.config.experimental.list_physical_devices('GPU')))
```

```
Num GPUs Available:  0
```

GOOGLE CLOUD

In [1]:

```
from collections import defaultdict
from glob import glob
from random import choice, sample
import matplotlib.pyplot as plt
from tqdm import tqdm
import numpy as np
import pandas as pd
import pickle
import gc, psutil, random, base64, h5py, os,shutil
#import cv2
pd.options.mode.chained_assignment = None
pd.options.display.max_columns = 9999
pd.options.display.float_format = '{:20, .2f}'.format
from keras import backend as K

from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Concatenate, Multiply
, Dropout, Subtract, Add
from keras_vggface.vggface import VGGFace
from pathlib import Path

from PIL import Image
from io import BytesIO
from IPython.display import HTML
import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
import plotly.tools as tls

import warnings
warnings.filterwarnings('ignore')
from keras.layers import Conv2D, ZeroPadding2D, Activation, concatenate
from keras.layers.pooling import MaxPooling2D, AveragePooling2D
from keras.layers.core import Lambda
import keras
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau,EarlyStopping
from keras.layers import Conv1D, BatchNormalization,Reshape, LeakyReLU
from keras.preprocessing import image
from keras.optimizers import Adam,SGD,rmsprop
from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Conv2D,MaxPooling1D,G
lobalAveragePooling1D,LSTM
from keras.models import Model
from keras.losses import binary_crossentropy
import tensorflow as tf
from sklearn.metrics import roc_auc_score
from keras import regularizers
%pylab inline
import matplotlib.image as mpimg
```

```
Using TensorFlow backend.
```

```
Populating the interactive namespace from numpy and matplotlib
```

In [6]:

```
!sudo pip install kaggle
```

```
Requirement already satisfied: kaggle in /usr/local/lib/python3.5/dist-packages (1.5.6)
Requirement already satisfied: certifi in /usr/local/lib/python3.5/dist-packages (from kaggle) (2019.11
.28)
Requirement already satisfied: python-dateutil in /usr/local/lib/python3.5/dist-packages (from kaggle)
(2.8.1)
Requirement already satisfied: requests in /usr/local/lib/python3.5/dist-packages (from kaggle) (2.22.0
)
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.5/dist-packages (from kaggle) (1.13.
0)
Requirement already satisfied: urllib3<1.25,>=1.21.1 in /usr/local/lib/python3.5/dist-packages (from ka
ggle) (1.24.2)
```

```
ggle) (1.24.2)
Requirement already satisfied: tqdm in /usr/local/lib/python3.5/dist-packages (from kaggle) (4.40.2)
Requirement already satisfied: python-slugify in /usr/local/lib/python3.5/dist-packages (from kaggle) (
4.0.0)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /usr/local/lib/python3.5/dist-packages (from re
quests->kaggle) (3.0.4)
Requirement already satisfied: idna<2.9,>=2.5 in /usr/local/lib/python3.5/dist-packages (from requests-
>kaggle) (2.8)
Requirement already satisfied: text-unidecode>=1.3 in /usr/local/lib/python3.5/dist-packages (from pyth
on-slugify->kaggle) (1.3)
WARNING: You are using pip version 19.3.1; however, version 20.0.2 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
```

In [7]:

```
!sudo mkdir .kaggle
```

```
mkdir: cannot create directory '.kaggle': File exists
```

In [8]:

```
!sudo mkdir ~/.kaggle
```

```
mkdir: cannot create directory '/root/.kaggle': File exists
```

In [9]:

```
!pwd
```

```
[redacted]
```

In [10]:

```
!ls
```

```
driver_installer.run      SAVE_MODEL

Northeastern_GCP_submit2_1_final_1.ipynb  TENSORBOARD

Northeastern_GCP_submit2_1_test.ipynb   test

recognizing-faces-in-the-wild    train

recognizing-faces-in-the-wild.zip
```

In [11]:

```
import json
token = {[redacted]
with open([redacted]
    json.dump(token, file)
```

In [12]:

```
!cp /home[redacted].kaggle/kaggle.json ~/.kaggle/kaggle.json
```

In [13]:

```
!sudo chmod 600 ~/.kaggle/kaggle.json
```

In [14]:

```
!kaggle datasets list
```

```
ref                                                      title
size  lastUpdated          downloadCount  voteCount  usabilityRating

-------------------------------------------------------  ----------------------------------------------
-----  -------------------  -------------  ---------  ---------------

chrisfilo/to-bee-or-no-to-bee                            To bee or not to bee
3GB  2020-02-04 22:55:31             47          9  0.8125

peijenlin/msl-m-rems-2-edr-v1.0                          MSL-M-REMS-2-EDR-V1.0
11GB  2020-02-05 00:55:52              4          4  0.8125

timoboz/superbowl-history-1967-2020                      Superbowl History 1967 - 2020
2KB  2020-02-03 23:41:14            279         31  1.0

timoboz/google-trends-data                               Google Trends Data
1MB  2020-02-04 17:31:10            244         20  0.85294116

fmejia21/demographics-of-academy-awards-oscars-winners  Demographics of Academy Awards (Oscars) Winners
20KB  2020-02-04 17:38:26            153         12  0.88235295

timoboz/big-list-of-naughty-strings                      Big List of Naughty Strings
66KB  2020-02-04 18:11:53             14         10  0.8125

fmejia21/2020-democratic-primary-endorsements            2020 Democratic Primary Endorsements
22KB  2020-02-04 18:05:34             43         10  0.88235295

prondeau/diceimages                                      Images of Board Game Dice
48MB  2020-02-04 18:24:40             14         10  0.8125

timoboz/python-data-science-handbook                     Python Data Science Handbook
15MB  2020-02-04 18:27:14             82         12  0.8235294

timoboz/clevr-dataset                                    CLEVR Dataset
18GB  2020-02-04 18:36:16              4          9  0.9375

peijenlin/surviving-mars-maps                            Surviving Mars Maps
4MB  2020-01-27 20:30:13             21         12  1.0

chrisfilo/urbansound8k                                   UrbanSound8K
6GB  2020-02-04 18:37:24              7         10  0.7647059

prondeau/350000-jeopardy-questions                       350,000+ Jeopardy Questions
19MB  2020-02-04 18:53:30             50         11  0.9411765

prondeau/superbowlads                                    Super Bowl Ads
36KB  2020-02-04 18:04:07             99         11  1.0

timoboz/can-james-bond-be-black                          Can James Bond Be Black?
345B  2020-02-04 19:08:57             18          9  1.0

devvret/farm-ads-binary-classification                   Farm Ads Binary Classification
5MB  2020-02-04 19:12:44             14         10  0.8125

fmejia21/nba-all-star-game-20002016                      NBA All Star Game 2000-2016
35KB  2020-02-04 19:21:13            102         13  0.88235295

timoboz/wikidata-jsons                                   Wikidata jsons
858MB  2020-02-04 19:19:08              4          8  0.8125

chrisfilo/firesense                                      FIRESENSE
784MB  2020-02-04 19:20:54             16         10  0.8125

prondeau/the-car-connection-picture-dataset              60,000+ Images of Cars
682MB  2020-02-04 19:19:07             24         12  0.9375
```

In [0]:

```
!kaggle competitions download -c recognizing-faces-in-the-wild -p [redacted]
```

```
Downloading recognizing-faces-in-the-wild.zip to [redacted]
 98%|[redacted] | 373M/381M [00:02<00:00, 142MB/s]
 98%|████████████████████████████████████████▉ | 373M/381M [00:02<00:00, 142MB/s]
100%|█████████████████████████████████████████| 381M/381M [00:02<00:00, 151MB/s]
```

In [0]:

```
!ls
```

```
Northeastern_GCP.ipynb recognizing-faces-in-the-wild.zip
```

In [0]:

```
!unzip recognizing-faces-in-the-wild.zip -d ./recognizing-faces-in-the-wild
```

```
Archive:  recognizing-faces-in-the-wild.zip
  inflating: ./recognizing-faces-in-the-wild/README.md
  inflating: ./recognizing-faces-in-the-wild/sample_submission.csv
  inflating: ./recognizing-faces-in-the-wild/test-faces.zip
  inflating: ./recognizing-faces-in-the-wild/test-private-faces.zip
  inflating: ./recognizing-faces-in-the-wild/test-private-labels.zip
  inflating: ./recognizing-faces-in-the-wild/test-private-lists.zip
  inflating: ./recognizing-faces-in-the-wild/test-public-faces.zip
  inflating: ./recognizing-faces-in-the-wild/test-public-lists.zip
  inflating: ./recognizing-faces-in-the-wild/test.zip
  inflating: ./recognizing-faces-in-the-wild/train-faces.zip
  inflating: ./recognizing-faces-in-the-wild/train.zip
  inflating: ./recognizing-faces-in-the-wild/train_relationships.csv
```

In [0]:

```
!ls
```

```
Northeastern_GCP.ipynb        recognizing-faces-in-the-wild.zip

recognizing-faces-in-the-wild
```

In [0]:

```
!sudo mkdir test
!sudo chmod 777 test
```

In [0]:

```
!sudo mkdir train
!sudo chmod 777 train
```

In [0]:

```
!unzip ./recognizing-faces-in-the-wild/test.zip -d ./test
```

In [0]:

```
!unzip ./recognizing-faces-in-the-wild/train.zip -d ./train
```

In [0]:

```
!ls
```

```
Northeastern_GCP.ipynb        recognizing-faces-in-the-wild.zip  train

recognizing-faces-in-the-wild  test
```

In [0]:

GOOGLE COLAB

In [6]:

```
!ls
```

```
sample_data
```

In [7]:

```
!pwd
```

```
/content
```

In [8]:

```
# we used VGGFace file from the below github which  has some models works better for Face Identificatio
n
!git clone https://github.com/rcmalli/keras-vggface.git
```

```
Cloning into 'keras-vggface'...
remote: Enumerating objects: 26, done.
remote: Counting objects: 100% (26/26), done.
remote: Compressing objects: 100% (24/24), done.
remote: Total 277 (delta 10), reused 6 (delta 2), pack-reused 251
Receiving objects: 100% (277/277), 6.03 MiB | 9.04 MiB/s, done.
Resolving deltas: 100% (129/129), done.
```

In [0]:

```
import os,shutil
for file_ in os.listdir('keras-vggface'):
    if '.*' not in file_:
        try:
            shutil.move('keras-vggface/'+file_,'.')
        except:
            print(9)
```

In [10]:

```
from collections import defaultdict
from glob import glob
from random import choice, sample
import matplotlib.pyplot as plt
from tqdm import tqdm
import numpy as np
import pandas as pd
import pickle
import gc, psutil, cv2, random, base64, h5py, os,shutil
pd.options.mode.chained_assignment = None
pd.options.display.max_columns = 9999
pd.options.display.float_format = '{:20, .2f}'.format
from keras import backend as K

from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Concatenate, Multiply
, Dropout, Subtract, Add
from keras_vggface.vggface import VGGFace
from pathlib import Path

from PIL import Image
from io import BytesIO
from IPython.display import HTML
import plotly.offline as py
py.init_notebook_mode(connected=True)
```

```
import plotly.graph_objs as go
import plotly.tools as tls

import warnings
warnings.filterwarnings('ignore')
from keras.layers import Conv2D, ZeroPadding2D, Activation, concatenate
from keras.layers.pooling import MaxPooling2D, AveragePooling2D
from keras.layers.core import Lambda
import keras
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau,EarlyStopping
from keras.layers import Conv1D, BatchNormalization,Reshape, LeakyReLU
from keras.preprocessing import image
from keras.optimizers import Adam,SGD,rmsprop
from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Conv2D,MaxPooling1D,G
lobalAveragePooling1D,LSTM
from keras.models import Model
from keras.losses import binary_crossentropy
import tensorflow as tf
from sklearn.metrics import roc_auc_score
from keras import regularizers
%pylab inline
import matplotlib.image as mpimg
```

```
Using TensorFlow backend.
```

```
Populating the interactive namespace from numpy and matplotlib
```

In [0]:

In [0]:

In [11]:

```
import shutil
from google.colab import drive
drive.mount('/content/drive',force_remount=True)
# path='drive/My Drive/AAIC/'
```

[redacted]

[redacted]

In [0]:

```
# UPLOADING model1.h5 from local PC to colab

#from google.colab import files
#uploaded = files.upload()

#import io
#df2 = pd.read_csv(io.BytesIO(uploaded['Filename.csv']))
# Dataset is now stored in a Pandas Dataframe
```

In [12]:

```
!ls
```

```
drive   keras_vggface  LICENSE.txt  README.md setup.py  tools
image   keras-vggface  MANIFEST.in  sample_data test.py   visualize
```

In [13]:

```
!pwd
```

```
/content
```

In [0]:

```

```

In [14]:

```
!pip install kaggle
```

```
Requirement already satisfied: kaggle in /usr/local/lib/python3.6/dist-packages (1.5.6)
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.6/dist-packages (from kaggle) (1.12.
0)
Requirement already satisfied: tqdm in /usr/local/lib/python3.6/dist-packages (from kaggle) (4.28.1)
Requirement already satisfied: certifi in /usr/local/lib/python3.6/dist-packages (from kaggle) (2019.11
.28)
Requirement already satisfied: requests in /usr/local/lib/python3.6/dist-packages (from kaggle) (2.21.0
)
Requirement already satisfied: python-dateutil in /usr/local/lib/python3.6/dist-packages (from kaggle)
(2.6.1)
Requirement already satisfied: python-slugify in /usr/local/lib/python3.6/dist-packages (from kaggle) (
4.0.0)
Requirement already satisfied: urllib3<1.25,>=1.21.1 in /usr/local/lib/python3.6/dist-packages (from ka
ggle) (1.24.3)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /usr/local/lib/python3.6/dist-packages (from re
quests->kaggle) (3.0.4)
Requirement already satisfied: idna<2.9,>=2.5 in /usr/local/lib/python3.6/dist-packages (from requests-
>kaggle) (2.8)
Requirement already satisfied: text-unidecode>=1.3 in /usr/local/lib/python3.6/dist-packages (from pyth
on-slugify->kaggle) (1.3)
```

In [0]:

```
!mkdir .kaggle
```

In [0]:

```
!mkdir ~/.kaggle
```

In [0]:

```

```

In [0]:

```
import json
token = {[redacted]
with open[redacted], 'w') as file:
  json.dump(token, file)
```

In [0]:

```
!cp /content/.kaggle/kaggle.json ~/.kaggle/kaggle.json
```

In [19]:

```
!kaggle config set -n path -v{/content}
```

```
Warning: Your Kaggle API key is readable by other users on this system! To fix this, you can run 'chmod
600 /root/.kaggle/kaggle.json'
- path is now set to: {/content}
```

In [0]:

```
!chmod 600 /root/.kaggle/kaggle.json
```

In [21]:

```
!kaggle datasets list
```

```
Warning: Looks like you're using an outdated API Version, please consider updating (server 1.5.6 / clie
nt 1.5.4)
ref                                                     title
size  lastUpdated          downloadCount
------------------------------------------------------  -----------------------------------------------
-----  -------------------  -------------
timoboz/google-trends-data                              Google Trends Data
1MB  2020-02-04 17:31:10            204
fmejia21/demographics-of-academy-awards-oscars-winners  Demographics of Academy Awards (Oscars) Winners
20KB  2020-02-04 17:38:26            135
timoboz/big-list-of-naughty-strings                     Big List of Naughty Strings
66KB  2020-02-04 18:11:53             13
fmejia21/2020-democratic-primary-endorsements           2020 Democratic Primary Endorsements
22KB  2020-02-04 18:05:34             38
prondeau/diceimages                                     Images of Board Game Dice
48MB  2020-02-04 18:24:40              8
timoboz/python-data-science-handbook                    Python Data Science Handbook
15MB  2020-02-04 18:27:14             67
timoboz/clevr-dataset                                   CLEVR Dataset
18GB  2020-02-04 18:36:16              4
peijenlin/surviving-mars-maps                           Surviving Mars Maps
4MB  2020-01-27 20:30:13             19
chrisfilo/urbansound8k                                  UrbanSound8K
6GB  2020-02-04 18:37:24              7
prondeau/350000-jeopardy-questions                      350,000+ Jeopardy Questions
19MB  2020-02-04 18:53:30             45
prondeau/superbowlads                                   Super Bowl Ads
36KB  2020-02-04 18:04:07             90
timoboz/can-james-bond-be-black                         Can James Bond Be Black?
345B  2020-02-04 19:08:57              9
devvret/farm-ads-binary-classification                  Farm Ads Binary Classification
5MB  2020-02-04 19:12:44             11
fmejia21/nba-all-star-game-20002016                     NBA All Star Game 2000-2016
35KB  2020-02-04 19:21:13             78
timoboz/wikidata-jsons                                  Wikidata jsons
858MB  2020-02-04 19:19:08              4
chrisfilo/firesense                                     FIRESENSE
784MB  2020-02-04 19:20:54             13
prondeau/the-car-connection-picture-dataset             60,000+ Images of Cars
682MB  2020-02-04 19:19:07             19
fmejia21/trump-impeachment-polls                        Trump Impeachment Polls
110KB  2020-02-04 19:29:05             48
chrisfilo/demand                                        DEMAND
7GB  2020-02-04 19:41:36              4
timoboz/data-science-cheat-sheets                       Data Science Cheat Sheets
596MB  2020-02-04 19:42:27            150
```

In [22]:

```
!kaggle datasets list -s Northeastern
```

```
Warning: Looks like you're using an outdated API Version, please consider updating (server 1.5.6 / clie
nt 1.5.4)
ref                                   title                              size  lastUpdated          down
loadCount
------------------------------------  -------------------------------  -----  -------------------  ----
---------
marcodena/mobile-phone-activity       Mobile phone activity in a city  242MB  2019-11-14 06:11:32
8084
theriley106/college-common-data-sets  College Common Data Sets         210MB  2018-01-20 22:14:19
```

```
689
theriley106/university-statistics     University Statistics            33KB  2018-01-21 23:03:55
1337
```

In [23]:

```
#!kaggle competitions download -c recognizing-faces-in-the-wild -p /content
!kaggle competitions download -c recognizing-faces-in-the-wild -p /content
```

```
Warning: Looks like you're using an outdated API Version, please consider updating (server 1.5.6 / clie
nt 1.5.4)
Downloading test-private-faces.zip to /content
 71% 21.0M/29.7M [00:00<00:00, 20.8MB/s]
100% 29.7M/29.7M [00:00<00:00, 60.6MB/s]
README.md: Skipping, found more recently modified local copy (use --force to force download)
Downloading test-public-faces.zip to /content
 89% 116M/130M [00:01<00:00, 63.5MB/s]
100% 130M/130M [00:01<00:00, 81.6MB/s]
Downloading test-public-lists.zip to /content
  0% 0.00/33.7k [00:00<?, ?B/s]
100% 33.7k/33.7k [00:00<00:00, 69.5MB/s]
Downloading test.zip to /content
 56% 19.0M/34.1M [00:00<00:00, 18.7MB/s]
100% 34.1M/34.1M [00:00<00:00, 58.0MB/s]
Downloading test-private-lists.zip to /content
  0% 0.00/354k [00:00<?, ?B/s]
100% 354k/354k [00:00<00:00, 111MB/s]
Downloading test-faces.zip to /content
 33% 9.00M/27.1M [00:00<00:01, 17.9MB/s]
100% 27.1M/27.1M [00:00<00:00, 46.3MB/s]
Downloading test-private-labels.zip to /content
  0% 0.00/8.67k [00:00<?, ?B/s]
100% 8.67k/8.67k [00:00<00:00, 8.88MB/s]
Downloading train_relationships.csv to /content
  0% 0.00/77.6k [00:00<?, ?B/s]
100% 77.6k/77.6k [00:00<00:00, 68.4MB/s]
Downloading train.zip to /content
 95% 65.0M/68.6M [00:00<00:00, 45.4MB/s]
100% 68.6M/68.6M [00:00<00:00, 78.5MB/s]
Downloading sample_submission.csv to /content
  0% 0.00/156k [00:00<?, ?B/s]
100% 156k/156k [00:00<00:00, 133MB/s]
Downloading train-faces.zip to /content
 83% 92.0M/111M [00:01<00:00, 38.3MB/s]
100% 111M/111M [00:01<00:00, 61.5MB/s]
```

In [24]:

```
!ls
```

```
drive         sample_submission.csv test.py
image         setup.py   test.zip
keras_vggface  test-faces.zip  tools
keras-vggface  test-private-faces.zip train-faces.zip
LICENSE.txt    test-private-labels.zip train_relationships.csv
MANIFEST.in    test-private-lists.zip train.zip
README.md      test-public-faces.zip visualize
sample_data    test-public-lists.zip
```

In [0]:

```
!mkdir test
!chmod 777 test
```

In [0]:

```
!mkdir train
!chmod 777 train
```

In [27]:

```
%%time
%%capture
!unzip test.zip -d test
```

```
CPU times: user 553 ms, sys: 53.2 ms, total: 606 ms
Wall time: 4.33 s
```

In [28]:

```
%%time
%%capture
!unzip train.zip -d train
```

```
CPU times: user 1.41 s, sys: 184 ms, total: 1.59 s
Wall time: 6.56 s
```

In [29]:

```
!ls
```

```
drive          setup.py   test.zip
image          test    tools
keras_vggface        test-faces.zip  train
keras-vggface        test-private-faces.zip train-faces.zip
LICENSE.txt        test-private-labels.zip train_relationships.csv
MANIFEST.in        test-private-lists.zip train.zip
README.md        test-public-faces.zip visualize
sample_data        test-public-lists.zip
sample_submission.csv  test.py
```

In [30]:

```
print("The no of train images:",len(os.listdir('train')))
# here it contains both the folders and pics
```

```
The no of train images: 470
```

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [4]:

```
!pip install git+https://github.com/rcmalli/keras-vggface.git
```

```
Collecting git+https://github.com/rcmalli/keras-vggface.git
  Cloning https://github.com/rcmalli/keras-vggface.git to /tmp/pip-req-build-awgjtkb8
  Running command git clone -q https://github.com/rcmalli/keras-vggface.git /tmp/pip-req-build-awgjtkb8
Requirement already satisfied (use --upgrade to upgrade): keras-vggface==0.6 from git+https://github.co
m/rcmalli/keras-vggface.git in /home/deeplearning/anaconda3/lib/python3.7/site-packages
Requirement already satisfied: numpy>=1.9.1 in /home/deeplearning/anaconda3/lib/python3.7/site-packages
(from keras-vggface==0.6) (1.17.2)
Requirement already satisfied: scipy>=0.14 in /home/deeplearning/anaconda3/lib/python3.7/site-packages
(from keras-vggface==0.6) (1.3.1)
Requirement already satisfied: h5py in /home/deeplearning/anaconda3/lib/python3.7/site-packages (from k
eras-vggface==0.6) (2.9.0)
Requirement already satisfied: pillow in /home/deeplearning/anaconda3/lib/python3.7/site-packages (from
keras-vggface==0.6) (6.2.0)
Requirement already satisfied: keras in /home/deeplearning/anaconda3/lib/python3.7/site-packages (from
keras-vggface==0.6) (2.3.1)
Requirement already satisfied: six>=1.9.0 in /home/deeplearning/anaconda3/lib/python3.7/site-packages (
from keras-vggface==0.6) (1.12.0)
Requirement already satisfied: pyyaml in /home/deeplearning/anaconda3/lib/python3.7/site-packages (from
keras-vggface==0.6) (5.1.2)
Requirement already satisfied: keras-preprocessing>=1.0.5 in /home/deeplearning/anaconda3/lib/python3.7
/site-packages (from keras->keras-vggface==0.6) (1.1.0)
Requirement already satisfied: keras-applications>=1.0.6 in /home/deeplearning/anaconda3/lib/python3.7/
site-packages (from keras->keras-vggface==0.6) (1.0.8)
Building wheels for collected packages: keras-vggface
  Building wheel for keras-vggface (setup.py) ... done
  Created wheel for keras-vggface: filename=keras_vggface-0.6-cp37-none-any.whl size=8312 sha256=827030
c2d77efc6ce7c2d83f73db137135a7eca1e6628a005d99f48b2193e2f5
  Stored in directory: /tmp/pip-ephem-wheel-cache-cqpp1b64/wheels/36/07/46/06c25ce8e9cd396dabe151ea1d8a
2bc28dafcb11321c1f3a6d
Successfully built keras-vggface
WARNING: You are using pip version 19.3.1; however, version 20.0.2 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
```

In [5]:

```python
from collections import defaultdict
from glob import glob
from random import choice, sample

#import cv2
import numpy as np
import pandas as pd
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau
from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Concatenate, Multiply
, Dropout, Subtract, Add, Conv2D
from keras.models import Model
from keras.preprocessing import image
from keras.optimizers import Adam
from keras_vggface.utils import preprocess_input
from keras_vggface.vggface import VGGFace

# import regularizer
from keras.regularizers import l1
from keras.regularizers import l2

import h5py
import itertools
```

EDA

In [6]:

```
import matplotlib.pyplot as plt
from PIL import Image
from pathlib import Path
import os

import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
import plotly.tools as tls

pd.options.mode.chained_assignment = None
pd.options.display.max_columns = 9999
pd.options.display.float_format = '{:20, .2f}'.format
```

In [9]:

```
files = [os.path.join(dp, f) for dp, dn, fn in os.walk(os.path.expanduser("./train")) for f in fn]
images_df = pd.DataFrame({
    'files': files,
    'familyId': [file.split('/')[2] for file in files],
    'kinId': [file.split('/')[3] for file in files],
    'uniqueId': [file.split('/')[2] + '/' + file.split('/')[3] for file in files]
})
images_df.head()
```

Out[9]:

	familyId	files	kinId	uniqueId
0	F0435	./train/F0435/MID5/P04588_face5.jpg	MID5	F0435/MID5
1	F0435	./train/F0435/MID5/P04592_face3.jpg	MID5	F0435/MID5
2	F0435	./train/F0435/MID5/P04585_face2.jpg	MID5	F0435/MID5
3	F0435	./train/F0435/MID5/P04584_face4.jpg	MID5	F0435/MID5
4	F0435	./train/F0435/MID5/P04583_face2.jpg	MID5	F0435/MID5

In [10]:

```
print("Total number of members in the dataset: {0}".format(images_df["uniqueId"].nunique()))
print("Total number of families in the dataset: {0}".format(images_df["familyId"].nunique()))
```

```
Total number of members in the dataset: 2316
Total number of families in the dataset: 470
```

In [11]:

```
family_with_most_pic = images_df["familyId"].value_counts()
kin_with_most_pic = images_df["uniqueId"].value_counts()
print("Family with maximum number of images: {0}, Image Count: {1}".format(family_with_most_pic.index[0
], family_with_most_pic[0]))
print("Member with maximum number of images: {0}, Image Count: {1}".format(kin_with_most_pic.index[0],
kin_with_most_pic[0]))
```

```
Family with maximum number of images: F0601, Image Count: 776
Member with maximum number of images: F0601/MID6, Image Count: 95
```

In [12]:

```
family_series = family_with_most_pic[:25]
labels = (np.array(family_series.index))
sizes = (np.array((family_series / family_with_most_pic.sum()) * 100))

trace = go.Pie(labels=labels, values=sizes)
layout = go.Layout(title='Pic Count by Families')
data = [trace]
fig = go.Figure(data=data, layout=layout)
py.iplot(fig, filename='Families')
```

PREPARING DATA

In [ ]:

In [13]:

```
#train_file_path = "./recognizing-faces-in-the-wild/train_relationships.csv"
train_file_path = "./train_relationships.csv"
train_folders_path = "./train/"
test_path = "./test/"
sample_sub_df=pd.read_csv('sample_submission.csv')
# Considering F09 for validation
val_families = "F09"
```

In [14]:

In [14]:

```
# all_images will have path of all jpg with
all_images = glob(train_folders_path + "*/*/*.jpg")
print(all_images[0])

# For windows 10 issue using following comment above
#from wcmatch import glob
#all_images = glob.glob(train_folders_path + "*\//*\//*.jpg")
#all_images = glob.glob(train_folders_path + "*/*/*.jpg")
#print(all_images[0])
```

```
./train/F0435/MID5/P04588_face5.jpg
```

In [15]:

```
# In train_images collecting path of all training data
# Path consist of Family name and person name as well
train_images = [x for x in all_images if val_families not in x]

# In val_images collecting path of all validation data
# Path consist of Family name and person name as well
val_images = [x for x in all_images if val_families in x]
```

In [16]:

```
# ppl contains Family name and person name
ppl = [x.split("/")[-3] + "/" + x.split("/")[-2] for x in all_images]
# For windows 10 comment above use following
#ppl = [x.split("\\")[-3] + "/" + x.split("\\")[-2] for x in all_images]
print(ppl[0])
```

```
F0435/MID5
```

In [17]:

```
# There are 2 for loops to create
# --> key as Family and person id
# --> value as image complete path of that person
train_person_to_images_map = defaultdict(list)
for x in train_images:
    train_person_to_images_map[x.split("/")[-3] + "/" + x.split("/")[-2]].append(x)
    #train_person_to_images_map[x.split("\\")[-3] + "/" + x.split("\\")[-2]].append(x)

val_person_to_images_map = defaultdict(list)
for x in val_images:
    val_person_to_images_map[x.split("/")[-3] + "/" + x.split("/")[-2]].append(x)
    #val_person_to_images_map[x.split("\\")[-3] + "/" + x.split("\\")[-2]].append(x)
```

In [18]:

```
#print(train_person_to_images_map.keys())
print(train_person_to_images_map['F0002/MID1'])
print(val_person_to_images_map['F0900/MID1'])
```

```
['./train/F0002/MID1/P00017_face3.jpg', './train/F0002/MID1/P00009_face3.jpg', './train/F0002/MID1/P000
10_face4.jpg', './train/F0002/MID1/P00011_face1.jpg', './train/F0002/MID1/P00016_face2.jpg', './train/F
0002/MID1/P00012_face2.jpg', './train/F0002/MID1/P00018_face1.jpg', './train/F0002/MID1/P00014_face2.jp
g', './train/F0002/MID1/P00015_face2.jpg', './train/F0002/MID1/P00013_face2.jpg']
['./train/F0900/MID1/P09509_face1.jpg', './train/F0900/MID1/P09513_face1.jpg', './train/F0900/MID1/P095
05_face1.jpg', './train/F0900/MID1/P09506_face1.jpg', './train/F0900/MID1/P09508_face1.jpg']
```

In [19]:

```
# Creating relationshop dataframe from train_relationships.csv
# Then considering only those values which are present in ppl
relationships = pd.read_csv(train_file_path)
print('#'* 50)
print(relationships.head())
relationships = list(zip(relationships.p1.values, relationships.p2.values))
```

```
print('#'* 50)
print(relationships[0])
print(len(relationships))
relationships = [x for x in relationships if x[0] in ppl and x[1] in ppl]
print('#'* 50)
print(relationships[0])
print(len(relationships))
print('#'* 50)
```

```
##################################################
          p1          p2
0  F0002/MID1  F0002/MID3
1  F0002/MID2  F0002/MID3
2  F0005/MID1  F0005/MID2
3  F0005/MID3  F0005/MID2
4  F0009/MID1  F0009/MID4
##################################################
('F0002/MID1', 'F0002/MID3')
3598
##################################################
('F0002/MID1', 'F0002/MID3')
3362
##################################################
```

In [20]:

```
# Now Creating train and val
train = [x for x in relationships if val_families not in x[0]]
print('train data sample', train[0])
val = [x for x in relationships if val_families in x[0]]
print('validation data sample', val[0])
```

```
train data sample ('F0002/MID1', 'F0002/MID3')
validation data sample ('F0900/MID2', 'F0900/MID1')
```

In [ ]:

```

```

In [ ]:

```

```

In [ ]:

```

```

In [21]:

```
batch_tuples_test = sample(train, 8)
print(batch_tuples_test)
labels_test = [1] * len(batch_tuples_test)
print(labels_test)
```

```
[('F0748/MID1', 'F0748/MID3'), ('F0829/MID6', 'F0829/MID1'), ('F0693/MID2', 'F0693/MID4'), ('F0376/MID8
', 'F0376/MID4'), ('F0421/MID3', 'F0421/MID7'), ('F0568/MID1', 'F0568/MID4'), ('F0174/MID5', 'F0174/MID
7'), ('F0287/MID3', 'F0287/MID4')]
[1, 1, 1, 1, 1, 1, 1, 1]
```

In [22]:

```
ppl_test = list(train_person_to_images_map.keys())
p1 = choice(ppl_test)
p2 = choice(ppl_test)

print(p1)
print(p2)
```

```
F0387/MID2
F0209/MID4
```

In [23]:

```
print((train_person_to_images_map['F0488/MID1']))
print(len(train_person_to_images_map['F0488/MID1']))
print(choice(train_person_to_images_map['F0488/MID1']))

print('*' * 50)
for x in batch_tuples_test:
    print(x[0])
    print((train_person_to_images_map[x[0]]))
    print(len(train_person_to_images_map[x[0]]))
    print(choice(train_person_to_images_map[x[0]]))
```

```
['./train/F0488/MID1/P05139_face3.jpg', './train/F0488/MID1/P05138_face2.jpg', './train/F0488/MID1/P051
32_face1.jpg', './train/F0488/MID1/P05134_face1.jpg', './train/F0488/MID1/P05133_face1.jpg']
5
./train/F0488/MID1/P05138_face2.jpg
**************************************************
F0748/MID1
['./train/F0748/MID1/P07842_face2.jpg', './train/F0748/MID1/P07840_face3.jpg', './train/F0748/MID1/P078
34_face2.jpg', './train/F0748/MID1/P07835_face2.jpg', './train/F0748/MID1/P07836_face2.jpg', './train/F
0748/MID1/P07841_face1.jpg', './train/F0748/MID1/P07837_face1.jpg', './train/F0748/MID1/P07838_face2.jp
g', './train/F0748/MID1/P07839_face2.jpg']
9
./train/F0748/MID1/P07842_face2.jpg
F0829/MID6
['./train/F0829/MID6/P08762_face2.jpg', './train/F0829/MID6/P08763_face2.jpg', './train/F0829/MID6/P087
70_face5.jpg', './train/F0829/MID6/P08766_face1.jpg']
4
./train/F0829/MID6/P08762_face2.jpg
F0693/MID2
['./train/F0693/MID2/P07225_face2.jpg', './train/F0693/MID2/P07220_face2.jpg', './train/F0693/MID2/P072
22_face2.jpg']
3
./train/F0693/MID2/P07220_face2.jpg
F0376/MID8
['./train/F0376/MID8/P10643_face1.jpg', './train/F0376/MID8/P10625_face4.jpg', './train/F0376/MID8/P106
29_face1.jpg', './train/F0376/MID8/P10640_face1.jpg', './train/F0376/MID8/P10630_face2.jpg', './train/F
0376/MID8/P10626_face5.jpg', './train/F0376/MID8/P10627_face1.jpg', './train/F0376/MID8/P10636_face1.jp
g', './train/F0376/MID8/P10642_face1.jpg', './train/F0376/MID8/P10637_face3.jpg', './train/F0376/MID8/P
10628_face1.jpg', './train/F0376/MID8/P10641_face4.jpg']
12
./train/F0376/MID8/P10627_face1.jpg
F0421/MID3
['./train/F0421/MID3/P04422_face7.jpg', './train/F0421/MID3/P04425_face3.jpg', './train/F0421/MID3/P044
28_face4.jpg', './train/F0421/MID3/P04429_face1.jpg', './train/F0421/MID3/P04423_face5.jpg', './train/F
0421/MID3/P04430_face3.jpg']
6
./train/F0421/MID3/P04425_face3.jpg
F0568/MID1
['./train/F0568/MID1/P05975_face1.jpg', './train/F0568/MID1/P05980_face2.jpg', './train/F0568/MID1/P059
71_face2.jpg', './train/F0568/MID1/P05977_face1.jpg', './train/F0568/MID1/P05972_face1.jpg', './train/F
0568/MID1/P05982_face2.jpg', './train/F0568/MID1/P05981_face1.jpg', './train/F0568/MID1/P05978_face1.jp
g', './train/F0568/MID1/P05979_face1.jpg', './train/F0568/MID1/P05973_face2.jpg', './train/F0568/MID1/P
05983_face3.jpg', './train/F0568/MID1/P05974_face1.jpg']
12
./train/F0568/MID1/P05977_face1.jpg
F0174/MID5
['./train/F0174/MID5/P01869_face4.jpg', './train/F0174/MID5/P01874_face1.jpg', './train/F0174/MID5/P018
73_face2.jpg']
3
./train/F0174/MID5/P01869_face4.jpg
F0287/MID3
['./train/F0287/MID3/P03069_face3.jpg', './train/F0287/MID3/P03063_face1.jpg', './train/F0287/MID3/P030
67_face2.jpg', './train/F0287/MID3/P03064_face1.jpg']
4
./train/F0287/MID3/P03064_face1.jpg
```

In [24]:

```
df = pd.read_csv("./train_relationships.csv")
df = pd.read_csv('./train_relationships.csv')
df.head()
```

Out[24]:

	p1	p2
0	F0002/MID1	F0002/MID3
1	F0002/MID2	F0002/MID3
2	F0005/MID1	F0005/MID2
3	F0005/MID3	F0005/MID2
4	F0009/MID1	F0009/MID4

In [25]:

```
new = df["p1"].str.split("/", n = 1, expand = True)

# making separate first name column from new data frame
df["Family"]= new[0]
# making separate last name column from new data frame
df["Person1"]= new[1]

# Dropping old Name columns
df.drop(columns =["p1"], inplace = True)

new = df["p2"].str.split("/", n = 1, expand = True)

# making separate first name column from new data frame
df["Family2"]= new[0]
# making separate last name column from new data frame
df["Person2"]= new[1]

# Dropping old Name columns
df.drop(columns =["p2"], inplace = True)
df.head()
```

Out[25]:

	Family	Person1	Family2	Person2
0	F0002	MID1	F0002	MID3
1	F0002	MID2	F0002	MID3
2	F0005	MID1	F0005	MID2
3	F0005	MID3	F0005	MID2
4	F0009	MID1	F0009	MID4

In [26]:

```
del df['Family2']
```

In [27]:

```
df['Related'] = 1

#Creating a dictionary, and storing members of each family
df_dict = {}
for index, row in df.iterrows():
    if row['Family'] in df_dict:
        df_dict[row['Family']].append(row['Person1'])
    else:
        df_dict[row['Family']] = [row['Person1']]

#For each family in this dictionary, we'll first make pairs of people
#For each pair, we'll check if they're related in our existing Dataset
#If they're not in the dataframe, means we'll create a row with both persons and related value 0
i=1
for key in df_dict:
    pair = list(itertools.combinations(df_dict[key], 2))
    for item in pair:
        if len(df[(df['Family']==key)&(df['Person1']==item[0])&(df['Person2']==item[1])])==0 \
        and len(df[(df['Family']==key)&(df['Person1']==item[1])&(df['Person2']==item[0])])==0:
            new = {'Family':key,'Person1':item[0],'Person2':item[1],'Related':0}
            df=df.append(new,ignore_index=True)

#Storing rows only where Person1 and Person2 are not same
df = df[(df['Person1']!=df['Person2'])]

#len(df[(df['Related']==1)])

print(df['Related'].value_counts())
```

```
1    3598
0    1720
Name: Related, dtype: int64
```

In [28]:

```
df.dtypes
```

Out[28]:

```
Family     object
Person1    object
Person2    object
Related     int64
dtype: object
```

In [29]:

```
df.describe
```

Out[29]:

```
<bound method NDFrame.describe of      Family Person1 Person2  Related
0     F0002    MID1    MID3        1
1     F0002    MID2    MID3        1
2     F0005    MID1    MID2        1
3     F0005    MID3    MID2        1
4     F0009    MID1    MID4        1
...     ...     ...     ...      ...
6285  F0203    MID4    MID8        0
6286  F0203    MID4    MID9        0
6289  F0203    MID7    MID8        0
6290  F0203    MID7    MID9        0
6292  F0203    MID8    MID9        0

[5318 rows x 4 columns]>
```

In [30]:

```
df[df['Related']==0].head()
```

Out[30]:

	Family	Person1	Person2	Related
3598	F0656	MID1	MID2	0
3599	F0656	MID1	MID4	0
3601	F0464	MID1	MID2	0
3603	F0821	MID1	MID3	0
3605	F0821	MID2	MID4	0

CHECKING FEW IMAGES

In [0]:

```
img_path = Path('./train/')
```

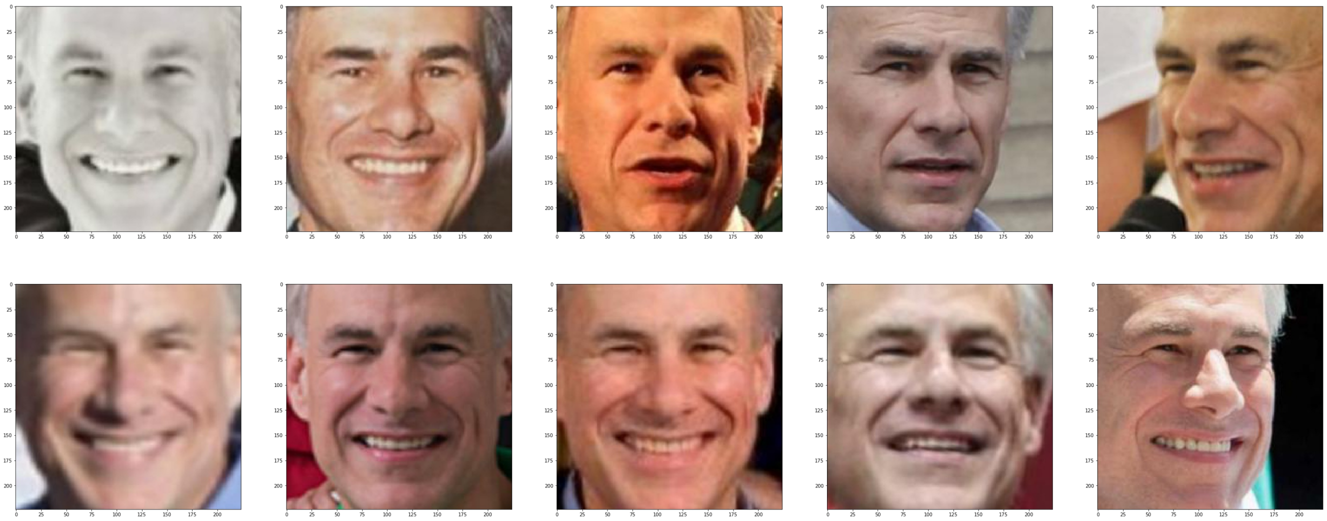
In [57]:

```
dir = os.path.join(img_path / df.Family[0] / df.Person1[0])
img_list = os.listdir(dir)
print(img_list)
```

```
['P00016_face2.jpg', 'P00015_face2.jpg', 'P00014_face2.jpg', 'P00011_face1.jpg', 'P00018_face1.jpg', 'P
00009_face3.jpg', 'P00013_face2.jpg', 'P00017_face3.jpg', 'P00010_face4.jpg', 'P00012_face2.jpg']
```

In [58]:

```
fig,ax = plt.subplots(2,5, figsize=(50,20))

for i in range(len(img_list)):
    with open(img_path / df.Family[0] / df.Person1[0] / img_list[i] ,'rb') as f:
        img = Image.open(f)
        ax[i%2][i//2].imshow(img)
fig.show()
```

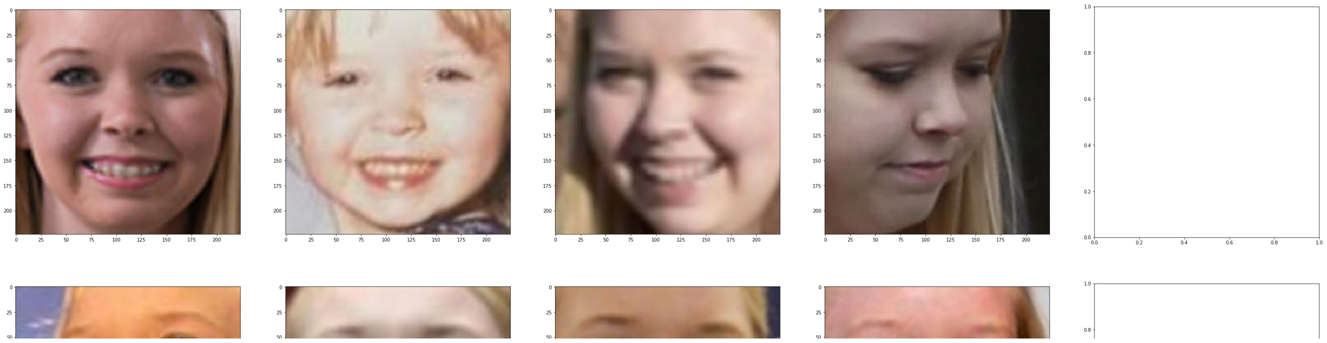


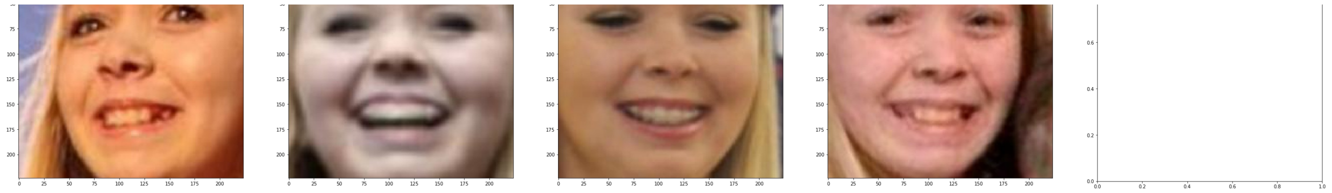
In [0]:

```
img_list = os.listdir(img_path / df.Family[0] / df.Person2[0])
```

In [60]:

```
fig,ax = plt.subplots(2,5, figsize=(50,20))

for i in range(len(img_list)):
    with open(img_path / df.Family[0] / df.Person2[0] / img_list[i] ,'rb') as f:
        img = Image.open(f)
        ax[i%2][i//2].imshow(img)
fig.show()
```





In [0]:

```

```

In [0]:

```

```

Defining functions

In [31]:

```
# Function for reading images
def read_img(path, IMG_SIZE):
    img = image.load_img(path, target_size=(IMG_SIZE, IMG_SIZE))
    img = np.array(img).astype(np.float)
    return preprocess_input(img, version=2)
```

In [32]:

```
def auc(y_true, y_pred):
    return tf.py_function(roc_auc_score, (y_true, y_pred), tf.double)
```

In [33]:

```
def get_thumbnail(path):
    if path and os.path.exists(path):
        i = Image.open(path)
        i.thumbnail((150, 150), Image.LANCZOS)
        return i

def image_base64(im):
    if isinstance(im, str):
        im = get_thumbnail(im)
    with BytesIO() as buffer:
        im.save(buffer, 'jpeg')
        return base64.b64encode(buffer.getvalue()).decode()

def add_image_path(x):
    image_path = 'train/' + x
    if os.path.exists(image_path):
        path = os.path.join(image_path, os.listdir(image_path)[0])
        return path
```

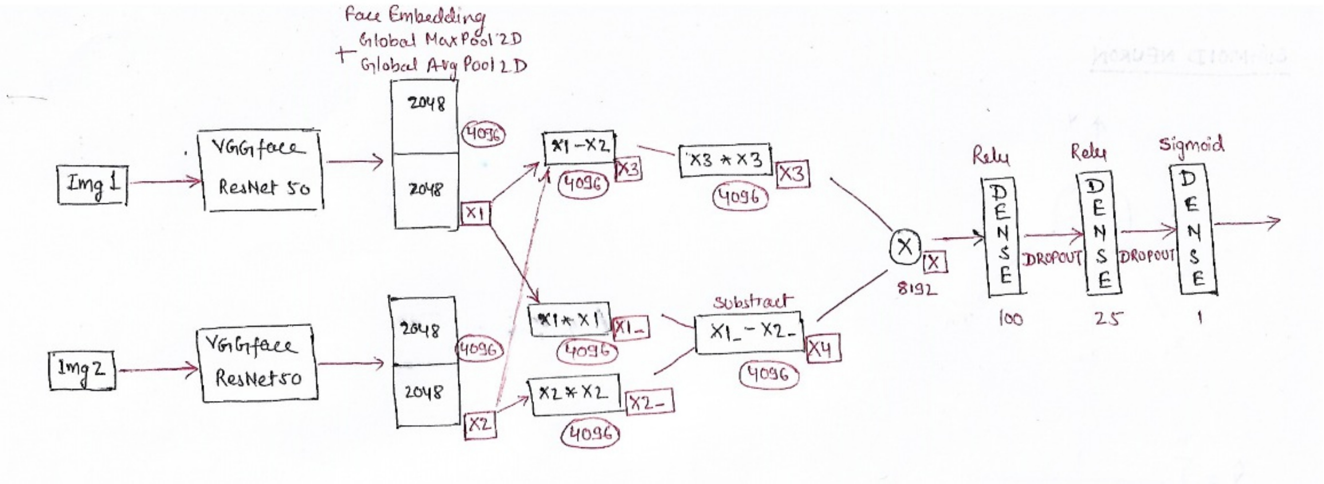
In [34]:

```
# https://www.kaggle.com/janpreets/just-another-feature-extractor-0-824-lb
def outer_product(x):
    """
    calculate outer-products of 2 tensors

        args
            x
                list of 2 tensors
                , assuming each of which has shape = (size_minibatch, total_pixels, size_filter)
    """
    return keras.backend.batch_dot(
                x[0]
                , x[1]
                , axes=[1,1]
            ) / x[0].get_shape().as_list()[1]

def signed_sqrt(x):
    """
    calculate element-wise signed square root

        args
            x
                a tensor
    """
    return keras.backend.sign(x) * keras.backend.sqrt(keras.backend.abs(x) + 1e-9)

def L2_norm(x, axis=-1):
    """
    calculate L2-norm

        args
            x
                a tensor
    """
    return keras.backend.l2_normalize(x, axis=axis)
```

In [0]:

```

```

MODEL 1



In [0]:

```
# Function gen is to generate data for label 0
# example --> gen(train, train_person_to_images_map, batch_size=16)
# train_person_to_images_map is dictionary with family and person as key and values are image path
#
def gen(list_tuples, person_to_images_map, batch_size=16):
    # All keys of dictionary stored in
    ppl = list(person_to_images_map.keys())
    while True:
        # list_tuples is train data from relationship csv file
        # batch_tuples contains sample from train data (from relationship.csv)
        batch_tuples = sample(list_tuples, batch_size // 2)
        # If batch_size is of size 16 then batch_tuples will have 8 samples
        labels = [1] * len(batch_tuples)
        while len(batch_tuples) < batch_size:
            # choice randomly picking data from ppl
            p1 = choice(ppl)
            p2 = choice(ppl)
            # Randomply Creating data with labels 0
            if p1 != p2 and (p1, p2) not in list_tuples and (p2, p1) not in list_tuples:
                batch_tuples.append((p1, p2))
                labels.append(0)
        # for each family person combination from batch_tuples
        # x[0] means considering first entry for ex --> ('F0488/MID1', 'F0488/MID4')  --> consider 'F04
88/MID1'
        # Check in dictioanry train_person_to_images_map
        for x in batch_tuples:
            if not len(person_to_images_map[x[0]]):
                print(x[0])

        # Considering randomly images using choice function
        X1 = [choice(person_to_images_map[x[0]]) for x in batch_tuples]
        X1 = np.array([read_img(x, 197) for x in X1])

        X2 = [choice(person_to_images_map[x[1]]) for x in batch_tuples]
        X2 = np.array([read_img(x, 197) for x in X2])

        yield [X1, X2], labels
```

In [0]:

```
# Preparing baseline_model1
def baseline_model1():
    input_1 = Input(shape=(197, 197, 3))
    input_2 = Input(shape=(197, 197, 3))

    base_model1 = VGGFace(model='resnet50', include_top=False)

    for x in base_model1.layers[:-3]:
        x.trainable = True
#    for x in base_model1.layers[-3:]:
#        x.trainable=False

    x1 = base_model1(input_1)
    x2 = base_model1(input_2)

    x1 = Concatenate(axis=-1)([GlobalMaxPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])

    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])

    x1_ = Multiply()([x1, x1])
    x2_ = Multiply()([x2, x2])
    x4 = Subtract()([x1_, x2_])
    x = Concatenate(axis=-1)([x4, x3])

    x = Dense(100, activation="relu", activity_regularizer=l1(0.001))(x)
    x = Dropout(0.3)(x)
    x = Dense(25, activation="relu", activity_regularizer=l1(0.001))(x)
    x = Dropout(0.01)(x)
    out = Dense(1, activation="sigmoid")(x)

    model1 = Model([input_1, input_2], out)

    model1.compile(loss="binary_crossentropy", metrics=['acc'], optimizer=Adam(0.00001))

    model1.summary()

    return model1
```

In [0]:

```
file_path1 = "./model1.h5"

checkpoint1 = ModelCheckpoint(file_path1, monitor='val_acc', verbose=1, save_best_only=True, mode='max'
)

reduce_on_plateau1 = ReduceLROnPlateau(monitor="val_acc", mode="max", factor=0.1, patience=20, verbose=
1)

callbacks_list1 = [checkpoint1, reduce_on_plateau1]

model1 = baseline_model1()
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:66:
The name tf.get_default_graph is deprecated. Please use tf.compat.v1.get_default_graph instead.
The name tf.get_default_graph is deprecated. Please use tf.compat.v1.get_default_graph instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:541:
The name tf.placeholder is deprecated. Please use tf.compat.v1.placeholder instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4432
: The name tf.random_uniform is deprecated. Please use tf.random.uniform instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:190:
The name tf.get_default_session is deprecated. Please use tf.compat.v1.get_default_session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:197:
The name tf.ConfigProto is deprecated. Please use tf.compat.v1.ConfigProto instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:203:
The name tf.Session is deprecated. Please use tf.compat.v1.Session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:207:
The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:216:
The name tf.is_variable_initialized is deprecated. Please use tf.compat.v1.is_variable_initialized inst
ead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:223:
The name tf.variables_initializer is deprecated. Please use tf.compat.v1.variables_initializer instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:2041
: The name tf.nn.fused_batch_norm is deprecated. Please use tf.compat.v1.nn.fused_batch_norm instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:148:
The name tf.placeholder_with_default is deprecated. Please use tf.compat.v1.placeholder_with_default in
stead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4267
: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4271
: The name tf.nn.avg_pool is deprecated. Please use tf.nn.avg_pool2d instead.

Downloading data from https://github.com/rcmalli/keras-vggface/releases/download/v2.0/rcmalli_vggface_t
f_notop_resnet50.h5
94699520/94694792 [==============================] - 1s 0us/step
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:3733
: calling dropout (from tensorflow.python.ops.nn_ops) with keep_prob is deprecated and will be removed
in a future version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/optimizers.py:793: The name tf.tra
in.Optimizer is deprecated. Please use tf.compat.v1.train.Optimizer instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:3657
: The name tf.log is deprecated. Please use tf.math.log instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorflow_core/python/ops/nn_impl.py:18
3: where (from tensorflow.python.ops.array_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
Model: "model_1"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            (None, 197, 197, 3)  0
__________________________________________________________________________________________________
input_2 (InputLayer)            (None, 197, 197, 3)  0
__________________________________________________________________________________________________
vggface_resnet50 (Model)        multiple             23561152    input_1[0][0]
                                                                 input_2[0][0]
__________________________________________________________________________________________________
global_max_pooling2d_1 (GlobalM (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_average_pooling2d_1 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_max_pooling2d_2 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
global_average_pooling2d_2 (Glo (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
concatenate_1 (Concatenate)     (None, 4096)         0           global_max_pooling2d_1[0][0]
```

```
concatenate_1 (Concatenate)     (None, 4096)         0           global_max_pooling2d_1[0][0]
                                                                 global_average_pooling2d_1[0][0]
__________________________________________________________________________________________________
concatenate_2 (Concatenate)     (None, 4096)         0           global_max_pooling2d_2[0][0]
                                                                 global_average_pooling2d_2[0][0]
__________________________________________________________________________________________________
multiply_2 (Multiply)           (None, 4096)         0           concatenate_1[0][0]
                                                                 concatenate_1[0][0]
__________________________________________________________________________________________________
multiply_3 (Multiply)           (None, 4096)         0           concatenate_2[0][0]
                                                                 concatenate_2[0][0]
__________________________________________________________________________________________________
subtract_1 (Subtract)           (None, 4096)         0           concatenate_1[0][0]
                                                                 concatenate_2[0][0]
__________________________________________________________________________________________________
subtract_2 (Subtract)           (None, 4096)         0           multiply_2[0][0]
                                                                 multiply_3[0][0]
__________________________________________________________________________________________________
multiply_1 (Multiply)           (None, 4096)         0           subtract_1[0][0]
                                                                 subtract_1[0][0]
__________________________________________________________________________________________________
concatenate_3 (Concatenate)     (None, 8192)         0           subtract_2[0][0]
                                                                 multiply_1[0][0]
__________________________________________________________________________________________________
dense_1 (Dense)                 (None, 100)          819300      concatenate_3[0][0]
__________________________________________________________________________________________________
dropout_1 (Dropout)             (None, 100)          0           dense_1[0][0]
__________________________________________________________________________________________________
dense_2 (Dense)                 (None, 25)           2525        dropout_1[0][0]
__________________________________________________________________________________________________
dropout_2 (Dropout)             (None, 25)           0           dense_2[0][0]
__________________________________________________________________________________________________
dense_3 (Dense)                 (None, 1)            26          dropout_2[0][0]
==================================================================================================
Total params: 24,383,003
Trainable params: 24,329,883
Non-trainable params: 53,120
__________________________________________________________________________________________________
```

In [0]:

```
model1.fit_generator(gen(train, train_person_to_images_map, batch_size=16), use_multiprocessing=True,
                    validation_data=gen(val, val_person_to_images_map, batch_size=16), epochs=100, verb
ose=1,
                    workers = 4, callbacks=callbacks_list1, steps_per_epoch=300, validation_steps=100)
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1033
: The name tf.assign_add is deprecated. Please use tf.compat.v1.assign_add instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1020
: The name tf.assign is deprecated. Please use tf.compat.v1.assign instead.

Epoch 1/100
300/300 [==============================] - 155s 517ms/step - loss: 4.2482 - acc: 0.5408 - val_loss: 4.5
607 - val_acc: 0.5850

Epoch 00001: val_acc improved from -inf to 0.58500, saving model to ./model1.h5
Epoch 2/100
300/300 [==============================] - 129s 430ms/step - loss: 0.9521 - acc: 0.5585 - val_loss: 1.6
322 - val_acc: 0.5944

Epoch 00002: val_acc improved from 0.58500 to 0.59437, saving model to ./model1.h5
Epoch 3/100
300/300 [==============================] - 129s 429ms/step - loss: 0.7854 - acc: 0.5708 - val_loss: 1.3
397 - val_acc: 0.5719

Epoch 00003: val_acc did not improve from 0.59437
Epoch 4/100
300/300 [==============================] - 129s 429ms/step - loss: 0.7376 - acc: 0.5721 - val_loss: 1.3
048 - val_acc: 0.5600

Epoch 00004: val_acc did not improve from 0.59437
Epoch 5/100
300/300 [==============================] - 129s 430ms/step - loss: 0.7180 - acc: 0.5767 - val_loss: 0.9
126 - val_acc: 0.5787
```

```
Epoch 00005: val_acc did not improve from 0.59437
Epoch 6/100
300/300 [==============================] - 129s 430ms/step - loss: 0.7020 - acc: 0.5985 - val_loss: 0.9
738 - val_acc: 0.5837

Epoch 00006: val_acc did not improve from 0.59437
Epoch 7/100
300/300 [==============================] - 129s 429ms/step - loss: 0.6957 - acc: 0.6127 - val_loss: 0.8
859 - val_acc: 0.5925

Epoch 00007: val_acc did not improve from 0.59437
Epoch 8/100
300/300 [==============================] - 128s 426ms/step - loss: 0.6853 - acc: 0.6231 - val_loss: 0.7
978 - val_acc: 0.6319

Epoch 00008: val_acc improved from 0.59437 to 0.63187, saving model to ./model1.h5
Epoch 9/100
300/300 [==============================] - 128s 425ms/step - loss: 0.6802 - acc: 0.6450 - val_loss: 0.8
370 - val_acc: 0.6269

Epoch 00009: val_acc did not improve from 0.63187
Epoch 10/100
300/300 [==============================] - 128s 426ms/step - loss: 0.6785 - acc: 0.6415 - val_loss: 0.7
451 - val_acc: 0.6362

Epoch 00010: val_acc improved from 0.63187 to 0.63625, saving model to ./model1.h5
Epoch 11/100
300/300 [==============================] - 128s 427ms/step - loss: 0.6742 - acc: 0.6473 - val_loss: 0.7
798 - val_acc: 0.6469

Epoch 00011: val_acc improved from 0.63625 to 0.64687, saving model to ./model1.h5
Epoch 12/100
300/300 [==============================] - 128s 427ms/step - loss: 0.6692 - acc: 0.6571 - val_loss: 0.7
349 - val_acc: 0.6425

Epoch 00012: val_acc did not improve from 0.64687
Epoch 13/100
300/300 [==============================] - 129s 429ms/step - loss: 0.6644 - acc: 0.6727 - val_loss: 0.6
964 - val_acc: 0.6438

Epoch 00013: val_acc did not improve from 0.64687
Epoch 14/100
300/300 [==============================] - 129s 430ms/step - loss: 0.6614 - acc: 0.6792 - val_loss: 0.7
064 - val_acc: 0.6488

Epoch 00014: val_acc improved from 0.64687 to 0.64875, saving model to ./model1.h5
Epoch 15/100
300/300 [==============================] - 129s 429ms/step - loss: 0.6586 - acc: 0.6931 - val_loss: 0.6
794 - val_acc: 0.6456

Epoch 00015: val_acc did not improve from 0.64875
Epoch 16/100
300/300 [==============================] - 128s 427ms/step - loss: 0.6553 - acc: 0.6977 - val_loss: 0.6
722 - val_acc: 0.6750

Epoch 00016: val_acc improved from 0.64875 to 0.67500, saving model to ./model1.h5
Epoch 17/100
300/300 [==============================] - 129s 430ms/step - loss: 0.6505 - acc: 0.7106 - val_loss: 0.6
761 - val_acc: 0.6919

Epoch 00017: val_acc improved from 0.67500 to 0.69188, saving model to ./model1.h5
Epoch 18/100
300/300 [==============================] - 128s 428ms/step - loss: 0.6471 - acc: 0.7127 - val_loss: 0.6
755 - val_acc: 0.7050

Epoch 00018: val_acc improved from 0.69188 to 0.70500, saving model to ./model1.h5
Epoch 19/100
300/300 [==============================] - 128s 428ms/step - loss: 0.6463 - acc: 0.7160 - val_loss: 0.6
808 - val_acc: 0.6863

Epoch 00019: val_acc did not improve from 0.70500
Epoch 20/100
300/300 [==============================] - 128s 427ms/step - loss: 0.6377 - acc: 0.7296 - val_loss: 0.6
669 - val_acc: 0.7244

Epoch 00020: val_acc improved from 0.70500 to 0.72437, saving model to ./model1.h5
Epoch 21/100
Epoch 21/100
300/300 [==============================] - 128s 427ms/step - loss: 0.6307 - acc: 0.7498 - val_loss: 0.6
530 - val_acc: 0.7181

Epoch 00021: val_acc did not improve from 0.72437
Epoch 22/100
300/300 [==============================] - 129s 428ms/step - loss: 0.6288 - acc: 0.7587 - val_loss: 0.6
589 - val_acc: 0.7050

Epoch 00022: val_acc did not improve from 0.72437
Epoch 23/100
300/300 [==============================] - 128s 427ms/step - loss: 0.6242 - acc: 0.7500 - val_loss: 0.6
598 - val_acc: 0.7300

Epoch 00023: val_acc improved from 0.72437 to 0.73000, saving model to ./model1.h5
Epoch 24/100
300/300 [==============================] - 129s 429ms/step - loss: 0.6244 - acc: 0.7533 - val_loss: 0.6
483 - val_acc: 0.7206

Epoch 00024: val_acc did not improve from 0.73000
Epoch 25/100
300/300 [==============================] - 128s 428ms/step - loss: 0.6176 - acc: 0.7663 - val_loss: 0.6
410 - val_acc: 0.7581

Epoch 00025: val_acc improved from 0.73000 to 0.75813, saving model to ./model1.h5
Epoch 26/100
300/300 [==============================] - 129s 429ms/step - loss: 0.6158 - acc: 0.7733 - val_loss: 0.6
311 - val_acc: 0.7638

Epoch 00026: val_acc improved from 0.75813 to 0.76375, saving model to ./model1.h5
Epoch 27/100
300/300 [==============================] - 128s 427ms/step - loss: 0.6122 - acc: 0.7758 - val_loss: 0.6
455 - val_acc: 0.7331

Epoch 00027: val_acc did not improve from 0.76375
Epoch 28/100
300/300 [==============================] - 129s 429ms/step - loss: 0.6057 - acc: 0.7906 - val_loss: 0.6
285 - val_acc: 0.7538

Epoch 00028: val_acc did not improve from 0.76375
Epoch 29/100
300/300 [==============================] - 129s 431ms/step - loss: 0.6094 - acc: 0.7748 - val_loss: 0.6
349 - val_acc: 0.7512

Epoch 00029: val_acc did not improve from 0.76375
Epoch 30/100
300/300 [==============================] - 129s 429ms/step - loss: 0.5974 - acc: 0.7967 - val_loss: 0.6
207 - val_acc: 0.7638

Epoch 00030: val_acc did not improve from 0.76375
Epoch 31/100
300/300 [==============================] - 129s 429ms/step - loss: 0.5969 - acc: 0.7944 - val_loss: 0.6
296 - val_acc: 0.7550

Epoch 00031: val_acc did not improve from 0.76375
Epoch 32/100
300/300 [==============================] - 129s 428ms/step - loss: 0.5907 - acc: 0.8096 - val_loss: 0.6
224 - val_acc: 0.7638

Epoch 00032: val_acc did not improve from 0.76375
Epoch 33/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5872 - acc: 0.8081 - val_loss: 0.6
253 - val_acc: 0.7719

Epoch 00033: val_acc improved from 0.76375 to 0.77187, saving model to ./model1.h5
Epoch 34/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5854 - acc: 0.8123 - val_loss: 0.6
266 - val_acc: 0.7550

Epoch 00034: val_acc did not improve from 0.77187
Epoch 35/100
300/300 [==============================] - 128s 425ms/step - loss: 0.5852 - acc: 0.8104 - val_loss: 0.5
975 - val_acc: 0.7925

Epoch 00035: val_acc improved from 0.77187 to 0.79250, saving model to ./model1.h5
Epoch 36/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5848 - acc: 0.8115 - val_loss: 0.6
091 - val_acc: 0.7619
```

```
091 - val_acc: 0.7619

Epoch 00036: val_acc did not improve from 0.79250
Epoch 37/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5727 - acc: 0.8242 - val_loss: 0.6
007 - val_acc: 0.7913

Epoch 00037: val_acc did not improve from 0.79250
Epoch 38/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5665 - acc: 0.8262 - val_loss: 0.6
043 - val_acc: 0.7819

Epoch 00038: val_acc did not improve from 0.79250
Epoch 39/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5618 - acc: 0.8242 - val_loss: 0.5
861 - val_acc: 0.7744

Epoch 00039: val_acc did not improve from 0.79250
Epoch 40/100
300/300 [==============================] - 128s 425ms/step - loss: 0.5561 - acc: 0.8346 - val_loss: 0.5
872 - val_acc: 0.7725

Epoch 00040: val_acc did not improve from 0.79250
Epoch 41/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5505 - acc: 0.8333 - val_loss: 0.6
160 - val_acc: 0.7438

Epoch 00041: val_acc did not improve from 0.79250
Epoch 42/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5491 - acc: 0.8377 - val_loss: 0.6
024 - val_acc: 0.7656

Epoch 00042: val_acc did not improve from 0.79250
Epoch 43/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5395 - acc: 0.8485 - val_loss: 0.5
810 - val_acc: 0.7856

Epoch 00043: val_acc did not improve from 0.79250
Epoch 44/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5415 - acc: 0.8465 - val_loss: 0.6
042 - val_acc: 0.7669

Epoch 00044: val_acc did not improve from 0.79250
Epoch 45/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5343 - acc: 0.8421 - val_loss: 0.5
973 - val_acc: 0.7731

Epoch 00045: val_acc did not improve from 0.79250
Epoch 46/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5278 - acc: 0.8525 - val_loss: 0.5
805 - val_acc: 0.7819

Epoch 00046: val_acc did not improve from 0.79250
Epoch 47/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5235 - acc: 0.8508 - val_loss: 0.5
891 - val_acc: 0.7738

Epoch 00047: val_acc did not improve from 0.79250
Epoch 48/100
300/300 [==============================] - 128s 427ms/step - loss: 0.5158 - acc: 0.8615 - val_loss: 0.5
995 - val_acc: 0.7781

Epoch 00048: val_acc did not improve from 0.79250
Epoch 49/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5201 - acc: 0.8544 - val_loss: 0.5
905 - val_acc: 0.7919

Epoch 00049: val_acc did not improve from 0.79250
Epoch 50/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5112 - acc: 0.8608 - val_loss: 0.5
837 - val_acc: 0.7750

Epoch 00050: val_acc did not improve from 0.79250
Epoch 51/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5040 - acc: 0.8677 - val_loss: 0.6
004 - val_acc: 0.7600

Epoch 00051: val_acc did not improve from 0.79250
Epoch 00051: val_acc did not improve from 0.79250
Epoch 52/100
300/300 [==============================] - 128s 426ms/step - loss: 0.4935 - acc: 0.8746 - val_loss: 0.5
881 - val_acc: 0.7688

Epoch 00052: val_acc did not improve from 0.79250
Epoch 53/100
300/300 [==============================] - 128s 426ms/step - loss: 0.5065 - acc: 0.8560 - val_loss: 0.5
871 - val_acc: 0.7719

Epoch 00053: val_acc did not improve from 0.79250
Epoch 54/100
300/300 [==============================] - 128s 427ms/step - loss: 0.4937 - acc: 0.8702 - val_loss: 0.5
628 - val_acc: 0.7950

Epoch 00054: val_acc improved from 0.79250 to 0.79500, saving model to ./model1.h5
Epoch 55/100
300/300 [==============================] - 128s 426ms/step - loss: 0.4875 - acc: 0.8765 - val_loss: 0.5
982 - val_acc: 0.7756

Epoch 00055: val_acc did not improve from 0.79500
Epoch 56/100
300/300 [==============================] - 128s 426ms/step - loss: 0.4818 - acc: 0.8744 - val_loss: 0.5
798 - val_acc: 0.7612

Epoch 00056: val_acc did not improve from 0.79500
Epoch 57/100
300/300 [==============================] - 128s 426ms/step - loss: 0.4845 - acc: 0.8700 - val_loss: 0.5
742 - val_acc: 0.7969

Epoch 00057: val_acc improved from 0.79500 to 0.79688, saving model to ./model1.h5
Epoch 58/100
300/300 [==============================] - 128s 428ms/step - loss: 0.4745 - acc: 0.8860 - val_loss: 0.6
051 - val_acc: 0.7669

Epoch 00058: val_acc did not improve from 0.79688
Epoch 59/100
300/300 [==============================] - 127s 425ms/step - loss: 0.4660 - acc: 0.8881 - val_loss: 0.5
793 - val_acc: 0.7812

Epoch 00059: val_acc did not improve from 0.79688
Epoch 60/100
300/300 [==============================] - 128s 426ms/step - loss: 0.4691 - acc: 0.8835 - val_loss: 0.5
726 - val_acc: 0.7850

Epoch 00060: val_acc did not improve from 0.79688
Epoch 61/100
300/300 [==============================] - 128s 425ms/step - loss: 0.4658 - acc: 0.8810 - val_loss: 0.5
958 - val_acc: 0.7506

Epoch 00061: val_acc did not improve from 0.79688
Epoch 62/100
300/300 [==============================] - 128s 425ms/step - loss: 0.4594 - acc: 0.8881 - val_loss: 0.5
752 - val_acc: 0.7856

Epoch 00062: val_acc did not improve from 0.79688
Epoch 63/100
300/300 [==============================] - 127s 425ms/step - loss: 0.4547 - acc: 0.8917 - val_loss: 0.5
800 - val_acc: 0.7844

Epoch 00063: val_acc did not improve from 0.79688
Epoch 64/100
300/300 [==============================] - 128s 426ms/step - loss: 0.4544 - acc: 0.8900 - val_loss: 0.5
865 - val_acc: 0.7706

Epoch 00064: val_acc did not improve from 0.79688
Epoch 65/100
300/300 [==============================] - 128s 426ms/step - loss: 0.4543 - acc: 0.8896 - val_loss: 0.5
769 - val_acc: 0.7788

Epoch 00065: val_acc did not improve from 0.79688
Epoch 66/100
300/300 [==============================] - 128s 425ms/step - loss: 0.4478 - acc: 0.8958 - val_loss: 0.5
975 - val_acc: 0.7556

Epoch 00066: val_acc did not improve from 0.79688
Epoch 67/100
300/300 [==============================] - 127s 425ms/step - loss: 0.4417 - acc: 0.8988 - val_loss: 0.6
```

```
300/300 [==============================] - 127s 425ms/step - loss: 0.4417 - acc: 0.8988 - val_loss: 0.6
224 - val_acc: 0.7394

Epoch 00067: val_acc did not improve from 0.79688
Epoch 68/100
300/300 [==============================] - 127s 424ms/step - loss: 0.4381 - acc: 0.8994 - val_loss: 0.6
112 - val_acc: 0.7450

Epoch 00068: val_acc did not improve from 0.79688
Epoch 69/100
300/300 [==============================] - 127s 424ms/step - loss: 0.4322 - acc: 0.9010 - val_loss: 0.6
082 - val_acc: 0.7562

Epoch 00069: val_acc did not improve from 0.79688
Epoch 70/100
300/300 [==============================] - 127s 425ms/step - loss: 0.4372 - acc: 0.9025 - val_loss: 0.5
923 - val_acc: 0.7619

Epoch 00070: val_acc did not improve from 0.79688
Epoch 71/100
300/300 [==============================] - 128s 426ms/step - loss: 0.4243 - acc: 0.9017 - val_loss: 0.5
851 - val_acc: 0.7700

Epoch 00071: val_acc did not improve from 0.79688
Epoch 72/100
300/300 [==============================] - 127s 425ms/step - loss: 0.4233 - acc: 0.9083 - val_loss: 0.5
632 - val_acc: 0.7869

Epoch 00072: val_acc did not improve from 0.79688
Epoch 73/100
300/300 [==============================] - 127s 425ms/step - loss: 0.4200 - acc: 0.9133 - val_loss: 0.5
976 - val_acc: 0.7675

Epoch 00073: val_acc did not improve from 0.79688
Epoch 74/100
300/300 [==============================] - 128s 425ms/step - loss: 0.4189 - acc: 0.9102 - val_loss: 0.6
240 - val_acc: 0.7400

Epoch 00074: val_acc did not improve from 0.79688
Epoch 75/100
300/300 [==============================] - 128s 425ms/step - loss: 0.4147 - acc: 0.9115 - val_loss: 0.6
113 - val_acc: 0.7450

Epoch 00075: val_acc did not improve from 0.79688
Epoch 76/100
300/300 [==============================] - 127s 425ms/step - loss: 0.4096 - acc: 0.9125 - val_loss: 0.5
819 - val_acc: 0.7731

Epoch 00076: val_acc did not improve from 0.79688
Epoch 77/100
300/300 [==============================] - 128s 425ms/step - loss: 0.4022 - acc: 0.9160 - val_loss: 0.5
917 - val_acc: 0.7475

Epoch 00077: val_acc did not improve from 0.79688

Epoch 00077: ReduceLROnPlateau reducing learning rate to 9.999999747378752e-07.
Epoch 78/100
300/300 [==============================] - 128s 425ms/step - loss: 0.4029 - acc: 0.9154 - val_loss: 0.5
991 - val_acc: 0.7588

Epoch 00078: val_acc did not improve from 0.79688
Epoch 79/100
300/300 [==============================] - 127s 425ms/step - loss: 0.3964 - acc: 0.9231 - val_loss: 0.6
198 - val_acc: 0.7300

Epoch 00079: val_acc did not improve from 0.79688
Epoch 80/100
300/300 [==============================] - 128s 426ms/step - loss: 0.3930 - acc: 0.9269 - val_loss: 0.6
234 - val_acc: 0.7331

Epoch 00080: val_acc did not improve from 0.79688
Epoch 81/100
300/300 [==============================] - 127s 425ms/step - loss: 0.3876 - acc: 0.9210 - val_loss: 0.6
119 - val_acc: 0.7362

Epoch 00081: val_acc did not improve from 0.79688
Epoch 82/100
300/300 [==============================] - 128s 426ms/step - loss: 0.3857 - acc: 0.9250 - val_loss: 0.6
```

```
300/300 [==============================] - 128s 426ms/step - loss: 0.3857 - acc: 0.9250 - val_loss: 0.6
117 - val_acc: 0.7400

Epoch 00082: val_acc did not improve from 0.79688
Epoch 83/100
300/300 [==============================] - 128s 426ms/step - loss: 0.3857 - acc: 0.9325 - val_loss: 0.5
927 - val_acc: 0.7588

Epoch 00083: val_acc did not improve from 0.79688
Epoch 84/100
300/300 [==============================] - 127s 425ms/step - loss: 0.3806 - acc: 0.9310 - val_loss: 0.6
024 - val_acc: 0.7388

Epoch 00084: val_acc did not improve from 0.79688
Epoch 85/100
300/300 [==============================] - 127s 425ms/step - loss: 0.3845 - acc: 0.9290 - val_loss: 0.6
155 - val_acc: 0.7544

Epoch 00085: val_acc did not improve from 0.79688
Epoch 86/100
300/300 [==============================] - 127s 424ms/step - loss: 0.3814 - acc: 0.9304 - val_loss: 0.5
907 - val_acc: 0.7631

Epoch 00086: val_acc did not improve from 0.79688
Epoch 87/100
300/300 [==============================] - 128s 425ms/step - loss: 0.3812 - acc: 0.9354 - val_loss: 0.5
776 - val_acc: 0.7669

Epoch 00087: val_acc did not improve from 0.79688
Epoch 88/100
300/300 [==============================] - 127s 425ms/step - loss: 0.3763 - acc: 0.9350 - val_loss: 0.6
244 - val_acc: 0.7412

Epoch 00088: val_acc did not improve from 0.79688
Epoch 89/100
300/300 [==============================] - 128s 425ms/step - loss: 0.3823 - acc: 0.9279 - val_loss: 0.6
221 - val_acc: 0.7288

Epoch 00089: val_acc did not improve from 0.79688
Epoch 90/100
300/300 [==============================] - 127s 425ms/step - loss: 0.3678 - acc: 0.9425 - val_loss: 0.6
074 - val_acc: 0.7356

Epoch 00090: val_acc did not improve from 0.79688
Epoch 91/100
300/300 [==============================] - 128s 428ms/step - loss: 0.3742 - acc: 0.9373 - val_loss: 0.6
088 - val_acc: 0.7525

Epoch 00091: val_acc did not improve from 0.79688
Epoch 92/100
300/300 [==============================] - 128s 428ms/step - loss: 0.3798 - acc: 0.9342 - val_loss: 0.5
930 - val_acc: 0.7494

Epoch 00092: val_acc did not improve from 0.79688
Epoch 93/100
300/300 [==============================] - 128s 427ms/step - loss: 0.3753 - acc: 0.9329 - val_loss: 0.5
945 - val_acc: 0.7538

Epoch 00093: val_acc did not improve from 0.79688
Epoch 94/100
300/300 [==============================] - 128s 427ms/step - loss: 0.3709 - acc: 0.9377 - val_loss: 0.5
990 - val_acc: 0.7575

Epoch 00094: val_acc did not improve from 0.79688
Epoch 95/100
300/300 [==============================] - 128s 426ms/step - loss: 0.3709 - acc: 0.9367 - val_loss: 0.6
093 - val_acc: 0.7525

Epoch 00095: val_acc did not improve from 0.79688
Epoch 96/100
300/300 [==============================] - 128s 428ms/step - loss: 0.3767 - acc: 0.9340 - val_loss: 0.5
994 - val_acc: 0.7575

Epoch 00096: val_acc did not improve from 0.79688
Epoch 97/100
300/300 [==============================] - 128s 428ms/step - loss: 0.3719 - acc: 0.9348 - val_loss: 0.6
103 - val_acc: 0.7550
```

```
Epoch 00097: val_acc did not improve from 0.79688

Epoch 00097: ReduceLROnPlateau reducing learning rate to 9.999999974752428e-08.
Epoch 98/100
300/300 [==============================] - 128s 427ms/step - loss: 0.3693 - acc: 0.9427 - val_loss: 0.6
068 - val_acc: 0.7544

Epoch 00098: val_acc did not improve from 0.79688
Epoch 99/100
300/300 [==============================] - 128s 427ms/step - loss: 0.3610 - acc: 0.9417 - val_loss: 0.6
356 - val_acc: 0.7362

Epoch 00099: val_acc did not improve from 0.79688
Epoch 100/100
300/300 [==============================] - 129s 429ms/step - loss: 0.3630 - acc: 0.9413 - val_loss: 0.6
205 - val_acc: 0.7469

Epoch 00100: val_acc did not improve from 0.79688
```

Out[0]:

```
<keras.callbacks.History at 0x7fb86d83de10>
```

At epoch 57 we can see Train accuracy is 87 and validation accuracy is approx 80 so our model was not overfit that time so considering epoch 57 for trained model

Epoch 57/100 => acc: 0.8700 - val_acc: 0.7969 --> saving model to ./model1.h5

In [0]:

```

```

In [0]:

```
!ls
```

```
drive          sample_submission.csv test.py
image          setup.py   test.zip
keras_vggface  test   tools
keras-vggface  test-faces.zip  train
LICENSE.txt    test-private-faces.zip train-faces.zip
MANIFEST.in    test-private-labels.zip train_relationships.csv
model1.h5      test-private-lists.zip train.zip
README.md      test-public-faces.zip visualize
sample_data    test-public-lists.zip
```

In [0]:

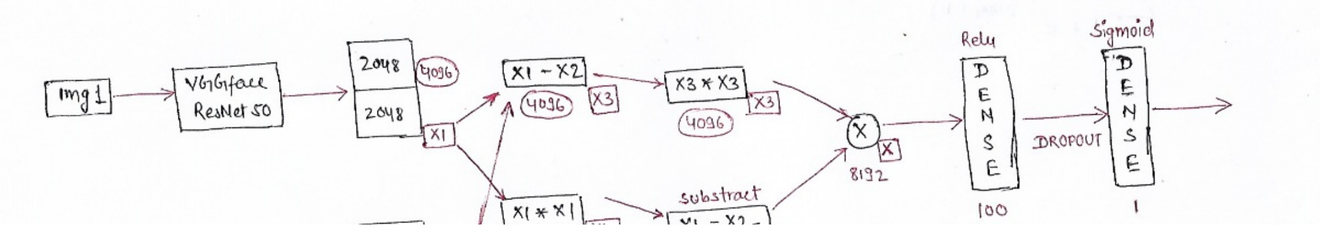
```
!pwd
```

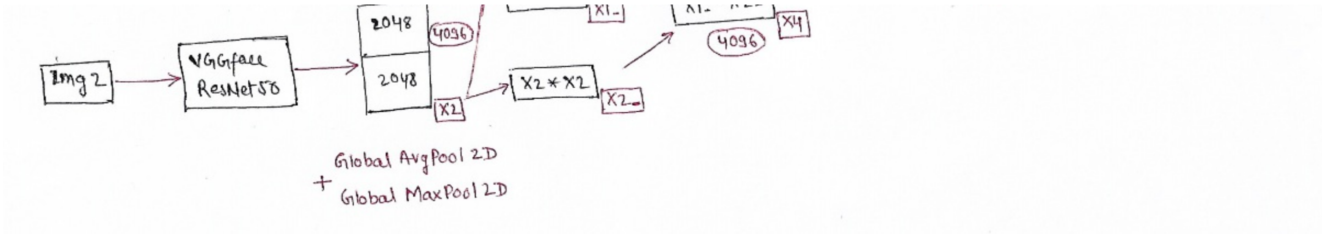
```
/content
```

In [0]:

```
!mv model1.h5 ./drive/My\ Drive/COLAB_FILES/
```

MODEL 2





In [0]:

```python
def gen(list_tuples, person_to_images_map, batch_size=16):
    ppl = list(person_to_images_map.keys())
    while True:
        batch_tuples = sample(list_tuples, batch_size // 2)
        labels = [1] * len(batch_tuples)
        while len(batch_tuples) < batch_size:
            p1 = choice(ppl)
            p2 = choice(ppl)

            if p1 != p2 and (p1, p2) not in list_tuples and (p2, p1) not in list_tuples:
                batch_tuples.append((p1, p2))
                labels.append(0)

        for x in batch_tuples:
            if not len(person_to_images_map[x[0]]):
                print(x[0])

        X1 = [choice(person_to_images_map[x[0]]) for x in batch_tuples]
        X1 = np.array([read_img(x, 197) for x in X1])

        X2 = [choice(person_to_images_map[x[1]]) for x in batch_tuples]
        X2 = np.array([read_img(x, 197) for x in X2])

        yield [X1, X2], labels
```

In [0]:

```python
def baseline_model2():
    input_1 = Input(shape=(197, 197, 3))
    input_2 = Input(shape=(197, 197, 3))

    base_model2 = VGGFace(model='resnet50', include_top=False)

    for x in base_model2.layers[:-3]:
        x.trainable = True

    x1 = base_model2(input_1)
    x2 = base_model2(input_2)

    x1 = Concatenate(axis=-1)([GlobalMaxPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])

    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])

    x1_ = Multiply()([x1, x1])
    x2_ = Multiply()([x2, x2])
    x4 = Subtract()([x1_, x2_])
    x = Concatenate(axis=-1)([x4, x3])
    x = Dropout(0.3)(x)
    x = Dense(100, activation="relu", activity_regularizer=l1(0.001))(x)
    x = Dropout(0.1)(x)
    out = Dense(1, activation="sigmoid")(x)

    model2 = Model([input_1, input_2], out)

    model2.compile(loss="binary_crossentropy", metrics=['acc', auc], optimizer=Adam(0.00001))

    model2.summary()

    return model2
```

In [0]:

```
file_path2 = "./model2.h5"

checkpoint2 = ModelCheckpoint(file_path2, monitor='val_acc', verbose=1, save_best_only=True, mode='max'
)

reduce_on_plateau2 = ReduceLROnPlateau(monitor="val_acc", mode="max", factor=0.1, patience=20, verbose=
1)

callbacks_list2 = [checkpoint2, reduce_on_plateau2]

model2 = baseline_model2()
#model.load_weights(file_path)
```

```
Model: "model_4"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_10 (InputLayer)           (None, 197, 197, 3)  0
__________________________________________________________________________________________________
input_11 (InputLayer)           (None, 197, 197, 3)  0
__________________________________________________________________________________________________
vggface_resnet50 (Model)        multiple             23561152    input_10[0][0]
                                                                 input_11[0][0]
__________________________________________________________________________________________________
global_max_pooling2d_7 (GlobalM (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_average_pooling2d_7 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_max_pooling2d_8 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
global_average_pooling2d_8 (Glo (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
concatenate_10 (Concatenate)    (None, 4096)         0           global_max_pooling2d_7[0][0]
                                                                 global_average_pooling2d_7[0][0]
__________________________________________________________________________________________________
concatenate_11 (Concatenate)    (None, 4096)         0           global_max_pooling2d_8[0][0]
                                                                 global_average_pooling2d_8[0][0]
__________________________________________________________________________________________________
multiply_11 (Multiply)          (None, 4096)         0           concatenate_10[0][0]
                                                                 concatenate_10[0][0]
__________________________________________________________________________________________________
multiply_12 (Multiply)          (None, 4096)         0           concatenate_11[0][0]
                                                                 concatenate_11[0][0]
__________________________________________________________________________________________________
subtract_7 (Subtract)           (None, 4096)         0           concatenate_10[0][0]
                                                                 concatenate_11[0][0]
__________________________________________________________________________________________________
subtract_8 (Subtract)           (None, 4096)         0           multiply_11[0][0]
                                                                 multiply_12[0][0]
__________________________________________________________________________________________________
multiply_10 (Multiply)          (None, 4096)         0           subtract_7[0][0]
                                                                 subtract_7[0][0]
__________________________________________________________________________________________________
concatenate_12 (Concatenate)    (None, 8192)         0           subtract_8[0][0]
                                                                 multiply_10[0][0]
__________________________________________________________________________________________________
dropout_7 (Dropout)             (None, 8192)         0           concatenate_12[0][0]
__________________________________________________________________________________________________
dense_8 (Dense)                 (None, 100)          819300      dropout_7[0][0]
__________________________________________________________________________________________________
dropout_8 (Dropout)             (None, 100)          0           dense_8[0][0]
__________________________________________________________________________________________________
dense_9 (Dense)                 (None, 1)            101         dropout_8[0][0]
==================================================================================================
Total params: 24,380,553
Trainable params: 24,327,433
Non-trainable params: 53,120
__________________________________________________________________________________________________
```

In [0]:

```
model2.fit_generator(gen(train, train_person_to_images_map, batch_size=16), use_multiprocessing=True,
                    validation_data=gen(val, val_person_to_images_map, batch_size=16), epochs=100, verb
                    validation_data=gen(val, val_person_to_images_map, batch_size=16), epochs=100, verb
ose=2,
                    workers=4, callbacks=callbacks_list2, steps_per_epoch=300, validation_steps=100)
```

```
Epoch 1/100
 - 131s - loss: 0.5935 - acc: 0.7660 - auc: 0.8304 - val_loss: 0.6011 - val_acc: 0.7388 - val_auc: 0.81
20

Epoch 00001: val_acc did not improve from 0.75750
Epoch 2/100
 - 127s - loss: 0.5863 - acc: 0.7702 - auc: 0.8344 - val_loss: 0.5661 - val_acc: 0.7969 - val_auc: 0.86
19

Epoch 00002: val_acc improved from 0.75750 to 0.79688, saving model to ./model2.h5
Epoch 3/100
 - 128s - loss: 0.5790 - acc: 0.7765 - auc: 0.8443 - val_loss: 0.6017 - val_acc: 0.7425 - val_auc: 0.81
69

Epoch 00003: val_acc did not improve from 0.79688
Epoch 4/100
 - 127s - loss: 0.5678 - acc: 0.7865 - auc: 0.8539 - val_loss: 0.5902 - val_acc: 0.7625 - val_auc: 0.82
67

Epoch 00004: val_acc did not improve from 0.79688
Epoch 5/100
 - 128s - loss: 0.5635 - acc: 0.7831 - auc: 0.8561 - val_loss: 0.5648 - val_acc: 0.7750 - val_auc: 0.85
41

Epoch 00005: val_acc did not improve from 0.79688
Epoch 6/100
 - 128s - loss: 0.5681 - acc: 0.7850 - auc: 0.8531 - val_loss: 0.5731 - val_acc: 0.7825 - val_auc: 0.84
64

Epoch 00006: val_acc did not improve from 0.79688
Epoch 7/100
 - 128s - loss: 0.5463 - acc: 0.7937 - auc: 0.8680 - val_loss: 0.5827 - val_acc: 0.7625 - val_auc: 0.83
36

Epoch 00007: val_acc did not improve from 0.79688
Epoch 8/100
 - 128s - loss: 0.5502 - acc: 0.7929 - auc: 0.8708 - val_loss: 0.5677 - val_acc: 0.7775 - val_auc: 0.84
94

Epoch 00008: val_acc did not improve from 0.79688
Epoch 9/100
 - 128s - loss: 0.5383 - acc: 0.8048 - auc: 0.8799 - val_loss: 0.5770 - val_acc: 0.7700 - val_auc: 0.84
02

Epoch 00009: val_acc did not improve from 0.79688
Epoch 10/100
 - 128s - loss: 0.5397 - acc: 0.8079 - auc: 0.8742 - val_loss: 0.5908 - val_acc: 0.7494 - val_auc: 0.82
34

Epoch 00010: val_acc did not improve from 0.79688
Epoch 11/100
 - 128s - loss: 0.5333 - acc: 0.8058 - auc: 0.8818 - val_loss: 0.5615 - val_acc: 0.7712 - val_auc: 0.85
25

Epoch 00011: val_acc did not improve from 0.79688
Epoch 12/100
 - 128s - loss: 0.5182 - acc: 0.8204 - auc: 0.8970 - val_loss: 0.5701 - val_acc: 0.7762 - val_auc: 0.83
97

Epoch 00012: val_acc did not improve from 0.79688
Epoch 13/100
 - 128s - loss: 0.5251 - acc: 0.8181 - auc: 0.8863 - val_loss: 0.5816 - val_acc: 0.7688 - val_auc: 0.83
53

Epoch 00013: val_acc did not improve from 0.79688
Epoch 14/100
 - 128s - loss: 0.5172 - acc: 0.8269 - auc: 0.8915 - val_loss: 0.5458 - val_acc: 0.7919 - val_auc: 0.87
38

Epoch 00014: val_acc did not improve from 0.79688
Epoch 15/100
 - 128s - loss: 0.5041 - acc: 0.8329 - auc: 0.9060 - val_loss: 0.5787 - val_acc: 0.7550 - val_auc: 0.82
86
```

```
00

Epoch 00015: val_acc did not improve from 0.79688
Epoch 16/100
 - 128s - loss: 0.5076 - acc: 0.8252 - auc: 0.8968 - val_loss: 0.5635 - val_acc: 0.7750 - val_auc: 0.85
77

Epoch 00016: val_acc did not improve from 0.79688
Epoch 17/100
 - 128s - loss: 0.5139 - acc: 0.8229 - auc: 0.8924 - val_loss: 0.5559 - val_acc: 0.7775 - val_auc: 0.84
80

Epoch 00017: val_acc did not improve from 0.79688
Epoch 18/100
 - 128s - loss: 0.5077 - acc: 0.8304 - auc: 0.8991 - val_loss: 0.5657 - val_acc: 0.7762 - val_auc: 0.84
70

Epoch 00018: val_acc did not improve from 0.79688
Epoch 19/100
 - 128s - loss: 0.4957 - acc: 0.8402 - auc: 0.9071 - val_loss: 0.5454 - val_acc: 0.7819 - val_auc: 0.85
48

Epoch 00019: val_acc did not improve from 0.79688
Epoch 20/100
 - 128s - loss: 0.4887 - acc: 0.8415 - auc: 0.9151 - val_loss: 0.5543 - val_acc: 0.7800 - val_auc: 0.86
00

Epoch 00020: val_acc did not improve from 0.79688
Epoch 21/100
 - 127s - loss: 0.4811 - acc: 0.8508 - auc: 0.9170 - val_loss: 0.5523 - val_acc: 0.7863 - val_auc: 0.85
94

Epoch 00021: val_acc did not improve from 0.79688
Epoch 22/100
 - 127s - loss: 0.4799 - acc: 0.8515 - auc: 0.9171 - val_loss: 0.5669 - val_acc: 0.7744 - val_auc: 0.84
44

Epoch 00022: val_acc did not improve from 0.79688

Epoch 00022: ReduceLROnPlateau reducing learning rate to 9.999999747378752e-07.
Epoch 23/100
 - 128s - loss: 0.4779 - acc: 0.8483 - auc: 0.9149 - val_loss: 0.5610 - val_acc: 0.7812 - val_auc: 0.85
09

Epoch 00023: val_acc did not improve from 0.79688
Epoch 24/100
 - 128s - loss: 0.4679 - acc: 0.8590 - auc: 0.9250 - val_loss: 0.5595 - val_acc: 0.7794 - val_auc: 0.85
47

Epoch 00024: val_acc did not improve from 0.79688
Epoch 25/100
 - 128s - loss: 0.4695 - acc: 0.8546 - auc: 0.9229 - val_loss: 0.5638 - val_acc: 0.7762 - val_auc: 0.85
50

Epoch 00025: val_acc did not improve from 0.79688
Epoch 26/100
 - 128s - loss: 0.4639 - acc: 0.8612 - auc: 0.9263 - val_loss: 0.5393 - val_acc: 0.8019 - val_auc: 0.87
03

Epoch 00026: val_acc improved from 0.79688 to 0.80188, saving model to ./model2.h5
Epoch 27/100
 - 128s - loss: 0.4583 - acc: 0.8633 - auc: 0.9292 - val_loss: 0.5679 - val_acc: 0.7769 - val_auc: 0.85
03

Epoch 00027: val_acc did not improve from 0.80188
Epoch 28/100
 - 128s - loss: 0.4534 - acc: 0.8696 - auc: 0.9351 - val_loss: 0.5590 - val_acc: 0.7850 - val_auc: 0.85
58

Epoch 00028: val_acc did not improve from 0.80188
Epoch 29/100
 - 128s - loss: 0.4571 - acc: 0.8610 - auc: 0.9319 - val_loss: 0.5296 - val_acc: 0.8031 - val_auc: 0.87
77

Epoch 00029: val_acc improved from 0.80188 to 0.80312, saving model to ./model2.h5
Epoch 30/100
 - 128s - loss: 0.4555 - acc: 0.8698 - auc: 0.9312 - val_loss: 0.5384 - val_acc: 0.8056 - val_auc: 0.87
42
```

```
42

Epoch 00030: val_acc improved from 0.80312 to 0.80563, saving model to ./model2.h5
Epoch 31/100
 - 128s - loss: 0.4538 - acc: 0.8675 - auc: 0.9333 - val_loss: 0.5303 - val_acc: 0.7994 - val_auc: 0.87
69

Epoch 00031: val_acc did not improve from 0.80563
Epoch 32/100
 - 128s - loss: 0.4534 - acc: 0.8706 - auc: 0.9331 - val_loss: 0.5415 - val_acc: 0.7925 - val_auc: 0.87
23

Epoch 00032: val_acc did not improve from 0.80563
Epoch 33/100
 - 128s - loss: 0.4434 - acc: 0.8813 - auc: 0.9397 - val_loss: 0.5376 - val_acc: 0.7963 - val_auc: 0.87
38

Epoch 00033: val_acc did not improve from 0.80563
Epoch 34/100
 - 128s - loss: 0.4462 - acc: 0.8771 - auc: 0.9389 - val_loss: 0.5681 - val_acc: 0.7756 - val_auc: 0.85
70

Epoch 00034: val_acc did not improve from 0.80563
Epoch 35/100
 - 128s - loss: 0.4433 - acc: 0.8794 - auc: 0.9388 - val_loss: 0.5457 - val_acc: 0.7931 - val_auc: 0.86
66

Epoch 00035: val_acc did not improve from 0.80563
Epoch 36/100
 - 128s - loss: 0.4452 - acc: 0.8723 - auc: 0.9400 - val_loss: 0.5154 - val_acc: 0.8175 - val_auc: 0.89
08

Epoch 00036: val_acc improved from 0.80563 to 0.81750, saving model to ./model2.h5
Epoch 37/100
 - 128s - loss: 0.4395 - acc: 0.8860 - auc: 0.9417 - val_loss: 0.5590 - val_acc: 0.7806 - val_auc: 0.86
00

Epoch 00037: val_acc did not improve from 0.81750
Epoch 38/100
 - 128s - loss: 0.4410 - acc: 0.8758 - auc: 0.9421 - val_loss: 0.5590 - val_acc: 0.7781 - val_auc: 0.85
55

Epoch 00038: val_acc did not improve from 0.81750
Epoch 39/100
 - 128s - loss: 0.4498 - acc: 0.8673 - auc: 0.9369 - val_loss: 0.5588 - val_acc: 0.7812 - val_auc: 0.86
17

Epoch 00039: val_acc did not improve from 0.81750
Epoch 40/100
 - 128s - loss: 0.4413 - acc: 0.8792 - auc: 0.9404 - val_loss: 0.5627 - val_acc: 0.7656 - val_auc: 0.85
56

Epoch 00040: val_acc did not improve from 0.81750
Epoch 41/100
 - 128s - loss: 0.4442 - acc: 0.8765 - auc: 0.9399 - val_loss: 0.5553 - val_acc: 0.7831 - val_auc: 0.85
86

Epoch 00041: val_acc did not improve from 0.81750
Epoch 42/100
 - 128s - loss: 0.4325 - acc: 0.8802 - auc: 0.9485 - val_loss: 0.5674 - val_acc: 0.7731 - val_auc: 0.85
48

Epoch 00042: val_acc did not improve from 0.81750
Epoch 43/100
 - 128s - loss: 0.4365 - acc: 0.8825 - auc: 0.9454 - val_loss: 0.5504 - val_acc: 0.7769 - val_auc: 0.86
56

Epoch 00043: val_acc did not improve from 0.81750
Epoch 44/100
 - 128s - loss: 0.4298 - acc: 0.8883 - auc: 0.9468 - val_loss: 0.5667 - val_acc: 0.7762 - val_auc: 0.86
28

Epoch 00044: val_acc did not improve from 0.81750
Epoch 45/100
 - 128s - loss: 0.4373 - acc: 0.8798 - auc: 0.9416 - val_loss: 0.5355 - val_acc: 0.8044 - val_auc: 0.87
69

Epoch 00045: val_acc did not improve from 0.81750
Epoch 00045: val_acc did not improve from 0.81750
Epoch 46/100
 - 128s - loss: 0.4371 - acc: 0.8808 - auc: 0.9424 - val_loss: 0.5636 - val_acc: 0.7775 - val_auc: 0.86
41

Epoch 00046: val_acc did not improve from 0.81750
Epoch 47/100
 - 128s - loss: 0.4344 - acc: 0.8823 - auc: 0.9432 - val_loss: 0.5785 - val_acc: 0.7562 - val_auc: 0.84
16

Epoch 00047: val_acc did not improve from 0.81750
Epoch 48/100
 - 128s - loss: 0.4270 - acc: 0.8915 - auc: 0.9486 - val_loss: 0.5528 - val_acc: 0.7825 - val_auc: 0.86
36

Epoch 00048: val_acc did not improve from 0.81750
Epoch 49/100
 - 128s - loss: 0.4401 - acc: 0.8767 - auc: 0.9394 - val_loss: 0.5397 - val_acc: 0.7944 - val_auc: 0.87
14

Epoch 00049: val_acc did not improve from 0.81750
Epoch 50/100
 - 128s - loss: 0.4367 - acc: 0.8804 - auc: 0.9443 - val_loss: 0.5570 - val_acc: 0.7806 - val_auc: 0.86
33

Epoch 00050: val_acc did not improve from 0.81750
Epoch 51/100
 - 128s - loss: 0.4272 - acc: 0.8850 - auc: 0.9469 - val_loss: 0.5656 - val_acc: 0.7750 - val_auc: 0.85
05

Epoch 00051: val_acc did not improve from 0.81750
Epoch 52/100
 - 128s - loss: 0.4259 - acc: 0.8883 - auc: 0.9466 - val_loss: 0.5543 - val_acc: 0.7806 - val_auc: 0.86
28

Epoch 00052: val_acc did not improve from 0.81750
Epoch 53/100
 - 128s - loss: 0.4374 - acc: 0.8817 - auc: 0.9431 - val_loss: 0.5646 - val_acc: 0.7719 - val_auc: 0.86
06

Epoch 00053: val_acc did not improve from 0.81750
Epoch 54/100
 - 128s - loss: 0.4363 - acc: 0.8792 - auc: 0.9426 - val_loss: 0.5645 - val_acc: 0.7831 - val_auc: 0.85
38

Epoch 00054: val_acc did not improve from 0.81750
Epoch 55/100
 - 128s - loss: 0.4275 - acc: 0.8840 - auc: 0.9452 - val_loss: 0.5621 - val_acc: 0.7750 - val_auc: 0.86
11

Epoch 00055: val_acc did not improve from 0.81750
Epoch 56/100
 - 128s - loss: 0.4262 - acc: 0.8915 - auc: 0.9491 - val_loss: 0.5851 - val_acc: 0.7662 - val_auc: 0.84
73

Epoch 00056: val_acc did not improve from 0.81750

Epoch 00056: ReduceLROnPlateau reducing learning rate to 9.999999974752428e-08.
Epoch 57/100
 - 128s - loss: 0.4261 - acc: 0.8917 - auc: 0.9456 - val_loss: 0.5582 - val_acc: 0.7831 - val_auc: 0.85
83

Epoch 00057: val_acc did not improve from 0.81750
Epoch 58/100
 - 128s - loss: 0.4240 - acc: 0.8867 - auc: 0.9497 - val_loss: 0.5648 - val_acc: 0.7906 - val_auc: 0.85
98

Epoch 00058: val_acc did not improve from 0.81750
Epoch 59/100
 - 128s - loss: 0.4344 - acc: 0.8829 - auc: 0.9419 - val_loss: 0.5409 - val_acc: 0.7969 - val_auc: 0.87
45

Epoch 00059: val_acc did not improve from 0.81750
Epoch 60/100
 - 128s - loss: 0.4274 - acc: 0.8844 - auc: 0.9485 - val_loss: 0.5798 - val_acc: 0.7681 - val_auc: 0.85
69

Epoch 00060: val_acc did not improve from 0.81750
Epoch 00060: val_acc did not improve from 0.81750
Epoch 61/100
 - 128s - loss: 0.4307 - acc: 0.8821 - auc: 0.9429 - val_loss: 0.5783 - val_acc: 0.7700 - val_auc: 0.84
83

Epoch 00061: val_acc did not improve from 0.81750
Epoch 62/100
 - 128s - loss: 0.4271 - acc: 0.8844 - auc: 0.9445 - val_loss: 0.5727 - val_acc: 0.7681 - val_auc: 0.84
56

Epoch 00062: val_acc did not improve from 0.81750
Epoch 63/100
 - 127s - loss: 0.4310 - acc: 0.8883 - auc: 0.9463 - val_loss: 0.5416 - val_acc: 0.8019 - val_auc: 0.87
86

Epoch 00063: val_acc did not improve from 0.81750
Epoch 64/100
 - 128s - loss: 0.4259 - acc: 0.8888 - auc: 0.9477 - val_loss: 0.5508 - val_acc: 0.7931 - val_auc: 0.86
42

Epoch 00064: val_acc did not improve from 0.81750
Epoch 65/100
 - 128s - loss: 0.4239 - acc: 0.8856 - auc: 0.9504 - val_loss: 0.5676 - val_acc: 0.7881 - val_auc: 0.85
52

Epoch 00065: val_acc did not improve from 0.81750
Epoch 66/100
 - 128s - loss: 0.4257 - acc: 0.8879 - auc: 0.9461 - val_loss: 0.5324 - val_acc: 0.7950 - val_auc: 0.88
11

Epoch 00066: val_acc did not improve from 0.81750
Epoch 67/100
 - 128s - loss: 0.4180 - acc: 0.8952 - auc: 0.9560 - val_loss: 0.5912 - val_acc: 0.7588 - val_auc: 0.84
38

Epoch 00067: val_acc did not improve from 0.81750
Epoch 68/100
 - 128s - loss: 0.4140 - acc: 0.8967 - auc: 0.9528 - val_loss: 0.5730 - val_acc: 0.7562 - val_auc: 0.84
70

Epoch 00068: val_acc did not improve from 0.81750
Epoch 69/100
 - 128s - loss: 0.4222 - acc: 0.8915 - auc: 0.9493 - val_loss: 0.5612 - val_acc: 0.7794 - val_auc: 0.86
81

Epoch 00069: val_acc did not improve from 0.81750
Epoch 70/100
 - 128s - loss: 0.4263 - acc: 0.8865 - auc: 0.9474 - val_loss: 0.5597 - val_acc: 0.7694 - val_auc: 0.86
22

Epoch 00070: val_acc did not improve from 0.81750
Epoch 71/100
 - 127s - loss: 0.4174 - acc: 0.8975 - auc: 0.9543 - val_loss: 0.5582 - val_acc: 0.7800 - val_auc: 0.86
45

Epoch 00071: val_acc did not improve from 0.81750
Epoch 72/100
 - 128s - loss: 0.4329 - acc: 0.8848 - auc: 0.9459 - val_loss: 0.5692 - val_acc: 0.7788 - val_auc: 0.85
92

Epoch 00072: val_acc did not improve from 0.81750
Epoch 73/100
 - 127s - loss: 0.4312 - acc: 0.8894 - auc: 0.9425 - val_loss: 0.5708 - val_acc: 0.7750 - val_auc: 0.84
77

Epoch 00073: val_acc did not improve from 0.81750
Epoch 74/100
 - 128s - loss: 0.4319 - acc: 0.8835 - auc: 0.9423 - val_loss: 0.5387 - val_acc: 0.8037 - val_auc: 0.88
52

Epoch 00074: val_acc did not improve from 0.81750
Epoch 75/100
 - 128s - loss: 0.4180 - acc: 0.8938 - auc: 0.9499 - val_loss: 0.5596 - val_acc: 0.7869 - val_auc: 0.86
58

Epoch 00075: val_acc did not improve from 0.81750
Epoch 76/100
 - 128s - loss: 0.4191 - acc: 0.8948 - auc: 0.9526 - val_loss: 0.5600 - val_acc: 0.7800 - val_auc: 0.87
 - 128s - loss: 0.4191 - acc: 0.8948 - auc: 0.9526 - val_loss: 0.5600 - val_acc: 0.7800 - val_auc: 0.87
03

Epoch 00076: val_acc did not improve from 0.81750

Epoch 00076: ReduceLROnPlateau reducing learning rate to 1.0000000116860975e-08.
Epoch 77/100
 - 128s - loss: 0.4243 - acc: 0.8919 - auc: 0.9497 - val_loss: 0.5819 - val_acc: 0.7569 - val_auc: 0.84
27

Epoch 00077: val_acc did not improve from 0.81750
Epoch 78/100
 - 128s - loss: 0.4219 - acc: 0.8881 - auc: 0.9498 - val_loss: 0.5611 - val_acc: 0.7662 - val_auc: 0.86
41

Epoch 00078: val_acc did not improve from 0.81750
Epoch 79/100
 - 127s - loss: 0.4185 - acc: 0.8904 - auc: 0.9533 - val_loss: 0.5641 - val_acc: 0.7744 - val_auc: 0.86
09

Epoch 00079: val_acc did not improve from 0.81750
Epoch 80/100
 - 128s - loss: 0.4230 - acc: 0.8867 - auc: 0.9501 - val_loss: 0.5682 - val_acc: 0.7762 - val_auc: 0.86
56

Epoch 00080: val_acc did not improve from 0.81750
Epoch 81/100
 - 128s - loss: 0.4331 - acc: 0.8835 - auc: 0.9420 - val_loss: 0.5668 - val_acc: 0.7738 - val_auc: 0.85
19

Epoch 00081: val_acc did not improve from 0.81750
Epoch 82/100
 - 128s - loss: 0.4207 - acc: 0.8942 - auc: 0.9494 - val_loss: 0.5576 - val_acc: 0.7837 - val_auc: 0.86
61

Epoch 00082: val_acc did not improve from 0.81750
Epoch 83/100
 - 127s - loss: 0.4220 - acc: 0.8910 - auc: 0.9513 - val_loss: 0.5615 - val_acc: 0.7750 - val_auc: 0.85
69

Epoch 00083: val_acc did not improve from 0.81750
Epoch 84/100
 - 128s - loss: 0.4176 - acc: 0.8948 - auc: 0.9513 - val_loss: 0.5678 - val_acc: 0.7594 - val_auc: 0.86
05

Epoch 00084: val_acc did not improve from 0.81750
Epoch 85/100
 - 127s - loss: 0.4196 - acc: 0.8923 - auc: 0.9497 - val_loss: 0.5681 - val_acc: 0.7662 - val_auc: 0.85
31

Epoch 00085: val_acc did not improve from 0.81750
Epoch 86/100
 - 127s - loss: 0.4268 - acc: 0.8894 - auc: 0.9470 - val_loss: 0.5839 - val_acc: 0.7669 - val_auc: 0.84
14

Epoch 00086: val_acc did not improve from 0.81750
Epoch 87/100
 - 127s - loss: 0.4263 - acc: 0.8904 - auc: 0.9483 - val_loss: 0.5694 - val_acc: 0.7731 - val_auc: 0.86
23

Epoch 00087: val_acc did not improve from 0.81750
Epoch 88/100
 - 127s - loss: 0.4236 - acc: 0.8890 - auc: 0.9483 - val_loss: 0.5586 - val_acc: 0.7881 - val_auc: 0.86
28

Epoch 00088: val_acc did not improve from 0.81750
Epoch 89/100
 - 127s - loss: 0.4260 - acc: 0.8896 - auc: 0.9476 - val_loss: 0.5649 - val_acc: 0.7762 - val_auc: 0.85
30

Epoch 00089: val_acc did not improve from 0.81750
Epoch 90/100
 - 127s - loss: 0.4279 - acc: 0.8856 - auc: 0.9436 - val_loss: 0.5719 - val_acc: 0.7694 - val_auc: 0.85
95

Epoch 00090: val_acc did not improve from 0.81750
Epoch 91/100
 - 128s - loss: 0.4180 - acc: 0.8975 - auc: 0.9511 - val_loss: 0.5727 - val_acc: 0.7825 - val_auc: 0.86
 - 128s - loss: 0.4189 - acc: 0.8975 - auc: 0.9511 - val_loss: 0.5727 - val_acc: 0.7825 - val_auc: 0.86
23

Epoch 00091: val_acc did not improve from 0.81750
Epoch 92/100
 - 128s - loss: 0.4243 - acc: 0.8929 - auc: 0.9481 - val_loss: 0.5695 - val_acc: 0.7725 - val_auc: 0.85
34

Epoch 00092: val_acc did not improve from 0.81750
Epoch 93/100
 - 127s - loss: 0.4200 - acc: 0.8921 - auc: 0.9485 - val_loss: 0.5695 - val_acc: 0.7725 - val_auc: 0.85
66

Epoch 00093: val_acc did not improve from 0.81750
Epoch 94/100
 - 127s - loss: 0.4186 - acc: 0.8950 - auc: 0.9498 - val_loss: 0.5583 - val_acc: 0.7775 - val_auc: 0.86
56

Epoch 00094: val_acc did not improve from 0.81750
Epoch 95/100
 - 128s - loss: 0.4164 - acc: 0.8983 - auc: 0.9526 - val_loss: 0.5486 - val_acc: 0.7825 - val_auc: 0.87
55

Epoch 00095: val_acc did not improve from 0.81750
Epoch 96/100
 - 128s - loss: 0.4275 - acc: 0.8906 - auc: 0.9456 - val_loss: 0.5781 - val_acc: 0.7675 - val_auc: 0.84
52

Epoch 00096: val_acc did not improve from 0.81750

Epoch 00096: ReduceLROnPlateau reducing learning rate to 9.999999939225292e-10.
Epoch 97/100
 - 127s - loss: 0.4313 - acc: 0.8896 - auc: 0.9435 - val_loss: 0.5556 - val_acc: 0.7738 - val_auc: 0.86
41

Epoch 00097: val_acc did not improve from 0.81750
Epoch 98/100
 - 128s - loss: 0.4251 - acc: 0.8917 - auc: 0.9481 - val_loss: 0.5690 - val_acc: 0.7769 - val_auc: 0.85
42

Epoch 00098: val_acc did not improve from 0.81750
Epoch 99/100
 - 127s - loss: 0.4215 - acc: 0.8944 - auc: 0.9489 - val_loss: 0.5546 - val_acc: 0.7700 - val_auc: 0.86
50

Epoch 00099: val_acc did not improve from 0.81750
Epoch 100/100
 - 128s - loss: 0.4252 - acc: 0.8910 - auc: 0.9471 - val_loss: 0.5749 - val_acc: 0.7844 - val_auc: 0.85
08

Epoch 00100: val_acc did not improve from 0.81750
```

Out[0]:

```
<keras.callbacks.History at 0x7fb6193dc518>
```

At epoch 36 we can see Train accuracy is 87 and validation accuracy is approx 82 so considering epoch 36 for trained model

Epoch 36/100 => acc: 0.8723 - val_acc: 0.8175 --> saving model to ./model2.h5

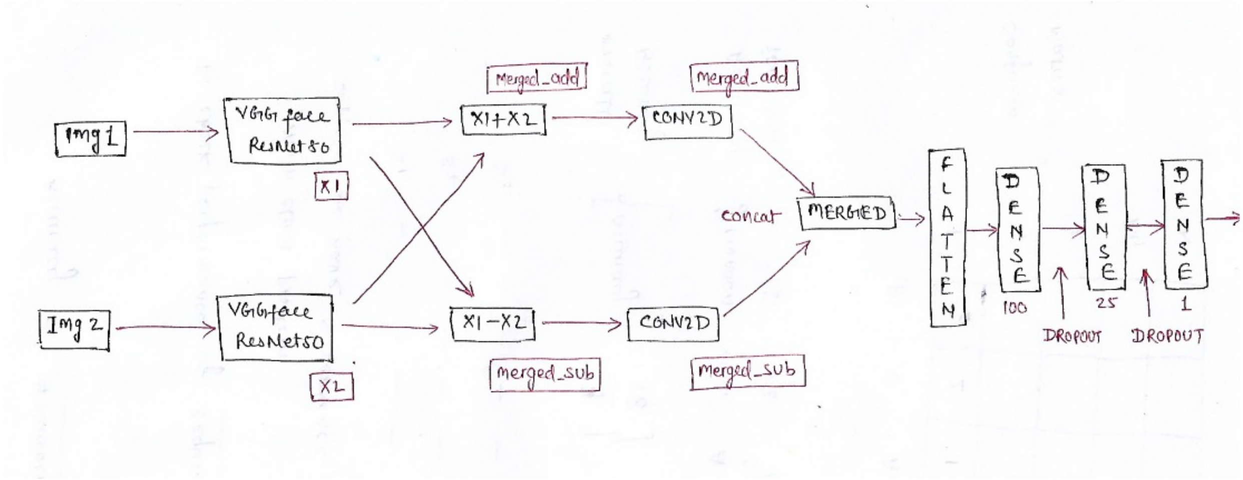
In [0]:

```
!mv model2.h5 ./drive/My\ Drive/COLAB_FILES/
```

In [0]:

```

```

MODEL 3



In [0]:

```
def read_img_cv(path):
    img = cv2.imread(path)
    img = np.array(img).astype(np.float)
    return preprocess_input(img, version=2)


def gen(list_tuples, person_to_images_map, batch_size=16):
    ppl = list(person_to_images_map.keys())
    while True:
        batch_tuples = sample(list_tuples, batch_size // 2)
        labels = [1] * len(batch_tuples)
        while len(batch_tuples) < batch_size:
            p1 = choice(ppl)
            p2 = choice(ppl)

            if p1 != p2 and (p1, p2) not in list_tuples and (p2, p1) not in list_tuples:
                batch_tuples.append((p1, p2))
                labels.append(0)

        for x in batch_tuples:
            if not len(person_to_images_map[x[0]]):
                print(x[0])

        X1 = [choice(person_to_images_map[x[0]]) for x in batch_tuples]
        X1 = np.array([read_img_cv(x) for x in X1])

        X2 = [choice(person_to_images_map[x[1]]) for x in batch_tuples]
        X2 = np.array([read_img_cv(x) for x in X2])

        yield [X1, X2], labels
```

In [0]:

In [0]:

```
def baseline_model3_1():
    input_1 = Input(shape=(224, 224, 3))
    input_2 = Input(shape=(224, 224, 3))

    base_model3 = VGGFace(model='resnet50', include_top=False)

    for layer in base_model3.layers[:-3]:
        layer.trainable = True

    x1 = base_model3(input_1)
    x2 = base_model3(input_2)

    merged_add = Add()([x1, x2])
    merged_sub = Subtract()([x1,x2])
    merged_add = Conv2D(100 , [1,1] )(merged_add)
    merged_sub = Conv2D(100 , [1,1] )(merged_sub)

    merged = Concatenate(axis=-1)([merged_add, merged_sub])

    merged = Flatten()(merged)
    merged = Dropout(0.2)(merged)
    merged = Dense(100, activation="relu", activity_regularizer=l1(0.001))(merged)
    merged = Dropout(0.2)(merged)
    merged = Dense(25, activation="relu", activity_regularizer=l1(0.001))(merged)
    merged = Dropout(0.2)(merged)
    out = Dense(1, activation="sigmoid")(merged)
    model3 = Model([input_1, input_2], out)
    model3.compile(loss="binary_crossentropy", metrics=['acc', auc], optimizer=Adam(0.00001))
    model3.summary()
    return model3
```

In [0]:

```
file_path3 = "./model3_1.h5"

checkpoint3 = ModelCheckpoint(file_path3, monitor='val_acc', verbose=1, save_best_only=True, mode='max'
)

reduce_on_plateau3 = ReduceLROnPlateau(monitor="val_acc", mode="max", factor=0.1, patience=20, verbose=
1)

callbacks_list3 = [checkpoint3, reduce_on_plateau3]

model3_1 = baseline_model3_1()
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:66:
The name tf.get_default_graph is deprecated. Please use tf.compat.v1.get_default_graph instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:541:
The name tf.placeholder is deprecated. Please use tf.compat.v1.placeholder instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4432
: The name tf.random_uniform is deprecated. Please use tf.random.uniform instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:190:
The name tf.get_default_session is deprecated. Please use tf.compat.v1.get_default_session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:197:
The name tf.ConfigProto is deprecated. Please use tf.compat.v1.ConfigProto instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:203:
The name tf.Session is deprecated. Please use tf.compat.v1.Session instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:207:
The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:216:
The name tf.is_variable_initialized is deprecated. Please use tf.compat.v1.is_variable_initialized inst
ead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:223:
The name tf.variables_initializer is deprecated. Please use tf.compat.v1.variables_initializer instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:2041
: The name tf.nn.fused_batch_norm is deprecated. Please use tf.compat.v1.nn.fused_batch_norm instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:148:
The name tf.placeholder_with_default is deprecated. Please use tf.compat.v1.placeholder_with_default in
stead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4267
: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:4271
: The name tf.nn.avg_pool is deprecated. Please use tf.nn.avg_pool2d instead.

Downloading data from https://github.com/rcmalli/keras-vggface/releases/download/v2.0/rcmalli_vggface_t
f_notop_resnet50.h5
94699520/94694792 [==============================] - 1s 0us/step
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:3733
: calling dropout (from tensorflow.python.ops.nn_ops) with keep_prob is deprecated and will be removed
in a future version.
Instructions for updating:
Please use `rate` instead of `keep_prob`. Rate should be set to `rate = 1 - keep_prob`.
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/optimizers.py:793: The name tf.tra
in.Optimizer is deprecated. Please use tf.compat.v1.train.Optimizer instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:3657
: The name tf.log is deprecated. Please use tf.math.log instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorflow_core/python/ops/nn_impl.py:18
3: where (from tensorflow.python.ops.array_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From <ipython-input-56-fb283a009ca9>:2: py_func (from tensorflow.python.ops.script_o
ps) is deprecated and will be removed in a future version.
Instructions for updating:
tf.py_func is deprecated in TF V2. Instead, there are two
    options available in V2.
    - tf.py_function takes a python function which manipulates tf eager
    tensors instead of numpy arrays. It's easy to convert a tf eager tensor to
    an ndarray (just call tensor.numpy()) but having access to eager tensors
    means `tf.py_function`s can use accelerators such as GPUs as well as
    being differentiable using a gradient tape.
    - tf.numpy_function maintains the semantics of the deprecated tf.py_func
    (it is not differentiable, and manipulates numpy arrays). It drops the
    stateful argument making all functions stateful.

Model: "model_1"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to                     
==================================================================================================
input_1 (InputLayer)            (None, 224, 224, 3)  0                                            
__________________________________________________________________________________________________
input_2 (InputLayer)            (None, 224, 224, 3)  0                                            
__________________________________________________________________________________________________
vggface_resnet50 (Model)        multiple             23561152    input_1[0][0]                    
                                                                 input_2[0][0]                    
__________________________________________________________________________________________________
add_17 (Add)                    (None, 1, 1, 2048)   0           vggface_resnet50[1][0]           
                                                                 vggface_resnet50[2][0]           
__________________________________________________________________________________________________
subtract_1 (Subtract)           (None, 1, 1, 2048)   0           vggface_resnet50[1][0]           
                                                                 vggface_resnet50[2][0]           
__________________________________________________________________________________________________
conv2d_1 (Conv2D)               (None, 1, 1, 100)    204900      add_17[0][0]                     
__________________________________________________________________________________________________
conv2d_2 (Conv2D)               (None, 1, 1, 100)    204900      subtract_1[0][0]                 
__________________________________________________________________________________________________
concatenate_1 (Concatenate)     (None, 1, 1, 200)    0           conv2d_1[0][0]                   
                                                                 conv2d_2[0][0]                   
__________________________________________________________________________________________________
flatten_1 (Flatten)             (None, 200)          0           concatenate_1[0][0]              
__________________________________________________________________________________________________
dropout_1 (Dropout)             (None, 200)          0           flatten_1[0][0]                  
__________________________________________________________________________________________________
dense_1 (Dense)                 (None, 100)          20100       dropout_1[0][0]                  
__________________________________________________________________________________________________
dropout_2 (Dropout)             (None, 100)          0           dense_1[0][0]                    
__________________________________________________________________________________________________
dense_2 (Dense)                 (None, 25)           2525        dropout_2[0][0]                  
__________________________________________________________________________________________________
dropout_3 (Dropout)             (None, 25)           0           dense_2[0][0]                    
__________________________________________________________________________________________________
dense_3 (Dense)                 (None, 1)            26          dropout_3[0][0]                  
==================================================================================================
Total params: 23,993,603
Trainable params: 23,940,483
Non-trainable params: 53,120
__________________________________________________________________________________________________
```

In [0]:

```
model3_1.fit_generator(gen(train, train_person_to_images_map, batch_size=16), use_multiprocessing=True,
                    validation data=gen(val, val person to images map, batch size=16), epochs=130, verb
ose=1,
                    workers = 4, callbacks=callbacks_list3, steps_per_epoch=300, validation_steps=100)
```

```
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1033
: The name tf.assign_add is deprecated. Please use tf.compat.v1.assign_add instead.

