# VGG16 Model 0

Hardware: Google Collab GPU

Software: Tensorflow, Keras

Dataset: 5,863 X-Ray images (JPEG) and 2 categories (Pneumonia/Normal). https://data.mendeley.com/datasets/rscbjbr9sj/2

This model objective is to classify pneumonia in chest x-ray. This model initial build from the guidance from community and resources such as Tensorflow Community, Google Colab Community, Medium and other resources. Specific project that are closely related to this can be found below.

### Reference

* Google Collab - rock, paper, scissors notebook: https://colab.research.google.com/github/lmoroney/dlaicourse/blob/master/Course%202%20-%20Part%208%20-%20Lesson%202%20-%20Notebook%20(RockPaperScissors).ipynb#scrollTo=LWTisYLQM1aM
* Easy to understand notebook: https://www.kaggle.com/joythabo33/99-accurate-cnn-that-detects-pneumonia/notebook
* Unit8 pneumonia git: https://github.com/unit8co/amld-workshop-pneumonia/tree/master/3\_pneumonia

!pip install tensorflow-gpu

Collecting tensorflow-gpu  
[?25l Downloading https://files.pythonhosted.org/packages/25/44/47f0722aea081697143fbcf5d2aa60d1aee4aaacb5869aee2b568974777b/tensorflow\_gpu-2.0.0-cp36-cp36m-manylinux2010\_x86\_64.whl (380.8MB)  
[K |████████████████████████████████| 380.8MB 84kB/s   
[?25hRequirement already satisfied: keras-preprocessing>=1.0.5 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (1.1.0)  
Requirement already satisfied: keras-applications>=1.0.8 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (1.0.8)  
Requirement already satisfied: gast==0.2.2 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (0.2.2)  
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (3.1.0)  
Requirement already satisfied: wrapt>=1.11.1 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (1.11.2)  
Requirement already satisfied: protobuf>=3.6.1 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (3.7.1)  
Collecting tensorboard<2.1.0,>=2.0.0 (from tensorflow-gpu)  
[?25l Downloading https://files.pythonhosted.org/packages/9b/a6/e8ffa4e2ddb216449d34cfcb825ebb38206bee5c4553d69e7bc8bc2c5d64/tensorboard-2.0.0-py3-none-any.whl (3.8MB)  
[K |████████████████████████████████| 3.8MB 27.0MB/s   
[?25hRequirement already satisfied: termcolor>=1.1.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (1.1.0)  
Requirement already satisfied: wheel>=0.26 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (0.33.6)  
Requirement already satisfied: google-pasta>=0.1.6 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (0.1.7)  
Requirement already satisfied: absl-py>=0.7.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (0.8.0)  
Collecting tensorflow-estimator<2.1.0,>=2.0.0 (from tensorflow-gpu)  
[?25l Downloading https://files.pythonhosted.org/packages/95/00/5e6cdf86190a70d7382d320b2b04e4ff0f8191a37d90a422a2f8ff0705bb/tensorflow\_estimator-2.0.0-py2.py3-none-any.whl (449kB)  
[K |████████████████████████████████| 450kB 48.7MB/s   
[?25hRequirement already satisfied: grpcio>=1.8.6 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (1.15.0)  
Requirement already satisfied: astor>=0.6.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (0.8.0)  
Requirement already satisfied: six>=1.10.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (1.12.0)  
Requirement already satisfied: numpy<2.0,>=1.16.0 in /usr/local/lib/python3.6/dist-packages (from tensorflow-gpu) (1.16.5)  
Requirement already satisfied: h5py in /usr/local/lib/python3.6/dist-packages (from keras-applications>=1.0.8->tensorflow-gpu) (2.8.0)  
Requirement already satisfied: setuptools in /usr/local/lib/python3.6/dist-packages (from protobuf>=3.6.1->tensorflow-gpu) (41.2.0)  
Requirement already satisfied: werkzeug>=0.11.15 in /usr/local/lib/python3.6/dist-packages (from tensorboard<2.1.0,>=2.0.0->tensorflow-gpu) (0.16.0)  
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.6/dist-packages (from tensorboard<2.1.0,>=2.0.0->tensorflow-gpu) (3.1.1)  
[31mERROR: tensorflow 1.15.0rc3 has requirement tensorboard<1.16.0,>=1.15.0, but you'll have tensorboard 2.0.0 which is incompatible.[0m  
[31mERROR: tensorflow 1.15.0rc3 has requirement tensorflow-estimator==1.15.1, but you'll have tensorflow-estimator 2.0.0 which is incompatible.[0m  
Installing collected packages: tensorboard, tensorflow-estimator, tensorflow-gpu  
 Found existing installation: tensorboard 1.15.0  
 Uninstalling tensorboard-1.15.0:  
 Successfully uninstalled tensorboard-1.15.0  
 Found existing installation: tensorflow-estimator 1.15.1  
 Uninstalling tensorflow-estimator-1.15.1:  
 Successfully uninstalled tensorflow-estimator-1.15.1  
Successfully installed tensorboard-2.0.0 tensorflow-estimator-2.0.0 tensorflow-gpu-2.0.0

import tensorflow as tf  
print(tf.\_\_version\_\_)

The default version of TensorFlow in Colab will soon switch to TensorFlow 2.x. We recommend you upgrade now or ensure your notebook will continue to use TensorFlow 1.x via the %tensorflow\_version 1.x magic: more info.

