Test

April 17, 2020

[2]:
```
import keras
import numpy as np
import pandas as pd
import os
import cv2
import math
import glob
import matplotlib.pyplot as plt
from scipy import stats as s
from keras.models import Sequential
from keras.applications.vgg16 import VGG16
from keras.layers import Dense, InputLayer, Dropout, Flatten
from keras.layers import Conv2D, MaxPooling2D, GlobalMaxPooling2D
from keras.preprocessing import image
from sklearn.model_selection import train_test_split
from tqdm import tqdm
```

[23]:
```
base_model = VGG16(weights = 'imagenet', include_top = False)
model = Sequential()
model.add(Dense(1024, activation='relu', input_shape=(25088,)))
model.add(Dropout(0.5))
model.add(Dense(512, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(256, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(11, activation='softmax'))
model.load_weights("weight.hdf5")
```

[24]:
```
test = pd.read_csv('testing.csv')
```

[25]:
```
test.head()
```

[25]:

	image	label
0	Dataset/0_656_org_voi.wav.png	org
1	Dataset/2_275_pia_tru.wav.png	pia

```
2  Dataset/2_709_gac_gel.wav.png   gac
3  Dataset/0_506_gac_vio.wav.png   gac
4  Dataset/0_352_gel_sax.wav.png   gel
```

[26]:
```
test_imgs = test['image']
```

[27]:
```
predict = []
#Get the ground truth values of the testing data
actual = test['label']
for i in range(len(test_imgs)):
    img = image.load_img(test_imgs[i], target_size=(224,224))
    img = image.img_to_array(img)
    img = img/255
    #Extract features from the base model
    pre = []
    pre.append(img)
    pre = np.array(pre)
    img_features = base_model.predict(pre)
    img_features = img_features.reshape(pre.shape[0], 7*7*512)
    #Input the extracted features to the model we built
    prediction = model.predict_classes(img_features)
    predict.append(y.columns.values[s.mode(prediction)[0][0]])
```

[28]:
```
from sklearn.metrics import accuracy_score
score = accuracy_score(predict, actual)*100
print("The accuracy of the model is %"+str(score))
```

```
The accuracy of the model is %54.6916890080429
```