en güncel deep learn 43.ipynb

Dosya Düzenle Görüntüle Ekle Çalıştırma zamanı Araçlar Yardım Son kaydedilme tarihi: 21.58

Yorum Paylaş

+ Kod + Metin

Bağlan Düzenleme

```
from keras.models import Sequential,Model
from keras.layers.core import Flatten, Dense, Dropout
from keras.layers import Concat0, MaxPooling2D,Input,Concatenate,AveragePooling2D,BatchNormalization
from keras.preprocessing.image import ImageDataGenerator
[5/20] from keras.models import Sequential,Model
from keras.layers.core import Flatten, Dense, Dropout
from keras.layers import Concat0, MaxPooling2D,Input
from keras.applications.vgg16 import *VGG16
from keras.preprocessing.image import ImageDataGenerator
from keras.optimizers import SGD,Adam
```

+ Kod + Metin

```
base_model=VGG16(input_shape=(224,224,3),include_top=False)
base_model.trainable=False

x=Flatten()(base_model.output)
x=dense(1200,activation='relu')(x)


out=Dense(3,activation='softmax')(x)
model=Model(inputs=base_model.inputs,output=out)

model.summary()
print(len(model.layers))
for layer in model.layers:
  print(layer.name, ',',layer.trainable)

for layer in model.layers[:10]:
  layer.trainable=False
```

```
 block2_conv2 (Conv2D)       (None, 112, 112, 128)   147584
 block2_pool (MaxPooling2D)  (None, 56, 56, 128)     0
 block3_conv1 (Conv2D)       (None, 56, 56, 256)     295168
```

en güncel deep learn 43.ipynb

Dosya Düzenle Görüntüle Ekle Çalıştırma zamanı Araçlar Yardım Son kaydedilme tarihi: 21.58

Yorum Paylaş

+ Kod + Metin

Bağlan Düzenleme

```
 block4_conv1 (Conv2D)       (None, 28, 28, 512)     1180160
 block4_conv2 (Conv2D)       (None, 28, 28, 512)     2359808
 block4_conv3 (Conv2D)       (None, 28, 28, 512)     2359808
 block4_pool (MaxPooling2D)  (None, 14, 14, 512)     0
 block5_conv1 (Conv2D)       (None, 14, 14, 512)     2359808
 block5_conv2 (Conv2D)       (None, 14, 14, 512)     2359808
 block5_conv3 (Conv2D)       (None, 14, 14, 512)     2359808
 block5_pool (MaxPooling2D)  (None, 7, 7, 512)       0
 Flatten_2 (Flatten)         (None, 25088)           0
 dense_4 (Dense)             (None, 1200)            30104800
 dense_5 (Dense)             (None, 3)               9603
=================================================================
Total params: 95,009,091
Trainable params: 80,294,403
Non-trainable params: 14,714,688
_________________________________________________________________
22
input_3   False
block1_conv1   False
block1_conv2   False
block1_pool   False
block2_conv1   False
block2_conv2   False
block2_pool   False
block3_conv1   False
block3_conv2   False
block3_conv3   False
block3_pool   False
block4_conv1   False
block4_conv2   False
block4_conv3   False
block4_pool   False
block5_conv1   False
block5_conv2   False
block5_conv3   False
block5_pool   False
Flatten_2   True
dense_4   True
dense_5   True
```

en güncel deep learn 43.ipynb

Dosya Düzenle Görüntüle Ekle Çalıştırma zamanı Araçlar Yardım Son kaydedilme tarihi: 21.58

Yorum Paylaş

+ Kod + Metin

Bağlan Düzenleme

```
trainpath='./drive/MyDrive/drn_lrn_fnl/chest_xray/train/'
testPath='./drive/MyDrive/drn_lrn_fnl/chest_xray/test/'

traindataGen=ImageDataGenerator(
    rescale=1./255,
    validation_split=0.1,
    zoom_range=0.1,
    horizontal_flip=True,
    vertical_flip='True',
    width_shift_range=0.1,
    height_shift_range=0.1,
    shear_range=0.35,
)

testDataGen=ImageDataGenerator(rescale=1./255)

trainGen=traindataGen.flow_from_directory(
    trainPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
    batch_size=128,
    subset='training ,
    shuffle='True
)

valGen=trainDataGen.flow_from_directory(
    trainPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
    batch_size=128,
    subset='validation',
    shuffle='True
)

testGen=testDataGen.flow_from_directory(
    testPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
```

en güncel deep learn 43.ipynb

Dosya Düzenle Görüntüle Ekle Çalıştırma zamanı Araçlar Yardım Son kaydedilme tarihi: 21.58