1.15.0

!ls /content/

sample\_data

!du -s /content/data/chest\_xray

1219956 /content/data/chest\_xray

!du /content/data/chest\_xray

du: cannot access '/content/data/chest\_xray': No such file or directory

!pwd

/content

## Make directory

To save the dataset

!mkdir /content/data/

## Download the dataset

Downloading from unit8 instead of from directly mendely database since this dataset is splitted and available to downloaded here.

!wget --no-check-certificate \  
 https://s3.eu-central-1.amazonaws.com/public.unit8.co/data/chest\_xray.tar.gz \  
 -O /content/data/chest\_xray.tar.gz

--2019-11-14 06:11:29-- https://s3.eu-central-1.amazonaws.com/public.unit8.co/data/chest\_xray.tar.gz  
Resolving s3.eu-central-1.amazonaws.com (s3.eu-central-1.amazonaws.com)... 52.219.72.143  
Connecting to s3.eu-central-1.amazonaws.com (s3.eu-central-1.amazonaws.com)|52.219.72.143|:443... connected.  
HTTP request sent, awaiting response... 200 OK  
Length: 1225393795 (1.1G) [application/x-gzip]  
Saving to: ‘/content/data/chest\_xray.tar.gz’  
  
/content/data/chest 100%[===================>] 1.14G 98.2MB/s in 12s   
  
2019-11-14 06:11:41 (95.9 MB/s) - ‘/content/data/chest\_xray.tar.gz’ saved [1225393795/1225393795]

## Extract the downloaded zip file

import os  
import tarfile  
  
tar = tarfile.open("data/chest\_xray.tar.gz")  
tar.extractall(path='./data/')  
os.remove('data/chest\_xray.tar.gz')

## Data preprocessing and manipulation

import keras\_preprocessing  
from keras\_preprocessing.image import ImageDataGenerator # Data preprocessing and augmentation  
  
TRAINING\_DIR = "/content/data/chest\_xray/train"  
VALIDATION\_DIR = "/content/data/chest\_xray/val"  
TEST\_DIR = "/content/data/chest\_xray/test"  
  
# TODO: Data augmentation - Fiddle with images for training  
training\_datagen = ImageDataGenerator(  
 rescale = 1./255,  
 rotation\_range=40,  
 width\_shift\_range=0.2,  
 height\_shift\_range=0.2,  
 shear\_range=0.2,  
 zoom\_range=0.2,  
 horizontal\_flip=True,  
 fill\_mode='nearest'  
)  
  
validation\_datagen = ImageDataGenerator()  
  
test\_datagen = ImageDataGenerator()  
  
# Create training data batch  
# TODO: Try grayscaling the image to see what will happen  
train\_generator = training\_datagen.flow\_from\_directory(  
 TRAINING\_DIR,  
 target\_size=(150,150), # Resize the image to 150px x 150px; Why? idk... Check Unit8 work..  
 class\_mode='categorical'  
)  
  
validation\_generator = validation\_datagen.flow\_from\_directory(  
 VALIDATION\_DIR,  
 target\_size=(150,150),  
 class\_mode='categorical'  
)  
  
test\_generator = test\_datagen.flow\_from\_directory(  
 TEST\_DIR,  
 target\_size=(150,150),  
 class\_mode='categorical'  
)

Found 5216 images belonging to 2 classes.  
Found 16 images belonging to 2 classes.  
Found 624 images belonging to 2 classes.

train\_generator.image\_shape

(150, 150, 3)

## Define the Model

VGG16 model for Keras

This is the Keras model of the 16-layer network used by the VGG team in the ILSVRC-2014 competition.

It has been obtained by directly converting the Caffe model provived by the authors.

Details about the network architecture can be found in the following arXiv paper:

Very Deep Convolutional Networks for Large-Scale Image Recognition K. Simonyan, A. Zisserman arXiv:1409.1556

In the paper, the VGG-16 model is denoted as configuration D. It achieves 7.5% top-5 error on ILSVRC-2012-val, 7.4% top-5 error on ILSVRC-2012-test.