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/tensorflow_backend.py:1020
: The name tf.assign is deprecated. Please use tf.compat.v1.assign instead.

Epoch 1/130
300/300 [==============================] - 167s 557ms/step - loss: 2.7493 - acc: 0.5581 - auc: 0.5769 -
val_loss: 0.9779 - val_acc: 0.5856 - val_auc: 0.6389

Epoch 00001: val_acc improved from -inf to 0.58562, saving model to ./model3_1.h5
Epoch 2/130
300/300 [==============================] - 141s 471ms/step - loss: 0.9386 - acc: 0.5667 - auc: 0.6057 -
val_loss: 0.7455 - val_acc: 0.6406 - val_auc: 0.6764

Epoch 00002: val_acc improved from 0.58562 to 0.64062, saving model to ./model3_1.h5
Epoch 3/130
300/300 [==============================] - 141s 471ms/step - loss: 0.8034 - acc: 0.5658 - auc: 0.6184 -
val_loss: 0.7196 - val_acc: 0.6388 - val_auc: 0.6834

Epoch 00003: val_acc did not improve from 0.64062
Epoch 4/130
300/300 [==============================] - 141s 470ms/step - loss: 0.7548 - acc: 0.6019 - auc: 0.6596 -
val_loss: 0.7005 - val_acc: 0.6331 - val_auc: 0.6786

Epoch 00004: val_acc did not improve from 0.64062
Epoch 5/130
300/300 [==============================] - 140s 468ms/step - loss: 0.7354 - acc: 0.5975 - auc: 0.6608 -
val_loss: 0.6916 - val_acc: 0.6569 - val_auc: 0.7083

Epoch 00005: val_acc improved from 0.64062 to 0.65687, saving model to ./model3_1.h5
Epoch 6/130
300/300 [==============================] - 141s 469ms/step - loss: 0.7224 - acc: 0.6042 - auc: 0.6709 -
val_loss: 0.6980 - val_acc: 0.6319 - val_auc: 0.6741

Epoch 00006: val_acc did not improve from 0.65687
Epoch 7/130
300/300 [==============================] - 141s 469ms/step - loss: 0.7099 - acc: 0.6135 - auc: 0.6919 -
val_loss: 0.6787 - val_acc: 0.6694 - val_auc: 0.7093

Epoch 00007: val_acc improved from 0.65687 to 0.66938, saving model to ./model3_1.h5
Epoch 8/130
300/300 [==============================] - 141s 469ms/step - loss: 0.7008 - acc: 0.6417 - auc: 0.7135 -
val_loss: 0.6776 - val_acc: 0.6531 - val_auc: 0.7084

Epoch 00008: val_acc did not improve from 0.66938
Epoch 9/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6939 - acc: 0.6219 - auc: 0.6960 -
val_loss: 0.6759 - val_acc: 0.6400 - val_auc: 0.7149

Epoch 00009: val_acc did not improve from 0.66938
Epoch 10/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6892 - acc: 0.6315 - auc: 0.7015 -
val_loss: 0.6709 - val_acc: 0.6519 - val_auc: 0.7117

Epoch 00010: val_acc did not improve from 0.66938
Epoch 11/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6812 - acc: 0.6485 - auc: 0.7156 -
val_loss: 0.6600 - val_acc: 0.6875 - val_auc: 0.7231

Epoch 00011: val_acc improved from 0.66938 to 0.68750, saving model to ./model3_1.h5
Epoch 12/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6748 - acc: 0.6425 - auc: 0.7003 -
val_loss: 0.6660 - val_acc: 0.6438 - val_auc: 0.6925

Epoch 00012: val_acc did not improve from 0.68750
Epoch 13/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6744 - acc: 0.6633 - auc: 0.7197 -
val_loss: 0.6634 - val_acc: 0.6538 - val_auc: 0.6844

Epoch 00013: val_acc did not improve from 0.68750
Epoch 14/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6718 - acc: 0.6544 - auc: 0.7045 -
```

```
val_loss: 0.6577 - val_acc: 0.6787 - val_auc: 0.7026

Epoch 00014: val_acc did not improve from 0.68750
Epoch 15/130
300/300 [==============================] - 142s 472ms/step - loss: 0.6691 - acc: 0.6627 - auc: 0.7093 -
val_loss: 0.6635 - val_acc: 0.6525 - val_auc: 0.7084

Epoch 00015: val_acc did not improve from 0.68750
Epoch 16/130
300/300 [==============================] - 142s 473ms/step - loss: 0.6626 - acc: 0.6771 - auc: 0.7221 -
val_loss: 0.6535 - val_acc: 0.6800 - val_auc: 0.7254

Epoch 00016: val_acc did not improve from 0.68750
Epoch 17/130
300/300 [==============================] - 142s 473ms/step - loss: 0.6596 - acc: 0.6869 - auc: 0.7290 -
val_loss: 0.6638 - val_acc: 0.6431 - val_auc: 0.6730

Epoch 00017: val_acc did not improve from 0.68750
Epoch 18/130
300/300 [==============================] - 142s 472ms/step - loss: 0.6601 - acc: 0.6823 - auc: 0.7263 -
val_loss: 0.6744 - val_acc: 0.6106 - val_auc: 0.6707

Epoch 00018: val_acc did not improve from 0.68750
Epoch 19/130
300/300 [==============================] - 141s 470ms/step - loss: 0.6552 - acc: 0.6873 - auc: 0.7282 -
val_loss: 0.6600 - val_acc: 0.6544 - val_auc: 0.6785

Epoch 00019: val_acc did not improve from 0.68750
Epoch 20/130
300/300 [==============================] - 141s 470ms/step - loss: 0.6503 - acc: 0.6898 - auc: 0.7185 -
val_loss: 0.6676 - val_acc: 0.6481 - val_auc: 0.6751

Epoch 00020: val_acc did not improve from 0.68750
Epoch 21/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6525 - acc: 0.6927 - auc: 0.7267 -
val_loss: 0.6552 - val_acc: 0.6700 - val_auc: 0.7130

Epoch 00021: val_acc did not improve from 0.68750
Epoch 22/130
300/300 [==============================] - 141s 468ms/step - loss: 0.6486 - acc: 0.6944 - auc: 0.7268 -
val_loss: 0.6538 - val_acc: 0.6694 - val_auc: 0.6868

Epoch 00022: val_acc did not improve from 0.68750
Epoch 23/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6467 - acc: 0.7008 - auc: 0.7310 -
val_loss: 0.6724 - val_acc: 0.6275 - val_auc: 0.6394

Epoch 00023: val_acc did not improve from 0.68750
Epoch 24/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6501 - acc: 0.6906 - auc: 0.7257 -
val_loss: 0.6504 - val_acc: 0.6719 - val_auc: 0.6993

Epoch 00024: val_acc did not improve from 0.68750
Epoch 25/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6472 - acc: 0.6992 - auc: 0.7279 -
val_loss: 0.6519 - val_acc: 0.6625 - val_auc: 0.6807

Epoch 00025: val_acc did not improve from 0.68750
Epoch 26/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6452 - acc: 0.6954 - auc: 0.7264 -
val_loss: 0.6636 - val_acc: 0.6338 - val_auc: 0.6507

Epoch 00026: val_acc did not improve from 0.68750
Epoch 27/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6423 - acc: 0.6973 - auc: 0.7324 -
val_loss: 0.6541 - val_acc: 0.6594 - val_auc: 0.6852

Epoch 00027: val_acc did not improve from 0.68750
Epoch 28/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6391 - acc: 0.7042 - auc: 0.7396 -
val_loss: 0.6793 - val_acc: 0.6519 - val_auc: 0.6957

Epoch 00028: val_acc did not improve from 0.68750
Epoch 29/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6374 - acc: 0.7125 - auc: 0.7433 -
val_loss: 0.6674 - val_acc: 0.6550 - val_auc: 0.6699
```

```
Epoch 00029: val_acc did not improve from 0.68750
Epoch 30/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6340 - acc: 0.7090 - auc: 0.7356 -
val_loss: 0.6648 - val_acc: 0.6538 - val_auc: 0.6917

Epoch 00030: val_acc did not improve from 0.68750
Epoch 31/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6315 - acc: 0.7150 - auc: 0.7396 -
val_loss: 0.6497 - val_acc: 0.6925 - val_auc: 0.7289

Epoch 00031: val_acc improved from 0.68750 to 0.69250, saving model to ./model3_1.h5
Epoch 32/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6299 - acc: 0.7188 - auc: 0.7429 -
val_loss: 0.6489 - val_acc: 0.6731 - val_auc: 0.7024

Epoch 00032: val_acc did not improve from 0.69250
Epoch 33/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6328 - acc: 0.7121 - auc: 0.7364 -
val_loss: 0.6576 - val_acc: 0.6562 - val_auc: 0.6737

Epoch 00033: val_acc did not improve from 0.69250
Epoch 34/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6303 - acc: 0.7198 - auc: 0.7461 -
val_loss: 0.6706 - val_acc: 0.6531 - val_auc: 0.6887

Epoch 00034: val_acc did not improve from 0.69250
Epoch 35/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6328 - acc: 0.7094 - auc: 0.7351 -
val_loss: 0.6602 - val_acc: 0.6456 - val_auc: 0.6690

Epoch 00035: val_acc did not improve from 0.69250
Epoch 36/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6208 - acc: 0.7262 - auc: 0.7446 -
val_loss: 0.6608 - val_acc: 0.6706 - val_auc: 0.7095

Epoch 00036: val_acc did not improve from 0.69250
Epoch 37/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6236 - acc: 0.7202 - auc: 0.7464 -
val_loss: 0.6351 - val_acc: 0.6937 - val_auc: 0.7191

Epoch 00037: val_acc improved from 0.69250 to 0.69375, saving model to ./model3_1.h5
Epoch 38/130
300/300 [==============================] - 140s 467ms/step - loss: 0.6265 - acc: 0.7171 - auc: 0.7350 -
val_loss: 0.6488 - val_acc: 0.6587 - val_auc: 0.6861

Epoch 00038: val_acc did not improve from 0.69375
Epoch 39/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6228 - acc: 0.7160 - auc: 0.7367 -
val_loss: 0.6574 - val_acc: 0.6669 - val_auc: 0.7067

Epoch 00039: val_acc did not improve from 0.69375
Epoch 40/130
300/300 [==============================] - 141s 468ms/step - loss: 0.6227 - acc: 0.7200 - auc: 0.7497 -
val_loss: 0.6380 - val_acc: 0.6887 - val_auc: 0.7133

Epoch 00040: val_acc did not improve from 0.69375
Epoch 41/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6197 - acc: 0.7246 - auc: 0.7476 -
val_loss: 0.6464 - val_acc: 0.6806 - val_auc: 0.7398

Epoch 00041: val_acc did not improve from 0.69375
Epoch 42/130
300/300 [==============================] - 140s 467ms/step - loss: 0.6194 - acc: 0.7181 - auc: 0.7527 -
val_loss: 0.6380 - val_acc: 0.6894 - val_auc: 0.7257

Epoch 00042: val_acc did not improve from 0.69375
Epoch 43/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6172 - acc: 0.7294 - auc: 0.7488 -
val_loss: 0.6439 - val_acc: 0.6844 - val_auc: 0.7069

Epoch 00043: val_acc did not improve from 0.69375
Epoch 44/130
300/300 [==============================] - 140s 468ms/step - loss: 0.6119 - acc: 0.7321 - auc: 0.7629 -
val_loss: 0.6438 - val_acc: 0.6619 - val_auc: 0.7079

Epoch 00044: val_acc did not improve from 0.69375
Epoch 45/130
```

```
300/300 [==============================] - 140s 468ms/step - loss: 0.6142 - acc: 0.7346 - auc: 0.7579 -
val_loss: 0.6334 - val_acc: 0.6994 - val_auc: 0.7334

Epoch 00045: val_acc improved from 0.69375 to 0.69937, saving model to ./model3_1.h5
Epoch 46/130
300/300 [==============================] - 140s 467ms/step - loss: 0.6153 - acc: 0.7227 - auc: 0.7515 -
val_loss: 0.6481 - val_acc: 0.6625 - val_auc: 0.6831

Epoch 00046: val_acc did not improve from 0.69937
Epoch 47/130
300/300 [==============================] - 140s 466ms/step - loss: 0.6150 - acc: 0.7258 - auc: 0.7575 -
val_loss: 0.6587 - val_acc: 0.6894 - val_auc: 0.7150

Epoch 00047: val_acc did not improve from 0.69937
Epoch 48/130
300/300 [==============================] - 141s 469ms/step - loss: 0.6145 - acc: 0.7277 - auc: 0.7469 -
val_loss: 0.6558 - val_acc: 0.6675 - val_auc: 0.6894

Epoch 00048: val_acc did not improve from 0.69937
Epoch 49/130
300/300 [==============================] - 140s 467ms/step - loss: 0.6019 - acc: 0.7525 - auc: 0.7844 -
val_loss: 0.6821 - val_acc: 0.6856 - val_auc: 0.7170

Epoch 00049: val_acc did not improve from 0.69937
Epoch 50/130
300/300 [==============================] - 140s 467ms/step - loss: 0.6042 - acc: 0.7423 - auc: 0.7699 -
val_loss: 0.6529 - val_acc: 0.6856 - val_auc: 0.7107

Epoch 00050: val_acc did not improve from 0.69937
Epoch 51/130
300/300 [==============================] - 140s 467ms/step - loss: 0.6024 - acc: 0.7473 - auc: 0.7743 -
val_loss: 0.6401 - val_acc: 0.6937 - val_auc: 0.7248

Epoch 00051: val_acc did not improve from 0.69937
Epoch 52/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5969 - acc: 0.7596 - auc: 0.7872 -
val_loss: 0.6963 - val_acc: 0.6575 - val_auc: 0.7373

Epoch 00052: val_acc did not improve from 0.69937
Epoch 53/130
300/300 [==============================] - 140s 467ms/step - loss: 0.6056 - acc: 0.7431 - auc: 0.7824 -
val_loss: 0.6472 - val_acc: 0.6794 - val_auc: 0.7122

Epoch 00053: val_acc did not improve from 0.69937
Epoch 54/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5999 - acc: 0.7515 - auc: 0.7768 -
val_loss: 0.6565 - val_acc: 0.6756 - val_auc: 0.7236

Epoch 00054: val_acc did not improve from 0.69937
Epoch 55/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5980 - acc: 0.7542 - auc: 0.7795 -
val_loss: 0.6308 - val_acc: 0.7075 - val_auc: 0.7541

Epoch 00055: val_acc improved from 0.69937 to 0.70750, saving model to ./model3_1.h5
Epoch 56/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5980 - acc: 0.7483 - auc: 0.7865 -
val_loss: 0.6472 - val_acc: 0.6987 - val_auc: 0.7624

Epoch 00056: val_acc did not improve from 0.70750
Epoch 57/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5913 - acc: 0.7579 - auc: 0.7965 -
val_loss: 0.6467 - val_acc: 0.7013 - val_auc: 0.7632

Epoch 00057: val_acc did not improve from 0.70750
Epoch 58/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5863 - acc: 0.7667 - auc: 0.8131 -
val_loss: 0.6311 - val_acc: 0.6919 - val_auc: 0.7485

Epoch 00058: val_acc did not improve from 0.70750
Epoch 59/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5868 - acc: 0.7648 - auc: 0.8025 -
val_loss: 0.6150 - val_acc: 0.7188 - val_auc: 0.7634

Epoch 00059: val_acc improved from 0.70750 to 0.71875, saving model to ./model3_1.h5
Epoch 60/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5813 - acc: 0.7750 - auc: 0.8131 -
val_loss: 0.6188 - val_acc: 0.7188 - val_auc: 0.7799
```

```
Epoch 00060: val_acc did not improve from 0.71875
Epoch 61/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5754 - acc: 0.7781 - auc: 0.8259 -
val_loss: 0.6152 - val_acc: 0.7344 - val_auc: 0.7845

Epoch 00061: val_acc improved from 0.71875 to 0.73438, saving model to ./model3_1.h5
Epoch 62/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5707 - acc: 0.7942 - auc: 0.8396 -
val_loss: 0.6223 - val_acc: 0.7063 - val_auc: 0.7625

Epoch 00062: val_acc did not improve from 0.73438
Epoch 63/130
300/300 [==============================] - 140s 468ms/step - loss: 0.5734 - acc: 0.7837 - auc: 0.8316 -
val_loss: 0.6223 - val_acc: 0.7294 - val_auc: 0.7702

Epoch 00063: val_acc did not improve from 0.73438
Epoch 64/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5791 - acc: 0.7754 - auc: 0.8273 -
val_loss: 0.6181 - val_acc: 0.7425 - val_auc: 0.7912

Epoch 00064: val_acc improved from 0.73438 to 0.74250, saving model to ./model3_1.h5
Epoch 65/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5657 - acc: 0.7915 - auc: 0.8379 -
val_loss: 0.6176 - val_acc: 0.7119 - val_auc: 0.7698

Epoch 00065: val_acc did not improve from 0.74250
Epoch 66/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5625 - acc: 0.7960 - auc: 0.8374 -
val_loss: 0.6093 - val_acc: 0.7169 - val_auc: 0.7712

Epoch 00066: val_acc did not improve from 0.74250
Epoch 67/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5640 - acc: 0.7979 - auc: 0.8365 -
val_loss: 0.6033 - val_acc: 0.7412 - val_auc: 0.7873

Epoch 00067: val_acc did not improve from 0.74250
Epoch 68/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5581 - acc: 0.8013 - auc: 0.8487 -
val_loss: 0.6089 - val_acc: 0.7469 - val_auc: 0.7852

Epoch 00068: val_acc improved from 0.74250 to 0.74687, saving model to ./model3_1.h5
Epoch 69/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5592 - acc: 0.7973 - auc: 0.8441 -
val_loss: 0.6152 - val_acc: 0.7306 - val_auc: 0.7803

Epoch 00069: val_acc did not improve from 0.74687
Epoch 70/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5618 - acc: 0.7960 - auc: 0.8324 -
val_loss: 0.5992 - val_acc: 0.7438 - val_auc: 0.8031

Epoch 00070: val_acc did not improve from 0.74687
Epoch 71/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5497 - acc: 0.8094 - auc: 0.8616 -
val_loss: 0.6027 - val_acc: 0.7400 - val_auc: 0.7990

Epoch 00071: val_acc did not improve from 0.74687
Epoch 72/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5521 - acc: 0.8031 - auc: 0.8516 -
val_loss: 0.6141 - val_acc: 0.7344 - val_auc: 0.8013

Epoch 00072: val_acc did not improve from 0.74687
Epoch 73/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5509 - acc: 0.8083 - auc: 0.8499 -
val_loss: 0.5845 - val_acc: 0.7519 - val_auc: 0.8148

Epoch 00073: val_acc improved from 0.74687 to 0.75187, saving model to ./model3_1.h5
Epoch 74/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5414 - acc: 0.8173 - auc: 0.8619 -
val_loss: 0.6462 - val_acc: 0.7269 - val_auc: 0.7998

Epoch 00074: val_acc did not improve from 0.75187
Epoch 75/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5486 - acc: 0.8096 - auc: 0.8547 -
val_loss: 0.6123 - val_acc: 0.7406 - val_auc: 0.7936

Epoch 00075: val acc did not improve from 0.75187
Epoch 76/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5421 - acc: 0.8140 - auc: 0.8611 -
val_loss: 0.6126 - val_acc: 0.7306 - val_auc: 0.7937

Epoch 00076: val_acc did not improve from 0.75187
Epoch 77/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5475 - acc: 0.8083 - auc: 0.8576 -
val_loss: 0.6061 - val_acc: 0.7400 - val_auc: 0.7874

Epoch 00077: val_acc did not improve from 0.75187
Epoch 78/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5420 - acc: 0.8121 - auc: 0.8590 -
val_loss: 0.5836 - val_acc: 0.7562 - val_auc: 0.8131

Epoch 00078: val_acc improved from 0.75187 to 0.75625, saving model to ./model3_1.h5
Epoch 79/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5366 - acc: 0.8194 - auc: 0.8647 -
val_loss: 0.5898 - val_acc: 0.7612 - val_auc: 0.8225

Epoch 00079: val_acc improved from 0.75625 to 0.76125, saving model to ./model3_1.h5
Epoch 80/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5319 - acc: 0.8269 - auc: 0.8682 -
val_loss: 0.5941 - val_acc: 0.7450 - val_auc: 0.7975

Epoch 00080: val_acc did not improve from 0.76125
Epoch 81/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5321 - acc: 0.8217 - auc: 0.8749 -
val_loss: 0.5881 - val_acc: 0.7531 - val_auc: 0.8130

Epoch 00081: val_acc did not improve from 0.76125
Epoch 82/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5282 - acc: 0.8269 - auc: 0.8712 -
val_loss: 0.5990 - val_acc: 0.7512 - val_auc: 0.8063

Epoch 00082: val_acc did not improve from 0.76125
Epoch 83/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5302 - acc: 0.8196 - auc: 0.8666 -
val_loss: 0.5860 - val_acc: 0.7612 - val_auc: 0.8231

Epoch 00083: val_acc did not improve from 0.76125
Epoch 84/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5262 - acc: 0.8279 - auc: 0.8765 -
val_loss: 0.5902 - val_acc: 0.7462 - val_auc: 0.8103

Epoch 00084: val_acc did not improve from 0.76125
Epoch 85/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5237 - acc: 0.8294 - auc: 0.8767 -
val_loss: 0.5860 - val_acc: 0.7800 - val_auc: 0.8431

Epoch 00085: val_acc improved from 0.76125 to 0.78000, saving model to ./model3_1.h5
Epoch 86/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5251 - acc: 0.8283 - auc: 0.8842 -
val_loss: 0.6211 - val_acc: 0.7269 - val_auc: 0.7824

Epoch 00086: val_acc did not improve from 0.78000
Epoch 87/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5154 - acc: 0.8356 - auc: 0.8889 -
val_loss: 0.5846 - val_acc: 0.7556 - val_auc: 0.8386

Epoch 00087: val_acc did not improve from 0.78000
Epoch 88/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5217 - acc: 0.8302 - auc: 0.8810 -
val_loss: 0.5996 - val_acc: 0.7406 - val_auc: 0.8045

Epoch 00088: val_acc did not improve from 0.78000
Epoch 89/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5165 - acc: 0.8354 - auc: 0.8784 -
val_loss: 0.6343 - val_acc: 0.7288 - val_auc: 0.8124

Epoch 00089: val_acc did not improve from 0.78000
Epoch 90/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5156 - acc: 0.8335 - auc: 0.8838 -
val_loss: 0.6046 - val_acc: 0.7406 - val_auc: 0.8202

Epoch 00090: val_acc did not improve from 0.78000
Epoch 91/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5040 - acc: 0.8483 - auc: 0.8999 -
```

```
val_loss: 0.6223 - val_acc: 0.7356 - val_auc: 0.7944

Epoch 00091: val_acc did not improve from 0.78000
Epoch 92/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5182 - acc: 0.8308 - auc: 0.8694 -
val_loss: 0.5841 - val_acc: 0.7438 - val_auc: 0.8010

Epoch 00092: val_acc did not improve from 0.78000
Epoch 93/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5096 - acc: 0.8352 - auc: 0.8857 -
val_loss: 0.6084 - val_acc: 0.7306 - val_auc: 0.8089

Epoch 00093: val_acc did not improve from 0.78000
Epoch 94/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5044 - acc: 0.8465 - auc: 0.8961 -
val_loss: 0.6103 - val_acc: 0.7294 - val_auc: 0.8149

Epoch 00094: val_acc did not improve from 0.78000
Epoch 95/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5096 - acc: 0.8352 - auc: 0.8843 -
val_loss: 0.6157 - val_acc: 0.7425 - val_auc: 0.8166

Epoch 00095: val_acc did not improve from 0.78000
Epoch 96/130
300/300 [==============================] - 140s 467ms/step - loss: 0.5004 - acc: 0.8510 - auc: 0.8940 -
val_loss: 0.5772 - val_acc: 0.7562 - val_auc: 0.8220

Epoch 00096: val_acc did not improve from 0.78000
Epoch 97/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5083 - acc: 0.8383 - auc: 0.8859 -
val_loss: 0.5922 - val_acc: 0.7519 - val_auc: 0.8105

Epoch 00097: val_acc did not improve from 0.78000
Epoch 98/130
300/300 [==============================] - 140s 466ms/step - loss: 0.4968 - acc: 0.8500 - auc: 0.8942 -
val_loss: 0.5956 - val_acc: 0.7544 - val_auc: 0.8295

Epoch 00098: val_acc did not improve from 0.78000
Epoch 99/130
300/300 [==============================] - 140s 466ms/step - loss: 0.4939 - acc: 0.8523 - auc: 0.8970 -
val_loss: 0.6376 - val_acc: 0.7275 - val_auc: 0.8004

Epoch 00099: val_acc did not improve from 0.78000
Epoch 100/130
300/300 [==============================] - 140s 466ms/step - loss: 0.5041 - acc: 0.8392 - auc: 0.8831 -
val_loss: 0.6006 - val_acc: 0.7338 - val_auc: 0.8057

Epoch 00100: val_acc did not improve from 0.78000
Epoch 101/130
300/300 [==============================] - 140s 467ms/step - loss: 0.4925 - acc: 0.8506 - auc: 0.8993 -
val_loss: 0.6389 - val_acc: 0.7325 - val_auc: 0.8216

Epoch 00101: val_acc did not improve from 0.78000
Epoch 102/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4912 - acc: 0.8496 - auc: 0.8996 -
val_loss: 0.6004 - val_acc: 0.7262 - val_auc: 0.7849

Epoch 00102: val_acc did not improve from 0.78000
Epoch 103/130
300/300 [==============================] - 141s 470ms/step - loss: 0.4846 - acc: 0.8602 - auc: 0.9010 -
val_loss: 0.6214 - val_acc: 0.7188 - val_auc: 0.7960

Epoch 00103: val_acc did not improve from 0.78000
Epoch 104/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4800 - acc: 0.8604 - auc: 0.9058 -
val_loss: 0.5993 - val_acc: 0.7450 - val_auc: 0.8124

Epoch 00104: val_acc did not improve from 0.78000
Epoch 105/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4794 - acc: 0.8640 - auc: 0.9112 -
val_loss: 0.6297 - val_acc: 0.7538 - val_auc: 0.8209

Epoch 00105: val_acc did not improve from 0.78000

Epoch 00105: ReduceLROnPlateau reducing learning rate to 9.999999747378752e-07.
Epoch 106/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4788 - acc: 0.8587 - auc: 0.9056 -
```

```
val_loss: 0.5917 - val_acc: 0.7450 - val_auc: 0.8180

Epoch 00106: val_acc did not improve from 0.78000
Epoch 107/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4784 - acc: 0.8615 - auc: 0.9049 -
val_loss: 0.5864 - val_acc: 0.7638 - val_auc: 0.8362

Epoch 00107: val_acc did not improve from 0.78000
Epoch 108/130
300/300 [==============================] - 142s 472ms/step - loss: 0.4733 - acc: 0.8631 - auc: 0.9116 -
val_loss: 0.5917 - val_acc: 0.7569 - val_auc: 0.8294

Epoch 00108: val_acc did not improve from 0.78000
Epoch 109/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4752 - acc: 0.8633 - auc: 0.9089 -
val_loss: 0.5833 - val_acc: 0.7762 - val_auc: 0.8398

Epoch 00109: val_acc did not improve from 0.78000
Epoch 110/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4691 - acc: 0.8708 - auc: 0.9135 -
val_loss: 0.5933 - val_acc: 0.7644 - val_auc: 0.8250

Epoch 00110: val_acc did not improve from 0.78000
Epoch 111/130
300/300 [==============================] - 142s 473ms/step - loss: 0.4745 - acc: 0.8642 - auc: 0.9104 -
val_loss: 0.6011 - val_acc: 0.7494 - val_auc: 0.8130

Epoch 00111: val_acc did not improve from 0.78000
Epoch 112/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4696 - acc: 0.8721 - auc: 0.9123 -
val_loss: 0.6082 - val_acc: 0.7425 - val_auc: 0.8241

Epoch 00112: val_acc did not improve from 0.78000
Epoch 113/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4607 - acc: 0.8833 - auc: 0.9233 -
val_loss: 0.5930 - val_acc: 0.7512 - val_auc: 0.8175

Epoch 00113: val_acc did not improve from 0.78000
Epoch 114/130
300/300 [==============================] - 141s 470ms/step - loss: 0.4688 - acc: 0.8723 - auc: 0.9183 -
val_loss: 0.6050 - val_acc: 0.7531 - val_auc: 0.8258

Epoch 00114: val_acc did not improve from 0.78000
Epoch 115/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4644 - acc: 0.8765 - auc: 0.9232 -
val_loss: 0.6032 - val_acc: 0.7519 - val_auc: 0.8220

Epoch 00115: val_acc did not improve from 0.78000
Epoch 116/130
300/300 [==============================] - 141s 470ms/step - loss: 0.4618 - acc: 0.8740 - auc: 0.9251 -
val_loss: 0.6046 - val_acc: 0.7369 - val_auc: 0.8150

Epoch 00116: val_acc did not improve from 0.78000
Epoch 117/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4576 - acc: 0.8821 - auc: 0.9259 -
val_loss: 0.6077 - val_acc: 0.7381 - val_auc: 0.8252

Epoch 00117: val_acc did not improve from 0.78000
Epoch 118/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4632 - acc: 0.8808 - auc: 0.9201 -
val_loss: 0.6052 - val_acc: 0.7456 - val_auc: 0.8241

Epoch 00118: val_acc did not improve from 0.78000
Epoch 119/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4646 - acc: 0.8763 - auc: 0.9160 -
val_loss: 0.6101 - val_acc: 0.7444 - val_auc: 0.8238

Epoch 00119: val_acc did not improve from 0.78000
Epoch 120/130
300/300 [==============================] - 141s 470ms/step - loss: 0.4618 - acc: 0.8781 - auc: 0.9190 -
val_loss: 0.5968 - val_acc: 0.7469 - val_auc: 0.8358

Epoch 00120: val_acc did not improve from 0.78000
Epoch 121/130
300/300 [==============================] - 141s 470ms/step - loss: 0.4590 - acc: 0.8804 - auc: 0.9245 -
val_loss: 0.6051 - val_acc: 0.7462 - val_auc: 0.8228
```

```
Epoch 00121: val_acc did not improve from 0.78000
Epoch 122/130
300/300 [==============================] - 141s 470ms/step - loss: 0.4512 - acc: 0.8858 - auc: 0.9302 -
val_loss: 0.6019 - val_acc: 0.7444 - val_auc: 0.8024

Epoch 00122: val_acc did not improve from 0.78000
Epoch 123/130
300/300 [==============================] - 141s 471ms/step - loss: 0.4544 - acc: 0.8873 - auc: 0.9287 -
val_loss: 0.5858 - val_acc: 0.7600 - val_auc: 0.8431

Epoch 00123: val_acc did not improve from 0.78000
Epoch 124/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4547 - acc: 0.8869 - auc: 0.9258 -
val_loss: 0.6109 - val_acc: 0.7294 - val_auc: 0.8048

Epoch 00124: val_acc did not improve from 0.78000
Epoch 125/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4559 - acc: 0.8813 - auc: 0.9252 -
val_loss: 0.5906 - val_acc: 0.7375 - val_auc: 0.8211

Epoch 00125: val_acc did not improve from 0.78000

Epoch 00125: ReduceLROnPlateau reducing learning rate to 9.999999974752428e-08.
Epoch 126/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4601 - acc: 0.8777 - auc: 0.9191 -
val_loss: 0.5997 - val_acc: 0.7475 - val_auc: 0.8170

Epoch 00126: val_acc did not improve from 0.78000
Epoch 127/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4515 - acc: 0.8877 - auc: 0.9279 -
val_loss: 0.6114 - val_acc: 0.7488 - val_auc: 0.8147

Epoch 00127: val_acc did not improve from 0.78000
Epoch 128/130
300/300 [==============================] - 141s 470ms/step - loss: 0.4549 - acc: 0.8865 - auc: 0.9286 -
val_loss: 0.6059 - val_acc: 0.7406 - val_auc: 0.8217

Epoch 00128: val_acc did not improve from 0.78000
Epoch 129/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4574 - acc: 0.8781 - auc: 0.9265 -
val_loss: 0.6136 - val_acc: 0.7381 - val_auc: 0.8189

Epoch 00129: val_acc did not improve from 0.78000
Epoch 130/130
300/300 [==============================] - 141s 469ms/step - loss: 0.4560 - acc: 0.8823 - auc: 0.9265 -
val_loss: 0.5985 - val_acc: 0.7531 - val_auc: 0.8325

Epoch 00130: val_acc did not improve from 0.78000
```

Out[0]:

```
<keras.callbacks.History at 0x7facac017be0>
```

At epoch 85 we can see Train accuracy is 82 and validation accuracy is approx 78 so considering epoch 85 for trained model

Epoch 85/130 => acc: 0.8294 - val_acc: 0.7800 --> saving model to ./model3_1.h5

In [0]:

```
!mv model3_1.h5 ./drive/My\ Drive/COLAB_FILES/
```

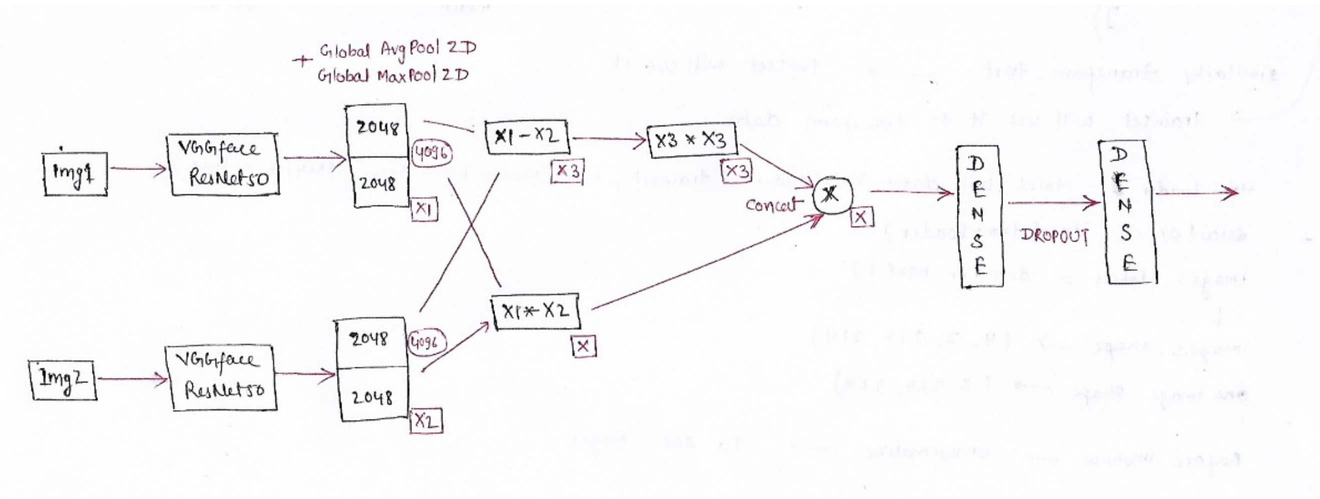
In [0]:

```

```

In [0]:

```

```

MODEL 4



In [0]:

```
!ls
```

```
drive          setup.py   test.zip
image          test    tools
keras_vggface        test-faces.zip  train
keras-vggface        test-private-faces.zip train-faces.zip
LICENSE.txt        test-private-labels.zip train_relationships.csv
MANIFEST.in        test-private-lists.zip train.zip
README.md        test-public-faces.zip visualize
sample_data        test-public-lists.zip
sample_submission.csv   test.py
```

In [0]:

```
def read_img_cv(path):
    img = cv2.imread(path)
    img = np.array(img).astype(np.float)
    return preprocess_input(img, version=2)


def gen(list_tuples, person_to_images_map, batch_size=16):
    ppl = list(person_to_images_map.keys())
    while True:
        batch_tuples = sample(list_tuples, batch_size // 2)
        labels = [1] * len(batch_tuples)
        while len(batch_tuples) < batch_size:
            p1 = choice(ppl)
            p2 = choice(ppl)

            if p1 != p2 and (p1, p2) not in list_tuples and (p2, p1) not in list_tuples:
                batch_tuples.append((p1, p2))
                labels.append(0)

        for x in batch_tuples:
            if not len(person_to_images_map[x[0]]):
                print(x[0])

        X1 = [choice(person_to_images_map[x[0]]) for x in batch_tuples]
        X1 = np.array([read_img_cv(x) for x in X1])

        X2 = [choice(person_to_images_map[x[1]]) for x in batch_tuples]
        X2 = np.array([read_img_cv(x) for x in X2])

        yield [X1, X2], labels
```

In [0]:

In [0]:

```
def baseline_model4():
    input_1 = Input(shape=(224, 224, 3))
    input_2 = Input(shape=(224, 224, 3))

    base_model4 = VGGFace(model='resnet50', include_top=False)

    for x in base_model4.layers[:-3]:
        x.trainable = True

    x1 = base_model4(input_1)
    x2 = base_model4(input_2)

    x1 = Concatenate(axis=-1)([GlobalMaxPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])

    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])

    x = Multiply()([x1, x2])

    x = Concatenate(axis=-1)([x, x3])
    x = BatchNormalization()(x)
    x = Dense(128,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.4)(x)
    x = Dense(64,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.4)(x)
    x = Dense(32,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.3)(x)
    x = Dense(16,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.3)(x)
    out = Dense(1, activation="sigmoid")(x)
    model4 = Model([input_1, input_2], out)
    model4.compile(loss="binary_crossentropy", metrics=['acc',auc], optimizer=Adam(0.00001))
    model4.summary()
    return model4
```

In [0]:

```
file_path4 = "./model4.h5"

checkpoint4 = ModelCheckpoint(file_path4, monitor='val_acc', verbose=1, save_best_only=True, mode='max')

reduce_on_plateau4 = ReduceLROnPlateau(monitor="val_acc", mode="max", factor=0.1, patience=20, verbose=1)

callbacks_list4 = [checkpoint4, reduce_on_plateau4]

model4 = baseline_model4()
```

```
Model: "model_2"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_4 (InputLayer)            (None, 224, 224, 3)  0
__________________________________________________________________________________________________
input_5 (InputLayer)            (None, 224, 224, 3)  0
__________________________________________________________________________________________________
vggface_resnet50 (Model)        multiple             23561152    input_4[0][0]
                                                                 input_5[0][0]
__________________________________________________________________________________________________
global_max_pooling2d_3 (GlobalM (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_average_pooling2d_3 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_max_pooling2d_4 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
```

```
global_average_pooling2d_4 (Glo (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
concatenate_4 (Concatenate)     (None, 4096)         0           global_max_pooling2d_3[0][0]
                                                                 global_average_pooling2d_3[0][0]
__________________________________________________________________________________________________
concatenate_5 (Concatenate)     (None, 4096)         0           global_max_pooling2d_4[0][0]
                                                                 global_average_pooling2d_4[0][0]
__________________________________________________________________________________________________
subtract_2 (Subtract)           (None, 4096)         0           concatenate_4[0][0]
                                                                 concatenate_5[0][0]
__________________________________________________________________________________________________
multiply_4 (Multiply)           (None, 4096)         0           concatenate_4[0][0]
                                                                 concatenate_5[0][0]
__________________________________________________________________________________________________
multiply_3 (Multiply)           (None, 4096)         0           subtract_2[0][0]
                                                                 subtract_2[0][0]
__________________________________________________________________________________________________
concatenate_6 (Concatenate)     (None, 8192)         0           multiply_4[0][0]
                                                                 multiply_3[0][0]
__________________________________________________________________________________________________
batch_normalization_2 (BatchNor (None, 8192)         32768       concatenate_6[0][0]
__________________________________________________________________________________________________
dense_6 (Dense)                 (None, 128)          1048704     batch_normalization_2[0][0]
__________________________________________________________________________________________________
activation_103 (Activation)     (None, 128)          0           dense_6[0][0]
__________________________________________________________________________________________________
dropout_2 (Dropout)             (None, 128)          0           activation_103[0][0]
__________________________________________________________________________________________________
dense_7 (Dense)                 (None, 64)           8256        dropout_2[0][0]
__________________________________________________________________________________________________
activation_104 (Activation)     (None, 64)           0           dense_7[0][0]
__________________________________________________________________________________________________
dropout_3 (Dropout)             (None, 64)           0           activation_104[0][0]
__________________________________________________________________________________________________
dense_8 (Dense)                 (None, 32)           2080        dropout_3[0][0]
__________________________________________________________________________________________________
activation_105 (Activation)     (None, 32)           0           dense_8[0][0]
__________________________________________________________________________________________________
dropout_4 (Dropout)             (None, 32)           0           activation_105[0][0]
__________________________________________________________________________________________________
dense_9 (Dense)                 (None, 16)           528         dropout_4[0][0]
__________________________________________________________________________________________________
activation_106 (Activation)     (None, 16)           0           dense_9[0][0]
__________________________________________________________________________________________________
dropout_5 (Dropout)             (None, 16)           0           activation_106[0][0]
__________________________________________________________________________________________________
dense_10 (Dense)                (None, 1)            17          dropout_5[0][0]
==================================================================================================
Total params: 24,653,505
Trainable params: 24,584,001
Non-trainable params: 69,504
__________________________________________________________________________________________________
```

In [0]:

```
model4.fit_generator(gen(train, train_person_to_images_map, batch_size=16), use_multiprocessing=True,
                    validation_data=gen(val, val_person_to_images_map, batch_size=16), epochs=100, verb
ose=1,
                    workers = 4, callbacks=callbacks_list4, steps_per_epoch=300, validation_steps=100)
```