Yorum Paylaş

+ Kod + Metin

Bağlan Düzenleme

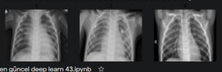
```
print(testGen.class_indices)
print(testGen.classes)

import matplotlib.pyplot as plt

plt.figure(figsize=(10,10))

for i in range(6):
  img_y=testGen.next()
  plt.subplot(2,3,i+1)
  plt.axis('off')
  plt.title(list(f[:199][0]).index(1))
  plt.imshow(img[0])
```

```
{'NORMAL': 0, 'bak': 1, 'vir': 2}
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2]
```



en güncel deep learn 43.ipynb

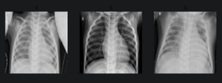
Dosya Düzenle Görüntüle Ekle Çalıştırma zamanı Araçlar Yardım Son kaydedilme tarihi: 21.58

Yorum Paylaş

+ Kod + Metin

Bağlan Düzenleme

```
[ ]
```



```
from keras.callbacks import EarlyStopping, ModelCheckpoint

callbacks=[EarlyStopping(monitor='val_loss',patience=1), ModelCheckpoint(filepath="bset_model.h5",monitor='val_loss',save_best_only='True')]

opt = SGD(learning_rate=0.00003,momentum=0.10)
model.compile(loss='categorical_crossentropy',optimizer=opt,metrics=['accuracy'])

history=model.fit(
    trainGen,
    steps_per_epoch=trainGen.samples//128,
    validation_data=valGen,
    epochs=50,
    validation_steps=valGen.samples//128,
    callbacks = callbacks
)
```

```
Epoch 8/50
35/35 [==============================] - 1486 s/step - loss: 0.9941 - accuracy: 0.4835 - val_loss: 0.9759 - val_accuracy: 0.5078
Epoch 9/50
35/35 [==============================] - 1331 s/step - loss: 0.9744 - accuracy: 0.5170 - val_loss: 0.9662 - val_accuracy: 0.5208
Epoch 10/50
35/35 [==============================] - 1241 s/step - loss: 0.9742 - accuracy: 0.5158 - val_loss: 0.9543 - val_accuracy: 0.5391
Epoch 11/50
35/35 [==============================] - 1275 s/step - loss: 0.9664 - accuracy: 0.5286 - val_loss: 0.9415 - val_accuracy: 0.5495
Epoch 12/50
35/35 [==============================] - 1381 s/step - loss: 0.9558 - accuracy: 0.5527 - val_loss: 0.9353 - val_accuracy: 0.5859
Epoch 13/50
35/35 [==============================] - 1341 s/step - loss: 0.9531 - accuracy: 0.5638 - val_loss: 0.9236 - val_accuracy: 0.5807
Epoch 14/50
35/35 [==============================] - 1411 s/step - loss: 0.9423 - accuracy: 0.5628 - val_loss: 0.9307 - val_accuracy: 0.5885
```

en güncel deep learn 43.ipynb

Dosya Düzenle Görüntüle Ekle Çalıştırma zamanı Araçlar Yardım Son kaydedilme tarihi: 21.58

Yorum Paylaş

+ Kod + Metin

Bağlan Düzenleme

```
Epoch 18/50
35/35 [==============================] - 1336 s/step - loss: 0.9253 - accuracy: 0.5958 - val_loss: 0.8996 - val_accuracy: 0.6224
Epoch 19/50
35/35 [==============================] - 1411 s/step - loss: 0.9227 - accuracy: 0.6094 - val_loss: 0.8963 - val_accuracy: 0.6172
Epoch 20/50
35/35 [==============================] - 1341 s/step - loss: 0.9168 - accuracy: 0.5908 - val_loss: 0.8862 - val_accuracy: 0.6458
Epoch 21/50
35/35 [==============================] - 1381 s/step - loss: 0.9109 - accuracy: 0.6104 - val_loss: 0.8963 - val_accuracy: 0.6250
Epoch 22/50
35/35 [==============================] - 1341 s/step - loss: 0.9078 - accuracy: 0.6189 - val_loss: 0.8903 - val_accuracy: 0.6328
Epoch 23/50
35/35 [==============================] - 1275 s/step - loss: 0.8978 - accuracy: 0.6243 - val_loss: 0.8755 - val_accuracy: 0.6354
Epoch 24/50
35/35 [==============================] - 1331 s/step - loss: 0.9009 - accuracy: 0.6175 - val_loss: 0.8733 - val_accuracy: 0.6302
Epoch 25/50
35/35 [==============================] - 1351 s/step - loss: 0.8975 - accuracy: 0.6249 - val_loss: 0.8812 - val_accuracy: 0.6068
Epoch 26/50
35/35 [==============================] - 1381 s/step - loss: 0.8866 - accuracy: 0.6353 - val_loss: 0.8766 - val_accuracy: 0.6146
Epoch 27/50
35/35 [==============================] - 1241 s/step - loss: 0.8985 - accuracy: 0.6276 - val_loss: 0.8616 - val_accuracy: 0.6380
Epoch 28/50
35/35 [==============================] - 1331 s/step - loss: 0.8881 - accuracy: 0.6232 - val_loss: 0.8539 - val_accuracy: 0.6484
Epoch 29/50
35/35 [==============================] - 1265 s/step - loss: 0.8784 - accuracy: 0.6378 - val_loss: 0.8513 - val_accuracy: 0.6667
Epoch 30/50
35/35 [==============================] - 1325 s/step - loss: 0.8690 - accuracy: 0.6451 - val_loss: 0.8438 - val_accuracy: 0.6797
Epoch 31/50
35/35 [==============================] - 1386 s/step - loss: 0.8643 - accuracy: 0.6478 - val_loss: 0.8494 - val_accuracy: 0.6382
Epoch 32/50
35/35 [==============================] - 1331 s/step - loss: 0.8692 - accuracy: 0.6369 - val_loss: 0.8592 - val_accuracy: 0.6354
Epoch 33/50
35/35 [==============================] - 1296 s/step - loss: 0.8598 - accuracy: 0.6406 - val_loss: 0.8143 - val_accuracy: 0.6823
Epoch 34/50
35/35 [==============================] - 1331 s/step - loss: 0.8614 - accuracy: 0.6477 - val_loss: 0.8402 - val_accuracy: 0.6250
Epoch 35/50
35/35 [==============================] - 1275 s/step - loss: 0.8459 - accuracy: 0.6685 - val_loss: 0.8248 - val_accuracy: 0.6693
Epoch 36/50
35/35 [==============================] - 1386 s/step - loss: 0.8495 - accuracy: 0.6486 - val_loss: 0.8197 - val_accuracy: 0.6667
Epoch 37/50
35/35 [==============================] - 1296 s/step - loss: 0.8458 - accuracy: 0.6467 - val_loss: 0.8294 - val_accuracy: 0.6380
Epoch 38/50
35/35 [==============================] - 1275 s/step - loss: 0.8471 - accuracy: 0.6402 - val_loss: 0.8263 - val_accuracy: 0.6484
```

```
[ ] import matplotlib.pyplot as plt

    plt.subplot(211)
    plt.title("Accuracy")
    plt.plot(history.history["accuracy"],color='r',label='train')
```