#VGG16 Model  
model = tf.keras.models.Sequential([  
 # First convolution layer  
 tf.keras.layers.ZeroPadding2D((1,1),input\_shape=train\_generator.image\_shape),  
 tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(64, (3, 3), activation='relu'),  
 tf.keras.layers.MaxPooling2D((2,2), strides=(2,2)),  
   
 # Second convolution layer  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(128, (3, 3), activation='relu'),  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(128, (3, 3), activation='relu'),  
 tf.keras.layers.MaxPooling2D((2,2), strides=(2,2)),  
   
 # Third convolution layer  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(256, (3, 3), activation='relu'),  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(256, (3, 3), activation='relu'),  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(256, (3, 3), activation='relu'),  
 tf.keras.layers.MaxPooling2D((2,2), strides=(2,2)),  
   
 # Fourth convolution layer  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(512, (3, 3), activation='relu'),  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(512, (3, 3), activation='relu'),  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(512, (3, 3), activation='relu'),  
 tf.keras.layers.MaxPooling2D((2,2), strides=(2,2)),  
   
 # Fifth convolution layer  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(512, (3, 3), activation='relu'),  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(512, (3, 3), activation='relu'),  
 tf.keras.layers.ZeroPadding2D((1,1)),  
 tf.keras.layers.Conv2D(512, (3, 3), activation='relu'),  
 tf.keras.layers.MaxPooling2D((2,2), strides=(2,2)),  
   
 # Flatten the results and feed into a DNN  
 tf.keras.layers.Flatten(),  
 tf.keras.layers.Dense(4096, activation='relu'),  
 tf.keras.layers.Dropout(0.5),  
 tf.keras.layers.Dense(4096, activation='relu'),  
 tf.keras.layers.Dropout(0.5),  
 tf.keras.layers.Dense(2, activation='softmax')  
])  
  
model.summary()

WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorflow\_core/python/ops/resource\_variable\_ops.py:1630: calling BaseResourceVariable.\_\_init\_\_ (from tensorflow.python.ops.resource\_variable\_ops) with constraint is deprecated and will be removed in a future version.  
Instructions for updating:  
If using Keras pass \*\_constraint arguments to layers.  
Model: "sequential"  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Layer (type) Output Shape Param #   
=================================================================  
zero\_padding2d (ZeroPadding2 (None, 152, 152, 3) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d (Conv2D) (None, 150, 150, 64) 1792   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_1 (ZeroPaddin (None, 152, 152, 64) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_1 (Conv2D) (None, 150, 150, 64) 36928   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
max\_pooling2d (MaxPooling2D) (None, 75, 75, 64) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_2 (ZeroPaddin (None, 77, 77, 64) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_2 (Conv2D) (None, 75, 75, 128) 73856   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_3 (ZeroPaddin (None, 77, 77, 128) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_3 (Conv2D) (None, 75, 75, 128) 147584   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
max\_pooling2d\_1 (MaxPooling2 (None, 37, 37, 128) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_4 (ZeroPaddin (None, 39, 39, 128) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_4 (Conv2D) (None, 37, 37, 256) 295168   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_5 (ZeroPaddin (None, 39, 39, 256) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_5 (Conv2D) (None, 37, 37, 256) 590080   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_6 (ZeroPaddin (None, 39, 39, 256) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_6 (Conv2D) (None, 37, 37, 256) 590080   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
max\_pooling2d\_2 (MaxPooling2 (None, 18, 18, 256) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_7 (ZeroPaddin (None, 20, 20, 256) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_7 (Conv2D) (None, 18, 18, 512) 1180160   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_8 (ZeroPaddin (None, 20, 20, 512) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_8 (Conv2D) (None, 18, 18, 512) 2359808   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_9 (ZeroPaddin (None, 20, 20, 512) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_9 (Conv2D) (None, 18, 18, 512) 2359808   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
max\_pooling2d\_3 (MaxPooling2 (None, 9, 9, 512) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_10 (ZeroPaddi (None, 11, 11, 512) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_10 (Conv2D) (None, 9, 9, 512) 2359808   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_11 (ZeroPaddi (None, 11, 11, 512) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_11 (Conv2D) (None, 9, 9, 512) 2359808   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
zero\_padding2d\_12 (ZeroPaddi (None, 11, 11, 512) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
conv2d\_12 (Conv2D) (None, 9, 9, 512) 2359808   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
max\_pooling2d\_4 (MaxPooling2 (None, 4, 4, 512) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
flatten (Flatten) (None, 8192) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense (Dense) (None, 4096) 33558528   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout (Dropout) (None, 4096) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_1 (Dense) (None, 4096) 16781312   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dropout\_1 (Dropout) (None, 4096) 0   
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
dense\_2 (Dense) (None, 2) 8194   
=================================================================  
Total params: 65,062,722  
Trainable params: 65,062,722  
Non-trainable params: 0  
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## Callbacks function

early\_stopping\_monitor = tf.keras.callbacks.EarlyStopping(patience = 3, monitor = "val\_accuracy", mode="max", verbose = 1)

## Compile the model

Here we use the “cross-entropy” loss function, which works well for learning probability distributions for classification.

