

selfie_retraining

Draft saved

File Edit View Run Add-ons Help

Share Save Version 0

Run All

Draft Session (2m)

HDD CPU RAM GPU

⏻ ↺ ⋮

I1: # Using a trained model for the next set of images

I1:

```
import keras
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten
from keras.preprocessing import image
from tensorflow.keras.applications.resnet50 import ResNet50
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from tqdm import tqdm
from keras.layers import Input, Lambda, Dense, Flatten
from keras.models import Model
import matplotlib.pyplot as plt
```

+ Code + Markdown

I1: df=pd.read_csv("../input/selfie-csv/selfie_dataset.csv")

I1: df.head()

Console

Type here to search

00:11 21-12-2020

Data

+ Add data

input (2.12 GB)

resnet10 resnet11 resnet55 resnet6 resnet7 resnetnew resnet91 saved-resnet50selfie selfie-csv selfie-resenet21h5 selfieresenet3 selfieresenet4 selfie2

output

/kaggle/working

Settings

Code Help

Find Code Help

Search for examples of how to do things

selfie_retraining

Draft saved

File Edit View Run Add-ons Help

Run All

Draft Session (2m)

HDD

CPU

RAM

GPU

Share

Save Version

0

```
[I]: df=df.drop(columns=['Unnamed: 0', 'score', 'partial_faces'])
df.head()

[I]: df=df.iloc[44000:46836]
df.shape

[I]: image_directory = '../input/selfie2/Selfie-dataset/images/'

[I]: SIZE = 224
X_dataset = []
for i in tqdm(range(44000,46836)):
    img = image.load_img(image_directory + df['image'][i] + '.jpg', target_size=(SIZE,SIZE,3))
    img = image.img_to_array(img)
    img = img/255.
    X_dataset.append(img)

X = np.array(X_dataset)
```

Data

+ Add data

input (2.12 GB)

resnet10

resnet11

resnet55

resnet6

resnet7

resnetnew

resnet91

saved-resnet50selfie

selfie-csv

selfie-resenet21h5

selfieresenet3

selfieresenet4

selfie2

output

/kaggle/working

Settings

Code Help

Find Code Help

Search for examples of how to do things

Console

Type here to search

00:11

21-12-2020

selfie_r x selfie_r x Selfie_r x (16) Cr x t81_55 x New Tab x Backgr x autoer x GAN L x New Tab x OMS- x Capabi x Shared x

kaggle.com/jaswanthreddysareddy/selfie-retraining/edit

selfie_retraining Draft saved

File Edit View Run Add-ons Help

+ Run All

Draft Session (3m)

HDD CPU RAM GPU

Share Save Version 0

11: y = np.array(df.drop(['image'], axis=1))
y.shape

11: from sklearn.model_selection import train_test_split

X_train, X_test, y_train, y_test = train_test_split(X, y, random_state=20, test_size=0.4)

+ Code + Markdown

Using the trained model

model=keras.models.load_model('../input/resnet11/selfie_resnet50_11.h5')
history = model.fit(X_train, y_train, epochs=10, validation_data=(X_test, y_test), batch_size=64)

11: #Validation on an image

img = image.load_img('../input/selfie2/Selfie-dataset/images/00a454da495e11e28a7322000a1fa414_6.jpg', target_size=(SIZE,SIZE,3))

img = image.img_to_array(img)
img = img/255.
plt.imshow(img)
img = np.expand_dims(img, axis=0)

classes = np.array(df.columns[1:]) #Get array of all classes
proba = model.predict(img) #Get probabilities for each class

Console

Data + Add data

input (2.12 GB)
resnet10
resnet11
resnet55
resnet6
resnet7
resnetnew
resnet91
saved-resnet50selfie
selfie-csv
selfie-resenet21h5
selfieresenet3
selfieresenet4
selfie2

output
/kaggle/working

Settings

Code Help

Find Code Help

Search for examples of how to do things

Type here to search

0 1 2 3 4

00:11
21-12-2020

selfie_r x selfie_r x Selfie_r x (16) Cr x t81_55 x New Tab x Backgr x autoer x GAN L x New Tab x OMS- x Capabi x Shared x

← → ↻ kaggle.com/jaswanthreddysareddy/selfie-retraining/edit

selfie_retraining Draft saved

File Edit View Run Add-ons Help

+ ▶ ▶▶ Run All

Draft Session (3m) H D D C P U R A M G P U

11: # Using the trained model

model=keras.models.load_model('../input/resnet11/selfie_resnet50_11.h5')
history = model.fit(X_train, y_train, epochs=10, validation_data=(X_test, y_test), batch_size=64)

▶ #Validation on an image

img = image.load_img('../input/selfie2/Selfie-dataset/images/00a454da495e11e28a7322000a1fa414_6.jpg', target_size=(SIZE,SIZE,3))

img = image.img_to_array(img)
img = img/255.
plt.imshow(img)
img = np.expand_dims(img, axis=0)|

classes = np.array(df.columns[1:]) #Get array of all classes
proba = model.predict(img) #Get probabilities for each class
sorted_categories = np.argsort(proba[0])[:-8:-1] #Get class names for top 8 categories

#Print classes and corresponding probabilities
for i in range(7):
 #print("{}".format(classes[sorted_categories[i]])+ " ({:.3})".format(proba[0][sorted_categories[i]]))
 print(f'The probability of this image to be in {classes[i]} category is: {round((proba[0][i]*100),2)}%')

+ Code + Markdown

Console

1

Type here to search

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99

00:11 21-12-2020

Share Save Version 0

Run All

Draft Session (3m)

Data + Add data

input (2.12 GB)

resnet10 resnet11 resnet55 resnet6 resnet7 resnetnew resnet91 saved-resnet50selfie selfie-csv selfie-resenet21h5 selfieresenet3 selfieresenet4 selfie2

output

/kaggle/working

Settings

Code Help

Find Code Help

Search for examples of how to do things