en güncel deep learn 43.ipynb

Dosya Düzenle Görüntüle Ekle Çalıştırma zamanı Araçlar Yardım Son kaydedilme tarihi: 21.58

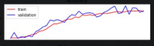
Yorum Paylaş

+ Kod + Metin

Bağlan Düzenleme

```
plt.plot(history.history["val_loss"],color='b',label='validation')
plt.legend(loc='best')


plt.tight_layout()
plt.show()
```



en güncel deep learn 43.ipynb

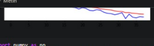
Dosya Düzenle Görüntüle Ekle Çalıştırma zamanı Araçlar Yardım Son kaydedilme tarihi: 21.58

Yorum Paylaş

+ Kod + Metin

Bağlan Düzenleme

```
[ ]
```



```
import numpy as np
pred = model.predict(testGen, steps=testGen.samples//1)
pred_cls = np.argmax(pred, axis=1)

tr_cls = testGen.classes
cls_lbl = list(testGen.class_indices.keys())


from sklearn.metrics import confusion_matrix, classification_report

print('Confusion matrisi')
cm_mx=confusion_matrix(tr_cls,pred_cls)
print(cm_mx,'\n')

print('Classification Bilgisi')
print(classification_report(tr_cls, pred_cls))
```

```
Confusion matrisi
[[ 99 134   1]
 [ 73 168   1]
 [ 55  93  92]]

Classification Bilgisi
              precision    recall  f1-score   support

           0       0.44      0.42      0.43       234
           1       0.43      0.69      0.53       242
           2       0.50      0.01      0.03       140

    accuracy                           0.43       624
   macro avg       0.45      0.38      0.33       624
weighted avg       0.45      0.43      0.37       624
```

ders13© son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 23:15

```
from keras.models import Sequential,Model
from keras.layers.core import Flatten, Dense, Dropout
from keras.layers import Conv2D, MaxPooling2D,Input,Concatenate,AveragePooling2D,BatchNormalization
from keras.preprocessing.image import ImageDataGenerator
from keras.models import Sequential,Model
from keras.layers.core import Flatten, Dense, Dropout
from keras.layers import Conv2D, MaxPooling2D,Input
from keras.applications.vgg16 import VGG16
from keras.preprocessing.image import ImageDataGenerator
```

```
base_model=VGG16(input_shape=(224,224,3),include_top=False)
base_model.trainable=False

x=Flatten()(base_model.output)
x=Dense(10000,activation='relu')(x)
Dropout(0.2)(x)
x=Dense(4100,activation='relu')(x)
Dropout(0.2)(x)
x=Dense(2048,activation='relu')(x)
Dropout(0.2)(x)
x=Dense(3200,activation='relu')(x)


out=Dense(3,activation='softmax')(x)
model=Model(inputs=base_model.inputs,outputs=out)

model.summary()
print(len(model.layers))
for layer in model.layers:
  print(layer.name," ",layer.trainable)

for layer in model.layers[:10]:
  layer.trainable=False
```