See e.g.: https://ml-cheatsheet.readthedocs.io/en/latest/loss\_functions.html#cross-entropy

# optimizer = tf.keras.optimizers.Adam(learning\_rate= 0.0001)  
optimizer = 'rmsprop'  
model.compile(loss='categorical\_crossentropy',   
 optimizer=optimizer,   
 metrics=['accuracy'])

Since the training set is un-balanced. Calculate the classweight to be used for weight balancing to solve accuracy and loss being stucked.

import sklearn  
import numpy as np  
  
classweight = sklearn.utils.class\_weight.compute\_class\_weight('balanced', np.unique(train\_generator.labels), train\_generator.labels)  
print(classweight)

[1.9448173 0.67303226]

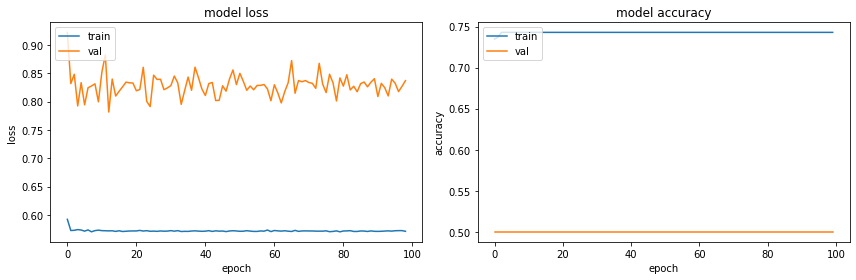
## Train the model

# Training process  
history = model.fit\_generator(  
 generator=train\_generator,   
 # steps\_per\_epoch=500,   
 epochs=100,  
 # callbacks=[early\_stopping\_monitor],  
 shuffle=True,   
 validation\_data=validation\_generator,   
 # validation\_steps=10,   
 class\_weight=classweight,  
 verbose = 1  
 )  
  
# model.save("pneumonia\_detection\_v1")  
  
### Plot training  
import matplotlib.pyplot as plt  
def plot\_learning\_curves(history):  
 plt.figure(figsize=(12,4))  
   
 plt.subplot(1,2,1)  
 plt.plot(history.history['loss'])  
 plt.plot(history.history['val\_loss'])  
 plt.title('model loss')  
 plt.ylabel('loss')  
 plt.xlabel('epoch')  
 plt.legend(['train', 'val'], loc='upper left')  
   
 plt.subplot(1,2,2)  
 plt.plot(history.history['acc'])  
 plt.plot(history.history['val\_acc'])  
 plt.title('model accuracy')  
 plt.ylabel('accuracy')  
 plt.xlabel('epoch')  
 plt.legend(['train', 'val'], loc='upper left')  
   
 plt.tight\_layout()  
   
plot\_learning\_curves(history)  
  
## Load best weight  
idx = np.argmin(history.history['val\_loss'])   
model.load\_weights("/content/data/model/weights.epoch\_{:02d}.hdf5".format(idx + 1))  
  
print("Loading the best model")  
print("epoch: {}, val\_loss: {}, val\_acc: {}".format(idx + 1, history.history['val\_loss'][idx], history.history['val\_acc'][idx]))  
  
## Evaluate the model  
test\_loss, test\_acc = model.evaluate\_generator(generator=test\_generator, verbose=1)  
  
## Analytics  
from sklearn.metrics import accuracy\_score, confusion\_matrix  
  
test\_generator.reset()  
test\_preds = model.predict\_generator(test\_generator, verbose=1)  
test\_preds = np.argmax(test\_preds,axis=1)  
  
acc = accuracy\_score(test\_generator.classes, test\_preds)\*100  
cm = confusion\_matrix(test\_generator.classes, test\_preds)  
tn, fp, fn, tp = cm.ravel()  
  
print('CONFUSION MATRIX ------------------')  
print(cm)  
# plot\_confusion\_matrix(cm, target\_names=['NORMAL', 'PNEUMONIA'], normalize=False)  
  
  
print('\nTEST METRICS ----------------------')  
precision = tp/(tp+fp)\*100  
recall = tp/(tp+fn)\*100  
print('Accuracy: {}%'.format(acc))  
print('Precision: {}%'.format(precision))  
print('Recall: {}%'.format(recall))  
print('F1-score: {}'.format(2\*precision\*recall/(precision+recall)))  
  
print('\nTRAIN METRIC ----------------------')  
print('Train acc: {}%'.format(np.round((history.history['acc'][-1])\*100, 14)))

Epoch 1/100  
162/163 [============================>.] - ETA: 0s - loss: 1197.9604 - acc: 0.7350Epoch 1/100  
163/163 [==============================] - 116s 712ms/step - loss: 1190.6147 - acc: 0.7349 - val\_loss: 0.8432 - val\_acc: 0.5000  
Epoch 2/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5927 - acc: 0.7365Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5921 - acc: 0.7370 - val\_loss: 0.9222 - val\_acc: 0.5000  
Epoch 3/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5728 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 110s 675ms/step - loss: 0.5723 - acc: 0.7429 - val\_loss: 0.8318 - val\_acc: 0.5000  
Epoch 4/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5734 - acc: 0.7423Epoch 1/100  
163/163 [==============================] - 110s 673ms/step - loss: 0.5727 - acc: 0.7429 - val\_loss: 0.8482 - val\_acc: 0.5000  
Epoch 5/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5742 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 669ms/step - loss: 0.5740 - acc: 0.7429 - val\_loss: 0.7923 - val\_acc: 0.5000  
Epoch 6/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5732 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5732 - acc: 0.7429 - val\_loss: 0.8334 - val\_acc: 0.5000  
Epoch 7/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5707 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 108s 666ms/step - loss: 0.5713 - acc: 0.7429 - val\_loss: 0.7943 - val\_acc: 0.5000  
Epoch 8/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5736 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 110s 674ms/step - loss: 0.5734 - acc: 0.7429 - val\_loss: 0.8244 - val\_acc: 0.5000  
Epoch 9/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5703 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 108s 661ms/step - loss: 0.5700 - acc: 0.7429 - val\_loss: 0.8276 - val\_acc: 0.5000  
Epoch 10/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5720 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 108s 664ms/step - loss: 0.5722 - acc: 0.7429 - val\_loss: 0.8314 - val\_acc: 0.5000  
Epoch 11/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5721 - acc: 0.7438Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5730 - acc: 0.7429 - val\_loss: 0.7996 - val\_acc: 0.5000  
Epoch 12/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5719 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5721 - acc: 0.7429 - val\_loss: 0.8517 - val\_acc: 0.5000  
Epoch 13/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5720 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5719 - acc: 0.7429 - val\_loss: 0.8824 - val\_acc: 0.5000  
Epoch 14/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5715 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5717 - acc: 0.7429 - val\_loss: 0.7816 - val\_acc: 0.5000  
Epoch 15/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5725 - acc: 0.7423Epoch 1/100  
163/163 [==============================] - 108s 663ms/step - loss: 0.5719 - acc: 0.7429 - val\_loss: 0.8398 - val\_acc: 0.5000  
Epoch 16/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5701 - acc: 0.7436Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5708 - acc: 0.7429 - val\_loss: 0.8098 - val\_acc: 0.5000  
Epoch 17/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5709 - acc: 0.7438Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5719 - acc: 0.7429 - val\_loss: 0.8182 - val\_acc: 0.5000  
Epoch 18/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5705 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 109s 669ms/step - loss: 0.5706 - acc: 0.7429 - val\_loss: 0.8260 - val\_acc: 0.5000  
Epoch 19/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5719 - acc: 0.7421Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5710 - acc: 0.7429 - val\_loss: 0.8343 - val\_acc: 0.5000  
Epoch 20/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5720 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5715 - acc: 0.7429 - val\_loss: 0.8331 - val\_acc: 0.5000  
Epoch 21/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5716 - acc: 0.7429 - val\_loss: 0.8327 - val\_acc: 0.5000  
Epoch 22/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5716 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 669ms/step - loss: 0.5716 - acc: 0.7429 - val\_loss: 0.8192 - val\_acc: 0.5000  
Epoch 23/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5717 - acc: 0.7436Epoch 1/100  
163/163 [==============================] - 109s 670ms/step - loss: 0.5725 - acc: 0.7429 - val\_loss: 0.8213 - val\_acc: 0.5000  
Epoch 24/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5724 - acc: 0.7421Epoch 1/100  
163/163 [==============================] - 109s 670ms/step - loss: 0.5714 - acc: 0.7429 - val\_loss: 0.8605 - val\_acc: 0.5000  
Epoch 25/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5708 - acc: 0.7442Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5721 - acc: 0.7429 - val\_loss: 0.8005 - val\_acc: 0.5000  
Epoch 26/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5699 - acc: 0.7440Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5710 - acc: 0.7429 - val\_loss: 0.7911 - val\_acc: 0.5000  
Epoch 27/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5713 - acc: 0.7429 - val\_loss: 0.8466 - val\_acc: 0.5000  
Epoch 28/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5709 - acc: 0.7429 - val\_loss: 0.8391 - val\_acc: 0.5000  
Epoch 29/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5716 - acc: 0.7429 - val\_loss: 0.8394 - val\_acc: 0.5000  
Epoch 30/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5714 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 669ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8210 - val\_acc: 0.5000  
Epoch 31/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5713 - acc: 0.7429 - val\_loss: 0.8241 - val\_acc: 0.5000  
Epoch 32/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5726 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 108s 664ms/step - loss: 0.5721 - acc: 0.7429 - val\_loss: 0.8282 - val\_acc: 0.5000  
Epoch 33/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5714 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8451 - val\_acc: 0.5000  
Epoch 34/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5715 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 108s 663ms/step - loss: 0.5721 - acc: 0.7429 - val\_loss: 0.8330 - val\_acc: 0.5000  
Epoch 35/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5707 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 108s 666ms/step - loss: 0.5706 - acc: 0.7429 - val\_loss: 0.7951 - val\_acc: 0.5000  
Epoch 36/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5704 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5709 - acc: 0.7429 - val\_loss: 0.8195 - val\_acc: 0.5000  
Epoch 37/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5717 - acc: 0.7421Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5708 - acc: 0.7429 - val\_loss: 0.8433 - val\_acc: 0.5000  
Epoch 38/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5709 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 108s 660ms/step - loss: 0.5715 - acc: 0.7429 - val\_loss: 0.8200 - val\_acc: 0.5000  
Epoch 39/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5716 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 108s 660ms/step - loss: 0.5718 - acc: 0.7429 - val\_loss: 0.8608 - val\_acc: 0.5000  
Epoch 40/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5722 - acc: 0.7421Epoch 1/100  
163/163 [==============================] - 108s 662ms/step - loss: 0.5713 - acc: 0.7429 - val\_loss: 0.8423 - val\_acc: 0.5000  
Epoch 41/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5710 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 108s 664ms/step - loss: 0.5709 - acc: 0.7429 - val\_loss: 0.8221 - val\_acc: 0.5000  
Epoch 42/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5713 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 108s 664ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8108 - val\_acc: 0.5000  
Epoch 43/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5717 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 107s 659ms/step - loss: 0.5721 - acc: 0.7429 - val\_loss: 0.8318 - val\_acc: 0.5000  
Epoch 44/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5704 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5708 - acc: 0.7429 - val\_loss: 0.8337 - val\_acc: 0.5000  
Epoch 45/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5714 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 108s 662ms/step - loss: 0.5718 - acc: 0.7429 - val\_loss: 0.8018 - val\_acc: 0.5000  
Epoch 46/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5711 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8023 - val\_acc: 0.5000  
Epoch 47/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5719 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 108s 664ms/step - loss: 0.5715 - acc: 0.7429 - val\_loss: 0.8279 - val\_acc: 0.5000  
Epoch 48/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5697 - acc: 0.7436Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5704 - acc: 0.7429 - val\_loss: 0.8184 - val\_acc: 0.5000  
Epoch 49/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5715 - acc: 0.7429 - val\_loss: 0.8399 - val\_acc: 0.5000  
Epoch 50/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5719 - acc: 0.7429 - val\_loss: 0.8559 - val\_acc: 0.5000  
Epoch 51/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 109s 670ms/step - loss: 0.5716 - acc: 0.7429 - val\_loss: 0.8300 - val\_acc: 0.5000  
Epoch 52/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 110s 672ms/step - loss: 0.5710 - acc: 0.7429 - val\_loss: 0.8499 - val\_acc: 0.5000  
Epoch 53/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5717 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5711 - acc: 0.7429 - val\_loss: 0.8358 - val\_acc: 0.5000  
Epoch 54/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5720 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5720 - acc: 0.7429 - val\_loss: 0.8200 - val\_acc: 0.5000  
Epoch 55/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5717 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 109s 669ms/step - loss: 0.5713 - acc: 0.7429 - val\_loss: 0.8275 - val\_acc: 0.5000  
Epoch 56/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5705 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5708 - acc: 0.7429 - val\_loss: 0.8209 - val\_acc: 0.5000  
Epoch 57/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5709 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5708 - acc: 0.7429 - val\_loss: 0.8285 - val\_acc: 0.5000  
Epoch 58/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5714 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5716 - acc: 0.7429 - val\_loss: 0.8287 - val\_acc: 0.5000  
Epoch 59/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5715 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 108s 665ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8302 - val\_acc: 0.5000  
Epoch 60/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5728 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5732 - acc: 0.7429 - val\_loss: 0.8224 - val\_acc: 0.5000  
Epoch 61/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5699 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5705 - acc: 0.7429 - val\_loss: 0.8015 - val\_acc: 0.5000  
Epoch 62/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5724 - acc: 0.7429 - val\_loss: 0.8297 - val\_acc: 0.5000  
Epoch 63/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5717 - acc: 0.7429 - val\_loss: 0.8152 - val\_acc: 0.5000  
Epoch 64/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5708 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5713 - acc: 0.7429 - val\_loss: 0.7977 - val\_acc: 0.5000  
Epoch 65/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 109s 669ms/step - loss: 0.5719 - acc: 0.7429 - val\_loss: 0.8175 - val\_acc: 0.5000  
Epoch 66/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5715 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8333 - val\_acc: 0.5000  
Epoch 67/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5702 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 109s 669ms/step - loss: 0.5707 - acc: 0.7429 - val\_loss: 0.8725 - val\_acc: 0.5000  
Epoch 68/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5720 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5726 - acc: 0.7429 - val\_loss: 0.8148 - val\_acc: 0.5000  
Epoch 69/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5709 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5709 - acc: 0.7429 - val\_loss: 0.8371 - val\_acc: 0.5000  
Epoch 70/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5714 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 109s 670ms/step - loss: 0.5716 - acc: 0.7429 - val\_loss: 0.8350 - val\_acc: 0.5000  
Epoch 71/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5716 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 108s 663ms/step - loss: 0.5717 - acc: 0.7429 - val\_loss: 0.8372 - val\_acc: 0.5000  
Epoch 72/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5722 - acc: 0.7423Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5715 - acc: 0.7429 - val\_loss: 0.8337 - val\_acc: 0.5000  
Epoch 73/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5715 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 666ms/step - loss: 0.5715 - acc: 0.7429 - val\_loss: 0.8322 - val\_acc: 0.5000  
Epoch 74/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5713 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8234 - val\_acc: 0.5000  
Epoch 75/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5717 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 110s 675ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8675 - val\_acc: 0.5000  
Epoch 76/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5720 - acc: 0.7421Epoch 1/100  
163/163 [==============================] - 110s 672ms/step - loss: 0.5712 - acc: 0.7429 - val\_loss: 0.8309 - val\_acc: 0.5000  
Epoch 77/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5722 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 110s 672ms/step - loss: 0.5718 - acc: 0.7429 - val\_loss: 0.8158 - val\_acc: 0.5000  
Epoch 78/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5692 - acc: 0.7438Epoch 1/100  
163/163 [==============================] - 110s 675ms/step - loss: 0.5703 - acc: 0.7429 - val\_loss: 0.8483 - val\_acc: 0.5000  
Epoch 79/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5708 - acc: 0.7429 - val\_loss: 0.8331 - val\_acc: 0.5000  
Epoch 80/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5709 - acc: 0.7436Epoch 1/100  
163/163 [==============================] - 110s 674ms/step - loss: 0.5717 - acc: 0.7429 - val\_loss: 0.8011 - val\_acc: 0.5000  
Epoch 81/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5698 - acc: 0.7431Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5700 - acc: 0.7429 - val\_loss: 0.8418 - val\_acc: 0.5000  
Epoch 82/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 110s 673ms/step - loss: 0.5715 - acc: 0.7429 - val\_loss: 0.8273 - val\_acc: 0.5000  
Epoch 83/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5720 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 110s 675ms/step - loss: 0.5717 - acc: 0.7429 - val\_loss: 0.8474 - val\_acc: 0.5000  
Epoch 84/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5721 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 110s 676ms/step - loss: 0.5721 - acc: 0.7429 - val\_loss: 0.8206 - val\_acc: 0.5000  
Epoch 85/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5707 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5707 - acc: 0.7429 - val\_loss: 0.8271 - val\_acc: 0.5000  
Epoch 86/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7423Epoch 1/100  
163/163 [==============================] - 109s 670ms/step - loss: 0.5707 - acc: 0.7429 - val\_loss: 0.8174 - val\_acc: 0.5000  
Epoch 87/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 110s 673ms/step - loss: 0.5715 - acc: 0.7429 - val\_loss: 0.8316 - val\_acc: 0.5000  
Epoch 88/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5714 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5714 - acc: 0.7429 - val\_loss: 0.8346 - val\_acc: 0.5000  
Epoch 89/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5707 - acc: 0.7429 - val\_loss: 0.8261 - val\_acc: 0.5000  
Epoch 90/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5725 - acc: 0.7421Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5716 - acc: 0.7429 - val\_loss: 0.8343 - val\_acc: 0.5000  
Epoch 91/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5706 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 109s 667ms/step - loss: 0.5709 - acc: 0.7429 - val\_loss: 0.8404 - val\_acc: 0.5000  
Epoch 92/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5703 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 110s 672ms/step - loss: 0.5708 - acc: 0.7429 - val\_loss: 0.8090 - val\_acc: 0.5000  
Epoch 93/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 109s 671ms/step - loss: 0.5710 - acc: 0.7429 - val\_loss: 0.8322 - val\_acc: 0.5000  
Epoch 94/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5708 - acc: 0.7434Epoch 1/100  
163/163 [==============================] - 110s 672ms/step - loss: 0.5714 - acc: 0.7429 - val\_loss: 0.8245 - val\_acc: 0.5000  
Epoch 95/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 110s 674ms/step - loss: 0.5717 - acc: 0.7429 - val\_loss: 0.8100 - val\_acc: 0.5000  
Epoch 96/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5713 - acc: 0.7429Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5713 - acc: 0.7429 - val\_loss: 0.8397 - val\_acc: 0.5000  
Epoch 97/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5725 - acc: 0.7425Epoch 1/100  
163/163 [==============================] - 109s 668ms/step - loss: 0.5720 - acc: 0.7429 - val\_loss: 0.8325 - val\_acc: 0.5000  
Epoch 98/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 110s 673ms/step - loss: 0.5721 - acc: 0.7429 - val\_loss: 0.8177 - val\_acc: 0.5000  
Epoch 99/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5718 - acc: 0.7432Epoch 1/100  
163/163 [==============================] - 109s 672ms/step - loss: 0.5722 - acc: 0.7429 - val\_loss: 0.8270 - val\_acc: 0.5000  
Epoch 100/100  
162/163 [============================>.] - ETA: 0s - loss: 0.5712 - acc: 0.7427Epoch 1/100  
163/163 [==============================] - 110s 673ms/step - loss: 0.5710 - acc: 0.7429 - val\_loss: 0.8370 - val\_acc: 0.5000  
  
  
  
---------------------------------------------------------------------------

modified\_loss = history.history['loss'[0:100]]  
modified\_loss = modified\_loss[1:100] #remove the first value recorded in loss since it's an outlier  
  
modified\_val\_loss = history.history['val\_loss']  
modified\_val\_loss = modified\_val\_loss[1:100]

### Plot training  
import matplotlib.pyplot as plt  
def plot\_learning\_curves(history):  
 plt.figure(figsize=(12,4))  
   
 plt.subplot(1,2,1)  
 # plt.plot(history.history['loss'])  
 # plt.plot(history.history['val\_loss'])  
 plt.plot(modified\_loss) # These changes are made because of the first record of loss is an outlier.  
 plt.plot(modified\_val\_loss)  
 plt.title('model loss')  
 plt.ylabel('loss')  
 plt.xlabel('epoch')  
 plt.legend(['train', 'val'], loc='upper left')  
   
 plt.subplot(1,2,2)  
 plt.plot(history.history['acc'])  
 plt.plot(history.history['val\_acc'])  
 plt.title('model accuracy')  
 plt.ylabel('accuracy')  
 plt.xlabel('epoch')  
 plt.legend(['train', 'val'], loc='upper left')  
   
 plt.tight\_layout()  
   
plot\_learning\_curves(history)



png

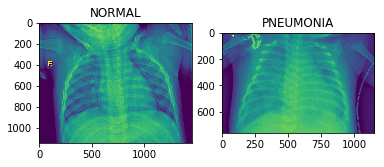
## Evaluate the model  
test\_loss, test\_acc = model.evaluate\_generator(generator=test\_generator, verbose=1)  
  
## Analytics  
from sklearn.metrics import accuracy\_score, confusion\_matrix  
  
test\_generator.reset()  
test\_preds = model.predict\_generator(test\_generator, verbose=1)  
test\_preds = np.argmax(test\_preds,axis=1)  
  
acc = accuracy\_score(test\_generator.classes, test\_preds)\*100  
cm = confusion\_matrix(test\_generator.classes, test\_preds)  
tn, fp, fn, tp = cm.ravel()  
  
print('CONFUSION MATRIX ------------------')  
print(cm)  
# plot\_confusion\_matrix(cm, target\_names=['NORMAL', 'PNEUMONIA'], normalize=False)  
  
  
print('\nTEST METRICS ----------------------')  
precision = tp/(tp+fp)\*100  
recall = tp/(tp+fn)\*100  
print('Accuracy: {}%'.format(acc))  
print('Precision: {}%'.format(precision))  
print('Recall: {}%'.format(recall))  
print('F1-score: {}'.format(2\*precision\*recall/(precision+recall)))  
  
print('\nTRAIN METRIC ----------------------')  
print('Train acc: {}%'.format(np.round((history.history['acc'][-1])\*100, 14)))

20/20 [==============================] - 7s 341ms/step - loss: 0.6980 - acc: 0.6250  
20/20 [==============================] - 7s 366ms/step  
CONFUSION MATRIX ------------------  
[[ 0 234]  
 [ 0 390]]  
  
TEST METRICS ----------------------  
Accuracy: 62.5%  
Precision: 62.5%  
Recall: 100.0%  
F1-score: 76.92307692307692  
  
TRAIN METRIC ----------------------  
Train acc: 74.29064512252808%

# Show images

import matplotlib.pyplot as plt  
  
plt.subplot(1,2,1).set\_title('NORMAL')  
plt.imshow(plt.imread('/content/data/chest\_xray/train/NORMAL/IM-0131-0001.jpeg'))  
  
plt.subplot(1,2,2).set\_title('PNEUMONIA')  
plt.imshow(plt.imread('/content/data/chest\_xray/train/PNEUMONIA/person1000\_bacteria\_2931.jpeg'))

<matplotlib.image.AxesImage at 0x7f6eb0ab8908>



png

### Graph

import matplotlib.pyplot as plt  
acc = history.history['accuracy']  
val\_acc = history.history['val\_accuracy']  
loss = history.history['loss']  
val\_loss = history.history['val\_loss']  
  
epochs = range(len(acc))  
  
plt.plot(epochs, acc, 'r', label='Training accuracy')  
plt.plot(epochs, val\_acc, 'b', label='Validation accuracy')  
plt.title('Training and validation accuracy')  
plt.legend(loc=0)  
plt.figure()  
  
  
plt.show()



png

<Figure size 432x288 with 0 Axes>