```
Epoch 1/100
300/300 [==============================] - 116s 388ms/step - loss: 1.2418 - acc: 0.5033 - auc: 0.5185 -
val_loss: 1.1100 - val_acc: 0.5181 - val_auc: 0.5350

Epoch 00001: val_acc improved from -inf to 0.51812, saving model to ./model4.h5
Epoch 2/100
300/300 [==============================] - 90s 301ms/step - loss: 1.2246 - acc: 0.4956 - auc: 0.5023 -
val_loss: 1.1092 - val_acc: 0.5306 - val_auc: 0.5273

Epoch 00002: val_acc improved from 0.51812 to 0.53063, saving model to ./model4.h5
Epoch 3/100
300/300 [==============================] - 91s 302ms/step - loss: 1.2009 - acc: 0.5129 - auc: 0.5223 -
val_loss: 1.0929 - val_acc: 0.5644 - val_auc: 0.5781

Epoch 00003: val_acc improved from 0.53063 to 0.56437, saving model to ./model4.h5
Epoch 4/100
300/300 [==============================] - 90s 300ms/step - loss: 1.1889 - acc: 0.5252 - auc: 0.5292 -
val_loss: 1.0857 - val_acc: 0.5506 - val_auc: 0.5734

Epoch 00004: val_acc did not improve from 0.56437
Epoch 5/100
300/300 [==============================] - 90s 299ms/step - loss: 1.1697 - acc: 0.5240 - auc: 0.5404 -
val_loss: 1.0731 - val_acc: 0.5763 - val_auc: 0.6123

Epoch 00005: val_acc improved from 0.56437 to 0.57625, saving model to ./model4.h5
Epoch 6/100
300/300 [==============================] - 90s 300ms/step - loss: 1.1756 - acc: 0.5196 - auc: 0.5206 -
val_loss: 1.0631 - val_acc: 0.6156 - val_auc: 0.6533

Epoch 00006: val_acc improved from 0.57625 to 0.61562, saving model to ./model4.h5
Epoch 7/100
300/300 [==============================] - 90s 301ms/step - loss: 1.1477 - acc: 0.5265 - auc: 0.5420 -
val_loss: 1.0733 - val_acc: 0.5950 - val_auc: 0.6373

Epoch 00007: val_acc did not improve from 0.61562
Epoch 8/100
300/300 [==============================] - 90s 301ms/step - loss: 1.1426 - acc: 0.5442 - auc: 0.5535 -
val_loss: 1.0741 - val_acc: 0.5794 - val_auc: 0.6197

Epoch 00008: val_acc did not improve from 0.61562
Epoch 9/100
300/300 [==============================] - 91s 303ms/step - loss: 1.1217 - acc: 0.5508 - auc: 0.5764 -
val_loss: 1.0535 - val_acc: 0.6025 - val_auc: 0.6577

Epoch 00009: val_acc did not improve from 0.61562
Epoch 10/100
300/300 [==============================] - 90s 300ms/step - loss: 1.1234 - acc: 0.5435 - auc: 0.5748 -
val_loss: 1.0139 - val_acc: 0.6494 - val_auc: 0.7362

Epoch 00010: val_acc improved from 0.61562 to 0.64938, saving model to ./model4.h5
Epoch 11/100
300/300 [==============================] - 90s 300ms/step - loss: 1.1116 - acc: 0.5533 - auc: 0.5861 -
val_loss: 1.0214 - val_acc: 0.6375 - val_auc: 0.7047

Epoch 00011: val_acc did not improve from 0.64938
Epoch 12/100
300/300 [==============================] - 91s 302ms/step - loss: 1.1136 - acc: 0.5613 - auc: 0.5901 -
val_loss: 1.0275 - val_acc: 0.6481 - val_auc: 0.7267

Epoch 00012: val_acc did not improve from 0.64938
Epoch 13/100
300/300 [==============================] - 90s 301ms/step - loss: 1.0968 - acc: 0.5723 - auc: 0.6023 -
val_loss: 1.0246 - val_acc: 0.6356 - val_auc: 0.6969

Epoch 00013: val_acc did not improve from 0.64938
Epoch 14/100
300/300 [==============================] - 90s 301ms/step - loss: 1.0844 - acc: 0.5827 - auc: 0.6223 -
val_loss: 1.0113 - val_acc: 0.6606 - val_auc: 0.7381

Epoch 00014: val_acc improved from 0.64938 to 0.66063, saving model to ./model4.h5
Epoch 15/100
300/300 [==============================] - 90s 300ms/step - loss: 1.0800 - acc: 0.5804 - auc: 0.6277 -
val_loss: 1.0078 - val_acc: 0.6488 - val_auc: 0.7209

Epoch 00015: val_acc did not improve from 0.66063
Epoch 16/100
300/300 [==============================] - 91s 303ms/step - loss: 1.0814 - acc: 0.5831 - auc: 0.6310 -
val_loss: 1.0042 - val_acc: 0.6562 - val_auc: 0.7273

Epoch 00016: val_acc did not improve from 0.66063
Epoch 17/100
300/300 [==============================] - 91s 303ms/step - loss: 1.0576 - acc: 0.5998 - auc: 0.6543 -
val_loss: 0.9937 - val_acc: 0.6787 - val_auc: 0.7517

Epoch 00017: val_acc improved from 0.66063 to 0.67875, saving model to ./model4.h5
Epoch 18/100
300/300 [==============================] - 90s 301ms/step - loss: 1.0547 - acc: 0.6019 - auc: 0.6517 -
val_loss: 0.9830 - val_acc: 0.6913 - val_auc: 0.7625

Epoch 00018: val_acc improved from 0.67875 to 0.69125, saving model to ./model4.h5
Epoch 19/100
300/300 [==============================] - 90s 302ms/step - loss: 1.0398 - acc: 0.6131 - auc: 0.6739 -
```

```
val_loss: 0.9673 - val_acc: 0.6800 - val_auc: 0.7795

Epoch 00019: val_acc did not improve from 0.69125
Epoch 20/100
300/300 [==============================] - 90s 301ms/step - loss: 1.0377 - acc: 0.6252 - auc: 0.6814 -
val_loss: 0.9614 - val_acc: 0.6850 - val_auc: 0.7856

Epoch 00020: val_acc did not improve from 0.69125
Epoch 21/100
300/300 [==============================] - 90s 300ms/step - loss: 1.0391 - acc: 0.6250 - auc: 0.6783 -
val_loss: 0.9561 - val_acc: 0.7194 - val_auc: 0.7927

Epoch 00021: val_acc improved from 0.69125 to 0.71937, saving model to ./model4.h5
Epoch 22/100
300/300 [==============================] - 92s 305ms/step - loss: 1.0172 - acc: 0.6442 - auc: 0.7071 -
val_loss: 0.9618 - val_acc: 0.7013 - val_auc: 0.7825

Epoch 00022: val_acc did not improve from 0.71937
Epoch 23/100
300/300 [==============================] - 91s 303ms/step - loss: 1.0091 - acc: 0.6371 - auc: 0.7102 -
val_loss: 0.9548 - val_acc: 0.7044 - val_auc: 0.7848

Epoch 00023: val_acc did not improve from 0.71937
Epoch 24/100
300/300 [==============================] - 92s 305ms/step - loss: 1.0146 - acc: 0.6442 - auc: 0.7165 -
val_loss: 0.9530 - val_acc: 0.7063 - val_auc: 0.7934

Epoch 00024: val_acc did not improve from 0.71937
Epoch 25/100
300/300 [==============================] - 91s 304ms/step - loss: 0.9949 - acc: 0.6642 - auc: 0.7284 -
val_loss: 0.9470 - val_acc: 0.7131 - val_auc: 0.7980

Epoch 00025: val_acc did not improve from 0.71937
Epoch 26/100
300/300 [==============================] - 91s 304ms/step - loss: 1.0099 - acc: 0.6496 - auc: 0.7197 -
val_loss: 0.9553 - val_acc: 0.6850 - val_auc: 0.7816

Epoch 00026: val_acc did not improve from 0.71937
Epoch 27/100
300/300 [==============================] - 91s 304ms/step - loss: 1.0050 - acc: 0.6579 - auc: 0.7211 -
val_loss: 0.9434 - val_acc: 0.6981 - val_auc: 0.8013

Epoch 00027: val_acc did not improve from 0.71937
Epoch 28/100
300/300 [==============================] - 92s 307ms/step - loss: 0.9926 - acc: 0.6635 - auc: 0.7380 -
val_loss: 0.9512 - val_acc: 0.6963 - val_auc: 0.7939

Epoch 00028: val_acc did not improve from 0.71937
Epoch 29/100
300/300 [==============================] - 91s 303ms/step - loss: 0.9960 - acc: 0.6627 - auc: 0.7327 -
val_loss: 0.9455 - val_acc: 0.6981 - val_auc: 0.7877

Epoch 00029: val_acc did not improve from 0.71937
Epoch 30/100
300/300 [==============================] - 91s 304ms/step - loss: 0.9757 - acc: 0.6767 - auc: 0.7502 -
val_loss: 0.9444 - val_acc: 0.7000 - val_auc: 0.7936

Epoch 00030: val_acc did not improve from 0.71937
Epoch 31/100
300/300 [==============================] - 91s 303ms/step - loss: 0.9703 - acc: 0.6794 - auc: 0.7554 -
val_loss: 0.9177 - val_acc: 0.7181 - val_auc: 0.8172

Epoch 00031: val_acc did not improve from 0.71937
Epoch 32/100
300/300 [==============================] - 91s 302ms/step - loss: 0.9710 - acc: 0.6783 - auc: 0.7609 -
val_loss: 0.9304 - val_acc: 0.7137 - val_auc: 0.8105

Epoch 00032: val_acc did not improve from 0.71937
Epoch 33/100
300/300 [==============================] - 91s 305ms/step - loss: 0.9657 - acc: 0.6898 - auc: 0.7627 -
val_loss: 0.9182 - val_acc: 0.7238 - val_auc: 0.8223

Epoch 00033: val_acc improved from 0.71937 to 0.72375, saving model to ./model4.h5
Epoch 34/100
300/300 [==============================] - 91s 303ms/step - loss: 0.9627 - acc: 0.6840 - auc: 0.7627 -
val_loss: 0.9401 - val_acc: 0.7069 - val_auc: 0.7983
```

```
Epoch 00034: val_acc did not improve from 0.72375
Epoch 35/100
300/300 [==============================] - 90s 302ms/step - loss: 0.9460 - acc: 0.6917 - auc: 0.7740 -
val_loss: 0.9454 - val_acc: 0.7231 - val_auc: 0.7978

Epoch 00035: val_acc did not improve from 0.72375
Epoch 36/100
300/300 [==============================] - 91s 303ms/step - loss: 0.9443 - acc: 0.7038 - auc: 0.7854 -
val_loss: 0.9280 - val_acc: 0.7013 - val_auc: 0.8070

Epoch 00036: val_acc did not improve from 0.72375
Epoch 37/100
300/300 [==============================] - 92s 307ms/step - loss: 0.9296 - acc: 0.7148 - auc: 0.7903 -
val_loss: 0.9134 - val_acc: 0.7106 - val_auc: 0.8208

Epoch 00037: val_acc did not improve from 0.72375
Epoch 38/100
300/300 [==============================] - 91s 303ms/step - loss: 0.9286 - acc: 0.7081 - auc: 0.7877 -
val_loss: 0.8897 - val_acc: 0.7406 - val_auc: 0.8389

Epoch 00038: val_acc improved from 0.72375 to 0.74062, saving model to ./model4.h5
Epoch 39/100
300/300 [==============================] - 91s 302ms/step - loss: 0.9296 - acc: 0.7117 - auc: 0.7892 -
val_loss: 0.9165 - val_acc: 0.7106 - val_auc: 0.8172

Epoch 00039: val_acc did not improve from 0.74062
Epoch 40/100
300/300 [==============================] - 90s 301ms/step - loss: 0.9171 - acc: 0.7302 - auc: 0.8057 -
val_loss: 0.9357 - val_acc: 0.7137 - val_auc: 0.8114

Epoch 00040: val_acc did not improve from 0.74062
Epoch 41/100
300/300 [==============================] - 90s 301ms/step - loss: 0.9019 - acc: 0.7235 - auc: 0.8105 -
val_loss: 0.9135 - val_acc: 0.7331 - val_auc: 0.8164

Epoch 00041: val_acc did not improve from 0.74062
Epoch 42/100
300/300 [==============================] - 91s 302ms/step - loss: 0.9077 - acc: 0.7285 - auc: 0.8075 -
val_loss: 0.8947 - val_acc: 0.7269 - val_auc: 0.8364

Epoch 00042: val_acc did not improve from 0.74062
Epoch 43/100
300/300 [==============================] - 91s 303ms/step - loss: 0.8980 - acc: 0.7379 - auc: 0.8129 -
val_loss: 0.9102 - val_acc: 0.7181 - val_auc: 0.8216

Epoch 00043: val_acc did not improve from 0.74062
Epoch 44/100
300/300 [==============================] - 90s 300ms/step - loss: 0.8891 - acc: 0.7413 - auc: 0.8268 -
val_loss: 0.8860 - val_acc: 0.7350 - val_auc: 0.8411

Epoch 00044: val_acc did not improve from 0.74062
Epoch 45/100
300/300 [==============================] - 90s 299ms/step - loss: 0.9032 - acc: 0.7321 - auc: 0.8145 -
val_loss: 0.9284 - val_acc: 0.7238 - val_auc: 0.8319

Epoch 00045: val_acc did not improve from 0.74062
Epoch 46/100
300/300 [==============================] - 90s 299ms/step - loss: 0.8807 - acc: 0.7392 - auc: 0.8244 -
val_loss: 0.9059 - val_acc: 0.7281 - val_auc: 0.8264

Epoch 00046: val_acc did not improve from 0.74062
Epoch 47/100
300/300 [==============================] - 90s 300ms/step - loss: 0.8784 - acc: 0.7452 - auc: 0.8286 -
val_loss: 0.9060 - val_acc: 0.7262 - val_auc: 0.8241

Epoch 00047: val_acc did not improve from 0.74062
Epoch 48/100
300/300 [==============================] - 90s 300ms/step - loss: 0.8616 - acc: 0.7637 - auc: 0.8465 -
val_loss: 0.9210 - val_acc: 0.7037 - val_auc: 0.8217

Epoch 00048: val_acc did not improve from 0.74062
Epoch 49/100
300/300 [==============================] - 90s 300ms/step - loss: 0.8693 - acc: 0.7625 - auc: 0.8441 -
val_loss: 0.9327 - val_acc: 0.7206 - val_auc: 0.8152

Epoch 00049: val_acc did not improve from 0.74062
Epoch 50/100
```

```
300/300 [==============================] - 90s 301ms/step - loss: 0.8670 - acc: 0.7654 - auc: 0.8445 -
val_loss: 0.8857 - val_acc: 0.7425 - val_auc: 0.8295

Epoch 00050: val_acc improved from 0.74062 to 0.74250, saving model to ./model4.h5
Epoch 51/100
300/300 [==============================] - 89s 297ms/step - loss: 0.8516 - acc: 0.7688 - auc: 0.8560 -
val_loss: 0.9308 - val_acc: 0.7212 - val_auc: 0.7941

Epoch 00051: val_acc did not improve from 0.74250
Epoch 52/100
300/300 [==============================] - 90s 299ms/step - loss: 0.8444 - acc: 0.7717 - auc: 0.8507 -
val_loss: 0.9076 - val_acc: 0.7244 - val_auc: 0.8395

Epoch 00052: val_acc did not improve from 0.74250
Epoch 53/100
300/300 [==============================] - 90s 300ms/step - loss: 0.8336 - acc: 0.7754 - auc: 0.8585 -
val_loss: 0.9352 - val_acc: 0.7056 - val_auc: 0.8263

Epoch 00053: val_acc did not improve from 0.74250
Epoch 54/100
300/300 [==============================] - 90s 299ms/step - loss: 0.8125 - acc: 0.7881 - auc: 0.8712 -
val_loss: 0.9096 - val_acc: 0.7244 - val_auc: 0.8189

Epoch 00054: val_acc did not improve from 0.74250
Epoch 55/100
300/300 [==============================] - 89s 298ms/step - loss: 0.8358 - acc: 0.7865 - auc: 0.8639 -
val_loss: 0.9338 - val_acc: 0.7156 - val_auc: 0.8070

Epoch 00055: val_acc did not improve from 0.74250
Epoch 56/100
300/300 [==============================] - 90s 299ms/step - loss: 0.8110 - acc: 0.7921 - auc: 0.8760 -
val_loss: 0.9426 - val_acc: 0.7056 - val_auc: 0.8178

Epoch 00056: val_acc did not improve from 0.74250
Epoch 57/100
300/300 [==============================] - 90s 299ms/step - loss: 0.8225 - acc: 0.7931 - auc: 0.8737 -
val_loss: 0.8981 - val_acc: 0.7431 - val_auc: 0.8530

Epoch 00057: val_acc improved from 0.74250 to 0.74313, saving model to ./model4.h5
Epoch 58/100
300/300 [==============================] - 90s 298ms/step - loss: 0.8082 - acc: 0.7992 - auc: 0.8817 -
val_loss: 0.9506 - val_acc: 0.7144 - val_auc: 0.8080

Epoch 00058: val_acc did not improve from 0.74313
Epoch 59/100
300/300 [==============================] - 90s 300ms/step - loss: 0.7974 - acc: 0.8077 - auc: 0.8855 -
val_loss: 0.9466 - val_acc: 0.7163 - val_auc: 0.8264

Epoch 00059: val_acc did not improve from 0.74313
Epoch 60/100
300/300 [==============================] - 90s 300ms/step - loss: 0.7993 - acc: 0.8013 - auc: 0.8794 -
val_loss: 0.9198 - val_acc: 0.7331 - val_auc: 0.8295

Epoch 00060: val_acc did not improve from 0.74313
Epoch 61/100
300/300 [==============================] - 90s 299ms/step - loss: 0.7895 - acc: 0.8073 - auc: 0.8848 -
val_loss: 0.9070 - val_acc: 0.7419 - val_auc: 0.8458

Epoch 00061: val_acc did not improve from 0.74313
Epoch 62/100
300/300 [==============================] - 90s 300ms/step - loss: 0.7597 - acc: 0.8217 - auc: 0.8999 -
val_loss: 0.9757 - val_acc: 0.7094 - val_auc: 0.8206

Epoch 00062: val_acc did not improve from 0.74313
Epoch 63/100
300/300 [==============================] - 92s 305ms/step - loss: 0.7785 - acc: 0.8169 - auc: 0.8909 -
val_loss: 0.9635 - val_acc: 0.7150 - val_auc: 0.8084

Epoch 00063: val_acc did not improve from 0.74313
Epoch 64/100
300/300 [==============================] - 89s 298ms/step - loss: 0.7685 - acc: 0.8204 - auc: 0.8982 -
val_loss: 0.9358 - val_acc: 0.7250 - val_auc: 0.8302

Epoch 00064: val_acc did not improve from 0.74313
Epoch 65/100
300/300 [==============================] - 89s 298ms/step - loss: 0.7638 - acc: 0.8206 - auc: 0.8963 -
val_loss: 0.9910 - val_acc: 0.7081 - val_auc: 0.8108
```

```
Epoch 00065: val_acc did not improve from 0.74313
Epoch 66/100
300/300 [==============================] - 89s 296ms/step - loss: 0.7384 - acc: 0.8335 - auc: 0.9105 -
val_loss: 0.9782 - val_acc: 0.7131 - val_auc: 0.8192

Epoch 00066: val_acc did not improve from 0.74313
Epoch 67/100
300/300 [==============================] - 89s 297ms/step - loss: 0.7505 - acc: 0.8308 - auc: 0.9070 -
val_loss: 0.9475 - val_acc: 0.7181 - val_auc: 0.8305

Epoch 00067: val_acc did not improve from 0.74313
Epoch 68/100
300/300 [==============================] - 89s 297ms/step - loss: 0.7518 - acc: 0.8262 - auc: 0.9030 -
val_loss: 1.0039 - val_acc: 0.6994 - val_auc: 0.8148

Epoch 00068: val_acc did not improve from 0.74313
Epoch 69/100
300/300 [==============================] - 89s 297ms/step - loss: 0.7443 - acc: 0.8375 - auc: 0.9099 -
val_loss: 0.9958 - val_acc: 0.7113 - val_auc: 0.8252

Epoch 00069: val_acc did not improve from 0.74313
Epoch 70/100
300/300 [==============================] - 89s 297ms/step - loss: 0.7282 - acc: 0.8408 - auc: 0.9134 -
val_loss: 0.9926 - val_acc: 0.7044 - val_auc: 0.8183

Epoch 00070: val_acc did not improve from 0.74313
Epoch 71/100
300/300 [==============================] - 88s 295ms/step - loss: 0.7271 - acc: 0.8512 - auc: 0.9149 -
val_loss: 1.0256 - val_acc: 0.7044 - val_auc: 0.8141

Epoch 00071: val_acc did not improve from 0.74313
Epoch 72/100
300/300 [==============================] - 89s 295ms/step - loss: 0.7365 - acc: 0.8373 - auc: 0.9123 -
val_loss: 0.9672 - val_acc: 0.7219 - val_auc: 0.8358

Epoch 00072: val_acc did not improve from 0.74313
Epoch 73/100
300/300 [==============================] - 88s 295ms/step - loss: 0.7297 - acc: 0.8394 - auc: 0.9170 -
val_loss: 0.9631 - val_acc: 0.7125 - val_auc: 0.8281

Epoch 00073: val_acc did not improve from 0.74313
Epoch 74/100
300/300 [==============================] - 89s 296ms/step - loss: 0.7067 - acc: 0.8575 - auc: 0.9230 -
val_loss: 0.9780 - val_acc: 0.7231 - val_auc: 0.8403

Epoch 00074: val_acc did not improve from 0.74313
Epoch 75/100
300/300 [==============================] - 89s 295ms/step - loss: 0.7100 - acc: 0.8558 - auc: 0.9232 -
val_loss: 0.9968 - val_acc: 0.7156 - val_auc: 0.8327

Epoch 00075: val_acc did not improve from 0.74313
Epoch 76/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6885 - acc: 0.8610 - auc: 0.9307 -
val_loss: 0.9604 - val_acc: 0.7225 - val_auc: 0.8466

Epoch 00076: val_acc did not improve from 0.74313
Epoch 77/100
300/300 [==============================] - 89s 296ms/step - loss: 0.7028 - acc: 0.8627 - auc: 0.9258 -
val_loss: 1.0680 - val_acc: 0.7131 - val_auc: 0.8197

Epoch 00077: val_acc did not improve from 0.74313

Epoch 00077: ReduceLROnPlateau reducing learning rate to 9.999999747378752e-07.
Epoch 78/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6858 - acc: 0.8712 - auc: 0.9329 -
val_loss: 1.0294 - val_acc: 0.7137 - val_auc: 0.8411

Epoch 00078: val_acc did not improve from 0.74313
Epoch 79/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6834 - acc: 0.8621 - auc: 0.9311 -
val_loss: 1.0052 - val_acc: 0.7044 - val_auc: 0.8170

Epoch 00079: val_acc did not improve from 0.74313
Epoch 80/100
300/300 [==============================] - 90s 298ms/step - loss: 0.6700 - acc: 0.8679 - auc: 0.9388 -
val loss: 1.0162 - val acc: 0.7119 - val auc: 0.8234
```

```
Epoch 00080: val_acc did not improve from 0.74313
Epoch 81/100
300/300 [==============================] - 89s 298ms/step - loss: 0.6709 - acc: 0.8710 - auc: 0.9371 -
val_loss: 1.0375 - val_acc: 0.7075 - val_auc: 0.8366

Epoch 00081: val_acc did not improve from 0.74313
Epoch 82/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6660 - acc: 0.8725 - auc: 0.9385 -
val_loss: 0.9707 - val_acc: 0.7181 - val_auc: 0.8444

Epoch 00082: val_acc did not improve from 0.74313
Epoch 83/100
300/300 [==============================] - 89s 296ms/step - loss: 0.6655 - acc: 0.8727 - auc: 0.9406 -
val_loss: 1.0353 - val_acc: 0.7000 - val_auc: 0.8300

Epoch 00083: val_acc did not improve from 0.74313
Epoch 84/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6806 - acc: 0.8675 - auc: 0.9356 -
val_loss: 1.0364 - val_acc: 0.7144 - val_auc: 0.8494

Epoch 00084: val_acc did not improve from 0.74313
Epoch 85/100
300/300 [==============================] - 89s 296ms/step - loss: 0.6654 - acc: 0.8777 - auc: 0.9412 -
val_loss: 1.0220 - val_acc: 0.7231 - val_auc: 0.8400

Epoch 00085: val_acc did not improve from 0.74313
Epoch 86/100
300/300 [==============================] - 89s 296ms/step - loss: 0.6592 - acc: 0.8775 - auc: 0.9432 -
val_loss: 1.0002 - val_acc: 0.7200 - val_auc: 0.8403

Epoch 00086: val_acc did not improve from 0.74313
Epoch 87/100
300/300 [==============================] - 89s 298ms/step - loss: 0.6619 - acc: 0.8700 - auc: 0.9392 -
val_loss: 1.0169 - val_acc: 0.7169 - val_auc: 0.8455

Epoch 00087: val_acc did not improve from 0.74313
Epoch 88/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6563 - acc: 0.8810 - auc: 0.9435 -
val_loss: 1.0063 - val_acc: 0.7019 - val_auc: 0.8305

Epoch 00088: val_acc did not improve from 0.74313
Epoch 89/100
300/300 [==============================] - 90s 299ms/step - loss: 0.6622 - acc: 0.8800 - auc: 0.9424 -
val_loss: 1.0016 - val_acc: 0.7131 - val_auc: 0.8392

Epoch 00089: val_acc did not improve from 0.74313
Epoch 90/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6458 - acc: 0.8840 - auc: 0.9460 -
val_loss: 1.0323 - val_acc: 0.7013 - val_auc: 0.8311

Epoch 00090: val_acc did not improve from 0.74313
Epoch 91/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6412 - acc: 0.8904 - auc: 0.9483 -
val_loss: 1.0452 - val_acc: 0.7100 - val_auc: 0.8256

Epoch 00091: val_acc did not improve from 0.74313
Epoch 92/100
300/300 [==============================] - 89s 296ms/step - loss: 0.6600 - acc: 0.8731 - auc: 0.9409 -
val_loss: 1.0238 - val_acc: 0.7031 - val_auc: 0.8372

Epoch 00092: val_acc did not improve from 0.74313
Epoch 93/100
300/300 [==============================] - 89s 298ms/step - loss: 0.6558 - acc: 0.8846 - auc: 0.9461 -
val_loss: 1.0364 - val_acc: 0.7025 - val_auc: 0.8286

Epoch 00093: val_acc did not improve from 0.74313
Epoch 94/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6608 - acc: 0.8754 - auc: 0.9409 -
val_loss: 1.0276 - val_acc: 0.7069 - val_auc: 0.8314

Epoch 00094: val_acc did not improve from 0.74313
Epoch 95/100
300/300 [==============================] - 89s 298ms/step - loss: 0.6418 - acc: 0.8915 - auc: 0.9511 -
val_loss: 1.0458 - val_acc: 0.7069 - val_auc: 0.8348

Epoch 00095: val_acc did not improve from 0.74313
Epoch 96/100
300/300 [==============================] - 89s 298ms/step - loss: 0.6368 - acc: 0.8842 - auc: 0.9489 -
val_loss: 1.0220 - val_acc: 0.7306 - val_auc: 0.8447

Epoch 00096: val_acc did not improve from 0.74313
Epoch 97/100
300/300 [==============================] - 89s 297ms/step - loss: 0.6506 - acc: 0.8835 - auc: 0.9469 -
val_loss: 1.0212 - val_acc: 0.7338 - val_auc: 0.8538

Epoch 00097: val_acc did not improve from 0.74313

Epoch 00097: ReduceLROnPlateau reducing learning rate to 9.999999974752428e-08.
Epoch 98/100
300/300 [==============================] - 89s 296ms/step - loss: 0.6315 - acc: 0.8946 - auc: 0.9521 -
val_loss: 1.0288 - val_acc: 0.7238 - val_auc: 0.8434

Epoch 00098: val_acc did not improve from 0.74313
Epoch 99/100
300/300 [==============================] - 90s 299ms/step - loss: 0.6443 - acc: 0.8917 - auc: 0.9490 -
val_loss: 1.0658 - val_acc: 0.7087 - val_auc: 0.8228

Epoch 00099: val_acc did not improve from 0.74313
Epoch 100/100
300/300 [==============================] - 89s 296ms/step - loss: 0.6381 - acc: 0.8944 - auc: 0.9523 -
val_loss: 1.0473 - val_acc: 0.7200 - val_auc: 0.8497

Epoch 00100: val_acc did not improve from 0.74313
```

Out[0]:

```
<keras.callbacks.History at 0x7f21e5ed86d8>
```

At epoch 57 we can see Train accuracy is 79 and validation accuracy is approx 74 so considering epoch 57 for trained model

Epoch 57/100 => acc: 0.7931 - val_acc: 0.7431 --> saving model to ./model4.h5

In [0]:

```
!mv model4.h5 ./drive/My\ Drive/COLAB/SIMILAR_FACES/
```

MODEL 5

In [36]:

```
from random import choice, sample
```

In [37]:

```
def read_img_cv(path):
    img = cv2.imread(path)
    img = np.array(img).astype(np.float)
    return preprocess_input(img, version=2)


def gen(list_tuples, person_to_images_map, batch_size=16):
    ppl = list(person_to_images_map.keys())
    while True:
        batch_tuples = sample(list_tuples, batch_size // 2)
        labels = [1] * len(batch_tuples)
        while len(batch_tuples) < batch_size:
            p1 = choice(ppl)
            p2 = choice(ppl)

            if p1 != p2 and (p1, p2) not in list_tuples and (p2, p1) not in list_tuples:
                batch_tuples.append((p1, p2))
                labels.append(0)

        for x in batch_tuples:
            if not len(person_to_images_map[x[0]]):
                print(x[0])

        X1 = [choice(person_to_images_map[x[0]]) for x in batch_tuples]
        X1 = np.array([read_img_cv(x) for x in X1])

        X2 = [choice(person_to_images_map[x[1]]) for x in batch_tuples]
        X2 = np.array([read_img_cv(x) for x in X2])

        yield [X1, X2], labels
```

In [38]:

```
import gc
import psutil
print("available RAM:", psutil.virtual_memory())
gc.collect()
print("available RAM:", psutil.virtual_memory())
```

```
available RAM: svmem(total=54881505280, available=53530177536, percent=2.5, used=788684800, free=533423
75936, active=781651968, inactive=473862144, buffers=72650752, cached=677793792, shared=12136448, slab=
82657280)
available RAM: svmem(total=54881505280, available=53530177536, percent=2.5, used=788676608, free=533423
75936, active=781651968, inactive=473862144, buffers=72658944, cached=677793792, shared=12136448, slab=
82657280)
```

In [39]:

```
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau,EarlyStopping
from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D,MaxPooling1D,GlobalAve
ragePooling1D,LSTM
from keras.layers import Concatenate, Multiply, Dropout, Subtract, Add, Conv2D,Conv1D
from keras.models import Model
from keras.layers import BatchNormalization,Reshape
from keras.preprocessing import image
from keras.optimizers import Adam,SGD,rmsprop
import h5py
from keras.regularizers import l2
from keras.losses import binary_crossentropy
import tensorflow as tf
from sklearn.metrics import roc_auc_score
from keras.layers import LeakyReLU
from keras import regularizers
from keras.initializers  import RandomNormal

def auc(y_true, y_pred):
    return tf.py_function(roc_auc_score, (y_true, y_pred), tf.double)
```

In [40]:

```
def siamese_model():
    input_1 = Input(shape=(224, 224, 3))
    input_2 = Input(shape=(224, 224, 3))
    base_model = VGGFace(model='resnet50', include_top=False,input_shape=(224, 224, 3))

    for x in base_model.layers[:-3]:
        x.trainable = True

    x1 = base_model(input_1)
    x2 = base_model(input_2)

    x1 = Concatenate(axis=-1)([GlobalAvgPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])
    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])
    x1_ = Multiply()([x1, x1])
    x2_ = Multiply()([x2, x2])
    x4  = Subtract()([x1_, x2_])
    x   = Concatenate(axis=-1)([x4, x3])
    x   = Reshape(input_shape=(8192,), target_shape=(8192, 1))(x)
    x = Conv1D(filters=100,
               kernel_size=8,
               strides=8,input_shape=( 8192,1)
```

```
                strides=8,input_shape=( 8192,1),
                activation='relu',
                padding='same')(x)
    x = Conv1D( 100,8,activation= 'relu')(x)
    x = LSTM(128,kernel_initializer='glorot_normal',recurrent_dropout=0.5,kernel_regularizer=l2(0.001),
return_sequences=True)(x)
    #x = LSTM(128, return_sequences=True)(x)
    x = Dropout(0.2)(x)
    x = LSTM(128,kernel_initializer='glorot_normal',recurrent_dropout=0.5,kernel_regularizer=l2(0.001),
return_sequences=True)(x)
    #x = LSTM(128, return_sequences=True)(x)
    x = BatchNormalization ()(x)
    x = Flatten()(x)
    x = Dense(100,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.2)(x)
    x = Dense(64,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.1)(x)
    x = Dense(32,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.1)(x)
    x = Dense(16,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.01)(x)
    out = Dense(1, activation="sigmoid")(x)
    model = Model([input_1, input_2], out)
    model.compile(loss="binary_crossentropy", metrics=['acc',auc], optimizer=Adam(0.00001,decay=1e-6))
    model.summary()

    return model
```

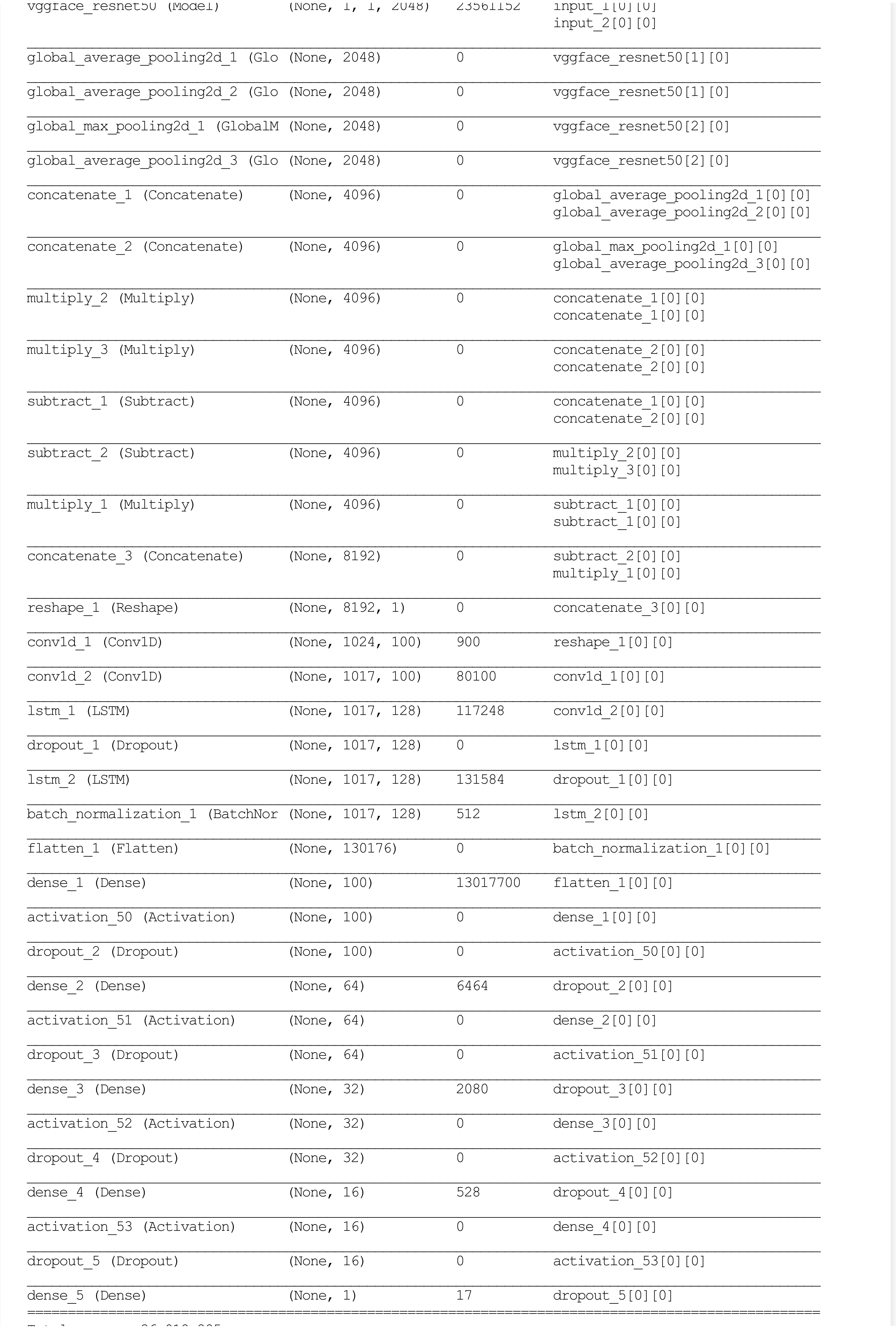
In [ ]:

```

```

In [41]:

```
file_path5 = "./model5.h5"

checkpoint5 = ModelCheckpoint(file_path5, monitor='val_acc', verbose=1, save_best_only=True, mode='max'
)
early_stopping5 = EarlyStopping(monitor='val_loss', min_delta=0.0001,  patience=20, verbose=0, mode='au
to')
reduce_on_plateau5 = ReduceLROnPlateau(monitor="val_loss", mode="max", factor=0.2, patience=10, verbose
=1)

callbacks_list5 = [checkpoint5, reduce_on_plateau5,early_stopping5]

model5 = siamese_model()
```

```
WARNING:tensorflow:From /usr/local/lib/python3.5/dist-packages/tensorflow_core/python/ops/resource_vari
able_ops.py:1630: calling BaseResourceVariable.__init__ (from tensorflow.python.ops.resource_variable_o
ps) with constraint is deprecated and will be removed in a future version.
Instructions for updating:
If using Keras pass *_constraint arguments to layers.
WARNING:tensorflow:From /usr/local/lib/python3.5/dist-packages/keras/backend/tensorflow_backend.py:4070
: The name tf.nn.max_pool is deprecated. Please use tf.nn.max_pool2d instead.

WARNING:tensorflow:From /usr/local/lib/python3.5/dist-packages/keras/backend/tensorflow_backend.py:4074
: The name tf.nn.avg_pool is deprecated. Please use tf.nn.avg_pool2d instead.

WARNING:tensorflow:From /usr/local/lib/python3.5/dist-packages/tensorflow_core/python/ops/nn_impl.py:18
3: where (from tensorflow.python.ops.array_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
Model: "model_1"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            (None, 224, 224, 3)  0

__________________________________________________________________________________________________
input_2 (InputLayer)            (None, 224, 224, 3)  0

__________________________________________________________________________________________________
    resnet50 (Model)            (None, 1, 1, 2048)   23561152    input_1[0][0]
```



```
vggface_resnet50 (Model)         (None, 1, 1, 2048)   23561152    input_1[0][0]
                                                                  input_2[0][0]
__________________________________________________________________________________________________
global_average_pooling2d_1 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_average_pooling2d_2 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_max_pooling2d_1 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
global_average_pooling2d_3 (Glo (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
concatenate_1 (Concatenate)     (None, 4096)         0           global_average_pooling2d_1[0][0]
                                                                 global_average_pooling2d_2[0][0]
__________________________________________________________________________________________________
concatenate_2 (Concatenate)     (None, 4096)         0           global_max_pooling2d_1[0][0]
                                                                 global_average_pooling2d_3[0][0]
__________________________________________________________________________________________________
multiply_2 (Multiply)           (None, 4096)         0           concatenate_1[0][0]
                                                                 concatenate_1[0][0]
__________________________________________________________________________________________________
multiply_3 (Multiply)           (None, 4096)         0           concatenate_2[0][0]
                                                                 concatenate_2[0][0]
__________________________________________________________________________________________________
subtract_1 (Subtract)           (None, 4096)         0           concatenate_1[0][0]
                                                                 concatenate_2[0][0]
__________________________________________________________________________________________________
subtract_2 (Subtract)           (None, 4096)         0           multiply_2[0][0]
                                                                 multiply_3[0][0]
__________________________________________________________________________________________________
multiply_1 (Multiply)           (None, 4096)         0           subtract_1[0][0]
                                                                 subtract_1[0][0]
__________________________________________________________________________________________________
concatenate_3 (Concatenate)     (None, 8192)         0           subtract_2[0][0]
                                                                 multiply_1[0][0]
__________________________________________________________________________________________________
reshape_1 (Reshape)             (None, 8192, 1)      0           concatenate_3[0][0]
__________________________________________________________________________________________________
conv1d_1 (Conv1D)               (None, 1024, 100)    900         reshape_1[0][0]
__________________________________________________________________________________________________
conv1d_2 (Conv1D)               (None, 1017, 100)    80100       conv1d_1[0][0]
__________________________________________________________________________________________________
lstm_1 (LSTM)                   (None, 1017, 128)    117248      conv1d_2[0][0]
__________________________________________________________________________________________________
dropout_1 (Dropout)             (None, 1017, 128)    0           lstm_1[0][0]
__________________________________________________________________________________________________
lstm_2 (LSTM)                   (None, 1017, 128)    131584      dropout_1[0][0]
__________________________________________________________________________________________________
batch_normalization_1 (BatchNor (None, 1017, 128)    512         lstm_2[0][0]
__________________________________________________________________________________________________
flatten_1 (Flatten)             (None, 130176)       0           batch_normalization_1[0][0]
__________________________________________________________________________________________________
dense_1 (Dense)                 (None, 100)          13017700    flatten_1[0][0]
__________________________________________________________________________________________________
activation_50 (Activation)      (None, 100)          0           dense_1[0][0]
__________________________________________________________________________________________________
dropout_2 (Dropout)             (None, 100)          0           activation_50[0][0]
__________________________________________________________________________________________________
dense_2 (Dense)                 (None, 64)           6464        dropout_2[0][0]
__________________________________________________________________________________________________
activation_51 (Activation)      (None, 64)           0           dense_2[0][0]
__________________________________________________________________________________________________
dropout_3 (Dropout)             (None, 64)           0           activation_51[0][0]
__________________________________________________________________________________________________
dense_3 (Dense)                 (None, 32)           2080        dropout_3[0][0]
__________________________________________________________________________________________________
activation_52 (Activation)      (None, 32)           0           dense_3[0][0]
__________________________________________________________________________________________________
dropout_4 (Dropout)             (None, 32)           0           activation_52[0][0]
__________________________________________________________________________________________________
dense_4 (Dense)                 (None, 16)           528         dropout_4[0][0]
__________________________________________________________________________________________________
activation_53 (Activation)      (None, 16)           0           dense_4[0][0]
__________________________________________________________________________________________________
dropout_5 (Dropout)             (None, 16)           0           activation_53[0][0]
__________________________________________________________________________________________________
dense_5 (Dense)                 (None, 1)            17          dropout_5[0][0]
==================================================================================================
Total params: 36,918,285
Total params: 36,918,285
Trainable params: 36,864,909
Non-trainable params: 53,376
_________________________________________________________________
```

In [42]:

```
model5.fit_generator(gen(train, train_person_to_images_map, batch_size=8), use_multiprocessing=True,
                    validation_data=gen(val, val_person_to_images_map, batch_size=8), epochs=30, verbo
se=1,
                    workers=16, callbacks=callbacks_list5, steps_per_epoch=300, validation_steps=100)
```

```
WARNING:tensorflow:From /usr/local/lib/python3.5/dist-packages/keras/backend/tensorflow_backend.py:422:
The name tf.global_variables is deprecated. Please use tf.compat.v1.global_variables instead.