ders13© son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 23:15

```
input_1 (InputLayer)        [(None, 224, 224, 3)]     0
_________________________________________________________________
block1_conv1 (Conv2D)       (None, 224, 224, 64)      1792
_________________________________________________________________
block1_conv2 (Conv2D)       (None, 224, 224, 64)      36928
_________________________________________________________________
block1_pool (MaxPooling2D)  (None, 112, 112, 64)      0
_________________________________________________________________
block2_conv1 (Conv2D)       (None, 112, 112, 128)     73856
_________________________________________________________________
block2_conv2 (Conv2D)       (None, 112, 112, 128)     147584
_________________________________________________________________
block2_pool (MaxPooling2D)  (None, 56, 56, 128)       0
_________________________________________________________________
block3_conv1 (Conv2D)       (None, 56, 56, 256)       295168
_________________________________________________________________
block3_conv2 (Conv2D)       (None, 56, 56, 256)       590080
_________________________________________________________________
block3_conv3 (Conv2D)       (None, 56, 56, 256)       590080
_________________________________________________________________
block3_pool (MaxPooling2D)  (None, 28, 28, 256)       0
_________________________________________________________________
block4_conv1 (Conv2D)       (None, 28, 28, 512)       1180160
_________________________________________________________________
block4_conv2 (Conv2D)       (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_conv3 (Conv2D)       (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_pool (MaxPooling2D)  (None, 14, 14, 512)       0
_________________________________________________________________
block5_conv1 (Conv2D)       (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv2 (Conv2D)       (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv3 (Conv2D)       (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_pool (MaxPooling2D)  (None, 7, 7, 512)         0
_________________________________________________________________
flatten (Flatten)           (None, 25088)             0
_________________________________________________________________
dense (Dense)               (None, 10000)             250890000
_________________________________________________________________
dense_1 (Dense)             (None, 4100)              41004100
_________________________________________________________________
dense_2 (Dense)             (None, 2048)              8398848
_________________________________________________________________
dense_3 (Dense)             (None, 3200)              6556800
_________________________________________________________________
dense_4 (Dense)             (None, 3)                 9603
```

ders13© son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 23:15

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
trainPath="./drive/MyDrive/drn_lrn_fnl/chest_xray/train/"
testPath="./drive/MyDrive/drn_lrn_fnl/chest_xray/test/"

trainDataGen=ImageDataGenerator(
    rescale=1./255,
    validation_split=0.1,
    width_shift_range=0.6,
    height_shift_range=0.6,
    shear_range=0.60,
)

testDataGen=ImageDataGenerator(rescale=1./255)

trainGen=trainDataGen.flow_from_directory(
    trainPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
    batch_size=128,
    subset='training'
)

valGen=trainDataGen.flow_from_directory(
    trainPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
    batch_size=128,
    subset='validation'
)

testGen=testDataGen.flow_from_directory(
    testPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
```

ders13© son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 23:15

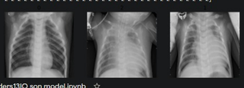
```
print(testGen.class_indices)
print(testGen.classes)

import matplotlib.pyplot as plt

plt.figure(figsize=(30,10))

for i in range(6):
  img,y=testGen.next()
  plt.subplot(2,3,i+1)
  plt.axis('off')
  plt.title(list(y[:189][0]).index(1))
  plt.imshow(img[0])
```

```
{'NORMAL': 0, 'bakteri': 1, 'virüs': 2}
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2]
```



ders13© son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 23:15

```
[ ]
```

```
from keras.callbacks import EarlyStopping, ModelCheckpoint

callbacks=[EarlyStopping(monitor='val_loss',patience=25), ModelCheckpoint(filepath="beset_model.h5",monitor='val_loss',save_best_only=True)]

model.compile(loss='categorical_crossentropy',optimizer='SGD',metrics=['accuracy'])

history=model.fit(
    trainGen,
    steps_per_epoch=trainGen.samples//128,
    validation_data=valGen,
    epochs=10,
    validation_steps=valGen.samples//128
)
```

```
Epoch 1/10
35/35 [==============================] - 103s 3s/step - loss: 1.2233 - accuracy: 0.4389 - val_loss: 0.9940 - val_accuracy: 0.5911
Epoch 2/10
35/35 [==============================] - 99s 3s/step - loss: 1.0169 - accuracy: 0.4989 - val_loss: 0.9332 - val_accuracy: 0.5547
Epoch 3/10
35/35 [==============================] - 99s 3s/step - loss: 1.0057 - accuracy: 0.4917 - val_loss: 0.9377 - val_accuracy: 0.6016
Epoch 4/10
35/35 [==============================] - 99s 3s/step - loss: 0.9654 - accuracy: 0.5574 - val_loss: 0.8836 - val_accuracy: 0.5521
Epoch 5/10
35/35 [==============================] - 99s 3s/step - loss: 0.9455 - accuracy: 0.5534 - val_loss: 1.0284 - val_accuracy: 0.4635
Epoch 6/10
35/35 [==============================] - 102s 3s/step - loss: 0.9457 - accuracy: 0.5464 - val_loss: 0.8488 - val_accuracy: 0.6458
Epoch 7/10
35/35 [==============================] - 102s 3s/step - loss: 0.8981 - accuracy: 0.5735 - val_loss: 0.9083 - val_accuracy: 0.5026
Epoch 8/10
35/35 [==============================] - 100s 3s/step - loss: 0.9389 - accuracy: 0.5451 - val_loss: 0.8593 - val_accuracy: 0.6172
Epoch 9/10
```

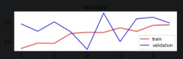
ders13© son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 23:15

```
plt.subplot(211)
plt.title("Accuracy")
plt.plot(history.history["accuracy"],color='r',label='train')
plt.plot(history.history["val_accuracy"],color='b',label='validation')
plt.legend(loc='best')

plt.subplot(212)
plt.title("Loss")
plt.plot(history.history["loss"],color='r',label='train')
plt.plot(history.history["val_loss"],color='b',label='validation')
plt.legend(loc='best')


plt.tight_layout()
plt.show()
```



ders13© son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 23:15

```
import numpy as np
pred = model.predict(testGen, steps=testGen.samples//1)
pred_cls = np.argmax(pred, axis=1)

tr_cls = testGen.classes
cls_lbl = list(testGen.class_indices.keys())


from sklearn.metrics import confusion_matrix, classification_report

print('Confusion Matrisi')
con_mx=confusion_matrix(tr_cls,pred_cls)
print(con_mx,'\n')

print('Classification Bilgisi')
print(classification_report(tr_cls, pred_cls))
```

```
Confusion Matrisi
[[ 67 167   0]
 [ 66 176   0]
 [ 52  96   0]]

Classification Bilgisi
              precision    recall  f1-score   support

           0       0.36      0.29      0.32       234
           1       0.40      0.73      0.52       242
           2       0.00      0.00      0.00       148

    accuracy                           0.39       624
   macro avg       0.25      0.34      0.28       624
weighted avg       0.29      0.39      0.32       624

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1272: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

39 ders13IO.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 21:58

Yorum Paylaş

+ Kod + Metin Bağlan Düzenleme

```
from keras.models import Sequential,Model
from keras.layers.core import Flatten, Dense, Dropout
from keras.layers import Conv2D, MaxPooling2D,Input,Concatenate,AveragePooling2D,BatchNormalization
from keras.preprocessing.image import ImageDataGenerator
from keras.models import Sequential,Model
from keras.layers.core import Flatten, Dense, Dropout
from keras.layers import Conv2D, MaxPooling2D,Input
from keras.applications.vgg16 import VGG16
from keras.preprocessing.image import ImageDataGenerator
from keras.optimizers import SGD,Adam
```

```
base_model=VGG16(input_shape=(224,224,3),include_top=False)
base_model.trainable=False

x=Flatten()(base_model.output)
x=Dense(3200,activation='relu')(x)


out=Dense(3,activation='softmax')(x)
model=Model(inputs=base_model.inputs,outputs=out)

model.summary()
print(len(model.layers))
for layer in model.layers:
  print(layer.name," ",layer.trainable)

for layer in model.layers[:10]:
  layer.trainable=False
```

```
Downloading data from https://storage.googleapis.com/tensorflow/keras-applications/vgg16/vgg16_weights_tf_dim_ordering_tf_kernels_notop.h5
58892288/58892256 [==============================] - 0s 0us/step
58900480/58892256 [==============================] - 0s 0us/step
Model: "model"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
_________________________________________________________________
block1_pool (MaxPooling2D)   (None, 112, 112, 64)      0
_________________________________________________________________
block2_conv1 (Conv2D)        (None, 112, 112, 128)     73856
_________________________________________________________________
block2_conv2 (Conv2D)        (None, 112, 112, 128)     147584
_________________________________________________________________
block2_pool (MaxPooling2D)   (None, 56, 56, 128)       0
_________________________________________________________________
block3_conv1 (Conv2D)        (None, 56, 56, 256)       295168
_________________________________________________________________
block3_conv2 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_conv3 (Conv2D)        (None, 56, 56, 256)       590080
_________________________________________________________________
block3_pool (MaxPooling2D)   (None, 28, 28, 256)       0
_________________________________________________________________
block4_conv1 (Conv2D)        (None, 28, 28, 512)       1180160
_________________________________________________________________
block4_conv2 (Conv2D)        (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_conv3 (Conv2D)        (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, 14, 14, 512)       0
_________________________________________________________________
block5_conv1 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv2 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv3 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, 7, 7, 512)         0
_________________________________________________________________
flatten (Flatten)            (None, 25088)             0
_________________________________________________________________
dense (Dense)                (None, 3200)              80204800
_________________________________________________________________
dense_1 (Dense)              (None, 3)                 9603
=================================================================
Total params: 99,009,091
Trainable params: 80,294,403
Non-trainable params: 14,714,688
_________________________________________________________________
3
input_1   False
block1_conv1   False
block1_conv2   False
block1_pool   False
block2_conv1   False
block2_conv2   False
```

39 ders13IO.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 21:58

Yorum Paylaş

+ Kod + Metin Bağlan Düzenleme

```
trainPath="./drive/MyDrive/drs_lrn_fnl/chest_xray/train/"
testPath="./drive/MyDrive/drs_lrn_fnl/chest_xray/test/"

trainDataGen=ImageDataGenerator(
    rescale=1./255,
    validation_split=0.1,
    zoom_range=0.2,
    horizontal_flip=True,
    vertical_flip=True,
    width_shift_range=0.1,
    height_shift_range=0.1,
    shear_range=0.35,
)

testDataGen=ImageDataGenerator(rescale=1./255)

trainGen=trainDataGen.flow_from_directory(
    trainPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
    batch_size=128,
    subset='training',
    shuffle=True
)

valGen=trainDataGen.flow_from_directory(
    trainPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
    batch_size=128,
    subset='validation',
    shuffle=True
)

testGen=testDataGen.flow_from_directory(
    testPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
```

39 ders13IO.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 21:58

Yorum Paylaş

+ Kod + Metin Bağlan Düzenleme

```
print(testGen.class_indices)
print(testGen.classes)

import matplotlib.pyplot as plt

plt.figure(figsize=(10,10))

for i in range(6):
  img,y=testGen.next()
  plt.subplot(2,3,i+1)
  plt.axis('off')
  plt.title(list(y[:195][0]).index(1))
  plt.imshow(img[0])
```

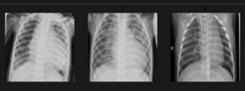
```
{'NORMAL': 0, 'bak': 1, 'vir': 2}
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2]
```

39 ders13IO.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 21:58

Yorum Paylaş

+ Kod + Metin Bağlan Düzenleme



```
from keras.callbacks import EarlyStopping, ModelCheckpoint

callbacks=[EarlyStopping(monitor='val_loss',patience=5), ModelCheckpoint(filepath="beset_model.h5",monitor='val_loss',save_best_only=True)]

opt = SGD(learning_rate=0.000001)
model.compile(loss="categorical_crossentropy",optimizer="SGD",metrics=["accuracy"])

history=model.fit(
    trainGen,
    steps_per_epoch=trainGen.samples//128,
    validation_data=valGen,
    epochs=10,
    validation_steps=valGen.samples//128,
    callbacks = callbacks
)
```

```
Epoch 1/10
35/35 [==============================] - 23345 65s/step - loss: 3.8547 - accuracy: 0.4344 - val_loss: 0.9023 - val_accuracy: 0.6484
Epoch 2/10
35/35 [==============================] - 1225 3s/step - loss: 0.9351 - accuracy: 0.5476 - val_loss: 0.9290 - val_accuracy: 0.4870
Epoch 3/10
35/35 [==============================] - 1205 3s/step - loss: 0.8734 - accuracy: 0.5971 - val_loss: 0.7426 - val_accuracy: 0.6667
Epoch 4/10
35/35 [==============================] - 1231 3s/step - loss: 0.7928 - accuracy: 0.6405 - val_loss: 0.6746 - val_accuracy: 0.6693
Epoch 5/10
35/35 [==============================] - 1205 3s/step - loss: 0.7843 - accuracy: 0.6226 - val_loss: 0.9489 - val_accuracy: 0.5260
Epoch 6/10
35/35 [==============================] - 1235 4s/step - loss: 0.7690 - accuracy: 0.6391 - val_loss: 0.8078 - val_accuracy: 0.6745
Epoch 7/10
35/35 [==============================] - 1205 3s/step - loss: 0.7283 - accuracy: 0.6576 - val_loss: 0.8429 - val_accuracy: 0.5599
Epoch 8/10
35/35 [==============================] - 1185 3s/step - loss: 0.7909 - accuracy: 0.6092 - val_loss: 0.7663 - val_accuracy: 0.6589
```

39 ders13IO.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 21:58

Yorum Paylaş

+ Kod + Metin Bağlan Düzenleme

```
import matplotlib.pyplot as plt

plt.subplot(211)
plt.title("accuracy")
plt.plot(history.history["accuracy"],color='r',label='train')
plt.plot(history.history["val_accuracy"],color='b',label='validation')
plt.legend(loc='best')

plt.subplot(212)
plt.title("loss")
plt.plot(history.history["loss"],color='r',label='train')
plt.plot(history.history["val_loss"],color='b',label='validation')
plt.legend(loc='best')


plt.tight_layout()
plt.show()
```

39 ders13IO.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Son kaydedilme tarihi: 21:58

Yorum Paylaş

+ Kod + Metin Bağlan Düzenleme

```
import numpy as np
pred = model.predict(testGen, steps=testGen.samples//1)
pred_cls = np.argmax(pred, axis=1)

tr_cls = testGen.classes
cls_lbl = list(testGen.class_indices.keys())


from sklearn.metrics import confusion_matrix, classification_report

print('Confusion Matrisi')
con_mx=confusion_matrix(tr_cls,pred_cls)
print(con_mx,"\n")

print('Classification Bilgisi')
print(classification_report(tr_cls, pred_cls))
```

```
Confusion Matrisi
[[ 83 123  28]
 [ 69 147  26]
 [ 46  86  16]]

Classification Bilgisi
              precision    recall  f1-score   support

           0       0.42      0.35      0.38       234
           1       0.41      0.61      0.49       242
           2       0.23      0.11      0.15       148

    accuracy                           0.39       624
   macro avg       0.35      0.36      0.34       624
weighted avg       0.37      0.39      0.37       624
```

ders13iO son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Tüm değişiklikler kaydedildi

Yorum Paylaş

+ Kod + Metin

Yeniden bağlan Düzenleme

```
[ ] from keras.models import Sequential,Model
    from keras.layers.core import Flatten, Dense, Dropout
    from keras.layers import Conv2D, MaxPooling2D,Input,Concatenate,AveragePooling2D,BatchNormalization
    from keras.preprocessing.image import ImageDataGenerator
    from keras.models import Sequential,Model
    from keras.layers.core import Flatten, Dense, Dropout
    from keras.layers import Conv2D, MaxPooling2D,Input
    from keras.applications.vgg16 import VGG16
    from keras.preprocessing.image import ImageDataGenerator
```

```
[ ] base_model=VGG16(input_shape=(224,224,3),include_top=False)
    base_model.trainable=False

    x=Flatten()(base_model.output)
    x=Dense(10000,activation='relu')(x)
    Dropout(0.2)(x)
    x=Dense(4100,activation='relu')(x)
    Dropout(0.2)(x)
    x=Dense(2048,activation='relu')(x)
    Dropout(0.2)(x)
    x=Dense(3200,activation='relu')(x)


    out=Dense(3,activation='softmax')(x)
    model=Model(inputs=base_model.inputs,outputs=out)

    model.summary()
    print(len(model.layers))
    for layer in model.layers:
      print(layer.name," ",layer.trainable)

    for layer in model.layers[:10]:
      layer.trainable=False
```

```
block3_conv3 (Conv2D)        (None, 56, 56, 256)       590080
```

ders13iO son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Tüm değişiklikler kaydedildi

Yorum Paylaş

+ Kod + Metin

Yeniden bağlan Düzenleme

```
block4_conv3 (Conv2D)        (None, 28, 28, 512)       2359808
_________________________________________________________________
block4_pool (MaxPooling2D)   (None, 14, 14, 512)       0
_________________________________________________________________
block5_conv1 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv2 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_conv3 (Conv2D)        (None, 14, 14, 512)       2359808
_________________________________________________________________
block5_pool (MaxPooling2D)   (None, 7, 7, 512)         0
_________________________________________________________________
flatten (Flatten)            (None, 25088)             0
_________________________________________________________________
dense (Dense)                (None, 10000)             250890000
_________________________________________________________________
dense_1 (Dense)              (None, 4100)              41004100
_________________________________________________________________
dense_2 (Dense)              (None, 2048)              8398848
_________________________________________________________________
dense_3 (Dense)              (None, 3200)              6556800
_________________________________________________________________
dense_4 (Dense)              (None, 3)                 9603
=================================================================
Total params: 323,574,039
Trainable params: 306,059,355
Non-trainable params: 14,714,688
_________________________________________________________________
25
input_1  False
block1_conv1  False
block1_conv2  False
block1_pool  False
block2_conv1  False
block2_conv2  False
block2_pool  False
block3_conv1  False
block3_conv2  False
block3_conv3  False
block3_pool  False
block4_conv1  False
block4_conv2  False
block4_conv3  False
block4_pool  False
block5_conv1  False
block5_conv2  False
block5_conv3  False
block5_pool  False
flatten  True
```

ders13iO son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Tüm değişiklikler kaydedildi

Yorum Paylaş

+ Kod + Metin

Yeniden bağlan Düzenleme

```
[ ] from google.colab import drive
    drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
trainPath="./drive/MyDrive/drn_lrn_fnl/chest_xray/train/"
testPath="./drive/MyDrive/drn_lrn_fnl/chest_xray/test/"

trainDataGen=ImageDataGenerator(
    rescale=1./255,
    validation_split=0.1,
    width_shift_range=0.0,
    height_shift_range=0.0,
    shear_range=0.02,
)

testDataGen=ImageDataGenerator(rescale=1./255)

trainGen=trainDataGen.flow_from_directory(
    trainPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
    batch_size=128,
    subset='training'
)

valGen=trainDataGen.flow_from_directory(
    trainPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
    batch_size=128,
    subset='validation'
)

testGen=testDataGen.flow_from_directory(
    testPath,
    target_size=(224,224),
    color_mode='rgb',
    class_mode='categorical',
```

ders13iO son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Tüm değişiklikler kaydedildi

Yorum Paylaş

+ Kod + Metin

Yeniden bağlan Düzenleme

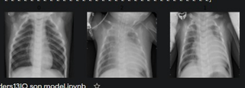
```
print(testGen.class_indices)
print(testGen.classes)

import matplotlib.pyplot as plt

plt.figure(figsize=(30,10))

for i in range(6):
  img,y=testGen.next()
  plt.subplot(2,3,i+1)
  plt.axis('off')
  plt.title(list(y[:189][0]).index(1))
  plt.imshow(img[0])
```

```
{'NORMAL': 0, 'bakteri': 1, 'virüs': 2}
[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2]
```



ders13iO son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Tüm değişiklikler kaydedildi

Yorum Paylaş

+ Kod + Metin

Yeniden bağlan Düzenleme

```
[ ]
```

```
from keras.callbacks import EarlyStopping, ModelCheckpoint

callbacks=[EarlyStopping(monitor='val_loss',patience=25), ModelCheckpoint(filepath="beset_model.h5",monitor='val_loss',save_best_only=True)]

model.compile(loss='categorical_crossentropy',optimizer='SGD',metrics=['accuracy'])

history=model.fit(
    trainGen,
    steps_per_epoch=trainGen.samples//128,
    validation_data=valGen,
    epochs=10,
    validation_steps=valGen.samples//128
)
```

```
Epoch 1/10
35/35 [==============================] - 103s 3s/step - loss: 1.2233 - accuracy: 0.4389 - val_loss: 0.9940 - val_accuracy: 0.5911
Epoch 2/10
35/35 [==============================] - 99s 3s/step - loss: 1.0169 - accuracy: 0.4989 - val_loss: 0.9332 - val_accuracy: 0.5547
Epoch 3/10
35/35 [==============================] - 99s 3s/step - loss: 1.0057 - accuracy: 0.4917 - val_loss: 0.9377 - val_accuracy: 0.6016
Epoch 4/10
35/35 [==============================] - 99s 3s/step - loss: 0.9654 - accuracy: 0.5574 - val_loss: 0.8836 - val_accuracy: 0.5521
Epoch 5/10
35/35 [==============================] - 99s 3s/step - loss: 0.9455 - accuracy: 0.5534 - val_loss: 1.0284 - val_accuracy: 0.4635
Epoch 6/10
35/35 [==============================] - 102s 3s/step - loss: 0.9457 - accuracy: 0.5464 - val_loss: 0.8488 - val_accuracy: 0.6458
Epoch 7/10
35/35 [==============================] - 102s 3s/step - loss: 0.8981 - accuracy: 0.5735 - val_loss: 0.9083 - val_accuracy: 0.5026
Epoch 8/10
35/35 [==============================] - 100s 3s/step - loss: 0.9389 - accuracy: 0.5451 - val_loss: 0.8593 - val_accuracy: 0.6172
Epoch 9/10
```

ders13iO son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Tüm değişiklikler kaydedildi

Yorum Paylaş

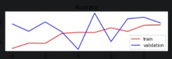
+ Kod + Metin

Yeniden bağlan Düzenleme

```
plt.subplot(211)
plt.title("Accuracy")
plt.plot(history.history["accuracy"],color='r',label='train')
plt.plot(history.history["val_accuracy"],color='b',label='validation')
plt.legend(loc='best')

plt.subplot(212)
plt.title("Loss")
plt.plot(history.history["loss"],color='r',label='train')
plt.plot(history.history["val_loss"],color='b',label='validation')
plt.legend(loc='best')


plt.tight_layout()
plt.show()
```



ders13iO son model.ipynb

Dosya Düzenle Göster Ekle Çalışma zamanı Araçlar Yardım Tüm değişiklikler kaydedildi

Yorum Paylaş

+ Kod + Metin

Yeniden bağlan Düzenleme

```
import numpy as np
pred = model.predict(testGen, steps=testGen.samples//1)
pred_cls = np.argmax(pred, axis=1)

tr_cls = testGen.classes
cls_lbl = list(testGen.class_indices.keys())


from sklearn.metrics import confusion_matrix, classification_report

print('Confusion Matrisi')
con_mx=confusion_matrix(tr_cls,pred_cls)
print(con_mx,'\n')

print('Classification Bilgisi')
print(classification_report(tr_cls, pred_cls))
```

```
Confusion Matrisi
[[ 67 167   0]
 [ 66 176   0]
 [ 52  96   0]]

Classification Bilgisi
              precision    recall  f1-score   support

           0       0.36      0.29      0.32       234
           1       0.40      0.73      0.52       242
           2       0.00      0.00      0.00       148

    accuracy                           0.39       624
   macro avg       0.25      0.34      0.28       624
weighted avg       0.29      0.39      0.32       624

/usr/local/lib/python3.7/dist-packages/sklearn/metrics/_classification.py:1272: UndefinedMetricWarning: Precision and F-score are ill-defined and being set to 0.0 in labels with no predicted samples. Use `zero_division` parameter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```