pneumonia-detection-using-deep-learning

April 19, 2020

[0]:

```
!pip install kaggle
from google.colab import files

uploaded = files.upload()

for fn in uploaded.keys():
  print('User uploaded file "{name}" with length {length} bytes'.format(
      name=fn, length=len(uploaded[fn])))

# Then move kaggle.json into the folder where the API expects to find it.
!mkdir -p ~/.kaggle/ && mv kaggle.json ~/.kaggle/ && chmod 600 ~/.kaggle/kaggle.
 ↪json
```

```
Requirement already satisfied: kaggle in /usr/local/lib/python3.6/dist-packages
(1.5.6)
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) (2.8.1)
Requirement already satisfied: python-slugify in /usr/local/lib/python3.6/dist-
packages (from kaggle) (4.0.0)
Requirement already satisfied: tqdm in /usr/local/lib/python3.6/dist-packages
(from kaggle) (4.38.0)
Requirement already satisfied: urllib3<1.25,>=1.21.1 in /usr/local/lib/python3.6
/dist-packages (from kaggle) (1.24.3)
Requirement already satisfied: certifi in /usr/local/lib/python3.6/dist-packages
(from kaggle) (2020.4.5.1)
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.6/dist-
packages (from kaggle) (1.12.0)
Requirement already satisfied: idna<2.9,>=2.5 in /usr/local/lib/python3.6/dist-
packages (from requests->kaggle) (2.8)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /usr/local/lib/python3.6
/dist-packages (from requests->kaggle) (3.0.4)
Requirement already satisfied: text-unidecode>=1.3 in /usr/local/lib/python3.6
/dist-packages (from python-slugify->kaggle) (1.3)
```

```
<IPython.core.display.HTML object>
```

```
Saving kaggle.json to kaggle.json
User uploaded file "kaggle.json" with length 65 bytes
```

[0]:

```
!kaggle datasets download paultimothymooney/chest-xray-pneumonia
!unzip chest-xray-pneumonia.zip
```

[0]:

[0]:

[0]:

```
!ls
```

```
chest_xray  chest-xray-pneumonia.zip  sample_data
```

This notebook tackles pneumonia classification using CNN (Convolutional Neural Network). In addition, this will also experiment with threshold values.

Accuracy on testing set: 0.9598976109215017 Precision on testing set: 0.9118541033434651 Recall on testing set: 0.9433962264150944

[0]:

[0]:

```
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow import keras
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Activation, Conv2D, MaxPooling2D, Flatten, Dropout, BatchNormalization
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import EarlyStopping
from sklearn.metrics import precision_recall_curve, roc_curve, accuracy_score, confusion_matrix, precision_score, recall_score
from sklearn.decomposition import PCA
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
import seaborn as sns
plt.style.use('fivethirtyeight')
import pickle
import os
import numpy as np
import cv2
%matplotlib inline
```

```
/usr/local/lib/python3.6/dist-packages/statsmodels/tools/_testing.py:19:
FutureWarning: pandas.util.testing is deprecated. Use the functions in the
public API at pandas.testing instead.
  import pandas.util.testing as tm
```

Process the images and resize them to the preferred size

[0]:

```
labels = ['PNEUMONIA', 'NORMAL']
img_size = 200
def get_training_data(data_dir):
    data = []
    for label in labels:
        path = os.path.join(data_dir, label)
        class_num = labels.index(label)
        for img in os.listdir(path):
            try:
                img_arr = cv2.imread(os.path.join(path, img), cv2.
 ↪IMREAD_GRAYSCALE)
                resized_arr = cv2.resize(img_arr, (img_size, img_size))
                data.append([resized_arr, class_num])
            except Exception as e:
                print(e)
    return np.array(data)
```

Preparing the training and testing data

[0]:

```
train = get_training_data('chest_xray/chest_xray/train')
test = get_training_data('chest_xray/chest_xray/test')
val = get_training_data('chest_xray/chest_xray/val')
```

```
OpenCV(4.1.2) /io/opencv/modules/imgproc/src/resize.cpp:3720: error:
(-215:Assertion failed) !ssize.empty() in function 'resize'

OpenCV(4.1.2) /io/opencv/modules/imgproc/src/resize.cpp:3720: error:
(-215:Assertion failed) !ssize.empty() in function 'resize'

OpenCV(4.1.2) /io/opencv/modules/imgproc/src/resize.cpp:3720: error:
(-215:Assertion failed) !ssize.empty() in function 'resize'

OpenCV(4.1.2) /io/opencv/modules/imgproc/src/resize.cpp:3720: error:
(-215:Assertion failed) !ssize.empty() in function 'resize'
```

[0]:

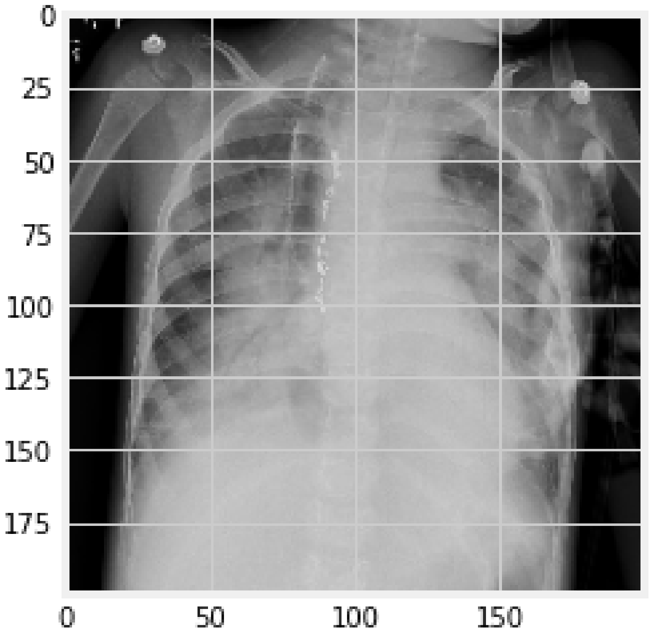
```
pnenumonia = 0
normal = 0

for i, j in train:
    if j == 0:
        pnenumonia+=1
    else:
        normal+=1

print('Pneumonia:', pnenumonia)
print('Normal:', normal)
print('Pneumonia - Normal:', pnenumonia-normal)
```

```
Pneumonia: 3875
Normal: 1341
Pneumonia - Normal: 2534
```

Visualize training images

[0]:
```
plt.imshow(train[1][0], cmap='gray')
print(labels[train[1][1]])
```

```
PNEUMONIA
```



We are incoprating the validation data into the training data because it does not contain enough examples.

[0]:
```
X = []
y = []

for feature, label in train:
    X.append(feature)
    y.append(label)

for feature, label in test:
    X.append(feature)
    y.append(label)

for feature, label in val:
    X.append(feature)
    y.append(label)
```