Epoch 1/30
300/300 [==============================] - 1400s 5s/step - loss: 1.4704 - acc: 0.4983 - auc: 0.4863 - v
al_loss: 1.4107 - val_acc: 0.5387 - val_auc: 0.5569

Epoch 00001: val_auc improved from -inf to 0.55687, saving model to ./model5.h5
Epoch 2/30
300/300 [==============================] - 1376s 5s/step - loss: 1.4383 - acc: 0.5208 - auc: 0.5354 - v
al_loss: 1.3606 - val_acc: 0.5962 - val_auc: 0.6494

Epoch 00002: val_auc improved from 0.55687 to 0.64938, saving model to ./model5.h5
Epoch 3/30
300/300 [==============================] - 1362s 5s/step - loss: 1.4213 - acc: 0.5412 - auc: 0.5569 - v
al_loss: 1.3343 - val_acc: 0.6275 - val_auc: 0.6687

Epoch 00003: val_auc improved from 0.64938 to 0.66875, saving model to ./model5.h5
Epoch 4/30
300/300 [==============================] - 1356s 5s/step - loss: 1.4031 - acc: 0.5592 - auc: 0.6006 - v
al_loss: 1.3580 - val_acc: 0.7013 - val_auc: 0.7681

Epoch 00004: val_auc improved from 0.66875 to 0.76812, saving model to ./model5.h5
Epoch 5/30
300/300 [==============================] - 1359s 5s/step - loss: 1.3732 - acc: 0.5942 - auc: 0.6317 - v
al_loss: 1.5030 - val_acc: 0.6725 - val_auc: 0.7256

Epoch 00005: val_auc did not improve from 0.76812
Epoch 6/30
300/300 [==============================] - 1349s 4s/step - loss: 1.3886 - acc: 0.5967 - auc: 0.6317 - v
al_loss: 1.2673 - val_acc: 0.6787 - val_auc: 0.7594

Epoch 00006: val_auc did not improve from 0.76812
Epoch 7/30
300/300 [==============================] - 1342s 4s/step - loss: 1.3453 - acc: 0.6229 - auc: 0.6702 - v
al_loss: 1.3428 - val_acc: 0.6550 - val_auc: 0.7406

Epoch 00007: val_auc did not improve from 0.76812
Epoch 8/30
300/300 [==============================] - 1341s 4s/step - loss: 1.3376 - acc: 0.6371 - auc: 0.6900 - v
al_loss: 1.1294 - val_acc: 0.6438 - val_auc: 0.7688

Epoch 00008: val_auc improved from 0.76812 to 0.76875, saving model to ./model5.h5
Epoch 9/30
300/300 [==============================] - 1382s 5s/step - loss: 1.3288 - acc: 0.6508 - auc: 0.6981 - v
al_loss: 1.2949 - val_acc: 0.6837 - val_auc: 0.8188

Epoch 00009: val_auc improved from 0.76875 to 0.81875, saving model to ./model5.h5
Epoch 10/30
300/300 [==============================] - 1403s 5s/step - loss: 1.3290 - acc: 0.6263 - auc: 0.6835 - v
al_loss: 1.2635 - val_acc: 0.6137 - val_auc: 0.7788

Epoch 00010: val_auc did not improve from 0.81875
Epoch 11/30
300/300 [==============================] - 1391s 5s/step - loss: 1.3033 - acc: 0.6571 - auc: 0.7233 - v
al_loss: 1.2668 - val_acc: 0.6125 - val_auc: 0.7925

Epoch 00011: val_auc did not improve from 0.81875
Epoch 12/30
300/300 [==============================] - 1405s 5s/step - loss: 1.3092 - acc: 0.6654 - auc: 0.7217 - v
al_loss: 1.6192 - val_acc: 0.6187 - val_auc: 0.8031

Epoch 00012: val_auc did not improve from 0.81875
Epoch 13/30
Epoch 13/30
300/300 [==============================] - 1392s 5s/step - loss: 1.2747 - acc: 0.6904 - auc: 0.7421 - v
al_loss: 1.3045 - val_acc: 0.6812 - val_auc: 0.8087

Epoch 00013: val_auc did not improve from 0.81875
Epoch 14/30
300/300 [==============================] - 1407s 5s/step - loss: 1.2737 - acc: 0.6854 - auc: 0.7535 - v
al_loss: 1.2144 - val_acc: 0.6888 - val_auc: 0.8181

Epoch 00014: val_auc did not improve from 0.81875
Epoch 15/30
300/300 [==============================] - 1402s 5s/step - loss: 1.2688 - acc: 0.6883 - auc: 0.7600 - v
al_loss: 1.4941 - val_acc: 0.6700 - val_auc: 0.8469

Epoch 00015: val_auc improved from 0.81875 to 0.84688, saving model to ./model5.h5
Epoch 16/30
300/300 [==============================] - 1413s 5s/step - loss: 1.2532 - acc: 0.7042 - auc: 0.7650 - v
al_loss: 1.0344 - val_acc: 0.6637 - val_auc: 0.8594

Epoch 00016: val_auc improved from 0.84688 to 0.85938, saving model to ./model5.h5
Epoch 17/30
300/300 [==============================] - 1401s 5s/step - loss: 1.2357 - acc: 0.7054 - auc: 0.7763 - v
al_loss: 1.5083 - val_acc: 0.6587 - val_auc: 0.8500

Epoch 00017: val_auc did not improve from 0.85938
Epoch 18/30
300/300 [==============================] - 1393s 5s/step - loss: 1.2434 - acc: 0.7088 - auc: 0.7704 - v
al_loss: 1.3254 - val_acc: 0.6888 - val_auc: 0.8388

Epoch 00018: val_auc did not improve from 0.85938
Epoch 19/30
300/300 [==============================] - 1403s 5s/step - loss: 1.2278 - acc: 0.6988 - auc: 0.7725 - v
al_loss: 0.9712 - val_acc: 0.6575 - val_auc: 0.8425

Epoch 00019: val_auc did not improve from 0.85938
Epoch 20/30
300/300 [==============================] - 1403s 5s/step - loss: 1.2278 - acc: 0.7192 - auc: 0.7858 - v
al_loss: 1.2096 - val_acc: 0.6913 - val_auc: 0.8319

Epoch 00020: val_auc did not improve from 0.85938
Epoch 21/30
300/300 [==============================] - 1387s 5s/step - loss: 1.2153 - acc: 0.7063 - auc: 0.7904 - v
al_loss: 1.5849 - val_acc: 0.7212 - val_auc: 0.8381

Epoch 00021: val_auc did not improve from 0.85938
Epoch 22/30
300/300 [==============================] - 1384s 5s/step - loss: 1.1996 - acc: 0.7325 - auc: 0.8087 - v
al_loss: 1.6508 - val_acc: 0.6712 - val_auc: 0.8419

Epoch 00022: val_auc did not improve from 0.85938
Epoch 23/30
300/300 [==============================] - 1401s 5s/step - loss: 1.1929 - acc: 0.7258 - auc: 0.8096 - v
al_loss: 1.0912 - val_acc: 0.6800 - val_auc: 0.8363

Epoch 00023: val_auc did not improve from 0.85938
Epoch 24/30
300/300 [==============================] - 1403s 5s/step - loss: 1.1646 - acc: 0.7442 - auc: 0.8198 - v
al_loss: 1.0079 - val_acc: 0.6913 - val_auc: 0.8506

Epoch 00024: val_auc did not improve from 0.85938
Epoch 25/30
300/300 [==============================] - 1402s 5s/step - loss: 1.1683 - acc: 0.7271 - auc: 0.8121 - v
al_loss: 0.8623 - val_acc: 0.7013 - val_auc: 0.8431

Epoch 00025: val_auc did not improve from 0.85938
Epoch 26/30
300/300 [==============================] - 1401s 5s/step - loss: 1.1551 - acc: 0.7412 - auc: 0.8171 - v
al_loss: 0.9408 - val_acc: 0.6950 - val_auc: 0.8456

Epoch 00026: val_auc did not improve from 0.85938
Epoch 27/30
300/300 [==============================] - 1413s 5s/step - loss: 1.1627 - acc: 0.7367 - auc: 0.8079 - v
al_loss: 1.2624 - val_acc: 0.6850 - val_auc: 0.8350

Epoch 00027: val_auc did not improve from 0.85938
Epoch 28/30
300/300 [==============================] - 1408s 5s/step - loss: 1.1403 - acc: 0.7417 - auc: 0.8290 - v
al_loss: 1.0550 - val_acc: 0.6925 - val_auc: 0.8378
```

```
al_loss: 1.0559 - val_acc: 0.6925 - val_auc: 0.8378

Epoch 00028: val_auc did not improve from 0.85938
Epoch 29/30
300/300 [==============================] - 1414s 5s/step - loss: 1.1394 - acc: 0.7387 - auc: 0.8129 - v
al_loss: 1.3482 - val_acc: 0.6913 - val_auc: 0.8406

Epoch 00029: val_auc did not improve from 0.85938
Epoch 30/30
300/300 [==============================] - 1401s 5s/step - loss: 1.1142 - acc: 0.7433 - auc: 0.8256 - v
al_loss: 1.1110 - val_acc: 0.7038 - val_auc: 0.8300

Epoch 00030: val_auc did not improve from 0.85938
```

Out[42]:

```
<keras.callbacks.callbacks.History at 0x7f5a27bfc080>
```

In [44]:

```python
def siamese_model_1():
    input_1 = Input(shape=(224, 224, 3))
    input_2 = Input(shape=(224, 224, 3))
    base_model = VGGFace(model='resnet50', include_top=False,input_shape=(224, 224, 3))

    for x in base_model.layers[:-3]:
        x.trainable = True

    x1 = base_model(input_1)
    x2 = base_model(input_2)

    x1 = Concatenate(axis=-1)([GlobalAvgPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])
    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])
    x1_ = Multiply()([x1, x1])
    x2_ = Multiply()([x2, x2])
    x4  = Subtract()([x1_, x2_])
    x   = Concatenate(axis=-1)([x4, x3])
    x   = Reshape(input_shape=(8192,), target_shape=(8192, 1))(x)
    x = Conv1D(filters=100,
               kernel_size=8,
               strides=8,input_shape=( 8192,1),
               activation='relu',
               padding='same')(x)
    x = Conv1D( 100,8,activation= 'relu')(x)
    x = LSTM(128,kernel_initializer='glorot_normal',recurrent_dropout=0.5,kernel_regularizer=l2(0.001),
return_sequences=True)(x)
    #x = LSTM(128, return_sequences=True)(x)
    x = Dropout(0.4)(x)
    x = LSTM(128,kernel_initializer='glorot_normal',recurrent_dropout=0.5,kernel_regularizer=l2(0.001),
return_sequences=True)(x)
    #x = LSTM(128, return_sequences=True)(x)
    x = BatchNormalization ()(x)
    x = Flatten()(x)
    x = Dense(100,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.3)(x)
    x = Dense(64,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.2)(x)
    x = Dense(32,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.2)(x)
    x = Dense(16,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.1)(x)
    out = Dense(1, activation="sigmoid")(x)
    model = Model([input_1, input_2], out)
    model.compile(loss="binary_crossentropy", metrics=['acc',auc], optimizer=Adam(0.00001,decay=1e-6))
    model.summary()
    return model
```

In [ ]:

In [47]:

```
file_path5 = "./model5_1.h5"

checkpoint5_1 = ModelCheckpoint(file_path5, monitor='val_acc', verbose=1, save_best_only=True, mode='max')
early_stopping5_1 = EarlyStopping(monitor='val_loss', min_delta=0.0001,  patience=50, verbose=0, mode='auto')
reduce_on_plateau5_1 = ReduceLROnPlateau(monitor="val_loss", mode="max", factor=0.2, patience=10, verbose=1)

callbacks_list5_1 = [checkpoint5_1, reduce_on_plateau5_1,early_stopping5_1]

model5_1 = siamese_model_1()
```

```
Model: "model_3"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_7 (InputLayer)            (None, 224, 224, 3)  0
__________________________________________________________________________________________________
input_8 (InputLayer)            (None, 224, 224, 3)  0
__________________________________________________________________________________________________
vggface_resnet50 (Model)        (None, 1, 1, 2048)   23561152    input_7[0][0]
                                                                 input_8[0][0]
__________________________________________________________________________________________________
global_average_pooling2d_7 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_average_pooling2d_8 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_max_pooling2d_3 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
global_average_pooling2d_9 (Glo (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
concatenate_7 (Concatenate)     (None, 4096)         0           global_average_pooling2d_7[0][0]
                                                                 global_average_pooling2d_8[0][0]
__________________________________________________________________________________________________
concatenate_8 (Concatenate)     (None, 4096)         0           global_max_pooling2d_3[0][0]
                                                                 global_average_pooling2d_9[0][0]
__________________________________________________________________________________________________
multiply_8 (Multiply)           (None, 4096)         0           concatenate_7[0][0]
                                                                 concatenate_7[0][0]
__________________________________________________________________________________________________
multiply_9 (Multiply)           (None, 4096)         0           concatenate_8[0][0]
                                                                 concatenate_8[0][0]
__________________________________________________________________________________________________
subtract_5 (Subtract)           (None, 4096)         0           concatenate_7[0][0]
                                                                 concatenate_8[0][0]
__________________________________________________________________________________________________
subtract_6 (Subtract)           (None, 4096)         0           multiply_8[0][0]
                                                                 multiply_9[0][0]
__________________________________________________________________________________________________
multiply_7 (Multiply)           (None, 4096)         0           subtract_5[0][0]
                                                                 subtract_5[0][0]
__________________________________________________________________________________________________
concatenate_9 (Concatenate)     (None, 8192)         0           subtract_6[0][0]
                                                                 multiply_7[0][0]
__________________________________________________________________________________________________
reshape_3 (Reshape)             (None, 8192, 1)      0           concatenate_9[0][0]
__________________________________________________________________________________________________
conv1d_5 (Conv1D)               (None, 1024, 100)    900         reshape_3[0][0]
__________________________________________________________________________________________________
conv1d_6 (Conv1D)               (None, 1017, 100)    80100       conv1d_5[0][0]
__________________________________________________________________________________________________
lstm_5 (LSTM)                   (None, 1017, 128)    117248      conv1d_6[0][0]
__________________________________________________________________________________________________
dropout_11 (Dropout)            (None, 1017, 128)    0           lstm_5[0][0]
__________________________________________________________________________________________________
lstm_6 (LSTM)                   (None, 1017, 128)    131584      dropout_11[0][0]
__________________________________________________________________________________________________
batch_normalization_3 (BatchNor (None, 1017, 128)    512         lstm_6[0][0]
__________________________________________________________________________________________________
flatten_3 (Flatten)             (None, 130176)       0           batch_normalization_3[0][0]
```

```
flatten_3 (Flatten)             (None, 130176)       0           batch_normalization_3[0][0]
__________________________________________________________________________________________________
dense_11 (Dense)                (None, 100)          13017700    flatten_3[0][0]
__________________________________________________________________________________________________
activation_156 (Activation)     (None, 100)          0           dense_11[0][0]
__________________________________________________________________________________________________
dropout_12 (Dropout)            (None, 100)          0           activation_156[0][0]
__________________________________________________________________________________________________
dense_12 (Dense)                (None, 64)           6464        dropout_12[0][0]
__________________________________________________________________________________________________
activation_157 (Activation)     (None, 64)           0           dense_12[0][0]
__________________________________________________________________________________________________
dropout_13 (Dropout)            (None, 64)           0           activation_157[0][0]
__________________________________________________________________________________________________
dense_13 (Dense)                (None, 32)           2080        dropout_13[0][0]
__________________________________________________________________________________________________
activation_158 (Activation)     (None, 32)           0           dense_13[0][0]
__________________________________________________________________________________________________
dropout_14 (Dropout)            (None, 32)           0           activation_158[0][0]
__________________________________________________________________________________________________
dense_14 (Dense)                (None, 16)           528         dropout_14[0][0]
__________________________________________________________________________________________________
activation_159 (Activation)     (None, 16)           0           dense_14[0][0]
__________________________________________________________________________________________________
dropout_15 (Dropout)            (None, 16)           0           activation_159[0][0]
__________________________________________________________________________________________________
dense_15 (Dense)                (None, 1)            17          dropout_15[0][0]
==================================================================================================
Total params: 36,918,285
Trainable params: 36,864,909
Non-trainable params: 53,376
__________________________________________________________________________________________________
```

In [49]:

```
model5_1.fit_generator(gen(train, train_person_to_images_map, batch_size=16), use_multiprocessing=True,
                    validation_data=gen(val, val_person_to_images_map, batch_size=16), epochs=50, verb
ose=1,
                    workers=16, callbacks=callbacks_list5_1, steps_per_epoch=300, validation_steps=100
)
```

```
Epoch 1/50
300/300 [==============================] - 2412s 8s/step - loss: 1.4332 - acc: 0.5325 - auc: 0.5481 - v
al_loss: 1.3648 - val_acc: 0.6044 - val_auc: 0.6998

Epoch 00001: val_acc improved from -inf to 0.60438, saving model to ./model5_1.h5
Epoch 2/50
300/300 [==============================] - 2408s 8s/step - loss: 1.3899 - acc: 0.5738 - auc: 0.6127 - v
al_loss: 1.4123 - val_acc: 0.6244 - val_auc: 0.7108

Epoch 00002: val_acc improved from 0.60438 to 0.62437, saving model to ./model5_1.h5
Epoch 3/50
300/300 [==============================] - 2409s 8s/step - loss: 1.3729 - acc: 0.5960 - auc: 0.6380 - v
al_loss: 1.5730 - val_acc: 0.6150 - val_auc: 0.7177

Epoch 00003: val_acc did not improve from 0.62437
Epoch 4/50
300/300 [==============================] - 2412s 8s/step - loss: 1.3514 - acc: 0.6183 - auc: 0.6661 - v
al_loss: 1.4229 - val_acc: 0.6344 - val_auc: 0.7167

Epoch 00004: val_acc improved from 0.62437 to 0.63437, saving model to ./model5_1.h5
Epoch 5/50
300/300 [==============================] - 2411s 8s/step - loss: 1.3341 - acc: 0.6354 - auc: 0.6838 - v
al_loss: 1.3731 - val_acc: 0.6612 - val_auc: 0.7200

Epoch 00005: val_acc improved from 0.63437 to 0.66125, saving model to ./model5_1.h5
Epoch 6/50
300/300 [==============================] - 2428s 8s/step - loss: 1.3323 - acc: 0.6319 - auc: 0.6779 - v
al_loss: 1.2908 - val_acc: 0.6463 - val_auc: 0.7120

Epoch 00006: val_acc did not improve from 0.66125
Epoch 7/50
300/300 [==============================] - 2421s 8s/step - loss: 1.3067 - acc: 0.6488 - auc: 0.7086 - v
al_loss: 1.2434 - val_acc: 0.6650 - val_auc: 0.7425

Epoch 00007: val_acc improved from 0.66125 to 0.66500, saving model to ./model5_1.h5
Epoch 00007: val_acc improved from 0.66125 to 0.66500, saving model to ./model5_1.h5
Epoch 8/50
300/300 [==============================] - 2434s 8s/step - loss: 1.3046 - acc: 0.6554 - auc: 0.7046 - v
al_loss: 1.3182 - val_acc: 0.6956 - val_auc: 0.7788

Epoch 00008: val_acc improved from 0.66500 to 0.69563, saving model to ./model5_1.h5
Epoch 9/50
300/300 [==============================] - 2449s 8s/step - loss: 1.2894 - acc: 0.6660 - auc: 0.7302 - v
al_loss: 1.2035 - val_acc: 0.6956 - val_auc: 0.7677

Epoch 00009: val_acc did not improve from 0.69563
Epoch 10/50
300/300 [==============================] - 2457s 8s/step - loss: 1.2601 - acc: 0.6888 - auc: 0.7574 - v
al_loss: 1.2242 - val_acc: 0.7088 - val_auc: 0.7806

Epoch 00010: val_acc improved from 0.69563 to 0.70875, saving model to ./model5_1.h5
Epoch 11/50
300/300 [==============================] - 2452s 8s/step - loss: 1.2319 - acc: 0.7171 - auc: 0.7835 - v
al_loss: 1.1536 - val_acc: 0.7669 - val_auc: 0.8366

Epoch 00011: val_acc improved from 0.70875 to 0.76688, saving model to ./model5_1.h5
Epoch 12/50
300/300 [==============================] - 2452s 8s/step - loss: 1.2137 - acc: 0.7360 - auc: 0.8021 - v
al_loss: 1.1006 - val_acc: 0.7563 - val_auc: 0.8238

Epoch 00012: val_acc did not improve from 0.76688
Epoch 13/50
300/300 [==============================] - 2417s 8s/step - loss: 1.2017 - acc: 0.7333 - auc: 0.8048 - v
al_loss: 1.0603 - val_acc: 0.7825 - val_auc: 0.8727

Epoch 00013: val_acc improved from 0.76688 to 0.78250, saving model to ./model5_1.h5

Epoch 00013: ReduceLROnPlateau reducing learning rate to 1.9999999494757505e-06.
Epoch 14/50
300/300 [==============================] - 2379s 8s/step - loss: 1.1694 - acc: 0.7519 - auc: 0.8269 - v
al_loss: 1.1775 - val_acc: 0.7950 - val_auc: 0.8672

Epoch 00014: val_acc improved from 0.78250 to 0.79500, saving model to ./model5_1.h5
Epoch 15/50
300/300 [==============================] - 2408s 8s/step - loss: 1.1418 - acc: 0.7631 - auc: 0.8474 - v
al_loss: 1.1670 - val_acc: 0.7744 - val_auc: 0.8558

Epoch 00015: val_acc did not improve from 0.79500
Epoch 16/50
300/300 [==============================] - 2392s 8s/step - loss: 1.1494 - acc: 0.7646 - auc: 0.8412 - v
al_loss: 1.3725 - val_acc: 0.7769 - val_auc: 0.8577

Epoch 00016: val_acc did not improve from 0.79500
Epoch 17/50
300/300 [==============================] - 2377s 8s/step - loss: 1.1512 - acc: 0.7635 - auc: 0.8360 - v
al_loss: 1.1046 - val_acc: 0.7806 - val_auc: 0.8589

Epoch 00017: val_acc did not improve from 0.79500
Epoch 18/50
300/300 [==============================] - 2372s 8s/step - loss: 1.1449 - acc: 0.7696 - auc: 0.8458 - v
al_loss: 1.1398 - val_acc: 0.7950 - val_auc: 0.8673

Epoch 00018: val_acc did not improve from 0.79500
Epoch 19/50
300/300 [==============================] - 2379s 8s/step - loss: 1.1292 - acc: 0.7704 - auc: 0.8536 - v
al_loss: 1.1963 - val_acc: 0.7700 - val_auc: 0.8561

Epoch 00019: val_acc did not improve from 0.79500
Epoch 20/50
300/300 [==============================] - 2412s 8s/step - loss: 1.1284 - acc: 0.7823 - auc: 0.8575 - v
al_loss: 1.0475 - val_acc: 0.7756 - val_auc: 0.8678

Epoch 00020: val_acc did not improve from 0.79500
Epoch 21/50
300/300 [==============================] - 2402s 8s/step - loss: 1.0996 - acc: 0.7973 - auc: 0.8761 - v
al_loss: 1.2326 - val_acc: 0.7887 - val_auc: 0.8745

Epoch 00021: val_acc did not improve from 0.79500
Epoch 22/50
300/300 [==============================] - 2454s 8s/step - loss: 1.0984 - acc: 0.7946 - auc: 0.8736 - v
al_loss: 1.1242 - val_acc: 0.7775 - val_auc: 0.8689

Epoch 00022: val_acc did not improve from 0.79500
Epoch 00022: val_acc did not improve from 0.79500
Epoch 23/50
300/300 [==============================] - 2452s 8s/step - loss: 1.0939 - acc: 0.7954 - auc: 0.8720 - v
al_loss: 1.2493 - val_acc: 0.7969 - val_auc: 0.8823

Epoch 00023: val_acc improved from 0.79500 to 0.79688, saving model to ./model5_1.h5

Epoch 00023: ReduceLROnPlateau reducing learning rate to 3.999999989900971e-07.
Epoch 24/50
300/300 [==============================] - 2446s 8s/step - loss: 1.0879 - acc: 0.8031 - auc: 0.8758 - v
al_loss: 1.2046 - val_acc: 0.7994 - val_auc: 0.8900

Epoch 00024: val_acc improved from 0.79688 to 0.79937, saving model to ./model5_1.h5
Epoch 25/50
300/300 [==============================] - 2452s 8s/step - loss: 1.1012 - acc: 0.7912 - auc: 0.8664 - v
al_loss: 1.1487 - val_acc: 0.7763 - val_auc: 0.8687

Epoch 00025: val_acc did not improve from 0.79937
Epoch 26/50
300/300 [==============================] - 2451s 8s/step - loss: 1.0909 - acc: 0.8008 - auc: 0.8720 - v
al_loss: 1.1911 - val_acc: 0.7719 - val_auc: 0.8656

Epoch 00026: val_acc did not improve from 0.79937
Epoch 27/50
300/300 [==============================] - 2473s 8s/step - loss: 1.0735 - acc: 0.8067 - auc: 0.8867 - v
al_loss: 1.0655 - val_acc: 0.7887 - val_auc: 0.8712

Epoch 00027: val_acc did not improve from 0.79937
Epoch 28/50
300/300 [==============================] - 2479s 8s/step - loss: 1.0813 - acc: 0.8006 - auc: 0.8783 - v
al_loss: 0.9317 - val_acc: 0.7919 - val_auc: 0.8722

Epoch 00028: val_acc did not improve from 0.79937
Epoch 29/50
300/300 [==============================] - 2465s 8s/step - loss: 1.0790 - acc: 0.8073 - auc: 0.8802 - v
al_loss: 1.0364 - val_acc: 0.7738 - val_auc: 0.8553

Epoch 00029: val_acc did not improve from 0.79937
Epoch 30/50
300/300 [==============================] - 2468s 8s/step - loss: 1.0745 - acc: 0.8117 - auc: 0.8859 - v
al_loss: 1.0505 - val_acc: 0.8012 - val_auc: 0.8731

Epoch 00030: val_acc improved from 0.79937 to 0.80125, saving model to ./model5_1.h5
Epoch 31/50
300/300 [==============================] - 2478s 8s/step - loss: 1.0897 - acc: 0.7946 - auc: 0.8733 - v
al_loss: 0.8719 - val_acc: 0.8175 - val_auc: 0.8997

Epoch 00031: val_acc improved from 0.80125 to 0.81750, saving model to ./model5_1.h5
Epoch 32/50
300/300 [==============================] - 2474s 8s/step - loss: 1.0733 - acc: 0.8133 - auc: 0.8852 - v
al_loss: 1.3500 - val_acc: 0.7806 - val_auc: 0.8703

Epoch 00032: val_acc did not improve from 0.81750
Epoch 33/50
300/300 [==============================] - 2472s 8s/step - loss: 1.0834 - acc: 0.8135 - auc: 0.8829 - v
al_loss: 1.1542 - val_acc: 0.7625 - val_auc: 0.8645

Epoch 00033: val_acc did not improve from 0.81750

Epoch 00033: ReduceLROnPlateau reducing learning rate to 8.00000009348878e-08.
Epoch 34/50
300/300 [==============================] - 2480s 8s/step - loss: 1.0882 - acc: 0.8025 - auc: 0.8780 - v
al_loss: 1.0896 - val_acc: 0.7850 - val_auc: 0.8763

Epoch 00034: val_acc did not improve from 0.81750
Epoch 35/50
300/300 [==============================] - 2484s 8s/step - loss: 1.0825 - acc: 0.8031 - auc: 0.8778 - v
al_loss: 1.3321 - val_acc: 0.7850 - val_auc: 0.8791

Epoch 00035: val_acc did not improve from 0.81750
Epoch 36/50
300/300 [==============================] - 2493s 8s/step - loss: 1.0744 - acc: 0.8090 - auc: 0.8836 - v
al_loss: 1.5253 - val_acc: 0.7931 - val_auc: 0.8813

Epoch 00036: val_acc did not improve from 0.81750
Epoch 37/50
300/300 [==============================] - 2450s 8s/step - loss: 1.0814 - acc: 0.8035 - auc: 0.8772 - v
al_loss: 0.9447 - val_acc: 0.7931 - val_auc: 0.8822
```

```
al_loss: 0.9447 - val_acc: 0.7951 - val_auc: 0.8622

Epoch 00037: val_acc did not improve from 0.81750
Epoch 38/50
300/300 [==============================] - 2472s 8s/step - loss: 1.0754 - acc: 0.8044 - auc: 0.8795 - v
al_loss: 1.3675 - val_acc: 0.7919 - val_auc: 0.8808

Epoch 00038: val_acc did not improve from 0.81750
Epoch 39/50
300/300 [==============================] - 2458s 8s/step - loss: 1.0764 - acc: 0.8104 - auc: 0.8824 - v
al_loss: 1.6520 - val_acc: 0.7788 - val_auc: 0.8748

Epoch 00039: val_acc did not improve from 0.81750
Epoch 40/50
300/300 [==============================] - 2454s 8s/step - loss: 1.0646 - acc: 0.8106 - auc: 0.8910 - v
al_loss: 1.0561 - val_acc: 0.7919 - val_auc: 0.8830

Epoch 00040: val_acc did not improve from 0.81750
Epoch 41/50
300/300 [==============================] - 2452s 8s/step - loss: 1.0790 - acc: 0.8158 - auc: 0.8863 - v
al_loss: 1.0373 - val_acc: 0.7925 - val_auc: 0.8734

Epoch 00041: val_acc did not improve from 0.81750
Epoch 42/50
300/300 [==============================] - 2477s 8s/step - loss: 1.0599 - acc: 0.8179 - auc: 0.8925 - v
al_loss: 1.1467 - val_acc: 0.7869 - val_auc: 0.8922

Epoch 00042: val_acc did not improve from 0.81750
Epoch 43/50
300/300 [==============================] - 2468s 8s/step - loss: 1.0724 - acc: 0.8092 - auc: 0.8825 - v
al_loss: 0.9809 - val_acc: 0.7894 - val_auc: 0.8811

Epoch 00043: val_acc did not improve from 0.81750
Epoch 44/50
300/300 [==============================] - 2443s 8s/step - loss: 1.0700 - acc: 0.8092 - auc: 0.8867 - v
al_loss: 1.1400 - val_acc: 0.8019 - val_auc: 0.8883

Epoch 00044: val_acc did not improve from 0.81750
Epoch 45/50
300/300 [==============================] - 2421s 8s/step - loss: 1.0757 - acc: 0.8081 - auc: 0.8815 - v
al_loss: 1.8584 - val_acc: 0.7912 - val_auc: 0.8827

Epoch 00045: val_acc did not improve from 0.81750
Epoch 46/50
300/300 [==============================] - 2453s 8s/step - loss: 1.0729 - acc: 0.8046 - auc: 0.8837 - v
al_loss: 1.3261 - val_acc: 0.7919 - val_auc: 0.8823

Epoch 00046: val_acc did not improve from 0.81750
Epoch 47/50
300/300 [==============================] - 2411s 8s/step - loss: 1.0605 - acc: 0.8202 - auc: 0.8899 - v
al_loss: 0.9758 - val_acc: 0.7663 - val_auc: 0.8692

Epoch 00047: val_acc did not improve from 0.81750
Epoch 48/50
300/300 [==============================] - 2354s 8s/step - loss: 1.0503 - acc: 0.8221 - auc: 0.8948 - v
al_loss: 1.0674 - val_acc: 0.7875 - val_auc: 0.8794

Epoch 00048: val_acc did not improve from 0.81750
Epoch 49/50
300/300 [==============================] - 2396s 8s/step - loss: 1.0632 - acc: 0.8179 - auc: 0.8892 - v
al_loss: 1.1514 - val_acc: 0.7763 - val_auc: 0.8628

Epoch 00049: val_acc did not improve from 0.81750
Epoch 50/50
300/300 [==============================] - 2438s 8s/step - loss: 1.0701 - acc: 0.8129 - auc: 0.8831 - v
al_loss: 1.1050 - val_acc: 0.7750 - val_auc: 0.8680

Epoch 00050: val_acc did not improve from 0.81750
```

Out[49]:

```
<keras.callbacks.callbacks.History at 0x7f5a27beca20>
```

In [51]:

```
!mv ./model5_1.h5 ./drive/My\ Drive/COLAB/SIMILAR_FACES/
```

In [53]:

```
!ls
```

```
driver_installer.run     sample_submission.csv

model5_1.h5      SAVE_MODEL

model5.h5      TENSORBOARD

Northeastern_GCP_submit2_1_final_1.ipynb  test

Northeastern_GCP_submit2_1_test.ipynb   train

recognizing-faces-in-the-wild    train_relationships.csv

recognizing-faces-in-the-wild.zip
```

In [0]:

In [0]:

In [0]:

In [0]:

MODEL 6

In [0]:

```
#https://www.kaggle.com/janpreets/just-another-feature-extractor-0-824-lb
```

In [34]:

```
def initialize_bias(shape, name=None):
    """
        The paper, http://www.cs.utoronto.ca/~gkoch/files/msc-thesis.pdf
        suggests to initialize CNN layer bias with mean as 0.5 and standard deviation of 0.01
    """
    return np.random.normal(loc = 0.5, scale = 1e-2, size = shape)

def initialize_weights(shape, name=None):
    """
        The paper, http://www.cs.utoronto.ca/~gkoch/files/msc-thesis.pdf
        suggests to initialize CNN layer weights with mean as 0.0 and standard deviation of 0.01
    """
    return np.random.normal(loc = 0.0, scale = 1e-2, size = shape)
```

In [0]:

```
from keras.layers import Conv2D, ZeroPadding2D, Activation, Input, concatenate
from keras.layers.core import Lambda, Flatten, Dense
from keras.layers.normalization import BatchNormalization
from keras.layers.pooling import MaxPooling2D, AveragePooling2D
from keras.models import Model
from keras.models import Model
from keras import backend as K
import pandas as pd
import numpy as np
import keras
from keras.models import *
from keras.layers import *
from keras.preprocessing import *
from keras.callbacks import *
from keras.optimizers import *
from tqdm import tqdm
import glob
from keras.applications.resnet50 import preprocess_input, decode_predictions

# from keras.applications.nasnet import preprocess_input, decode_predictions
# from keras.applications.densenet import preprocess_input, decode_predictions
from keras.preprocessing import image
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import sys

import cv2
from random import choice, sample
import datetime
from sklearn.metrics import roc_auc_score
from keras.callbacks import ReduceLROnPlateau, ModelCheckpoint, EarlyStopping
import threading

from keras.initializers import glorot_normal

import tensorflow as tf
import numpy as np
import os

from numpy import genfromtxt
from keras.layers import Conv2D, ZeroPadding2D, Activation
from keras.layers.normalization import BatchNormalization
```

In [ ]:

```
import tensorflow as tf
from tensorflow.python.keras import backend as K
sess = K.get_session()
#array = sess.run(your_tensor)
```

In [ ]:

```
# https://github.com/rcmalli/keras-vggface/blob/master/keras_vggface/utils.py
def preprocess_input(x, data_format=None, version=1):
    '''
    preprocess_input will take the numpy array of image and remove the unnecessary values
    The out ut numpy only contains thermal image of face which will have the umporatant chracters
    Here we subtract the values from image numpy array after subtraction redundant pixels are removed

    '''
    x_temp = np.copy(x)
    if data_format is None:
        data_format = K.image_data_format()
    assert data_format in {'channels_last', 'channels_first'}

    if version == 1:
        if data_format == 'channels_first':
            x_temp = x_temp[:, ::-1, ...]
            x_temp[:, 0, :, :] -= 93.5940
            x_temp[:, 1, :, :] -= 104.7624
            x_temp[:, 2, :, :] -= 129.1863
        else:
            x_temp = x_temp[..., ::-1]
            x_temp[..., 0] -= 93.5940
            x_temp[..., 1] -= 104.7624
            x_temp[..., 2] -= 129.1863

    elif version == 2:
        if data_format == 'channels_first':
            x_temp = x_temp[:, ::-1, ...]
            x_temp[:, 0, :, :] -= 91.4953
            x_temp[:, 1, :, :] -= 103.8827
            x_temp[:, 2, :, :] -= 131.0912
        else:
            x_temp = x_temp[..., ::-1]
            x_temp[..., 0] -= 91.4953
            x_temp[..., 1] -= 103.8827
            x_temp[..., 2] -= 131.0912
    else:
        raise NotImplementedError
    return x_temp
```

In [ ]:

```
# Function for reading images
def read_img(path, IMG_SIZE):
    img = image.load_img(path, target_size=(IMG_SIZE, IMG_SIZE))
    img = np.array(img).astype(np.float)
    return preprocess_input(img, version=2)

def gen(list_tuples, person_to_images_map, batch_size=16):
    ppl = list(person_to_images_map.keys())
    while True:
        batch_tuples = sample(list_tuples, batch_size // 2)
        labels = [1] * len(batch_tuples)
        while len(batch_tuples) < batch_size:
            p1 = choice(ppl)
            p2 = choice(ppl)
            if p1 != p2 and (p1, p2) not in list_tuples and (p2, p1) not in list_tuples:
                batch_tuples.append((p1, p2))
                labels.append(0)
        for x in batch_tuples:
            if not len(person_to_images_map[x[0]]):
                print(x[0])
        X1 = [choice(person_to_images_map[x[0]]) for x in batch_tuples]
        X1 = np.array([read_img(x,96) for x in X1])
        X2 = [choice(person_to_images_map[x[1]]) for x in batch_tuples]
        X2 = np.array([read_img(x,96) for x in X2])
        yield [X1, X2], labels
```

In [0]:

```
def inception_based_model():
    myInput = Input(shape=(96, 96, 3))

    x = ZeroPadding2D(padding=(3, 3), input_shape=(96, 96, 3))(myInput)
    x = Conv2D(64, (7, 7), strides=(2, 2), name='conv1')(x)
    x = BatchNormalization(axis=3, epsilon=0.00001, name='bn1')(x)
    x = Activation('relu')(x)
    x = ZeroPadding2D(padding=(1, 1))(x)
    x = MaxPooling2D(pool_size=3, strides=2)(x)
    x = Lambda(LRN2D, name='lrn_1')(x)
    x = Conv2D(64, (1, 1), name='conv2')(x)
    x = BatchNormalization(axis=3, epsilon=0.00001, name='bn2')(x)
    x = Activation('relu')(x)
    x = ZeroPadding2D(padding=(1, 1))(x)
    x = Conv2D(192, (3, 3), name='conv3')(x)
    x = BatchNormalization(axis=3, epsilon=0.00001, name='bn3')(x)
    x = Activation('relu')(x)
    x = Lambda(LRN2D, name='lrn_2')(x)
    x = ZeroPadding2D(padding=(1, 1))(x)
    x = MaxPooling2D(pool_size=3, strides=2)(x)

    # Inception3a
    inception_3a_3x3 = Conv2D(96, (1, 1), name='inception_3a_3x3_conv1')(x)
    inception_3a_3x3 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_3x3_bn1')(incepti
on_3a_3x3)
    inception_3a_3x3 = Activation('relu')(inception_3a_3x3)
    inception_3a_3x3 = ZeroPadding2D(padding=(1, 1))(inception_3a_3x3)
    inception_3a_3x3 = Conv2D(128, (3, 3), name='inception_3a_3x3_conv2')(inception_3a_3x3)
    inception_3a_3x3 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_3x3_bn2')(incepti
on_3a_3x3)
```

