

# day1-042-ipy

June 25, 2024

```
[ ]: import tensorflow as tf
from tensorflow import keras
from tensorflow.keras import layers
from tensorflow.keras.preprocessing.image import ImageDataGenerator
IMG_SIZE = 244
BATCH_SIZE = 32
```

```
[ ]: train_datagen = ImageDataGenerator(rescale=1./255,validation_split=0.2)
train_generator = train_datagen.flow_from_directory(
    '/content/drive/MyDrive/Lung X-Ray Image',
    target_size=(IMG_SIZE,IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical',
    subset='training'
)

val_generator = train_datagen.flow_from_directory(
    '/content/drive/MyDrive/Lung X-Ray Image',
    target_size=(IMG_SIZE,IMG_SIZE),
    batch_size=BATCH_SIZE,
    class_mode='categorical',
    subset='validation'
)
```

Found 2788 images belonging to 1 classes.

Found 697 images belonging to 1 classes.

```
[ ]: # Define the model
model = keras.Sequential([
    layers.Conv2D(32,␣
    ↪(3,3),activation='relu',input_shape=(IMG_SIZE,IMG_SIZE,3)),
    layers.MaxPooling2D(2,2),
    layers.Conv2D(64,(3,3),activation='relu'),
    layers.MaxPooling2D(2,2),
    layers.Conv2D(128,(3,3),activation='relu'),
    layers.MaxPooling2D(2,2),
    layers.Flatten(),
    layers.Dense(128,activation='relu'),
```

```
layers.Dense(1,activation='sigmoid') #output layer
])
```

```
[ ]: 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).

```
[ ]: model.compile(optimizer='adam', loss='binary_crossentropy',
metrics=['accuracy'])
```

```
[ ]: model.fit(train_generator,validation_data=val_generator,epochs=5)
```

Epoch 1/5

69/69 [=====] - 1547s 22s/step - loss: 0.0105 - accuracy: 0.0087 - val\_loss: 0.0000e+00 - val\_accuracy: 0.0000e+00

Epoch 2/5

69/69 [=====] - 340s 5s/step - loss: 0.0000e+00 - accuracy: 0.0000e+00 - val\_loss: 0.0000e+00 - val\_accuracy: 0.0000e+00

Epoch 3/5

69/69 [=====] - 392s 6s/step - loss: 0.0000e+00 - accuracy: 0.0000e+00 - val\_loss: 0.0000e+00 - val\_accuracy: 0.0000e+00

Epoch 4/5

69/69 [=====] - 362s 5s/step - loss: 0.0000e+00 - accuracy: 0.0000e+00 - val\_loss: 0.0000e+00 - val\_accuracy: 0.0000e+00

Epoch 5/5

69/69 [=====] - 348s 5s/step - loss: 0.0000e+00 - accuracy: 0.0000e+00 - val\_loss: 0.0000e+00 - val\_accuracy: 0.0000e+00

```
[ ]: <keras.src.callbacks.History at 0x7e222d462050>
```

```
[ ]: model.save("Model.h5","label.text")
```

/usr/local/lib/python3.10/dist-packages/keras/src/engine/training.py:3103: UserWarning: You are saving your model as an HDF5 file via `model.save()`. This file format is considered legacy. We recommend using instead the native Keras format, e.g. `model.save('my\_model.keras')`.

```
saving_api.save_model(
```

WARNING:tensorflow:Compiled the loaded model, but the compiled metrics have yet to be built. `model.compile\_metrics` will be empty until you train or evaluate the model.

```
[ ]: from tensorflow.keras.models import load_model
from tensorflow.keras.preprocessing import image
import numpy as np
```

```

model = load_model('/content/drive/MyDrive/Model.h5')
test_image_path = '/content/drive/MyDrive/Lung X-Ray Image/Lung X-Ray Image/
↳Normal/1001.jpg'
img = image.load_img(test_image_path, target_size=(244,244))
img_array = image.img_to_array(img)
img_array = np.expand_dims(img_array,axis=0)

```

WARNING:tensorflow:No training configuration found in the save file, so the model was \*not\* compiled. Compile it manually.

```

[ ]: img_array /= 255.
      prediction = model.predict(img_array)
      print(prediction)

```

```

1/1 [=====] - 0s 119ms/step
[[0.5041871]]

```

```

[ ]: if prediction < 0.33:
      print("prediction : lung_opacity (probability:" , prediction[0][0])
    elif prediction < 0.66:
      print("prediction : normal (probability:" , prediction[0][0])
    else:
      print("prediction : viral peenomonina (probability:", prediction[0][0])

```

```

prediction : normal (probability: 0.5041871

```

```

[ ]:

```