```
    inception_3a_3x3 = Activation('relu')(inception_3a_3x3)

    inception_3a_5x5 = Conv2D(16, (1, 1), name='inception_3a_5x5_conv1')(x)
    inception_3a_5x5 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_5x5_bn1')(incepti
on_3a_5x5)
    inception_3a_5x5 = Activation('relu')(inception_3a_5x5)
    inception_3a_5x5 = ZeroPadding2D(padding=(2, 2))(inception_3a_5x5)
    inception_3a_5x5 = Conv2D(32, (5, 5), name='inception_3a_5x5_conv2')(inception_3a_5x5)
    inception_3a_5x5 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_5x5_bn2')(incepti
on_3a_5x5)
    inception_3a_5x5 = Activation('relu')(inception_3a_5x5)

    inception_3a_pool = MaxPooling2D(pool_size=3, strides=2)(x)
    inception_3a_pool = Conv2D(32, (1, 1), name='inception_3a_pool_conv')(inception_3a_pool)
    inception_3a_pool = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_pool_bn')(incept
ion_3a_pool)
    inception_3a_pool = Activation('relu')(inception_3a_pool)
    inception_3a_pool = ZeroPadding2D(padding=((3, 4), (3, 4)))(inception_3a_pool)

    inception_3a_1x1 = Conv2D(64, (1, 1), name='inception_3a_1x1_conv')(x)
    inception_3a_1x1 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_1x1_bn')(inceptio
n_3a_1x1)
    inception_3a_1x1 = Activation('relu')(inception_3a_1x1)

    inception_3a = concatenate([inception_3a_3x3, inception_3a_5x5, inception_3a_pool, inception_3a_1x1
], axis=3)

    # Inception3b
    inception_3b_3x3 = Conv2D(96, (1, 1), name='inception_3b_3x3_conv1')(inception_3a)
    inception_3b_3x3 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_3x3_bn1')(incepti
on_3b_3x3)
    inception_3b_3x3 = Activation('relu')(inception_3b_3x3)
    inception_3b_3x3 = ZeroPadding2D(padding=(1, 1))(inception_3b_3x3)
    inception_3b_3x3 = Conv2D(128, (3, 3), name='inception_3b_3x3_conv2')(inception_3b_3x3)
    inception_3b_3x3 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_3x3_bn2')(incepti
on_3b_3x3)
    inception_3b_3x3 = Activation('relu')(inception_3b_3x3)

    inception_3b_5x5 = Conv2D(32, (1, 1), name='inception_3b_5x5_conv1')(inception_3a)
    inception_3b_5x5 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_5x5_bn1')(incepti
on_3b_5x5)
    inception_3b_5x5 = Activation('relu')(inception_3b_5x5)
    inception_3b_5x5 = ZeroPadding2D(padding=(2, 2))(inception_3b_5x5)
    inception_3b_5x5 = Conv2D(64, (5, 5), name='inception_3b_5x5_conv2')(inception_3b_5x5)
    inception_3b_5x5 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_5x5_bn2')(incepti
on_3b_5x5)
    inception_3b_5x5 = Activation('relu')(inception_3b_5x5)

    inception_3b_pool = AveragePooling2D(pool_size=(3, 3), strides=(3, 3))(inception_3a)
    inception_3b_pool = Conv2D(64, (1, 1), name='inception_3b_pool_conv')(inception_3b_pool)
    inception_3b_pool = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_pool_bn')(incept
ion_3b_pool)
    inception_3b_pool = Activation('relu')(inception_3b_pool)
    inception_3b_pool = ZeroPadding2D(padding=(4, 4))(inception_3b_pool)

    inception_3b_1x1 = Conv2D(64, (1, 1), name='inception_3b_1x1_conv')(inception_3a)
    inception_3b_1x1 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_1x1_bn')(inceptio
n_3b_1x1)
    inception_3b_1x1 = Activation('relu')(inception_3b_1x1)

    inception_3b = concatenate([inception_3b_3x3, inception_3b_5x5, inception_3b_pool, inception_3b_1x1
], axis=3)

    # Inception3c
    inception_3c_3x3 = conv2d_bn(inception_3b,
                                       layer='inception_3c_3x3',
                                       cv1_out=128,
                                       cv1_filter=(1, 1),
                                       cv2_out=256,
                                       cv2_filter=(3, 3),
                                       cv2_strides=(2, 2),
                                       padding=(1, 1))

    inception_3c_5x5 = conv2d_bn(inception_3b,
                                       layer='inception_3c_5x5',
                                       cv1_out=32,
                                       cv1_filter=(1, 1),
                                       cv2_out=64,
                                       cv2_filter=(5, 5),
                                       cv2_strides=(2, 2),
                                       padding=(2, 2))

    inception_3c_pool = MaxPooling2D(pool_size=3, strides=2)(inception_3b)
    inception_3c_pool = ZeroPadding2D(padding=((0, 1), (0, 1)))(inception_3c_pool)

    inception_3c = concatenate([inception_3c_3x3, inception_3c_5x5, inception_3c_pool], axis=3)

    #inception 4a
    inception_4a_3x3 = conv2d_bn(inception_3c,
                                       layer='inception_4a_3x3',
                                       cv1_out=96,
                                       cv1_filter=(1, 1),
                                       cv2_out=192,
                                       cv2_filter=(3, 3),
                                       cv2_strides=(1, 1),
                                       padding=(1, 1))
    inception_4a_5x5 = conv2d_bn(inception_3c,
                                       layer='inception_4a_5x5',
                                       cv1_out=32,
                                       cv1_filter=(1, 1),
                                       cv2_out=64,
                                       cv2_filter=(5, 5),
                                       cv2_strides=(1, 1),
                                       padding=(2, 2))

    inception_4a_pool = AveragePooling2D(pool_size=(3, 3), strides=(3, 3))(inception_3c)
    inception_4a_pool = conv2d_bn(inception_4a_pool,
                                        layer='inception_4a_pool',
                                        cv1_out=128,
                                        cv1_filter=(1, 1),
                                        padding=(2, 2))
    inception_4a_1x1 = conv2d_bn(inception_3c,
                                       layer='inception_4a_1x1',
                                       cv1_out=256,
                                       cv1_filter=(1, 1))
    inception_4a = concatenate([inception_4a_3x3, inception_4a_5x5, inception_4a_pool, inception_4a_1x1
], axis=3)

    #inception4e
    inception_4e_3x3 = conv2d_bn(inception_4a,
                                       layer='inception_4e_3x3',
                                       cv1_out=160,
                                       cv1_filter=(1, 1),
                                       cv2_out=256,
                                       cv2_filter=(3, 3),
                                       cv2_strides=(2, 2),
                                       padding=(1, 1))
    inception_4e_5x5 = conv2d_bn(inception_4a,
                                       layer='inception_4e_5x5',
                                       cv1_out=64,
                                       cv1_filter=(1, 1),
                                       cv2_out=128,
                                       cv2_filter=(5, 5),
                                       cv2_strides=(2, 2),
                                       padding=(2, 2))
    inception_4e_pool = MaxPooling2D(pool_size=3, strides=2)(inception_4a)
    inception_4e_pool = ZeroPadding2D(padding=((0, 1), (0, 1)))(inception_4e_pool)

    inception_4e = concatenate([inception_4e_3x3, inception_4e_5x5, inception_4e_pool], axis=3)

    #inception5a
    inception_5a_3x3 = conv2d_bn(inception_4e,
                                       layer='inception_5a_3x3',
                                       cv1_out=96,
                                       cv1_filter=(1, 1),
                                       cv2_out=384,
                                       cv2_filter=(3, 3),
                                       cv2_strides=(1, 1),
                                       padding=(1, 1))

    inception_5a_pool = AveragePooling2D(pool_size=(3, 3), strides=(3, 3))(inception_4e)
    inception_5a_pool = conv2d_bn(inception_5a_pool,
                                        layer='inception_5a_pool',
                                        cv1_out=96,
                                        cv1_out=96,
                                        cv1_filter=(1, 1),
                                        padding=(1, 1))
    inception_5a_1x1 = conv2d_bn(inception_4e,
                                       layer='inception_5a_1x1',
                                       cv1_out=256,
                                       cv1_filter=(1, 1))

    inception_5a = concatenate([inception_5a_3x3, inception_5a_pool, inception_5a_1x1], axis=3)

    #inception_5b
    inception_5b_3x3 = conv2d_bn(inception_5a,
                                       layer='inception_5b_3x3',
                                       cv1_out=96,
                                       cv1_filter=(1, 1),
                                       cv2_out=384,
                                       cv2_filter=(3, 3),
                                       cv2_strides=(1, 1),
                                       padding=(1, 1))
    inception_5b_pool = MaxPooling2D(pool_size=3, strides=2)(inception_5a)
    inception_5b_pool = conv2d_bn(inception_5b_pool,
                                        layer='inception_5b_pool',
                                        cv1_out=96,
                                        cv1_filter=(1, 1))
    inception_5b_pool = ZeroPadding2D(padding=(1, 1))(inception_5b_pool)

    inception_5b_1x1 = conv2d_bn(inception_5a,
                                       layer='inception_5b_1x1',
                                       cv1_out=256,
                                       cv1_filter=(1, 1))
    inception_5b = concatenate([inception_5b_3x3, inception_5b_pool, inception_5b_1x1], axis=3)

    av_pool = AveragePooling2D(pool_size=(3, 3), strides=(1, 1))(inception_5b)
    reshape_layer = Flatten()(av_pool)
    dense_layer = Dense(128, name='dense_layer')(reshape_layer)
    norm_layer = Lambda(lambda  x: K.l2_normalize(x, axis=1), name='norm_layer')(dense_layer)

    return Model(inputs=[myInput], outputs=norm_layer)
```

In [0]:

```
def LRN2D(x):
    return tf.nn.lrn(x, alpha=1e-4, beta=0.75)

def conv2d_bn(
  x,
  layer=None,
  cv1_out=None,
  cv1_filter=(1, 1),
  cv1_strides=(1, 1),
  cv2_out=None,
  cv2_filter=(3, 3),
  cv2_strides=(1, 1),
  padding=None,
):
    num = '' if cv2_out == None else '1'
    tensor = Conv2D(cv1_out, cv1_filter, strides=cv1_strides, name=layer+'_conv'+num)(x)
    tensor = BatchNormalization(axis=3, epsilon=0.00001, name=layer+'_bn'+num)(tensor)
    tensor = Activation('relu')(tensor)
    if padding == None:
        return tensor
    tensor = ZeroPadding2D(padding=padding)(tensor)
    if cv2_out == None:
        return tensor
    tensor = Conv2D(cv2_out, cv2_filter, strides=cv2_strides, name=layer+'_conv'+'2')(tensor)
    tensor = BatchNormalization(axis=3, epsilon=0.00001, name=layer+'_bn'+'2')(tensor)
    tensor = Activation('relu')(tensor)
    return tensor

def create_base_network(input_shape):
    mod = inception_based_model()
    new_mod = Model(mod.input,mod.layers[-5].output)
    return new_mod

def get_model6():
    input_shape = (96,96,3)
    input_shape = (96,96,3)
    input_a = Input(shape=input_shape)
    input_b = Input(shape=input_shape)
    base_network = create_base_network(input_shape)
    # because we re-use the same instance `base_network`,
    # the weights of the network
    # will be shared across the two branches
    processed_a = base_network(input_a)
    processed_b = base_network(input_b)
    # extract features from detector
    x_detector = processed_a
    shape_detector = processed_a.shape
    # extract features from extractor , same with detector for symmetry DxD model
    shape_extractor = processed_b.shape
    x_extractor = processed_b
    # rehape to (minibatch_size, total_pixels, filter_size)
    x_detector = keras.layers.Reshape([shape_detector[1] * shape_detector[2] , shape_detector[-1]])(x_d
etector)
    x_extractor = keras.layers.Reshape([shape_extractor[1] * shape_extractor[2] , shape_extractor[-1]])
(x_extractor)
    # outer products of features, output shape=(minibatch_size, filter_size_detector*filter_size_extrac
tor)
    x = keras.layers.Lambda(outer_product)([x_detector, x_extractor])
    # rehape to (minibatch_size, filter_size_detector*filter_size_extractor)
    x = keras.layers.Reshape([shape_detector[-1]*shape_extractor[-1]])(x)
    # signed square-root
    x = keras.layers.Lambda(signed_sqrt)(x)
    # L2 normalization
    x = keras.layers.Lambda(L2_norm)(x)
    ###
    ### attach FC-Layer
    ###
    x = Dense(100, activation="relu")(x)
    x = Dropout(0.01)(x)
    #out = Dense(units=1,kernel_regularizer=keras.regularizers.l2(1e-8),kernel_initializer='glorot_norm
al',activation="sigmoid")(x)
    out = Dense(units=1,kernel_initializer='glorot_normal',activation="sigmoid")(x)
    model = Model([input_a, input_b], out)
    model.compile(loss="binary_crossentropy", metrics=['acc',auc], optimizer=Adam(0.001))
    print(model.summary())
    return model
```

In [0]:

```

```

In [0]:

```
file_path6 = "./model6.h5"

callbacks_list6 = [keras.callbacks.EarlyStopping(monitor='val_loss', patience=50, verbose=1, min_delta=
1e-4),
                   keras.callbacks.ReduceLROnPlateau(monitor='val_loss', factor=0.1, patience=20, verbo
se=1, epsilon=1e-4),
                   keras.callbacks.ModelCheckpoint(file_path6,save_best_only=True, verbose=1, save_weig
hts_only=True,monitor='val_auc',mode='max'),
                   ]

#callbacks_list6 = [checkpoint6, reduce_on_plateau6, early_stopping6]
#callbacks_list6 = [checkpoint6]

model6 = get_model6()
```

In [0]:

```
model6_hist = model6.fit_generator(gen(train, train_person_to_images_map, batch_size=16), use_multiproc
essing=True,
                                   validation_data=gen(val, val_person_to_images_map, batch_size=16), e
pochs=150, verbose=1,
                                   workers = 16, callbacks=callbacks_list6, steps_per_epoch=300, valida
tion_steps=200)
```

```
Epoch 00133: val auc did not improve from 0.76961
Epoch 134/150
300/300 [==============================] - 55s 182ms/step - loss: 0.4515 - acc: 0.7856 - auc: 0.8625 - val_loss: 0.5869 - val_acc: 0.7006 - val_auc: 0.7702
300/300 [==============================] - 55s 182ms/step - loss: 0.4515 - acc: 0.7856 - auc: 0.8625 - val_loss: 0.5869 - val_acc: 0.7006 - val_auc: 0.7702

Epoch 00134: val_auc improved from 0.76961 to 0.77016, saving model to ./model6.h5

Epoch 00134: val_auc improved from 0.76961 to 0.77016, saving model to ./model6.h5
Epoch 135/150
  1/300 [..............................] - ETA: 47s - loss: 0.5908 - acc: 0.7500 - auc: 0.8281Epoch 135/150
300/300 [==============================] - 51s 172ms/step - loss: 0.4592 - acc: 0.7800 - auc: 0.8529 - val_loss: 0.5900 - val_acc: 0.6947 - val_auc: 0.7600
300/300 [==============================] - 51s 172ms/step - loss: 0.4592 - acc: 0.7800 - auc: 0.8529 - val_loss: 0.5900 - val_acc: 0.6947 - val_auc: 0.7600

Epoch 00135: val_auc did not improve from 0.77016
Epoch 136/150

Epoch 00135: val_auc did not improve from 0.77016
Epoch 136/150
300/300 [==============================] - 52s 172ms/step - loss: 0.4572 - acc: 0.7779 - auc: 0.8541 - val_loss: 0.5977 - val_acc: 0.6934 - val_auc: 0.7534
300/300 [==============================] - 52s 172ms/step - loss: 0.4572 - acc: 0.7779 - auc: 0.8541 - val_loss: 0.5977 - val_acc: 0.6934 - val_auc: 0.7534

Epoch 00136: val_auc did not improve from 0.77016
Epoch 137/150
  1/300 [..............................] - ETA: 56s - loss: 0.5061 - acc: 0.7500 - auc: 0.8594
Epoch 00136: val_auc did not improve from 0.77016
Epoch 137/150
300/300 [==============================] - 52s 174ms/step - loss: 0.4610 - acc: 0.7765 - auc: 0.8529 - val_loss: 0.5922 - val_acc: 0.6913 - val_auc: 0.7585
300/300 [==============================] - 52s 174ms/step - loss: 0.4610 - acc: 0.7765 - auc: 0.8529 - val_loss: 0.5922 - val_acc: 0.6913 - val_auc: 0.7585

Epoch 00137: val_auc did not improve from 0.77016
Epoch 138/150

Epoch 00137: val_auc did not improve from 0.77016
Epoch 138/150
300/300 [==============================] - 52s 175ms/step - loss: 0.4494 - acc: 0.7790 - auc: 0.8625 - val_loss: 0.6292 - val_acc: 0.6684 - val_auc: 0.7419
300/300 [==============================] - 52s 175ms/step - loss: 0.4494 - acc: 0.7790 - auc: 0.8625 - val_loss: 0.6292 - val_acc: 0.6684 - val_auc: 0.7419

Epoch 00138: val_auc did not improve from 0.77016
Epoch 00138: early stopping

Epoch 00138: val_auc did not improve from 0.77016
Epoch 00138: early stopping
```

In [0]:

Ensembling with best models

In [7]:

```python
from collections import defaultdict
from glob import glob
from random import choice, sample

#import cv2
import numpy as np
import pandas as pd
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau
from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Concatenate, Multiply
, Dropout, Subtract, Add, Conv2D
from keras.models import Model
from keras.preprocessing import image
from keras.optimizers import Adam
from keras_vggface.utils import preprocess_input
from keras_vggface.vggface import VGGFace

# import regularizer
from keras.regularizers import l1
from keras.regularizers import l2

import h5py
import itertools
```

In [8]:

```python
from collections import defaultdict
from glob import glob
from random import choice, sample
import matplotlib.pyplot as plt
from tqdm import tqdm
import numpy as np
import pandas as pd
import pickle
import gc, psutil, random, base64, h5py, os,shutil
#import cv2
pd.options.mode.chained_assignment = None
pd.options.display.max_columns = 9999
pd.options.display.float_format = '{:20, .2f}'.format
from keras import backend as K

from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Concatenate, Multiply
, Dropout, Subtract, Add
from keras_vggface.vggface import VGGFace
from pathlib import Path

from PIL import Image
from io import BytesIO
from IPython.display import HTML
import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
import plotly.tools as tls

import warnings
warnings.filterwarnings('ignore')
from keras.layers import Conv2D, ZeroPadding2D, Activation, concatenate
from keras.layers.pooling import MaxPooling2D, AveragePooling2D
from keras.layers.core import Lambda
import keras
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau,EarlyStopping
from keras.layers import Conv1D, BatchNormalization,Reshape, LeakyReLU
from keras.preprocessing import image
from keras.optimizers import Adam,SGD,rmsprop
from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Conv2D,MaxPooling1D,G
lobalAveragePooling1D,LSTM
from keras.models import Model
from keras.losses import binary_crossentropy
import tensorflow as tf
from sklearn.metrics import roc_auc_score
from keras import regularizers
%pylab inline
import matplotlib.image as mpimg
```

```
Populating the interactive namespace from numpy and matplotlib
```

In [9]:

```
def auc(y_true, y_pred):
    return tf.py_function(roc_auc_score, (y_true, y_pred), tf.double)
```

In [10]:

```
def gen(list_tuples, person_to_images_map, batch_size=16):
    ppl = list(person_to_images_map.keys())
    while True:
        batch_tuples = sample(list_tuples, batch_size // 2)
        labels = [1] * len(batch_tuples)
        while len(batch_tuples) < batch_size:
            p1 = choice(ppl)
            p2 = choice(ppl)

            if p1 != p2 and (p1, p2) not in list_tuples and (p2, p1) not in list_tuples:
                batch_tuples.append((p1, p2))
                labels.append(0)

        for x in batch_tuples:
            if not len(person_to_images_map[x[0]]):
                print(x[0])

        X1 = [choice(person_to_images_map[x[0]]) for x in batch_tuples]
        X1 = np.array([read_img(x, 197) for x in X1])

        X2 = [choice(person_to_images_map[x[1]]) for x in batch_tuples]
        X2 = np.array([read_img(x, 197) for x in X2])
        yield [X1, X2], labels
```

MODEL 1

In [11]:

```
# Preparing baseline_model1
def baseline_model1():
    input_1 = Input(shape=(197, 197, 3))
    input_2 = Input(shape=(197, 197, 3))

    base_model1 = VGGFace(model='resnet50', include_top=False)

    for x in base_model1.layers[:-3]:
        x.trainable = True
#    for x in base_model1.layers[-3:]:
#        x.trainable=False

    x1 = base_model1(input_1)
    x2 = base_model1(input_2)

    x1 = Concatenate(axis=-1)([GlobalMaxPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])

    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])

    x1_ = Multiply()([x1, x1])
    x2_ = Multiply()([x2, x2])
    x4 = Subtract()([x1_, x2_])
    x = Concatenate(axis=-1)([x4, x3])

    x = Dense(100, activation="relu", activity_regularizer=l1(0.001))(x)
    x = Dropout(0.3)(x)
    x = Dense(25, activation="relu", activity_regularizer=l1(0.001))(x)
    x = Dropout(0.01)(x)
    out = Dense(1, activation="sigmoid")(x)

    model1 = Model([input_1, input_2], out)

    model1.compile(loss="binary_crossentropy", metrics=['acc'], optimizer=Adam(0.00001))

    model1.summary()

    return model1
```

In [12]:

```
model1 = baseline_model1()
```

```
Model: "model_1"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_1 (InputLayer)            (None, 197, 197, 3)  0
__________________________________________________________________________________________________
input_2 (InputLayer)            (None, 197, 197, 3)  0
__________________________________________________________________________________________________
vggface_resnet50 (Model)        multiple             23561152    input_1[0][0]
                                                                 input_2[0][0]
__________________________________________________________________________________________________
global_max_pooling2d_1 (GlobalM (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_average_pooling2d_1 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_max_pooling2d_2 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
global_average_pooling2d_2 (Glo (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
concatenate_1 (Concatenate)     (None, 4096)         0           global_max_pooling2d_1[0][0]
                                                                 global_average_pooling2d_1[0][0]
```

```
__________________________________________________________________________________________________
concatenate_2 (Concatenate)      (None, 4096)         0           global_max_pooling2d_2[0][0]
                                                                  global_average_pooling2d_2[0][0]
__________________________________________________________________________________________________
multiply_2 (Multiply)            (None, 4096)         0           concatenate_1[0][0]
                                                                  concatenate_1[0][0]
__________________________________________________________________________________________________
multiply_3 (Multiply)            (None, 4096)         0           concatenate_2[0][0]
                                                                  concatenate_2[0][0]
__________________________________________________________________________________________________
subtract_1 (Subtract)            (None, 4096)         0           concatenate_1[0][0]
                                                                  concatenate_2[0][0]
__________________________________________________________________________________________________
subtract_2 (Subtract)            (None, 4096)         0           multiply_2[0][0]
                                                                  multiply_3[0][0]
__________________________________________________________________________________________________
multiply_1 (Multiply)            (None, 4096)         0           subtract_1[0][0]
                                                                  subtract_1[0][0]
__________________________________________________________________________________________________
concatenate_3 (Concatenate)      (None, 8192)         0           subtract_2[0][0]
                                                                  multiply_1[0][0]
__________________________________________________________________________________________________
dense_1 (Dense)                  (None, 100)          819300      concatenate_3[0][0]
__________________________________________________________________________________________________
dropout_1 (Dropout)              (None, 100)          0           dense_1[0][0]
__________________________________________________________________________________________________
dense_2 (Dense)                  (None, 25)           2525        dropout_1[0][0]
__________________________________________________________________________________________________
dropout_2 (Dropout)              (None, 25)           0           dense_2[0][0]
__________________________________________________________________________________________________
dense_3 (Dense)                  (None, 1)            26          dropout_2[0][0]
==================================================================================================
Total params: 24,383,003
Trainable params: 24,329,883
Non-trainable params: 53,120
__________________________________________________________________________________________________
```

In [ ]:

```

```

In [ ]:

```

```

MODEL 2

In [ ]:

```

```

In [13]:

```
def baseline_model2():
    input_1 = Input(shape=(197, 197, 3))
    input_2 = Input(shape=(197, 197, 3))

    base_model2 = VGGFace(model='resnet50', include_top=False)

    for x in base_model2.layers[:-3]:
        x.trainable = True

    x1 = base_model2(input_1)
    x2 = base_model2(input_2)

    x1 = Concatenate(axis=-1)([GlobalMaxPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])

    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])

    x1_ = Multiply()([x1, x1])
    x2_ = Multiply()([x2, x2])
    x4 = Subtract()([x1_, x2_])
    x = Concatenate(axis=-1)([x4, x3])
    x = Dropout(0.3)(x)
    x = Dense(100, activation="relu", activity_regularizer=l1(0.001))(x)
    x = Dropout(0.1)(x)
    out = Dense(1, activation="sigmoid")(x)

    model2 = Model([input_1, input_2], out)

    model2.compile(loss="binary_crossentropy", metrics=['acc', auc], optimizer=Adam(0.00001))

    model2.summary()

    return model2
```

In [14]:

```
model2 = baseline_model2()
#model.load_weights(file_path)
```

```
Model: "model_2"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to                     
==================================================================================================
input_4 (InputLayer)            (None, 197, 197, 3)  0                                            
__________________________________________________________________________________________________
input_5 (InputLayer)            (None, 197, 197, 3)  0                                            
__________________________________________________________________________________________________
vggface_resnet50 (Model)        multiple             23561152    input_4[0][0]                    
                                                                 input_5[0][0]                    
__________________________________________________________________________________________________
global_max_pooling2d_3 (GlobalM (None, 2048)         0           vggface_resnet50[1][0]           
__________________________________________________________________________________________________
global_average_pooling2d_3 (Glo (None, 2048)         0           vggface_resnet50[1][0]           
__________________________________________________________________________________________________
global_max_pooling2d_4 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]           
__________________________________________________________________________________________________
global_average_pooling2d_4 (Glo (None, 2048)         0           vggface_resnet50[2][0]           
__________________________________________________________________________________________________
concatenate_4 (Concatenate)     (None, 4096)         0           global_max_pooling2d_3[0][0]     
                                                                 global_average_pooling2d_3[0][0] 
__________________________________________________________________________________________________
concatenate_5 (Concatenate)     (None, 4096)         0           global_max_pooling2d_4[0][0]     
                                                                 global_average_pooling2d_4[0][0] 
__________________________________________________________________________________________________
multiply_5 (Multiply)           (None, 4096)         0           concatenate_4[0][0]              
                                                                 concatenate_4[0][0]              
__________________________________________________________________________________________________
multiply_6 (Multiply)           (None, 4096)         0           concatenate_5[0][0]              
                                                                 concatenate_5[0][0]              
__________________________________________________________________________________________________
subtract_3 (Subtract)           (None, 4096)         0           concatenate_4[0][0]              
                                                                 concatenate_5[0][0]              
__________________________________________________________________________________________________
subtract_4 (Subtract)           (None, 4096)         0           multiply_5[0][0]                 
                                                                 multiply_6[0][0]                 
__________________________________________________________________________________________________
multiply_4 (Multiply)           (None, 4096)         0           subtract_3[0][0]                 
                                                                 subtract_3[0][0]                 
__________________________________________________________________________________________________
concatenate_6 (Concatenate)     (None, 8192)         0           subtract_4[0][0]                 
                                                                 multiply_4[0][0]                 
__________________________________________________________________________________________________
dropout_3 (Dropout)             (None, 8192)         0           concatenate_6[0][0]              
__________________________________________________________________________________________________
dense_4 (Dense)                 (None, 100)          819300      dropout_3[0][0]                  
__________________________________________________________________________________________________
dropout_4 (Dropout)             (None, 100)          0           dense_4[0][0]                    
__________________________________________________________________________________________________
dense_5 (Dense)                 (None, 1)            101         dropout_4[0][0]                  
==================================================================================================
Total params: 24,380,553
Trainable params: 24,327,433
Non-trainable params: 53,120
```

```
________________________________________________________________________________________________________
```

In [ ]:

```

```

MODEL 3

In [ ]:

```

```

In [15]:

```python
def baseline_model3_1():
    input_1 = Input(shape=(224, 224, 3))
    input_2 = Input(shape=(224, 224, 3))

    base_model3 = VGGFace(model='resnet50', include_top=False)

    for layer in base_model3.layers[:-3]:
        layer.trainable = True

    x1 = base_model3(input_1)
    x2 = base_model3(input_2)

    merged_add = Add()([x1, x2])
    merged_sub = Subtract()([x1,x2])

    merged_add = Conv2D(100 , [1,1] )(merged_add)
    merged_sub = Conv2D(100 , [1,1] )(merged_sub)

    merged = Concatenate(axis=-1)([merged_add, merged_sub])

    merged = Flatten()(merged)
    merged = Dropout(0.2)(merged)
    merged = Dense(100, activation="relu", activity_regularizer=l1(0.001))(merged)
    merged = Dropout(0.2)(merged)
    merged = Dense(25, activation="relu", activity_regularizer=l1(0.001))(merged)
    merged = Dropout(0.2)(merged)
    out = Dense(1, activation="sigmoid")(merged)
    model3 = Model([input_1, input_2], out)
    model3.compile(loss="binary_crossentropy", metrics=['acc', auc], optimizer=Adam(0.00001))
    model3.summary()
    return model3
```

In [16]:

```python
model3 = baseline_model3_1()
```

```
Model: "model_3"
________________________________________________________________________________________________________
Layer (type)                     Output Shape          Param #     Connected to
========================================================================================================
input_7 (InputLayer)             (None, 224, 224, 3)   0
________________________________________________________________________________________________________
input_8 (InputLayer)             (None, 224, 224, 3)   0
________________________________________________________________________________________________________
vggface_resnet50 (Model)         multiple              23561152    input_7[0][0]
                                                                   input_8[0][0]
________________________________________________________________________________________________________
add_49 (Add)                     (None, 1, 1, 2048)    0           vggface_resnet50[1][0]
                                                                   vggface_resnet50[2][0]
________________________________________________________________________________________________________
subtract_5 (Subtract)            (None, 1, 1, 2048)    0           vggface_resnet50[1][0]
                                                                   vggface_resnet50[2][0]
________________________________________________________________________________________________________
conv2d_1 (Conv2D)                (None, 1, 1, 100)     204900      add_49[0][0]
________________________________________________________________________________________________________
conv2d_2 (Conv2D)                (None, 1, 1, 100)     204900      subtract_5[0][0]
```

```
________________________________________________________________________________________________
concatenate_7 (Concatenate)     (None, 1, 1, 200)    0           conv2d_1[0][0]
                                                                 conv2d_2[0][0]
________________________________________________________________________________________________
flatten_1 (Flatten)             (None, 200)          0           concatenate_7[0][0]
________________________________________________________________________________________________
dropout_5 (Dropout)             (None, 200)          0           flatten_1[0][0]
________________________________________________________________________________________________
dense_6 (Dense)                 (None, 100)          20100       dropout_5[0][0]
________________________________________________________________________________________________
dropout_6 (Dropout)             (None, 100)          0           dense_6[0][0]
________________________________________________________________________________________________
dense_7 (Dense)                 (None, 25)           2525        dropout_6[0][0]
________________________________________________________________________________________________
dropout_7 (Dropout)             (None, 25)           0           dense_7[0][0]
________________________________________________________________________________________________
dense_8 (Dense)                 (None, 1)            26          dropout_7[0][0]
================================================================================================
Total params: 23,993,603
Trainable params: 23,940,483
Non-trainable params: 53,120
________________________________________________________________________________________________
```

In [ ]:

```

```

In [ ]:

```

```

MODEL 4

In [ ]:

```

```

In [17]:

```python
def baseline_model4():
    input_1 = Input(shape=(224, 224, 3))
    input_2 = Input(shape=(224, 224, 3))

    base_model4 = VGGFace(model='resnet50', include_top=False)

    for x in base_model4.layers[:-3]:
        x.trainable = True

    x1 = base_model4(input_1)
    x2 = base_model4(input_2)

    x1 = Concatenate(axis=-1)([GlobalMaxPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])

    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])

    x = Multiply()([x1, x2])

    x = Concatenate(axis=-1)([x, x3])
    x = BatchNormalization()(x)
    x = Dense(128,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.4)(x)
    x = Dense(64,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.4)(x)
    x = Dense(32,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.3)(x)
    x = Dense(16,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Activation('relu')(x)
    x = Dropout(0.3)(x)
    out = Dense(1, activation="sigmoid")(x)
    model4 = Model([input_1, input_2], out)
    model4.compile(loss="binary_crossentropy", metrics=['acc',auc], optimizer=Adam(0.00001))
    model4.summary()
    return model4
```

In [18]:

```
model4 = baseline_model4()
```

```
Model: "model_4"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_10 (InputLayer)           (None, 224, 224, 3)  0
__________________________________________________________________________________________________
input_11 (InputLayer)           (None, 224, 224, 3)  0
__________________________________________________________________________________________________
vggface_resnet50 (Model)        multiple             23561152    input_10[0][0]
                                                                 input_11[0][0]
__________________________________________________________________________________________________
global_max_pooling2d_5 (GlobalM (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_average_pooling2d_5 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_max_pooling2d_6 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
global_average_pooling2d_6 (Glo (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
concatenate_8 (Concatenate)     (None, 4096)         0           global_max_pooling2d_5[0][0]
                                                                 global_average_pooling2d_5[0][0]
__________________________________________________________________________________________________
concatenate_9 (Concatenate)     (None, 4096)         0           global_max_pooling2d_6[0][0]
                                                                 global_average_pooling2d_6[0][0]
__________________________________________________________________________________________________
subtract_6 (Subtract)           (None, 4096)         0           concatenate_8[0][0]
                                                                 concatenate_9[0][0]
__________________________________________________________________________________________________
multiply_8 (Multiply)           (None, 4096)         0           concatenate_8[0][0]
                                                                 concatenate_9[0][0]
__________________________________________________________________________________________________
multiply_7 (Multiply)           (None, 4096)         0           subtract_6[0][0]
                                                                 subtract_6[0][0]
__________________________________________________________________________________________________
concatenate_10 (Concatenate)    (None, 8192)         0           multiply_8[0][0]
                                                                 multiply_7[0][0]
__________________________________________________________________________________________________
batch_normalization_1 (BatchNor (None, 8192)         32768       concatenate_10[0][0]
__________________________________________________________________________________________________
dense_9 (Dense)                 (None, 128)          1048704     batch_normalization_1[0][0]
__________________________________________________________________________________________________
activation_197 (Activation)     (None, 128)          0           dense_9[0][0]
__________________________________________________________________________________________________
dropout_8 (Dropout)             (None, 128)          0           activation_197[0][0]
__________________________________________________________________________________________________
dense_10 (Dense)                (None, 64)           8256        dropout_8[0][0]
__________________________________________________________________________________________________
activation_198 (Activation)     (None, 64)           0           dense_10[0][0]
__________________________________________________________________________________________________
dropout_9 (Dropout)             (None, 64)           0           activation_198[0][0]
__________________________________________________________________________________________________
dense_11 (Dense)                (None, 32)           2080        dropout_9[0][0]
__________________________________________________________________________________________________
activation_199 (Activation)     (None, 32)           0           dense_11[0][0]
__________________________________________________________________________________________________
dropout_10 (Dropout)            (None, 32)           0           activation_199[0][0]
__________________________________________________________________________________________________
dense_12 (Dense)                (None, 16)           528         dropout_10[0][0]
__________________________________________________________________________________________________
activation_200 (Activation)     (None, 16)           0           dense_12[0][0]
__________________________________________________________________________________________________
dropout_11 (Dropout)            (None, 16)           0           activation_200[0][0]
__________________________________________________________________________________________________
dense_13 (Dense)                (None, 1)            17          dropout_11[0][0]
```

```
dense_13 (Dense)                (None, 1)            17          dropout_11[0][0]
==================================================================================================
Total params: 24,653,505
Trainable params: 24,584,001
Non-trainable params: 69,504
__________________________________________________________________________________________________
```

In [ ]:

MODEL 5

In [ ]:

In [19]:

```
from random import choice, sample
```

In [20]:

```
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau,EarlyStopping
from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D,MaxPooling1D,GlobalAve
ragePooling1D,LSTM
from keras.layers import Concatenate, Multiply, Dropout, Subtract, Add, Conv2D,Conv1D
from keras.models import Model
from keras.layers import BatchNormalization,Reshape
from keras.preprocessing import image
from keras.optimizers import Adam,SGD,rmsprop
import h5py
from keras.regularizers import l2
from keras.losses import binary_crossentropy
import tensorflow as tf
from sklearn.metrics import roc_auc_score
from keras.layers import LeakyReLU
from keras import regularizers
from keras.initializers  import RandomNormal

def auc(y_true, y_pred):
    return tf.py_function(roc_auc_score, (y_true, y_pred), tf.double)
```

In [21]:

```
def siamese_model_1():
    input_1 = Input(shape=(224, 224, 3))
    input_2 = Input(shape=(224, 224, 3))
    base_model = VGGFace(model='resnet50', include_top=False,input_shape=(224, 224, 3))

    for x in base_model.layers[:-3]:
        x.trainable = True

    x1 = base_model(input_1)
    x2 = base_model(input_2)

    x1 = Concatenate(axis=-1)([GlobalAvgPool2D()(x1), GlobalAvgPool2D()(x1)])
    x2 = Concatenate(axis=-1)([GlobalMaxPool2D()(x2), GlobalAvgPool2D()(x2)])
    x3 = Subtract()([x1, x2])
    x3 = Multiply()([x3, x3])
    x1_ = Multiply()([x1, x1])
    x2_ = Multiply()([x2, x2])
    x4  = Subtract()([x1_, x2_])
    x   = Concatenate(axis=-1)([x4, x3])
    x   = Reshape(input_shape=(8192,), target_shape=(8192, 1))(x)
    x = Conv1D(filters=100,
               kernel_size=8,
               strides=8,input_shape=( 8192,1),
               activation='relu',
               padding='same')(x)
               padding='same')(x)
    x = Conv1D( 100,8,activation= 'relu')(x)
    x = LSTM(128,kernel_initializer='glorot_normal',recurrent_dropout=0.5,kernel_regularizer=l2(0.001),
return_sequences=True)(x)
    #x = LSTM(128, return_sequences=True)(x)
    x = Dropout(0.4)(x)
    x = LSTM(128,kernel_initializer='glorot_normal',recurrent_dropout=0.5,kernel_regularizer=l2(0.001),
return_sequences=True)(x)
    #x = LSTM(128, return_sequences=True)(x)
    x = BatchNormalization ()(x)
    x = Flatten()(x)
    x = Dense(100,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.3)(x)
    x = Dense(64,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.2)(x)
    x = Dense(32,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.2)(x)
    x = Dense(16,kernel_initializer='glorot_normal',kernel_regularizer=l2(0.001))(x)
    x = Activation('relu')(x)
    x = Dropout(0.1)(x)
    out = Dense(1, activation="sigmoid")(x)
    model = Model([input_1, input_2], out)
    model.compile(loss="binary_crossentropy", metrics=['acc',auc], optimizer=Adam(0.00001,decay=1e-6))
    model.summary()
    return model
```

In [ ]:

```

```

In [22]:

```
model5 = siamese_model_1()
```

```
Model: "model_5"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_13 (InputLayer)           (None, 224, 224, 3)  0
__________________________________________________________________________________________________
input_14 (InputLayer)           (None, 224, 224, 3)  0
__________________________________________________________________________________________________
vggface_resnet50 (Model)        (None, 1, 1, 2048)   23561152    input_13[0][0]
                                                                 input_14[0][0]
__________________________________________________________________________________________________
global_average_pooling2d_7 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_average_pooling2d_8 (Glo (None, 2048)         0           vggface_resnet50[1][0]
__________________________________________________________________________________________________
global_max_pooling2d_7 (GlobalM (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
global_average_pooling2d_9 (Glo (None, 2048)         0           vggface_resnet50[2][0]
__________________________________________________________________________________________________
concatenate_11 (Concatenate)    (None, 4096)         0           global_average_pooling2d_7[0][0]
                                                                 global_average_pooling2d_8[0][0]
__________________________________________________________________________________________________
concatenate_12 (Concatenate)    (None, 4096)         0           global_max_pooling2d_7[0][0]
                                                                 global_average_pooling2d_9[0][0]
__________________________________________________________________________________________________
multiply_10 (Multiply)          (None, 4096)         0           concatenate_11[0][0]
                                                                 concatenate_11[0][0]
__________________________________________________________________________________________________
multiply_11 (Multiply)          (None, 4096)         0           concatenate_12[0][0]
                                                                 concatenate_12[0][0]
__________________________________________________________________________________________________
subtract_7 (Subtract)           (None, 4096)         0           concatenate_11[0][0]
                                                                 concatenate_12[0][0]
__________________________________________________________________________________________________
subtract_8 (Subtract)           (None, 4096)         0           multiply_10[0][0]
                                                                 multiply_11[0][0]
__________________________________________________________________________________________________
multiply_9 (Multiply)           (None, 4096)         0           subtract_7[0][0]
```

```
multiply_9 (Multiply)            (None, 4096)         0           subtract_7[0][0]
                                                                  subtract_7[0][0]
__________________________________________________________________________________________________
concatenate_13 (Concatenate)     (None, 8192)         0           subtract_8[0][0]
                                                                  multiply_9[0][0]
__________________________________________________________________________________________________
reshape_1 (Reshape)              (None, 8192, 1)      0           concatenate_13[0][0]
__________________________________________________________________________________________________
conv1d_1 (Conv1D)                (None, 1024, 100)    900         reshape_1[0][0]
__________________________________________________________________________________________________
conv1d_2 (Conv1D)                (None, 1017, 100)    80100       conv1d_1[0][0]
__________________________________________________________________________________________________
lstm_1 (LSTM)                    (None, 1017, 128)    117248      conv1d_2[0][0]
__________________________________________________________________________________________________
dropout_12 (Dropout)             (None, 1017, 128)    0           lstm_1[0][0]
__________________________________________________________________________________________________
lstm_2 (LSTM)                    (None, 1017, 128)    131584      dropout_12[0][0]
__________________________________________________________________________________________________
batch_normalization_2 (BatchNor (None, 1017, 128)    512         lstm_2[0][0]
__________________________________________________________________________________________________
flatten_2 (Flatten)              (None, 130176)       0           batch_normalization_2[0][0]
__________________________________________________________________________________________________
dense_14 (Dense)                 (None, 100)          13017700    flatten_2[0][0]
__________________________________________________________________________________________________
activation_250 (Activation)      (None, 100)          0           dense_14[0][0]
__________________________________________________________________________________________________
dropout_13 (Dropout)             (None, 100)          0           activation_250[0][0]
__________________________________________________________________________________________________
dense_15 (Dense)                 (None, 64)           6464        dropout_13[0][0]
__________________________________________________________________________________________________
activation_251 (Activation)      (None, 64)           0           dense_15[0][0]
__________________________________________________________________________________________________
dropout_14 (Dropout)             (None, 64)           0           activation_251[0][0]
__________________________________________________________________________________________________
dense_16 (Dense)                 (None, 32)           2080        dropout_14[0][0]
__________________________________________________________________________________________________
activation_252 (Activation)      (None, 32)           0           dense_16[0][0]
__________________________________________________________________________________________________
dropout_15 (Dropout)             (None, 32)           0           activation_252[0][0]
__________________________________________________________________________________________________
dense_17 (Dense)                 (None, 16)           528         dropout_15[0][0]
__________________________________________________________________________________________________
activation_253 (Activation)      (None, 16)           0           dense_17[0][0]
__________________________________________________________________________________________________
dropout_16 (Dropout)             (None, 16)           0           activation_253[0][0]
__________________________________________________________________________________________________
dense_18 (Dense)                 (None, 1)            17          dropout_16[0][0]
==================================================================================================
Total params: 36,918,285
Trainable params: 36,864,909
Non-trainable params: 53,376
__________________________________________________________________________________________________
```

In [ ]:

```

```

MODEL 6

In [23]:

```
from collections import defaultdict
from glob import glob
from random import choice, sample
import matplotlib.pyplot as plt
from tqdm import tqdm
import numpy as np
import pandas as pd
import pickle
import gc, psutil, random, base64, h5py, os,shutil
#import cv2
pd.options.mode.chained_assignment = None
pd.options.display.max_columns = 9999
pd.options.display.float_format = '{:20,.2f}'.format
pd.options.display.float_format = '{:20,.2f}'.format
from keras import backend as K

from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Concatenate, Multiply, Dropout, Subtract, Add
from keras_vggface.vggface import VGGFace
from pathlib import Path

from PIL import Image
from io import BytesIO
from IPython.display import HTML
import plotly.offline as py
py.init_notebook_mode(connected=True)
import plotly.graph_objs as go
import plotly.tools as tls

import warnings
warnings.filterwarnings('ignore')
from keras.layers import Conv2D, ZeroPadding2D, Activation, concatenate
from keras.layers.pooling import MaxPooling2D, AveragePooling2D
from keras.layers.core import Lambda
import keras
from keras.callbacks import ModelCheckpoint, ReduceLROnPlateau,EarlyStopping
from keras.layers import Conv1D, BatchNormalization,Reshape, LeakyReLU
from keras.preprocessing import image
from keras.optimizers import Adam,SGD,rmsprop
from keras.layers import Input, Dense, Flatten, GlobalMaxPool2D, GlobalAvgPool2D, Conv2D,MaxPooling1D,GlobalAveragePooling1D,LSTM
from keras.models import Model
from keras.losses import binary_crossentropy
import tensorflow as tf
from sklearn.metrics import roc_auc_score
from keras import regularizers
%pylab inline
import matplotlib.image as mpimg
```

```
Populating the interactive namespace from numpy and matplotlib
```

In [24]:

```
from keras.layers import Conv2D, ZeroPadding2D, Activation, Input, concatenate
from keras.layers.core import Lambda, Flatten, Dense
from keras.layers.normalization import BatchNormalization
from keras.layers.pooling import MaxPooling2D, AveragePooling2D
from keras.models import Model
from keras import backend as K
import pandas as pd
import numpy as np
import keras
from keras.models import *
from keras.layers import *
from keras.preprocessing import *
from keras.callbacks import *
from keras.optimizers import *
from tqdm import tqdm
import glob
from keras.applications.resnet50 import preprocess_input, decode_predictions

# from keras.applications.nasnet import preprocess_input, decode_predictions
# from keras.applications.densenet import preprocess_input, decode_predictions
from keras.preprocessing import image
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import sys

#import cv2
from random import choice, sample
import datetime
from sklearn.metrics import roc_auc_score
from keras.callbacks import ReduceLROnPlateau, ModelCheckpoint, EarlyStopping
import threading
```

```
from keras.initializers import glorot_normal

import tensorflow as tf
import numpy as np
import os

from numpy import genfromtxt
from keras.layers import Conv2D, ZeroPadding2D, Activation
from keras.layers.normalization import BatchNormalization
```

In [25]:

```
import tensorflow as tf
from tensorflow.python.keras import backend as K
sess = K.get_session()
#array = sess.run(your_tensor)
```

In [26]:

```
# https://github.com/rcmalli/keras-vggface/blob/master/keras_vggface/utils.py
def preprocess_input(x, data_format=None, version=1):
    '''
    preprocess_input will take the numpy array of image and remove the unnecessary values
    The out ut numpy only contains thermal image of face which will have the umporatant chracters
    Here we subtract the values from image numpy array after subtraction redundant pixels are removed

    '''
    x_temp = np.copy(x)
    if data_format is None:
        data_format = K.image_data_format()
    assert data_format in {'channels_last', 'channels_first'}

    if version == 1:
        if data_format == 'channels_first':
            x_temp = x_temp[:, ::-1, ...]
            x_temp[:, 0, :, :] -= 93.5940
            x_temp[:, 1, :, :] -= 104.7624
            x_temp[:, 2, :, :] -= 129.1863
        else:
            x_temp = x_temp[..., ::-1]
            x_temp[..., 0] -= 93.5940
            x_temp[..., 1] -= 104.7624
            x_temp[..., 2] -= 129.1863

    elif version == 2:
        if data_format == 'channels_first':
            x_temp = x_temp[:, ::-1, ...]
            x_temp[:, 0, :, :] -= 91.4953
            x_temp[:, 1, :, :] -= 103.8827
            x_temp[:, 2, :, :] -= 131.0912
        else:
            x_temp = x_temp[..., ::-1]
            x_temp[..., 0] -= 91.4953
            x_temp[..., 1] -= 103.8827
            x_temp[..., 2] -= 131.0912
    else:
        raise NotImplementedError
    return x_temp
```

In [27]:

```
# Function for reading images
def read_img(path, IMG_SIZE):
    img = image.load_img(path, target_size=(IMG_SIZE, IMG_SIZE))
    img = np.array(img).astype(np.float)
    return preprocess_input(img, version=2)
```

In [28]:

```
def auc(y_true, y_pred):
    return tf.py_function(roc_auc_score, (y_true, y_pred), tf.double)
```

In [29]:

```
def get_thumbnail(path):
    if path and os.path.exists(path):
        i = Image.open(path)
        i.thumbnail((150, 150), Image.LANCZOS)
        return i

def image_base64(im):
    if isinstance(im, str):
        im = get_thumbnail(im)
    with BytesIO() as buffer:
        im.save(buffer, 'jpeg')
        return base64.b64encode(buffer.getvalue()).decode()

def add_image_path(x):
    image_path = 'train/' + x
    if os.path.exists(image_path):
        path = os.path.join(image_path, os.listdir(image_path)[0])
        return path
```

In [30]:

```
# https://www.kaggle.com/janpreets/just-another-feature-extractor-0-824-lb
def outer_product(x):
    """
    calculate outer-products of 2 tensors

        args
            x
                list of 2 tensors
                , assuming each of which has shape = (size_minibatch, total_pixels, size_filter)
    """
    return keras.backend.batch_dot(
                x[0]
                , x[1]
                , axes=[1,1]
            ) / x[0].get_shape().as_list()[1]

def signed_sqrt(x):
    """
    calculate element-wise signed square root

        args
            x
                a tensor
    """
    return keras.backend.sign(x) * keras.backend.sqrt(keras.backend.abs(x) + 1e-9)

def L2_norm(x, axis=-1):
    """
    calculate L2-norm

        args
            x
                a tensor
    """
    return keras.backend.l2_normalize(x, axis=axis)
```

In [ ]:

In [32]:

```
def inception_based_model():
    myInput = Input(shape=(96, 96, 3))

    x = ZeroPadding2D(padding=(3, 3), input_shape=(96, 96, 3))(myInput)
    x = Conv2D(64, (7, 7), strides=(2, 2), name='conv1')(x)
    x = BatchNormalization(axis=3, epsilon=0.00001, name='bn1')(x)
    x = BatchNormalization(axis=3, epsilon=0.00001, name='bn1')(x)
    x = Activation('relu')(x)
    x = ZeroPadding2D(padding=(1, 1))(x)
    x = MaxPooling2D(pool_size=3, strides=2)(x)
    x = Lambda(LRN2D, name='lrn_1')(x)
    x = Conv2D(64, (1, 1), name='conv2')(x)
    x = BatchNormalization(axis=3, epsilon=0.00001, name='bn2')(x)
    x = Activation('relu')(x)
    x = ZeroPadding2D(padding=(1, 1))(x)
    x = Conv2D(192, (3, 3), name='conv3')(x)
    x = BatchNormalization(axis=3, epsilon=0.00001, name='bn3')(x)
    x = Activation('relu')(x)
    x = Lambda(LRN2D, name='lrn_2')(x)
    x = ZeroPadding2D(padding=(1, 1))(x)
    x = MaxPooling2D(pool_size=3, strides=2)(x)

    # Inception3a
    inception_3a_3x3 = Conv2D(96, (1, 1), name='inception_3a_3x3_conv1')(x)
    inception_3a_3x3 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_3x3_bn1')(inception_3a_3x3)
    inception_3a_3x3 = Activation('relu')(inception_3a_3x3)
    inception_3a_3x3 = ZeroPadding2D(padding=(1, 1))(inception_3a_3x3)
    inception_3a_3x3 = Conv2D(128, (3, 3), name='inception_3a_3x3_conv2')(inception_3a_3x3)
    inception_3a_3x3 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_3x3_bn2')(inception_3a_3x3)
    inception_3a_3x3 = Activation('relu')(inception_3a_3x3)

    inception_3a_5x5 = Conv2D(16, (1, 1), name='inception_3a_5x5_conv1')(x)
    inception_3a_5x5 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_5x5_bn1')(inception_3a_5x5)
    inception_3a_5x5 = Activation('relu')(inception_3a_5x5)
    inception_3a_5x5 = ZeroPadding2D(padding=(2, 2))(inception_3a_5x5)
    inception_3a_5x5 = Conv2D(32, (5, 5), name='inception_3a_5x5_conv2')(inception_3a_5x5)
    inception_3a_5x5 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_5x5_bn2')(inception_3a_5x5)
    inception_3a_5x5 = Activation('relu')(inception_3a_5x5)

    inception_3a_pool = MaxPooling2D(pool_size=3, strides=2)(x)
    inception_3a_pool = Conv2D(32, (1, 1), name='inception_3a_pool_conv')(inception_3a_pool)
    inception_3a_pool = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_pool_bn')(inception_3a_pool)
    inception_3a_pool = Activation('relu')(inception_3a_pool)
    inception_3a_pool = ZeroPadding2D(padding=((3, 4), (3, 4)))(inception_3a_pool)

    inception_3a_1x1 = Conv2D(64, (1, 1), name='inception_3a_1x1_conv')(x)
    inception_3a_1x1 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3a_1x1_bn')(inception_3a_1x1)
    inception_3a_1x1 = Activation('relu')(inception_3a_1x1)

    inception_3a = concatenate([inception_3a_3x3, inception_3a_5x5, inception_3a_pool, inception_3a_1x1], axis=3)

    # Inception3b
    inception_3b_3x3 = Conv2D(96, (1, 1), name='inception_3b_3x3_conv1')(inception_3a)
    inception_3b_3x3 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_3x3_bn1')(inception_3b_3x3)
    inception_3b_3x3 = Activation('relu')(inception_3b_3x3)
    inception_3b_3x3 = ZeroPadding2D(padding=(1, 1))(inception_3b_3x3)
    inception_3b_3x3 = Conv2D(128, (3, 3), name='inception_3b_3x3_conv2')(inception_3b_3x3)
    inception_3b_3x3 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_3x3_bn2')(inception_3b_3x3)
    inception_3b_3x3 = Activation('relu')(inception_3b_3x3)

    inception_3b_5x5 = Conv2D(32, (1, 1), name='inception_3b_5x5_conv1')(inception_3a)
    inception_3b_5x5 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_5x5_bn1')(inception_3b_5x5)
    inception_3b_5x5 = Activation('relu')(inception_3b_5x5)
    inception_3b_5x5 = ZeroPadding2D(padding=(2, 2))(inception_3b_5x5)
    inception_3b_5x5 = Conv2D(64, (5, 5), name='inception_3b_5x5_conv2')(inception_3b_5x5)
    inception_3b_5x5 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_5x5_bn2')(inception_3b_5x5)
    inception_3b_5x5 = Activation('relu')(inception_3b_5x5)

    inception_3b_pool = AveragePooling2D(pool_size=(3, 3), strides=(3, 3))(inception_3a)
    inception_3b_pool = Conv2D(64, (1, 1), name='inception_3b_pool_conv')(inception_3b_pool)
    inception_3b_pool = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_pool_bn')(inception_3b_pool)
    inception_3b_pool = Activation('relu')(inception_3b_pool)
    inception_3b_pool = Activation('relu')(inception_3b_pool)
    inception_3b_pool = ZeroPadding2D(padding=(4, 4))(inception_3b_pool)

    inception_3b_1x1 = Conv2D(64, (1, 1), name='inception_3b_1x1_conv')(inception_3a)
    inception_3b_1x1 = BatchNormalization(axis=3, epsilon=0.00001, name='inception_3b_1x1_bn')(inceptio
n_3b_1x1)
    inception_3b_1x1 = Activation('relu')(inception_3b_1x1)

    inception_3b = concatenate([inception_3b_3x3, inception_3b_5x5, inception_3b_pool, inception_3b_1x1
], axis=3)

    # Inception3c
    inception_3c_3x3 = conv2d_bn(inception_3b,
                                       layer='inception_3c_3x3',
                                       cv1_out=128,
                                       cv1_filter=(1, 1),
                                       cv2_out=256,
                                       cv2_filter=(3, 3),
                                       cv2_strides=(2, 2),
                                       padding=(1, 1))

    inception_3c_5x5 = conv2d_bn(inception_3b,
                                       layer='inception_3c_5x5',
                                       cv1_out=32,
                                       cv1_filter=(1, 1),
                                       cv2_out=64,
                                       cv2_filter=(5, 5),
                                       cv2_strides=(2, 2),
                                       padding=(2, 2))

    inception_3c_pool = MaxPooling2D(pool_size=3, strides=2)(inception_3b)
    inception_3c_pool = ZeroPadding2D(padding=((0, 1), (0, 1)))(inception_3c_pool)

    inception_3c = concatenate([inception_3c_3x3, inception_3c_5x5, inception_3c_pool], axis=3)

    #inception 4a
    inception_4a_3x3 = conv2d_bn(inception_3c,
                                       layer='inception_4a_3x3',
                                       cv1_out=96,
                                       cv1_filter=(1, 1),
                                       cv2_out=192,
                                       cv2_filter=(3, 3),
                                       cv2_strides=(1, 1),
                                       padding=(1, 1))
    inception_4a_5x5 = conv2d_bn(inception_3c,
                                       layer='inception_4a_5x5',
                                       cv1_out=32,
                                       cv1_filter=(1, 1),
                                       cv2_out=64,
                                       cv2_filter=(5, 5),
                                       cv2_strides=(1, 1),
                                       padding=(2, 2))

    inception_4a_pool = AveragePooling2D(pool_size=(3, 3), strides=(3, 3))(inception_3c)
    inception_4a_pool = conv2d_bn(inception_4a_pool,
                                        layer='inception_4a_pool',
                                        cv1_out=128,
                                        cv1_filter=(1, 1),
                                        padding=(2, 2))
    inception_4a_1x1 = conv2d_bn(inception_3c,
                                       layer='inception_4a_1x1',
                                       cv1_out=256,
                                       cv1_filter=(1, 1))
    inception_4a = concatenate([inception_4a_3x3, inception_4a_5x5, inception_4a_pool, inception_4a_1x1
], axis=3)

    #inception4e
    inception_4e_3x3 = conv2d_bn(inception_4a,
                                       layer='inception_4e_3x3',
                                       cv1_out=160,
                                       cv1_filter=(1, 1),
                                       cv2_out=256,
                                       cv2_filter=(3, 3),
                                       cv2_strides=(2, 2),
                                       padding=(1, 1))
    inception_4e_5x5 = conv2d_bn(inception_4a,
                                       layer='inception_4e_5x5',
                                       cv1_out=64
```

```
                                        cv1_out=64,
                                        cv1_filter=(1, 1),
                                        cv2_out=128,
                                        cv2_filter=(5, 5),
                                        cv2_strides=(2, 2),
                                        padding=(2, 2))
    inception_4e_pool = MaxPooling2D(pool_size=3, strides=2)(inception_4a)
    inception_4e_pool = ZeroPadding2D(padding=((0, 1), (0, 1)))(inception_4e_pool)

    inception_4e = concatenate([inception_4e_3x3, inception_4e_5x5, inception_4e_pool], axis=3)

    #inception5a
    inception_5a_3x3 = conv2d_bn(inception_4e,
                                        layer='inception_5a_3x3',
                                        cv1_out=96,
                                        cv1_filter=(1, 1),
                                        cv2_out=384,
                                        cv2_filter=(3, 3),
                                        cv2_strides=(1, 1),
                                        padding=(1, 1))

    inception_5a_pool = AveragePooling2D(pool_size=(3, 3), strides=(3, 3))(inception_4e)
    inception_5a_pool = conv2d_bn(inception_5a_pool,
                                         layer='inception_5a_pool',
                                         cv1_out=96,
                                         cv1_filter=(1, 1),
                                         padding=(1, 1))
    inception_5a_1x1 = conv2d_bn(inception_4e,
                                        layer='inception_5a_1x1',
                                        cv1_out=256,
                                        cv1_filter=(1, 1))

    inception_5a = concatenate([inception_5a_3x3, inception_5a_pool, inception_5a_1x1], axis=3)

    #inception_5b
    inception_5b_3x3 = conv2d_bn(inception_5a,
                                        layer='inception_5b_3x3',
                                        cv1_out=96,
                                        cv1_filter=(1, 1),
                                        cv2_out=384,
                                        cv2_filter=(3, 3),
                                        cv2_strides=(1, 1),
                                        padding=(1, 1))
    inception_5b_pool = MaxPooling2D(pool_size=3, strides=2)(inception_5a)
    inception_5b_pool = conv2d_bn(inception_5b_pool,
                                         layer='inception_5b_pool',
                                         cv1_out=96,
                                         cv1_filter=(1, 1))
    inception_5b_pool = ZeroPadding2D(padding=(1, 1))(inception_5b_pool)

    inception_5b_1x1 = conv2d_bn(inception_5a,
                                        layer='inception_5b_1x1',
                                        cv1_out=256,
                                        cv1_filter=(1, 1))
    inception_5b = concatenate([inception_5b_3x3, inception_5b_pool, inception_5b_1x1], axis=3)

    av_pool = AveragePooling2D(pool_size=(3, 3), strides=(1, 1))(inception_5b)
    reshape_layer = Flatten()(av_pool)
    dense_layer = Dense(128, name='dense_layer')(reshape_layer)
    norm_layer = Lambda(lambda  x: K.l2_normalize(x, axis=1), name='norm_layer')(dense_layer)

    return Model(inputs=[myInput], outputs=norm_layer)
```

In [33]:

```
def LRN2D(x):
    return tf.nn.lrn(x, alpha=1e-4, beta=0.75)

def conv2d_bn(
  x,
  layer=None,
  cv1_out=None,
  cv1_filter=(1, 1),
  cv1_strides=(1, 1),
  cv2_out=None,
  cv2_filter=(3, 3),
  cv2_strides=(1, 1),
  padding=None,
):
    num = '' if cv2_out == None else '1'
    tensor = Conv2D(cv1_out, cv1_filter, strides=cv1_strides, name=layer+'_conv'+num)(x)
    tensor = BatchNormalization(axis=3, epsilon=0.00001, name=layer+'_bn'+num)(tensor)
    tensor = Activation('relu')(tensor)
    if padding == None:
        return tensor
    tensor = ZeroPadding2D(padding=padding)(tensor)
    if cv2_out == None:
        return tensor
    tensor = Conv2D(cv2_out, cv2_filter, strides=cv2_strides, name=layer+'_conv'+'2')(tensor)
    tensor = BatchNormalization(axis=3, epsilon=0.00001, name=layer+'_bn'+'2')(tensor)
    tensor = Activation('relu')(tensor)
    return tensor

def create_base_network(input_shape):
    mod = inception_based_model()
    new_mod = Model(mod.input,mod.layers[-5].output)
    return new_mod

def get_model6():
    input_shape = (96,96,3)
    input_a = Input(shape=input_shape)
    input_b = Input(shape=input_shape)
    base_network = create_base_network(input_shape)

    # because we re-use the same instance `base_network`,
    # the weights of the network
    # will be shared across the two branches
    processed_a = base_network(input_a)
    processed_b = base_network(input_b)


    # extract features from detector
    x_detector = processed_a
    shape_detector = processed_a.shape

    # extract features from extractor , same with detector for symmetry DxD model
    shape_extractor = processed_b.shape
    x_extractor = processed_b

    # rehape to (minibatch_size, total_pixels, filter_size)
    x_detector = keras.layers.Reshape([shape_detector[1] * shape_detector[2] , shape_detector[-1]])(x_d
etector)
    x_extractor = keras.layers.Reshape([shape_extractor[1] * shape_extractor[2] , shape_extractor[-1]])
(x_extractor)

    # outer products of features, output shape=(minibatch_size, filter_size_detector*filter_size_extrac
tor)
    x = keras.layers.Lambda(outer_product)([x_detector, x_extractor])

    # rehape to (minibatch_size, filter_size_detector*filter_size_extractor)
    x = keras.layers.Reshape([shape_detector[-1]*shape_extractor[-1]])(x)
    # signed square-root

    x = keras.layers.Lambda(signed_sqrt)(x)
    # L2 normalization
    x = keras.layers.Lambda(L2_norm)(x)

    ###
    ### attach FC-Layer
    ###

    x = Dense(100, activation="relu")(x)
    x = Dropout(0.01)(x)

    out = Dense(units=1,kernel_regularizer=keras.regularizers.l2(1e-8),kernel_initializer='glorot_norma
l',activation="sigmoid")(x)



    model = Model([input_a, input_b], out)

    model.compile(loss="binary_crossentropy", metrics=['acc',auc], optimizer=Adam(0.001))
    print(model.summary())

    return model
```

In [ ]:

```

```

In [34]:

```
model6 = get_model6()
```

```
Model: "model_8"
__________________________________________________________________________________________________
Layer (type)                    Output Shape         Param #     Connected to
==================================================================================================
input_16 (InputLayer)           (None, 96, 96, 3)    0
__________________________________________________________________________________________________
input_17 (InputLayer)           (None, 96, 96, 3)    0
__________________________________________________________________________________________________
model_7 (Model)                 (None, 3, 3, 736)    3648944     input_16[0][0]
                                                                 input_17[0][0]
__________________________________________________________________________________________________
reshape_2 (Reshape)             (None, 9, 736)       0           model_7[1][0]
__________________________________________________________________________________________________
reshape_3 (Reshape)             (None, 9, 736)       0           model_7[2][0]
__________________________________________________________________________________________________
lambda_1 (Lambda)               (None, 736, 736)     0           reshape_2[0][0]
                                                                 reshape_3[0][0]
__________________________________________________________________________________________________
reshape_4 (Reshape)             (None, 541696)       0           lambda_1[0][0]
__________________________________________________________________________________________________
lambda_2 (Lambda)               (None, 541696)       0           reshape_4[0][0]
__________________________________________________________________________________________________
lambda_3 (Lambda)               (None, 541696)       0           lambda_2[0][0]
__________________________________________________________________________________________________
dense_19 (Dense)                (None, 100)          54169700    lambda_3[0][0]
__________________________________________________________________________________________________
dropout_17 (Dropout)            (None, 100)          0           dense_19[0][0]
__________________________________________________________________________________________________
dense_20 (Dense)                (None, 1)            101         dropout_17[0][0]
==================================================================================================
Total params: 57,818,745
Trainable params: 57,809,433
Non-trainable params: 9,312
__________________________________________________________________________________________________
None
```

In [ ]:

```

```

FINAL MODEL ENSEMBLING

In [35]:

```
import numpy as np
import pandas as pd
```

In [36]:

```
# Function for reading images
def read_img(path, IMG_SIZE):
    img = image.load_img(path, target_size=(IMG_SIZE, IMG_SIZE))
    img = np.array(img).astype(np.float)
    return preprocess_input(img, version=2)
```

In [ ]:

```

```

In [37]:

```python
from tqdm import tqdm
test_path = "../test/"

def final_model(test_path):
    def chunker(seq, size=32):
        return (seq[pos:pos + size] for pos in range(0, len(seq), size))

    submission1 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission2 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission3 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission4 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission5 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission6 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')

    predictions1 = []
    predictions2 = []
    predictions3 = []
    predictions4 = []
    predictions5 = []
    predictions6 = []

    print('Model1 predictions ....')
    IMG_SIZE = 197
    for batch in tqdm(chunker(submission1.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model1.load_weights('../SAVE_MODEL1/model1/model1.h5')
        pred1 = model1.predict([X1, X2]).ravel().tolist()
        predictions1 += pred1

    print('Model2 predictions ....')
    IMG_SIZE = 197
    for batch in tqdm(chunker(submission2.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        pred2 = model2.predict([X1, X2]).ravel().tolist()
        model2.load_weights('../SAVE_MODEL1/model2/model2.h5')
        predictions2 += pred2

    print('Model3 predictions ....')
    IMG_SIZE = 224
    for batch in tqdm(chunker(submission3.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model3.load_weights('../SAVE_MODEL1/model3/model3.h5')
        pred3 = model3.predict([X1, X2]).ravel().tolist()
        predictions3 += pred3

    print('Model4 predictions ....')
    IMG_SIZE = 224
    for batch in tqdm(chunker(submission4.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model4.load_weights('../SAVE_MODEL1/model4/model4.h5')
        pred4 = model4.predict([X1, X2]).ravel().tolist()
        predictions4 += pred4

    print('Model5 predictions ....')
    IMG_SIZE = 224
    for batch in tqdm(chunker(submission5.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model5.load_weights('../SAVE_MODEL1/model5/model5.h5')
        pred5 = model5.predict([X1, X2]).ravel().tolist()
        predictions5 += pred5

    print('Model6 predictions ....')
    IMG_SIZE = 96
    for batch in tqdm(chunker(submission6.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model6.load_weights('../SAVE_MODEL1/model6/model6.h5')
        pred6 = model6.predict([X1, X2]).ravel().tolist()
        predictions6 += pred6

    submission1['is_related'] = predictions1
    submission2['is_related'] = predictions2
    submission3['is_related'] = predictions3
    submission4['is_related'] = predictions4
    submission5['is_related'] = predictions5
    submission6['is_related'] = predictions6

    submission1.to_csv("model1.csv", index=False)
    submission2.to_csv("model2.csv", index=False)
    submission3.to_csv("model3.csv", index=False)
    submission4.to_csv("model4.csv", index=False)
    submission5.to_csv("model5.csv", index=False)
    submission6.to_csv("model6.csv", index=False)

    sub1 = pd.read_csv('./model1.csv')
    sub2 = pd.read_csv('./model2.csv')
    sub3 = pd.read_csv('./model3.csv')
    sub4 = pd.read_csv('./model4.csv')
    sub5 = pd.read_csv('./model5.csv')
    sub6 = pd.read_csv('./model6.csv')

    #final=pd.read_csv('./model2.csv')

    #print('Combining Models predictions ....')
    #final['is_related'] = 0.18*sub2['is_related'] + 0.16*sub3['is_related'] + 0.22*sub4['is_related']
+ 0.22*sub5['is_related'] + 0.22*sub6['is_related']
    #final.to_csv('submission_final1.csv', index=False )

    return sub1, sub2, sub3, sub4, sub5, sub6
```

In [38]:

```
sub_1, sub_2, sub_3, sub_4, sub_5, sub_6 = final_model(test_path)
```

```
0it [00:00, ?it/s]
```

```
Model1 predictions ....
```

```
166it [24:52,  8.99s/it]
0it [00:00, ?it/s]
```

```
Model2 predictions ....
```

```
166it [10:46,  3.89s/it]
0it [00:00, ?it/s]
```

```
Model3 predictions ....
```

```
166it [11:56,  4.31s/it]
0it [00:00, ?it/s]
```

```
Model4 predictions ....
```

```
166it [11:54,  4.31s/it]
0it [00:00, ?it/s]
```

```
Model5 predictions ....
```

```
166it [13:41,  4.95s/it]
0it [00:00, ?it/s]
```

```
Model6 predictions ....
```

```
166it [03:18,  1.19s/it]
```

In [ ]:

```
final=pd.read_csv('./model2.csv')
print('Combining Models predictions ....')
final['is_related'] = 0.18*sub_2['is_related'] + 0.16*sub_3['is_related'] + 0.22*sub_4['is_related'] +
0.22*sub_5['is_related'] + 0.22*sub_6['is_related']
final.to_csv('submission_final1.csv', index=False )
```

In [39]:

```
final=pd.read_csv('./model2.csv')
print('Combining Models predictions ....')
final['is_related'] = 0.17*sub_1['is_related'] + 0.17*sub_2['is_related'] + 0.16*sub_3['is_related'] +
0.16*sub_4['is_related'] + 0.17*sub_5['is_related'] + 0.17*sub_6['is_related']
final.to_csv('submission_final2.csv', index=False )
```

```
Combining Models predictions ....
```

In [40]:

```
from tqdm import tqdm
test_path = "../test/"

def final_model1(test_path):
    def chunker(seq, size=32):
        return (seq[pos:pos + size] for pos in range(0, len(seq), size))

    submission1 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission2 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission3 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission4 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission5 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission6 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')

    predictions1 = []
    predictions2 = []
    predictions3 = []
    predictions4 = []
    predictions5 = []
    predictions6 = []

    print('Model1 predictions ....')
    IMG_SIZE = 197
    for batch in tqdm(chunker(submission1.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model1.load_weights('../SAVE_MODEL1/model1/model1_0.h5')
        pred1 = model1.predict([X1, X2]).ravel().tolist()
        predictions1 += pred1

    print('Model2 predictions ....')
    IMG_SIZE = 197
    for batch in tqdm(chunker(submission2.img_pair.values)):
```

```
for batch in tqdm(chunker(submission2.img_pair.values)):
    X1 = [x.split("-")[0] for x in batch]
    X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
    X2 = [x.split("-")[1] for x in batch]
    X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
    pred2 = model2.predict([X1, X2]).ravel().tolist()
    model2.load_weights('../SAVE_MODEL1/model2/model2_0.h5')
    predictions2 += pred2

print('Model3 predictions ....')
IMG_SIZE = 224
for batch in tqdm(chunker(submission3.img_pair.values)):
    X1 = [x.split("-")[0] for x in batch]
    X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
    X2 = [x.split("-")[1] for x in batch]
    X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
    model3.load_weights('../SAVE_MODEL1/model3/model3.h5')
    pred3 = model3.predict([X1, X2]).ravel().tolist()
    predictions3 += pred3

print('Model4 predictions ....')
IMG_SIZE = 224
for batch in tqdm(chunker(submission4.img_pair.values)):
    X1 = [x.split("-")[0] for x in batch]
    X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
    X2 = [x.split("-")[1] for x in batch]
    X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
    model4.load_weights('../SAVE_MODEL1/model4/model4.h5')
    pred4 = model4.predict([X1, X2]).ravel().tolist()
    predictions4 += pred4

print('Model5 predictions ....')
IMG_SIZE = 224
for batch in tqdm(chunker(submission5.img_pair.values)):
    X1 = [x.split("-")[0] for x in batch]
    X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
    X2 = [x.split("-")[1] for x in batch]
    X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
    model5.load_weights('../SAVE_MODEL1/model5/model5.h5')
    pred5 = model5.predict([X1, X2]).ravel().tolist()
    predictions5 += pred5

print('Model6 predictions ....')
IMG_SIZE = 96
for batch in tqdm(chunker(submission6.img_pair.values)):
    X1 = [x.split("-")[0] for x in batch]
    X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
    X2 = [x.split("-")[1] for x in batch]
    X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
    model6.load_weights('../SAVE_MODEL1/model6/model6.h5')
    pred6 = model6.predict([X1, X2]).ravel().tolist()
    predictions6 += pred6

submission1['is_related'] = predictions1
submission2['is_related'] = predictions2
submission3['is_related'] = predictions3
submission4['is_related'] = predictions4
submission5['is_related'] = predictions5
submission6['is_related'] = predictions6

submission1.to_csv("model1.csv", index=False)
submission2.to_csv("model2.csv", index=False)
submission3.to_csv("model3.csv", index=False)
submission4.to_csv("model4.csv", index=False)
submission5.to_csv("model5.csv", index=False)
submission6.to_csv("model6.csv", index=False)

sub1 = pd.read_csv('./model1.csv')
sub2 = pd.read_csv('./model2.csv')
sub3 = pd.read_csv('./model3.csv')
sub4 = pd.read_csv('./model4.csv')
sub5 = pd.read_csv('./model5.csv')
sub6 = pd.read_csv('./model6.csv')

#final=pd.read_csv('./model2.csv')
```

```
    #print('Combining Models predictions ....')
    #final['is_related'] = 0.18*sub2['is_related'] + 0.16*sub3['is_related'] + 0.22*sub4['is_related']
+ 0.22*sub5['is_related'] + 0.22*sub6['is_related']
    #final.to_csv('submission_final1.csv', index=False )

    return sub1, sub2, sub3, sub4, sub5, sub6
```

In [ ]:

```

```

In [41]:

```
sub1_1, sub1_2, sub1_3, sub1_4, sub1_5, sub1_6 = final_model1(test_path)
```

```
0it [00:00, ?it/s]
```

```
Model1 predictions ....
```

```
166it [10:06,  3.66s/it]
0it [00:00, ?it/s]
```

```
Model2 predictions ....
```

```
166it [10:14,  3.70s/it]
0it [00:00, ?it/s]
```

```
Model3 predictions ....
```

```
166it [11:59,  4.33s/it]
0it [00:00, ?it/s]
```

```
Model4 predictions ....
```

```
166it [11:52,  4.29s/it]
0it [00:00, ?it/s]
```

```
Model5 predictions ....
```

```
166it [13:50,  5.00s/it]
0it [00:00, ?it/s]
```

```
Model6 predictions ....
```

```
166it [03:20,  1.21s/it]
```

In [42]:

```
final1=pd.read_csv('./model2.csv')
print('Combining Models predictions ....')
final1['is_related'] = 0.18*sub1_2['is_related'] + 0.16*sub1_3['is_related'] + 0.22*sub1_4['is_related'
] + 0.22*sub1_5['is_related'] + 0.22*sub1_6['is_related']
final1.to_csv('submission_final3.csv', index=False )
```

```
Combining Models predictions ....
```

In [43]:

```
final1=pd.read_csv('./model2.csv')
print('Combining Models predictions ....')
final1['is_related'] = 0.17*sub1_1['is_related'] + 0.17*sub1_2['is_related'] + 0.16*sub1_3['is_related'
] + 0.16*sub1_4['is_related'] + 0.17*sub1_5['is_related'] + 0.17*sub1_6['is_related']
final1.to_csv('submission_final4.csv', index=False )
```

```
final1.to_csv('submission_final4.csv', index=False)
```

Combining Models predictions ....

In [ ]:

```

```

In [44]:

```
from tqdm import tqdm
test_path = "../test/"

def final_model2(test_path):
    def chunker(seq, size=32):
        return (seq[pos:pos + size] for pos in range(0, len(seq), size))

    submission1 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission2 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission3 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission4 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission5 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')
    submission6 = pd.read_csv('../recognizing-faces-in-the-wild/sample_submission.csv')

    predictions1 = []
    predictions2 = []
    predictions3 = []
    predictions4 = []
    predictions5 = []
    predictions6 = []

    print('Model1 predictions ....')
    IMG_SIZE = 197
    for batch in tqdm(chunker(submission1.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model1.load_weights('../SAVE_MODEL1/model1/model1_0.h5')
        pred1 = model1.predict([X1, X2]).ravel().tolist()
        predictions1 += pred1

    print('Model2 predictions ....')
    IMG_SIZE = 197
    for batch in tqdm(chunker(submission2.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        pred2 = model2.predict([X1, X2]).ravel().tolist()
        model2.load_weights('../SAVE_MODEL1/model2/model2_0.h5')
        predictions2 += pred2

    print('Model3 predictions ....')
    IMG_SIZE = 224
    for batch in tqdm(chunker(submission3.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model3.load_weights('../SAVE_MODEL1/model3/model3.h5')
        pred3 = model3.predict([X1, X2]).ravel().tolist()
        predictions3 += pred3

    print('Model4 predictions ....')
    IMG_SIZE = 224
    for batch in tqdm(chunker(submission4.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model4.load_weights('../SAVE_MODEL1/model4/model4.h5')
        pred4 = model4.predict([X1, X2]).ravel().tolist()
        predictions4 += pred4

    print('Model5 predictions ....')
    IMG_SIZE = 224
    for batch in tqdm(chunker(submission5.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model5.load_weights('../SAVE_MODEL1/model5/model5.h5')
        pred5 = model5.predict([X1, X2]).ravel().tolist()
        predictions5 += pred5

    print('Model6 predictions ....')
    IMG_SIZE = 96
    for batch in tqdm(chunker(submission6.img_pair.values)):
        X1 = [x.split("-")[0] for x in batch]
        X1 = np.array([read_img(test_path + x, IMG_SIZE) for x in X1])
        X2 = [x.split("-")[1] for x in batch]
        X2 = np.array([read_img(test_path + x, IMG_SIZE) for x in X2])
        model6.load_weights('../SAVE_MODEL1/model6/model6_0.h5')
        pred6 = model6.predict([X1, X2]).ravel().tolist()
        predictions6 += pred6

    submission1['is_related'] = predictions1
    submission2['is_related'] = predictions2
    submission3['is_related'] = predictions3
    submission4['is_related'] = predictions4
    submission5['is_related'] = predictions5
    submission6['is_related'] = predictions6

    submission1.to_csv("model1.csv", index=False)
    submission2.to_csv("model2.csv", index=False)
    submission3.to_csv("model3.csv", index=False)
    submission4.to_csv("model4.csv", index=False)
    submission5.to_csv("model5.csv", index=False)
    submission6.to_csv("model6.csv", index=False)

    sub1 = pd.read_csv('./model1.csv')
    sub2 = pd.read_csv('./model2.csv')
    sub3 = pd.read_csv('./model3.csv')
    sub4 = pd.read_csv('./model4.csv')
    sub5 = pd.read_csv('./model5.csv')
    sub6 = pd.read_csv('./model6.csv')

    #final=pd.read_csv('./model2.csv')

    #print('Combining Models predictions ....')
    #final['is_related'] = 0.18*sub2['is_related'] + 0.16*sub3['is_related'] + 0.22*sub4['is_related']
+ 0.22*sub5['is_related'] + 0.22*sub6['is_related']
    #final.to_csv('submission_final1.csv', index=False )

    return sub1, sub2, sub3, sub4, sub5, sub6
```

In [ ]:

In [45]:

```
sub2_1, sub2_2, sub2_3, sub2_4, sub2_5, sub2_6 = final_model2(test_path)
```

```
0it [00:00, ?it/s]
```

```
Model1 predictions ....
```

```
166it [10:15,  3.71s/it]
0it [00:00, ?it/s]
```

```
Model2 predictions ....
```

```
166it [10:10,  3.68s/it]
0it [00:00, ?it/s]
```

```
Model3 predictions ....
```

```
166it [11:46,  4.25s/it]
0it [00:00, ?it/s]
```

```
Model4 predictions ....
```

```
166it [11:50,  4.28s/it]
0it [00:00, ?it/s]
```

```
Model5 predictions ....
```

```
166it [13:44,  4.97s/it]
0it [00:00, ?it/s]
```

```
Model6 predictions ....
```

```
166it [03:18,  1.20s/it]
```

In [46]:

```
final2=pd.read_csv('./model2.csv')
print('Combining Models predictions ....')
final2['is_related'] = 0.18*sub2_2['is_related'] + 0.16*sub2_3['is_related'] + 0.22*sub2_4['is_related'
] + 0.22*sub2_5['is_related'] + 0.22*sub2_6['is_related']
final2.to_csv('submission_final5.csv', index=False )
```

```
Combining Models predictions ....
```

In [50]:

```
final2=pd.read_csv('./model2.csv')
print('Combining Models predictions ....')
final2['is_related'] = 0.17*sub2_1['is_related'] + 0.17*sub2_2['is_related'] + 0.16*sub2_3['is_related'
] + 0.16*sub2_4['is_related'] + 0.17*sub2_5['is_related'] + 0.17*sub2_6['is_related']
final2.to_csv('submission_final6.csv', index=False )
```

```
Combining Models predictions ....
```

In [49]:

```
final2=pd.read_csv('./model2.csv')
print('Combining Models predictions ....')
final2['is_related'] = 0.15*sub2_1['is_related'] + 0.15*sub2_2['is_related'] + 0.15*sub2_3['is_related'
] + 0.15*sub2_4['is_related'] + 0.15*sub2_5['is_related'] + 0.25*sub2_6['is_related']
final2.to_csv('submission_final7.csv', index=False )
```

```
Combining Models predictions ....
```

References: -->

```
https://www.kaggle.com/shivamsarawagi/wildimagedetection-0-875
https://www.kaggle.com/hsinwenchang/vggface-baseline-197x197
https://www.kaggle.com/arjunrao2000/kinship-detection-with-vgg16
https://www.kaggle.com/leonbora/kinship-recognition-transfer-learning-vggface
https://www.kaggle.com/gowrishankarin/eda-with-plotly-smart-cute-and-pretty-people
```

In [ ]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]:

In [0]: