data_aug_adam_val___recall_86_06

November 7, 2021

1 Covid Classifier Model

1.0.1 Goals

Classify: - Normal CXR - Viral Pneumonia CXR - COVID CXR

1.1 Create Directories for Dataset

Separate the data to use later as generators.

[ ]:
```
# Aumentar threshold de Early Stop.
# Aumentar las rotaciones y escalas.
# Jugar con las metricas AUC y recall.
import os

BASE_PATH = '/home/hivini/learn/research/new-covid'
ORIGINAL_DATASET_DIR = os.path.join(BASE_PATH, 'COVID-19_Radiography_Dataset')
ORIGINAL_VIRAL_DIR = os.path.join(ORIGINAL_DATASET_DIR, 'Viral Pneumonia')
ORIGINAL_COVID_DIR = os.path.join(ORIGINAL_DATASET_DIR, 'COVID')
ORIGINAL_NORMAL_DIR = os.path.join(ORIGINAL_DATASET_DIR, 'Normal')
DATASET_DIR = os.path.join(BASE_PATH, 'small_dataset')
TRAIN_DIR = os.path.join(DATASET_DIR, 'train')
VALIDATION_DIR = os.path.join(DATASET_DIR, 'validation')
TEST_DIR = os.path.join(DATASET_DIR, 'test')
TRAIN_VIRAL_DIR = os.path.join(TRAIN_DIR, 'viral_pneumonia')
TRAIN_COVID_DIR = os.path.join(TRAIN_DIR, 'covid')
TRAIN_NORMAL_DIR = os.path.join(TRAIN_DIR, 'normal')
VALIDATION_VIRAL_DIR = os.path.join(VALIDATION_DIR, 'viral_pneumonia')
VALIDATION_COVID_DIR = os.path.join(VALIDATION_DIR, 'covid')
VALIDATION_NORMAL_DIR = os.path.join(VALIDATION_DIR, 'normal')
TEST_VIRAL_DIR = os.path.join(TEST_DIR, 'viral_pneumonia')
TEST_COVID_DIR = os.path.join(TEST_DIR, 'covid')
TEST_NORMAL_DIR = os.path.join(TEST_DIR, 'normal')


def createDir(path: str) -> None:
    if not os.path.exists(path):
        os.mkdir(path)
```

```
createDir(DATASET_DIR)
createDir(TRAIN_DIR)
createDir(VALIDATION_DIR)
createDir(TEST_DIR)
createDir(TRAIN_VIRAL_DIR)
createDir(TRAIN_COVID_DIR)
createDir(TRAIN_NORMAL_DIR)
createDir(VALIDATION_VIRAL_DIR)
createDir(VALIDATION_COVID_DIR)
createDir(VALIDATION_NORMAL_DIR)
createDir(TEST_VIRAL_DIR)
createDir(TEST_COVID_DIR)
createDir(TEST_NORMAL_DIR)
```

[ ]:
```
import numpy as np
import shutil


def generate_sets(source: str):
    allFiles = os.listdir(source)
    np.random.shuffle(allFiles)
    return np.split(np.array(allFiles), [int(len(allFiles)*0.7),␣
 ↪int(len(allFiles)*0.85)])


def saveAndSeparateFiles(src_dir: str, train_dir: str, val_dir: str, test_dir):
    train_fnames, val_fnames, test_fnames = generate_sets(src_dir)
    for fname in train_fnames:
        src = os.path.join(src_dir, fname)
        dst = os.path.join(train_dir, fname)
        shutil.copyfile(src, dst)

    for fname in val_fnames:
        src = os.path.join(src_dir, fname)
        dst = os.path.join(val_dir, fname)
        shutil.copyfile(src, dst)

    for fname in test_fnames:
        src = os.path.join(src_dir, fname)
        dst = os.path.join(test_dir, fname)
        shutil.copyfile(src, dst)

create = False
if create:
    saveAndSeparateFiles(ORIGINAL_NORMAL_DIR, TRAIN_NORMAL_DIR,
                         VALIDATION_NORMAL_DIR, TEST_NORMAL_DIR)
```

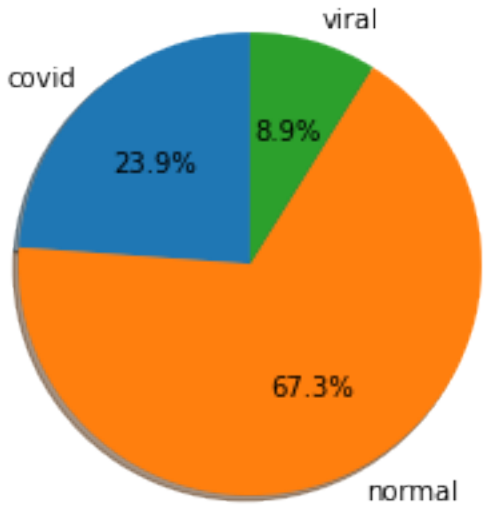
```
    saveAndSeparateFiles(ORIGINAL_COVID_DIR, TRAIN_COVID_DIR,
                         VALIDATION_COVID_DIR, TEST_COVID_DIR)
    saveAndSeparateFiles(ORIGINAL_VIRAL_DIR, TRAIN_VIRAL_DIR,
                         VALIDATION_VIRAL_DIR, TEST_VIRAL_DIR)
```

1.2 Counting our images

[ ]:
```
import tensorflow as tf
import matplotlib.pyplot as plt
normal_train = tf.io.gfile.glob(TRAIN_NORMAL_DIR + '/*')
viral_train = tf.io.gfile.glob(TRAIN_VIRAL_DIR + '/*')
covid_train = tf.io.gfile.glob(TRAIN_COVID_DIR + '/*')

# Plotting Distribution of Each Classes
image_count = {'covid': len(covid_train), 'normal': len(
    normal_train), 'viral': len(viral_train)}
print(image_count)
fig1, ax1 = plt.subplots()
ax1.pie(image_count.values(),
        labels=image_count.keys(),
        shadow=True,
        autopct='%1.1f%%',
        startangle=90)
plt.show()
```

```
2021-11-07 00:47:45.832720: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcudart.so.10.1
```

```
{'covid': 2531, 'normal': 7134, 'viral': 941}
```



1.3 Create our Covnet Model

In this case we are doing a multi class classification, our total clases are 3: - Viral CXR - Covid CXR - Normal CXR

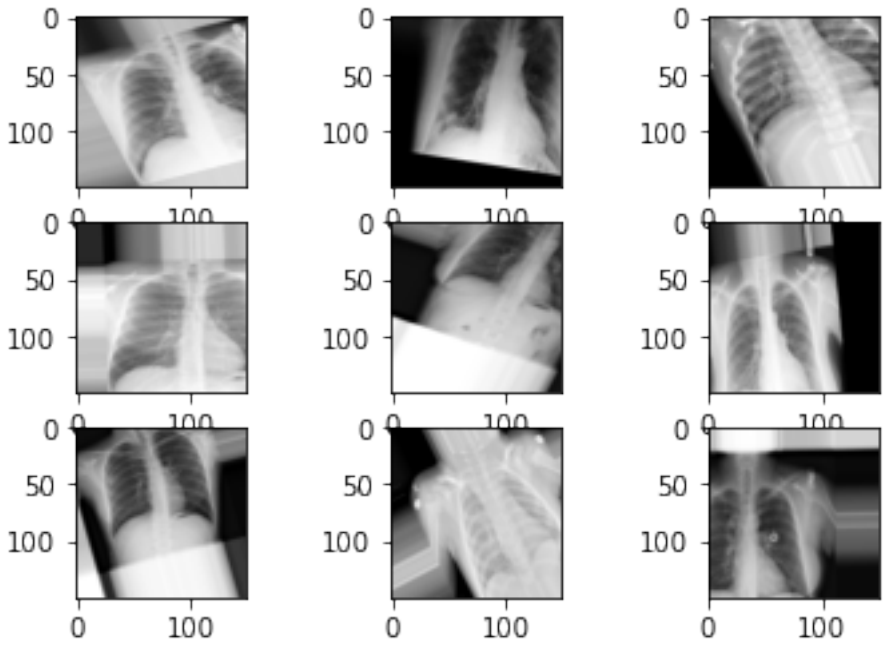
Our neural network will output neurons as 3 classes that will calculate the probability of being one using the softmax function.

[ ]:
```
from keras.preprocessing.image import ImageDataGenerator

train_datagen = ImageDataGenerator(
    rescale=1./255,
    featurewise_center=False,  # set input mean to 0 over the dataset
    samplewise_center=False,  # set each sample mean to 0
    featurewise_std_normalization=False,  # divide inputs by std of the dataset
    samplewise_std_normalization=False,  # divide each input by its std
    zca_whitening=False,  # apply ZCA whitening
    # randomly rotate images in the range (degrees, 0 to 180)
    rotation_range=30,
    zoom_range=0.25,  # Randomly zoom image
    # randomly shift images horizontally (fraction of total width)
    width_shift_range=0.25,
    # randomly shift images vertically (fraction of total height)
    height_shift_range=0.25,
    horizontal_flip=False,  # randomly flip images
    vertical_flip=False  # randomly flip images
)

# train_datagen = ImageDataGenerator(rescale=1./255)
test_datagen = ImageDataGenerator(rescale=1./255)
evaluate_datagen = ImageDataGenerator(rescale=1./255)

train_generator = train_datagen.flow_from_directory(
    TRAIN_DIR,
    target_size=(150, 150),
    batch_size=32,
    class_mode='categorical',
    color_mode='grayscale'
)

print(train_generator.class_indices)

validation_generator = test_datagen.flow_from_directory(
    VALIDATION_DIR,
    target_size=(150, 150),
```

```
    batch_size=32,
    class_mode='categorical',
    color_mode='grayscale'
)

print(validation_generator.class_indices)

test_generator = evaluate_datagen.flow_from_directory(
    TEST_DIR,
    target_size=(150, 150),
    batch_size=32,
    class_mode='categorical',
    color_mode='grayscale'
)

print(test_generator.class_indices)
```

```
Found 10606 images belonging to 3 classes.
{'covid': 0, 'normal': 1, 'viral_pneumonia': 2}
Found 2273 images belonging to 3 classes.
{'covid': 0, 'normal': 1, 'viral_pneumonia': 2}
Found 2274 images belonging to 3 classes.
{'covid': 0, 'normal': 1, 'viral_pneumonia': 2}
```

[ ]:
```
for X_batch, y_batch in train_generator:
        # create a grid of 3x3 images
        for i in range(0, 9):
                plt.subplot(330 + 1 + i)
                plt.imshow(X_batch[i].reshape(150, 150), cmap=plt.
 ↪get_cmap('gray'))
        # show the plot
        plt.show()
        break
```



[ ]:
```
from keras.layers import Conv2D, BatchNormalization, MaxPooling2D, Dropout,␣
↪Flatten, Dense
from keras.models import Sequential
from keras import backend

# We want to make sure we start from the start when training our model␣
↪everytime we run it.
backend.clear_session()

model = Sequential()
model.add(Conv2D(64, (3, 3), activation='relu', input_shape=(150, 150, 1)))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(BatchNormalization())
model.add(MaxPooling2D((2, 2)))
model.add(Flatten())
model.add(Dropout(0.5))
model.add(Dense(512, activation='relu'))
```

```
model.add(Dense(64, activation='relu'))
model.add(Dense(3, activation='softmax'))
model.summary()
```

```
2021-11-07 00:47:50.753426: I tensorflow/compiler/jit/xla_cpu_device.cc:41] Not
creating XLA devices, tf_xla_enable_xla_devices not set
2021-11-07 00:47:50.764836: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcuda.so.1
2021-11-07 00:47:51.160238: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
Your kernel may have been built without NUMA support.
2021-11-07 00:47:51.160304: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1720] Found device 0 with
properties:
pciBusID: 0000:01:00.0 name: NVIDIA GeForce RTX 2080 with Max-Q Design
computeCapability: 7.5
coreClock: 1.215GHz coreCount: 46 deviceMemorySize: 8.00GiB
deviceMemoryBandwidth: 357.69GiB/s
2021-11-07 00:47:51.160329: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcudart.so.10.1
2021-11-07 00:47:51.216397: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcublas.so.10
2021-11-07 00:47:51.216494: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcublasLt.so.10
2021-11-07 00:47:51.250783: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcufft.so.10
2021-11-07 00:47:51.262189: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcurand.so.10
2021-11-07 00:47:51.328532: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcusolver.so.10
2021-11-07 00:47:51.361511: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcusparse.so.10
2021-11-07 00:47:51.453246: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcudnn.so.7
2021-11-07 00:47:51.454414: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
```

```
Your kernel may have been built without NUMA support.
2021-11-07 00:47:51.455217: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
Your kernel may have been built without NUMA support.
2021-11-07 00:47:51.455237: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1862] Adding visible gpu
devices: 0
2021-11-07 00:47:51.456551: I tensorflow/core/platform/cpu_feature_guard.cc:142]
This TensorFlow binary is optimized with oneAPI Deep Neural Network Library
(oneDNN) to use the following CPU instructions in performance-critical
operations:  SSE4.1 SSE4.2 AVX AVX2 FMA
To enable them in other operations, rebuild TensorFlow with the appropriate
compiler flags.
2021-11-07 00:47:51.458654: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
Your kernel may have been built without NUMA support.
2021-11-07 00:47:51.458684: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1720] Found device 0 with
properties:
pciBusID: 0000:01:00.0 name: NVIDIA GeForce RTX 2080 with Max-Q Design
computeCapability: 7.5
coreClock: 1.215GHz coreCount: 46 deviceMemorySize: 8.00GiB
deviceMemoryBandwidth: 357.69GiB/s
2021-11-07 00:47:51.458717: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcudart.so.10.1
2021-11-07 00:47:51.458751: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcublas.so.10
2021-11-07 00:47:51.458764: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcublasLt.so.10
2021-11-07 00:47:51.458776: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcufft.so.10
2021-11-07 00:47:51.458787: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcurand.so.10
2021-11-07 00:47:51.458799: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcusolver.so.10
2021-11-07 00:47:51.458811: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcusparse.so.10
2021-11-07 00:47:51.458824: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
```

```
opened dynamic library libcudnn.so.7
2021-11-07 00:47:51.459502: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
Your kernel may have been built without NUMA support.
2021-11-07 00:47:51.460207: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
Your kernel may have been built without NUMA support.
2021-11-07 00:47:51.460220: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1862] Adding visible gpu
devices: 0
2021-11-07 00:47:51.460724: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcudart.so.10.1
2021-11-07 00:47:53.455547: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1261] Device interconnect
StreamExecutor with strength 1 edge matrix:
2021-11-07 00:47:53.455575: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1267]      0
2021-11-07 00:47:53.455581: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1280] 0:   N
2021-11-07 00:47:53.491165: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
Your kernel may have been built without NUMA support.
2021-11-07 00:47:53.491211: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1489] Could not identify NUMA
node of platform GPU id 0, defaulting to 0.  Your kernel may not have been built
with NUMA support.
2021-11-07 00:47:53.492155: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
Your kernel may have been built without NUMA support.
2021-11-07 00:47:53.492858: E
tensorflow/stream_executor/cuda/cuda_gpu_executor.cc:927] could not open file to
read NUMA node: /sys/bus/pci/devices/0000:01:00.0/numa_node
Your kernel may have been built without NUMA support.
2021-11-07 00:47:53.492898: I
tensorflow/core/common_runtime/gpu/gpu_device.cc:1406] Created TensorFlow device
(/job:localhost/replica:0/task:0/device:GPU:0 with 6575 MB memory) -> physical
GPU (device: 0, name: NVIDIA GeForce RTX 2080 with Max-Q Design, pci bus id:
0000:01:00.0, compute capability: 7.5)
2021-11-07 00:47:53.496484: I tensorflow/compiler/jit/xla_gpu_device.cc:99] Not
creating XLA devices, tf_xla_enable_xla_devices not set

Model: "sequential"

_________________________________________________________________
```

```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d (Conv2D)              (None, 148, 148, 64)      640
_________________________________________________________________
batch_normalization (BatchNo (None, 148, 148, 64)      256
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 74, 74, 64)        0
_________________________________________________________________
conv2d_1 (Conv2D)            (None, 72, 72, 64)        36928
_________________________________________________________________
batch_normalization_1 (Batch (None, 72, 72, 64)        256
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 36, 36, 64)        0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 34, 34, 128)       73856
_________________________________________________________________
batch_normalization_2 (Batch (None, 34, 34, 128)       512
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 17, 17, 128)       0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 15, 15, 128)       147584
_________________________________________________________________
batch_normalization_3 (Batch (None, 15, 15, 128)       512
_________________________________________________________________
max_pooling2d_3 (MaxPooling2 (None, 7, 7, 128)         0
_________________________________________________________________
flatten (Flatten)            (None, 6272)              0
_________________________________________________________________
dropout (Dropout)            (None, 6272)              0
_________________________________________________________________
dense (Dense)                (None, 512)               3211776
_________________________________________________________________
dense_1 (Dense)              (None, 64)                32832
_________________________________________________________________
dense_2 (Dense)              (None, 3)                 195
=================================================================
Total params: 3,505,347
Trainable params: 3,504,579
Non-trainable params: 768
_________________________________________________________________
```

[ ]:
```python
from keras import optimizers

# opt = RMSprop(lr=0.0001, decay=1e-6)
lr_schedule = optimizers.schedules.ExponentialDecay(
    initial_learning_rate=1e-5,
    decay_steps=2000,
```

```
    decay_rate=0.9)
opt = optimizers.Adam(learning_rate=lr_schedule)

# try with metric categorical_crossentropy
model.compile(loss='categorical_crossentropy', optimizer=opt,␣
 ↪metrics=['accuracy', tf.keras.metrics.Recall()])
```

[ ]:
```
import numpy as np
from sklearn.utils import class_weight
from keras.callbacks import EarlyStopping
from keras.callbacks import ModelCheckpoint

classes = train_generator.classes
class_weights = class_weight.compute_class_weight(None,
                                                  np.unique(classes),
                                                  classes)
best_model_path = os.path.join(BASE_PATH, 'best_model.h5')
es = EarlyStopping(monitor='val_loss', mode='min', verbose=1, patience=50)
mc = ModelCheckpoint(best_model_path, monitor='val_accuracy', mode='max',␣
 ↪verbose=1, save_best_only=True)
history = model.fit(
    train_generator,
    steps_per_epoch=train_generator.n // 32,
    epochs=400,
    validation_data=validation_generator,
    class_weight=dict(zip(np.unique(classes), class_weights)),
    callbacks=[es, mc]
)
```

```
/home/hivini/anaconda3/envs/tf-gpu/lib/python3.9/site-
packages/sklearn/utils/validation.py:67: FutureWarning: Pass classes=[0 1 2],
y=[0 0 0 … 2 2 2] as keyword args. From version 0.25 passing these as
positional arguments will result in an error
  warnings.warn("Pass {} as keyword args. From version 0.25 "
```

```
Epoch 1/400
```

```
2021-11-07 00:49:36.765613: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcublas.so.10
2021-11-07 00:49:37.382782: I
tensorflow/stream_executor/platform/default/dso_loader.cc:49] Successfully
opened dynamic library libcudnn.so.7
2021-11-07 00:49:39.694844: W tensorflow/stream_executor/gpu/asm_compiler.cc:63]
Running ptxas --version returned 256
2021-11-07 00:49:39.796993: W
tensorflow/stream_executor/gpu/redzone_allocator.cc:314] Internal: ptxas exited
with non-zero error code 256, output:
```

```
Relying on driver to perform ptx compilation.
Modify $PATH to customize ptxas location.
This message will be only logged once.
331/331 [==============================] - 86s 204ms/step - loss: 1.0113 -
accuracy: 0.6222 - recall_1: 0.5893 - val_loss: 1.3243 - val_accuracy: 0.2385 -
val_recall_1: 0.2385

Epoch 00001: val_accuracy improved from -inf to 0.23845, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 2/400
331/331 [==============================] - 39s 118ms/step - loss: 0.7581 -
accuracy: 0.6706 - recall_1: 0.6364 - val_loss: 0.5890 - val_accuracy: 0.7589 -
val_recall_1: 0.6925

Epoch 00002: val_accuracy improved from 0.23845 to 0.75891, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 3/400
331/331 [==============================] - 39s 118ms/step - loss: 0.6746 -
accuracy: 0.7022 - recall_1: 0.6738 - val_loss: 0.5262 - val_accuracy: 0.7800 -
val_recall_1: 0.7448

Epoch 00003: val_accuracy improved from 0.75891 to 0.78003, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 4/400
331/331 [==============================] - 40s 119ms/step - loss: 0.6485 -
accuracy: 0.7138 - recall_1: 0.6843 - val_loss: 0.5095 - val_accuracy: 0.7765 -
val_recall_1: 0.7484

Epoch 00004: val_accuracy did not improve from 0.78003
Epoch 5/400
331/331 [==============================] - 39s 119ms/step - loss: 0.6194 -
accuracy: 0.7354 - recall_1: 0.7101 - val_loss: 0.4926 - val_accuracy: 0.7862 -
val_recall_1: 0.7677

Epoch 00005: val_accuracy improved from 0.78003 to 0.78619, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 6/400
331/331 [==============================] - 39s 119ms/step - loss: 0.6063 -
accuracy: 0.7260 - recall_1: 0.7028 - val_loss: 0.4693 - val_accuracy: 0.8007 -
val_recall_1: 0.7818

Epoch 00006: val_accuracy improved from 0.78619 to 0.80070, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 7/400
331/331 [==============================] - 39s 118ms/step - loss: 0.5726 -
accuracy: 0.7341 - recall_1: 0.7164 - val_loss: 0.4554 - val_accuracy: 0.8126 -
val_recall_1: 0.7972
```

```
Epoch 00007: val_accuracy improved from 0.80070 to 0.81258, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 8/400
331/331 [==============================] - 39s 118ms/step - loss: 0.5680 -
accuracy: 0.7412 - recall_1: 0.7216 - val_loss: 0.4471 - val_accuracy: 0.8187 -
val_recall_1: 0.7972

Epoch 00008: val_accuracy improved from 0.81258 to 0.81874, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 9/400
331/331 [==============================] - 39s 119ms/step - loss: 0.5612 -
accuracy: 0.7433 - recall_1: 0.7252 - val_loss: 0.4312 - val_accuracy: 0.8262 -
val_recall_1: 0.8117

Epoch 00009: val_accuracy improved from 0.81874 to 0.82622, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 10/400
331/331 [==============================] - 39s 118ms/step - loss: 0.5404 -
accuracy: 0.7629 - recall_1: 0.7440 - val_loss: 0.4156 - val_accuracy: 0.8306 -
val_recall_1: 0.8165

Epoch 00010: val_accuracy improved from 0.82622 to 0.83062, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 11/400
331/331 [==============================] - 39s 119ms/step - loss: 0.5175 -
accuracy: 0.7667 - recall_1: 0.7507 - val_loss: 0.4035 - val_accuracy: 0.8377 -
val_recall_1: 0.8249

Epoch 00011: val_accuracy improved from 0.83062 to 0.83766, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 12/400
331/331 [==============================] - 40s 121ms/step - loss: 0.5050 -
accuracy: 0.7777 - recall_1: 0.7642 - val_loss: 0.4000 - val_accuracy: 0.8359 -
val_recall_1: 0.8258

Epoch 00012: val_accuracy did not improve from 0.83766
Epoch 13/400
331/331 [==============================] - 39s 118ms/step - loss: 0.5033 -
accuracy: 0.7783 - recall_1: 0.7642 - val_loss: 0.3864 - val_accuracy: 0.8465 -
val_recall_1: 0.8350

Epoch 00013: val_accuracy improved from 0.83766 to 0.84646, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 14/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4906 -
accuracy: 0.7837 - recall_1: 0.7716 - val_loss: 0.3704 - val_accuracy: 0.8535 -
val_recall_1: 0.8447
```

```
Epoch 00014: val_accuracy improved from 0.84646 to 0.85350, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 15/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4859 -
accuracy: 0.7860 - recall_1: 0.7720 - val_loss: 0.3567 - val_accuracy: 0.8561 -
val_recall_1: 0.8482

Epoch 00015: val_accuracy improved from 0.85350 to 0.85614, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 16/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4705 -
accuracy: 0.7890 - recall_1: 0.7791 - val_loss: 0.3522 - val_accuracy: 0.8583 -
val_recall_1: 0.8513

Epoch 00016: val_accuracy improved from 0.85614 to 0.85834, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 17/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4624 -
accuracy: 0.7997 - recall_1: 0.7901 - val_loss: 0.3421 - val_accuracy: 0.8610 -
val_recall_1: 0.8561

Epoch 00017: val_accuracy improved from 0.85834 to 0.86098, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 18/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4425 -
accuracy: 0.8127 - recall_1: 0.8043 - val_loss: 0.3334 - val_accuracy: 0.8676 -
val_recall_1: 0.8614

Epoch 00018: val_accuracy improved from 0.86098 to 0.86758, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 19/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4393 -
accuracy: 0.8072 - recall_1: 0.7983 - val_loss: 0.3765 - val_accuracy: 0.8359 -
val_recall_1: 0.8293

Epoch 00019: val_accuracy did not improve from 0.86758
Epoch 20/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4480 -
accuracy: 0.8090 - recall_1: 0.8003 - val_loss: 0.3461 - val_accuracy: 0.8544 -
val_recall_1: 0.8469

Epoch 00020: val_accuracy did not improve from 0.86758
Epoch 21/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4157 -
accuracy: 0.8169 - recall_1: 0.8077 - val_loss: 0.3404 - val_accuracy: 0.8592 -
val_recall_1: 0.8509
```

```
Epoch 00021: val_accuracy did not improve from 0.86758
Epoch 22/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4194 -
accuracy: 0.8189 - recall_1: 0.8113 - val_loss: 0.3101 - val_accuracy: 0.8759 -
val_recall_1: 0.8715

Epoch 00022: val_accuracy improved from 0.86758 to 0.87593, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 23/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4095 -
accuracy: 0.8277 - recall_1: 0.8197 - val_loss: 0.3266 - val_accuracy: 0.8641 -
val_recall_1: 0.8597

Epoch 00023: val_accuracy did not improve from 0.87593
Epoch 24/400
331/331 [==============================] - 39s 118ms/step - loss: 0.4080 -
accuracy: 0.8302 - recall_1: 0.8241 - val_loss: 0.3611 - val_accuracy: 0.8438 -
val_recall_1: 0.8407

Epoch 00024: val_accuracy did not improve from 0.87593
Epoch 25/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3997 -
accuracy: 0.8279 - recall_1: 0.8212 - val_loss: 0.2974 - val_accuracy: 0.8773 -
val_recall_1: 0.8720

Epoch 00025: val_accuracy improved from 0.87593 to 0.87725, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 26/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3914 -
accuracy: 0.8365 - recall_1: 0.8292 - val_loss: 0.3144 - val_accuracy: 0.8720 -
val_recall_1: 0.8667

Epoch 00026: val_accuracy did not improve from 0.87725
Epoch 27/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3824 -
accuracy: 0.8446 - recall_1: 0.8391 - val_loss: 0.3508 - val_accuracy: 0.8539 -
val_recall_1: 0.8495

Epoch 00027: val_accuracy did not improve from 0.87725
Epoch 28/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3690 -
accuracy: 0.8461 - recall_1: 0.8419 - val_loss: 0.5417 - val_accuracy: 0.7624 -
val_recall_1: 0.7527

Epoch 00028: val_accuracy did not improve from 0.87725
Epoch 29/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3775 -
accuracy: 0.8503 - recall_1: 0.8433 - val_loss: 0.4224 - val_accuracy: 0.8209 -
```

```
val_recall_1: 0.8135

Epoch 00029: val_accuracy did not improve from 0.87725
Epoch 30/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3810 -
accuracy: 0.8442 - recall_1: 0.8374 - val_loss: 0.4170 - val_accuracy: 0.8218 -
val_recall_1: 0.8152

Epoch 00030: val_accuracy did not improve from 0.87725
Epoch 31/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3734 -
accuracy: 0.8476 - recall_1: 0.8406 - val_loss: 0.3514 - val_accuracy: 0.8575 -
val_recall_1: 0.8504

Epoch 00031: val_accuracy did not improve from 0.87725
Epoch 32/400
331/331 [==============================] - 39s 119ms/step - loss: 0.3461 -
accuracy: 0.8610 - recall_1: 0.8540 - val_loss: 0.3699 - val_accuracy: 0.8429 -
val_recall_1: 0.8372

Epoch 00032: val_accuracy did not improve from 0.87725
Epoch 33/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3681 -
accuracy: 0.8497 - recall_1: 0.8433 - val_loss: 0.2722 - val_accuracy: 0.8957 -
val_recall_1: 0.8887

Epoch 00033: val_accuracy improved from 0.87725 to 0.89573, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 34/400
331/331 [==============================] - 39s 119ms/step - loss: 0.3511 -
accuracy: 0.8555 - recall_1: 0.8495 - val_loss: 0.3274 - val_accuracy: 0.8693 -
val_recall_1: 0.8623

Epoch 00034: val_accuracy did not improve from 0.89573
Epoch 35/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3628 -
accuracy: 0.8508 - recall_1: 0.8477 - val_loss: 0.2710 - val_accuracy: 0.8940 -
val_recall_1: 0.8891

Epoch 00035: val_accuracy did not improve from 0.89573
Epoch 36/400
331/331 [==============================] - 39s 119ms/step - loss: 0.3488 -
accuracy: 0.8565 - recall_1: 0.8500 - val_loss: 0.4240 - val_accuracy: 0.8231 -
val_recall_1: 0.8161

Epoch 00036: val_accuracy did not improve from 0.89573
Epoch 37/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3471 -
```

```
accuracy: 0.8572 - recall_1: 0.8503 - val_loss: 0.3046 - val_accuracy: 0.8808 -
val_recall_1: 0.8755

Epoch 00037: val_accuracy did not improve from 0.89573
Epoch 38/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3269 -
accuracy: 0.8702 - recall_1: 0.8642 - val_loss: 0.2769 - val_accuracy: 0.8909 -
val_recall_1: 0.8852

Epoch 00038: val_accuracy did not improve from 0.89573
Epoch 39/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3414 -
accuracy: 0.8624 - recall_1: 0.8583 - val_loss: 0.3100 - val_accuracy: 0.8777 -
val_recall_1: 0.8724

Epoch 00039: val_accuracy did not improve from 0.89573
Epoch 40/400
331/331 [==============================] - 39s 119ms/step - loss: 0.3457 -
accuracy: 0.8603 - recall_1: 0.8553 - val_loss: 0.3630 - val_accuracy: 0.8531 -
val_recall_1: 0.8443

Epoch 00040: val_accuracy did not improve from 0.89573
Epoch 41/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3417 -
accuracy: 0.8603 - recall_1: 0.8551 - val_loss: 0.5026 - val_accuracy: 0.7840 -
val_recall_1: 0.7756

Epoch 00041: val_accuracy did not improve from 0.89573
Epoch 42/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3297 -
accuracy: 0.8654 - recall_1: 0.8616 - val_loss: 0.5795 - val_accuracy: 0.7536 -
val_recall_1: 0.7466

Epoch 00042: val_accuracy did not improve from 0.89573
Epoch 43/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3104 -
accuracy: 0.8786 - recall_1: 0.8728 - val_loss: 0.3314 - val_accuracy: 0.8720 -
val_recall_1: 0.8671

Epoch 00043: val_accuracy did not improve from 0.89573
Epoch 44/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3234 -
accuracy: 0.8776 - recall_1: 0.8737 - val_loss: 0.4477 - val_accuracy: 0.8104 -
val_recall_1: 0.7985

Epoch 00044: val_accuracy did not improve from 0.89573
Epoch 45/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3251 -
```

```
accuracy: 0.8662 - recall_1: 0.8604 - val_loss: 0.3279 - val_accuracy: 0.8768 -
val_recall_1: 0.8685

Epoch 00045: val_accuracy did not improve from 0.89573
Epoch 46/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3312 -
accuracy: 0.8632 - recall_1: 0.8589 - val_loss: 0.5603 - val_accuracy: 0.7607 -
val_recall_1: 0.7523

Epoch 00046: val_accuracy did not improve from 0.89573
Epoch 47/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3253 -
accuracy: 0.8672 - recall_1: 0.8628 - val_loss: 0.3844 - val_accuracy: 0.8447 -
val_recall_1: 0.8377

Epoch 00047: val_accuracy did not improve from 0.89573
Epoch 48/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3158 -
accuracy: 0.8772 - recall_1: 0.8726 - val_loss: 0.5945 - val_accuracy: 0.7571 -
val_recall_1: 0.7484

Epoch 00048: val_accuracy did not improve from 0.89573
Epoch 49/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3202 -
accuracy: 0.8735 - recall_1: 0.8676 - val_loss: 0.3613 - val_accuracy: 0.8544 -
val_recall_1: 0.8482

Epoch 00049: val_accuracy did not improve from 0.89573
Epoch 50/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3168 -
accuracy: 0.8701 - recall_1: 0.8664 - val_loss: 0.3639 - val_accuracy: 0.8601 -
val_recall_1: 0.8535

Epoch 00050: val_accuracy did not improve from 0.89573
Epoch 51/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3228 -
accuracy: 0.8680 - recall_1: 0.8620 - val_loss: 0.5631 - val_accuracy: 0.7664 -
val_recall_1: 0.7576

Epoch 00051: val_accuracy did not improve from 0.89573
Epoch 52/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3098 -
accuracy: 0.8767 - recall_1: 0.8724 - val_loss: 0.2645 - val_accuracy: 0.9023 -
val_recall_1: 0.8975

Epoch 00052: val_accuracy improved from 0.89573 to 0.90233, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 53/400
```

```
331/331 [==============================] - 39s 118ms/step - loss: 0.3184 -
accuracy: 0.8702 - recall_1: 0.8664 - val_loss: 0.2698 - val_accuracy: 0.9015 -
val_recall_1: 0.8962

Epoch 00053: val_accuracy did not improve from 0.90233
Epoch 54/400
331/331 [==============================] - 39s 119ms/step - loss: 0.3082 -
accuracy: 0.8739 - recall_1: 0.8698 - val_loss: 0.5126 - val_accuracy: 0.7923 -
val_recall_1: 0.7857

Epoch 00054: val_accuracy did not improve from 0.90233
Epoch 55/400
331/331 [==============================] - 39s 119ms/step - loss: 0.2968 -
accuracy: 0.8807 - recall_1: 0.8773 - val_loss: 0.4723 - val_accuracy: 0.8033 -
val_recall_1: 0.7945

Epoch 00055: val_accuracy did not improve from 0.90233
Epoch 56/400
331/331 [==============================] - 39s 118ms/step - loss: 0.3211 -
accuracy: 0.8726 - recall_1: 0.8676 - val_loss: 0.3378 - val_accuracy: 0.8693 -
val_recall_1: 0.8627

Epoch 00056: val_accuracy did not improve from 0.90233
Epoch 57/400
331/331 [==============================] - 39s 119ms/step - loss: 0.3132 -
accuracy: 0.8774 - recall_1: 0.8737 - val_loss: 0.3273 - val_accuracy: 0.8742 -
val_recall_1: 0.8693

Epoch 00057: val_accuracy did not improve from 0.90233
Epoch 58/400
331/331 [==============================] - 40s 121ms/step - loss: 0.3147 -
accuracy: 0.8797 - recall_1: 0.8747 - val_loss: 0.2671 - val_accuracy: 0.9032 -
val_recall_1: 0.8971

Epoch 00058: val_accuracy improved from 0.90233 to 0.90321, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 59/400
331/331 [==============================] - 40s 121ms/step - loss: 0.3056 -
accuracy: 0.8807 - recall_1: 0.8756 - val_loss: 0.4302 - val_accuracy: 0.8205 -
val_recall_1: 0.8157

Epoch 00059: val_accuracy did not improve from 0.90321
Epoch 60/400
331/331 [==============================] - 40s 121ms/step - loss: 0.3024 -
accuracy: 0.8774 - recall_1: 0.8735 - val_loss: 0.4137 - val_accuracy: 0.8311 -
val_recall_1: 0.8240

Epoch 00060: val_accuracy did not improve from 0.90321
```

```
Epoch 61/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2855 -
accuracy: 0.8921 - recall_1: 0.8885 - val_loss: 0.2914 - val_accuracy: 0.8909 -
val_recall_1: 0.8874

Epoch 00061: val_accuracy did not improve from 0.90321
Epoch 62/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2994 -
accuracy: 0.8798 - recall_1: 0.8750 - val_loss: 0.2730 - val_accuracy: 0.8971 -
val_recall_1: 0.8922

Epoch 00062: val_accuracy did not improve from 0.90321
Epoch 63/400
331/331 [==============================] - 40s 121ms/step - loss: 0.3019 -
accuracy: 0.8843 - recall_1: 0.8804 - val_loss: 0.2909 - val_accuracy: 0.8935 -
val_recall_1: 0.8874

Epoch 00063: val_accuracy did not improve from 0.90321
Epoch 64/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2961 -
accuracy: 0.8876 - recall_1: 0.8846 - val_loss: 0.2995 - val_accuracy: 0.8856 -
val_recall_1: 0.8808

Epoch 00064: val_accuracy did not improve from 0.90321
Epoch 65/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2892 -
accuracy: 0.8887 - recall_1: 0.8829 - val_loss: 0.3746 - val_accuracy: 0.8561 -
val_recall_1: 0.8465

Epoch 00065: val_accuracy did not improve from 0.90321
Epoch 66/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2886 -
accuracy: 0.8868 - recall_1: 0.8836 - val_loss: 0.3330 - val_accuracy: 0.8755 -
val_recall_1: 0.8698

Epoch 00066: val_accuracy did not improve from 0.90321
Epoch 67/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2915 -
accuracy: 0.8858 - recall_1: 0.8825 - val_loss: 0.3705 - val_accuracy: 0.8504 -
val_recall_1: 0.8447

Epoch 00067: val_accuracy did not improve from 0.90321
Epoch 68/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2806 -
accuracy: 0.8861 - recall_1: 0.8818 - val_loss: 0.2831 - val_accuracy: 0.8966 -
val_recall_1: 0.8896

Epoch 00068: val_accuracy did not improve from 0.90321
```

```
Epoch 69/400
331/331 [==============================] - 40s 120ms/step - loss: 0.2971 -
accuracy: 0.8789 - recall_1: 0.8747 - val_loss: 0.2557 - val_accuracy: 0.8984 -
val_recall_1: 0.8957

Epoch 00069: val_accuracy did not improve from 0.90321
Epoch 70/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2979 -
accuracy: 0.8789 - recall_1: 0.8755 - val_loss: 0.2887 - val_accuracy: 0.8909 -
val_recall_1: 0.8883

Epoch 00070: val_accuracy did not improve from 0.90321
Epoch 71/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2816 -
accuracy: 0.8889 - recall_1: 0.8857 - val_loss: 0.4672 - val_accuracy: 0.8099 -
val_recall_1: 0.8038

Epoch 00071: val_accuracy did not improve from 0.90321
Epoch 72/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2950 -
accuracy: 0.8839 - recall_1: 0.8791 - val_loss: 0.4792 - val_accuracy: 0.8051 -
val_recall_1: 0.7981

Epoch 00072: val_accuracy did not improve from 0.90321
Epoch 73/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2957 -
accuracy: 0.8855 - recall_1: 0.8822 - val_loss: 0.3305 - val_accuracy: 0.8702 -
val_recall_1: 0.8676

Epoch 00073: val_accuracy did not improve from 0.90321
Epoch 74/400
331/331 [==============================] - 40s 120ms/step - loss: 0.2839 -
accuracy: 0.8859 - recall_1: 0.8828 - val_loss: 0.3332 - val_accuracy: 0.8680 -
val_recall_1: 0.8641

Epoch 00074: val_accuracy did not improve from 0.90321
Epoch 75/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2783 -
accuracy: 0.8928 - recall_1: 0.8891 - val_loss: 0.3348 - val_accuracy: 0.8658 -
val_recall_1: 0.8605

Epoch 00075: val_accuracy did not improve from 0.90321
Epoch 76/400
331/331 [==============================] - 40s 120ms/step - loss: 0.2865 -
accuracy: 0.8845 - recall_1: 0.8813 - val_loss: 0.3367 - val_accuracy: 0.8645 -
val_recall_1: 0.8605

Epoch 00076: val_accuracy did not improve from 0.90321
```

```
Epoch 77/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2976 -
accuracy: 0.8810 - recall_1: 0.8781 - val_loss: 0.2685 - val_accuracy: 0.8953 -
val_recall_1: 0.8922

Epoch 00077: val_accuracy did not improve from 0.90321
Epoch 78/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2838 -
accuracy: 0.8911 - recall_1: 0.8870 - val_loss: 0.3028 - val_accuracy: 0.8839 -
val_recall_1: 0.8812

Epoch 00078: val_accuracy did not improve from 0.90321
Epoch 79/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2689 -
accuracy: 0.8922 - recall_1: 0.8886 - val_loss: 0.3127 - val_accuracy: 0.8764 -
val_recall_1: 0.8742

Epoch 00079: val_accuracy did not improve from 0.90321
Epoch 80/400
331/331 [==============================] - 40s 122ms/step - loss: 0.2798 -
accuracy: 0.8893 - recall_1: 0.8853 - val_loss: 0.2737 - val_accuracy: 0.8975 -
val_recall_1: 0.8949

Epoch 00080: val_accuracy did not improve from 0.90321
Epoch 81/400
331/331 [==============================] - 40s 122ms/step - loss: 0.2833 -
accuracy: 0.8914 - recall_1: 0.8886 - val_loss: 0.4232 - val_accuracy: 0.8306 -
val_recall_1: 0.8236

Epoch 00081: val_accuracy did not improve from 0.90321
Epoch 82/400
331/331 [==============================] - 40s 122ms/step - loss: 0.2726 -
accuracy: 0.8952 - recall_1: 0.8917 - val_loss: 0.2437 - val_accuracy: 0.9063 -
val_recall_1: 0.9037

Epoch 00082: val_accuracy improved from 0.90321 to 0.90629, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 83/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2745 -
accuracy: 0.8946 - recall_1: 0.8905 - val_loss: 0.3182 - val_accuracy: 0.8759 -
val_recall_1: 0.8724

Epoch 00083: val_accuracy did not improve from 0.90629
Epoch 84/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2867 -
accuracy: 0.8864 - recall_1: 0.8826 - val_loss: 0.2350 - val_accuracy: 0.9129 -
val_recall_1: 0.9098
```

```
Epoch 00084: val_accuracy improved from 0.90629 to 0.91289, saving model to
/home/hivini/learn/research/new-covid/best_model.h5
Epoch 85/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2663 -
accuracy: 0.8928 - recall_1: 0.8896 - val_loss: 0.2267 - val_accuracy: 0.9107 -
val_recall_1: 0.9076

Epoch 00085: val_accuracy did not improve from 0.91289
Epoch 86/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2841 -
accuracy: 0.8935 - recall_1: 0.8906 - val_loss: 0.2894 - val_accuracy: 0.8918 -
val_recall_1: 0.8883

Epoch 00086: val_accuracy did not improve from 0.91289
Epoch 87/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2766 -
accuracy: 0.8923 - recall_1: 0.8891 - val_loss: 0.4083 - val_accuracy: 0.8359 -
val_recall_1: 0.8315

Epoch 00087: val_accuracy did not improve from 0.91289
Epoch 88/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2789 -
accuracy: 0.8879 - recall_1: 0.8848 - val_loss: 0.2484 - val_accuracy: 0.9063 -
val_recall_1: 0.9041

Epoch 00088: val_accuracy did not improve from 0.91289
Epoch 89/400
331/331 [==============================] - 41s 123ms/step - loss: 0.2700 -
accuracy: 0.8929 - recall_1: 0.8907 - val_loss: 0.3365 - val_accuracy: 0.8641 -
val_recall_1: 0.8623

Epoch 00089: val_accuracy did not improve from 0.91289
Epoch 90/400
331/331 [==============================] - 41s 123ms/step - loss: 0.2796 -
accuracy: 0.8893 - recall_1: 0.8853 - val_loss: 0.2784 - val_accuracy: 0.8966 -
val_recall_1: 0.8935

Epoch 00090: val_accuracy did not improve from 0.91289
Epoch 91/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2675 -
accuracy: 0.8930 - recall_1: 0.8894 - val_loss: 0.2390 - val_accuracy: 0.9081 -
val_recall_1: 0.9059

Epoch 00091: val_accuracy did not improve from 0.91289
Epoch 92/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2660 -
accuracy: 0.8971 - recall_1: 0.8939 - val_loss: 0.6176 - val_accuracy: 0.7585 -
val_recall_1: 0.7558
```

```
Epoch 00092: val_accuracy did not improve from 0.91289
Epoch 93/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2610 -
accuracy: 0.8976 - recall_1: 0.8941 - val_loss: 0.2555 - val_accuracy: 0.9028 -
val_recall_1: 0.9006

Epoch 00093: val_accuracy did not improve from 0.91289
Epoch 94/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2775 -
accuracy: 0.8914 - recall_1: 0.8868 - val_loss: 0.2706 - val_accuracy: 0.8962 -
val_recall_1: 0.8940

Epoch 00094: val_accuracy did not improve from 0.91289
Epoch 95/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2680 -
accuracy: 0.8978 - recall_1: 0.8931 - val_loss: 0.2646 - val_accuracy: 0.9001 -
val_recall_1: 0.8988

Epoch 00095: val_accuracy did not improve from 0.91289
Epoch 96/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2563 -
accuracy: 0.9038 - recall_1: 0.9012 - val_loss: 0.2596 - val_accuracy: 0.9006 -
val_recall_1: 0.8984

Epoch 00096: val_accuracy did not improve from 0.91289
Epoch 97/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2837 -
accuracy: 0.8890 - recall_1: 0.8848 - val_loss: 0.3674 - val_accuracy: 0.8557 -
val_recall_1: 0.8526

Epoch 00097: val_accuracy did not improve from 0.91289
Epoch 98/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2708 -
accuracy: 0.8935 - recall_1: 0.8898 - val_loss: 0.3797 - val_accuracy: 0.8482 -
val_recall_1: 0.8465

Epoch 00098: val_accuracy did not improve from 0.91289
Epoch 99/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2629 -
accuracy: 0.8964 - recall_1: 0.8935 - val_loss: 0.2601 - val_accuracy: 0.8993 -
val_recall_1: 0.8971

Epoch 00099: val_accuracy did not improve from 0.91289
Epoch 100/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2635 -
accuracy: 0.8961 - recall_1: 0.8931 - val_loss: 0.2590 - val_accuracy: 0.8988 -
val_recall_1: 0.8971
```

```
Epoch 00100: val_accuracy did not improve from 0.91289
Epoch 101/400
331/331 [==============================] - 40s 122ms/step - loss: 0.2769 -
accuracy: 0.8953 - recall_1: 0.8912 - val_loss: 0.2675 - val_accuracy: 0.9006 -
val_recall_1: 0.8975

Epoch 00101: val_accuracy did not improve from 0.91289
Epoch 102/400
331/331 [==============================] - 40s 122ms/step - loss: 0.2553 -
accuracy: 0.8996 - recall_1: 0.8976 - val_loss: 0.3147 - val_accuracy: 0.8786 -
val_recall_1: 0.8773

Epoch 00102: val_accuracy did not improve from 0.91289
Epoch 103/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2811 -
accuracy: 0.8913 - recall_1: 0.8890 - val_loss: 0.3698 - val_accuracy: 0.8544 -
val_recall_1: 0.8526

Epoch 00103: val_accuracy did not improve from 0.91289
Epoch 104/400
331/331 [==============================] - 40s 122ms/step - loss: 0.2641 -
accuracy: 0.8981 - recall_1: 0.8947 - val_loss: 0.2452 - val_accuracy: 0.9063 -
val_recall_1: 0.9041

Epoch 00104: val_accuracy did not improve from 0.91289
Epoch 105/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2694 -
accuracy: 0.8946 - recall_1: 0.8920 - val_loss: 0.2402 - val_accuracy: 0.9067 -
val_recall_1: 0.9045

Epoch 00105: val_accuracy did not improve from 0.91289
Epoch 106/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2694 -
accuracy: 0.8960 - recall_1: 0.8938 - val_loss: 0.2775 - val_accuracy: 0.8966 -
val_recall_1: 0.8944

Epoch 00106: val_accuracy did not improve from 0.91289
Epoch 107/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2590 -
accuracy: 0.8996 - recall_1: 0.8966 - val_loss: 0.2530 - val_accuracy: 0.9028 -
val_recall_1: 0.8993

Epoch 00107: val_accuracy did not improve from 0.91289
Epoch 108/400
331/331 [==============================] - 40s 122ms/step - loss: 0.2579 -
accuracy: 0.8978 - recall_1: 0.8934 - val_loss: 0.2284 - val_accuracy: 0.9089 -
val_recall_1: 0.9076
```

```
Epoch 00108: val_accuracy did not improve from 0.91289
Epoch 109/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2530 -
accuracy: 0.9022 - recall_1: 0.8990 - val_loss: 0.2607 - val_accuracy: 0.9032 -
val_recall_1: 0.9006

Epoch 00109: val_accuracy did not improve from 0.91289
Epoch 110/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2539 -
accuracy: 0.9035 - recall_1: 0.9008 - val_loss: 0.3009 - val_accuracy: 0.8883 -
val_recall_1: 0.8834

Epoch 00110: val_accuracy did not improve from 0.91289
Epoch 111/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2565 -
accuracy: 0.8969 - recall_1: 0.8939 - val_loss: 0.2310 - val_accuracy: 0.9111 -
val_recall_1: 0.9089

Epoch 00111: val_accuracy did not improve from 0.91289
Epoch 112/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2635 -
accuracy: 0.8974 - recall_1: 0.8943 - val_loss: 0.2645 - val_accuracy: 0.9015 -
val_recall_1: 0.8993

Epoch 00112: val_accuracy did not improve from 0.91289
Epoch 113/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2448 -
accuracy: 0.9053 - recall_1: 0.9021 - val_loss: 0.2767 - val_accuracy: 0.8966 -
val_recall_1: 0.8953

Epoch 00113: val_accuracy did not improve from 0.91289
Epoch 114/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2538 -
accuracy: 0.9028 - recall_1: 0.9002 - val_loss: 0.2636 - val_accuracy: 0.9001 -
val_recall_1: 0.8984

Epoch 00114: val_accuracy did not improve from 0.91289
Epoch 115/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2533 -
accuracy: 0.8986 - recall_1: 0.8965 - val_loss: 0.2611 - val_accuracy: 0.9037 -
val_recall_1: 0.9006

Epoch 00115: val_accuracy did not improve from 0.91289
Epoch 116/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2611 -
accuracy: 0.8970 - recall_1: 0.8943 - val_loss: 0.3871 - val_accuracy: 0.8478 -
val_recall_1: 0.8451
```

```
Epoch 00116: val_accuracy did not improve from 0.91289
Epoch 117/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2570 -
accuracy: 0.8976 - recall_1: 0.8954 - val_loss: 0.2854 - val_accuracy: 0.8979 -
val_recall_1: 0.8935

Epoch 00117: val_accuracy did not improve from 0.91289
Epoch 118/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2658 -
accuracy: 0.8957 - recall_1: 0.8924 - val_loss: 0.2704 - val_accuracy: 0.8993 -
val_recall_1: 0.8966

Epoch 00118: val_accuracy did not improve from 0.91289
Epoch 119/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2471 -
accuracy: 0.9041 - recall_1: 0.8996 - val_loss: 0.2585 - val_accuracy: 0.9023 -
val_recall_1: 0.9001

Epoch 00119: val_accuracy did not improve from 0.91289
Epoch 120/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2626 -
accuracy: 0.8976 - recall_1: 0.8955 - val_loss: 0.3301 - val_accuracy: 0.8755 -
val_recall_1: 0.8720

Epoch 00120: val_accuracy did not improve from 0.91289
Epoch 121/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2655 -
accuracy: 0.8964 - recall_1: 0.8941 - val_loss: 0.2868 - val_accuracy: 0.8940 -
val_recall_1: 0.8927

Epoch 00121: val_accuracy did not improve from 0.91289
Epoch 122/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2520 -
accuracy: 0.8964 - recall_1: 0.8919 - val_loss: 0.4278 - val_accuracy: 0.8311 -
val_recall_1: 0.8284

Epoch 00122: val_accuracy did not improve from 0.91289
Epoch 123/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2605 -
accuracy: 0.9000 - recall_1: 0.8964 - val_loss: 0.3118 - val_accuracy: 0.8830 -
val_recall_1: 0.8812

Epoch 00123: val_accuracy did not improve from 0.91289
Epoch 124/400
331/331 [==============================] - 41s 123ms/step - loss: 0.2557 -
accuracy: 0.8997 - recall_1: 0.8972 - val_loss: 0.2505 - val_accuracy: 0.9045 -
val_recall_1: 0.9028
```

```
Epoch 00124: val_accuracy did not improve from 0.91289
Epoch 125/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2521 -
accuracy: 0.9010 - recall_1: 0.8987 - val_loss: 0.2871 - val_accuracy: 0.8913 -
val_recall_1: 0.8891

Epoch 00125: val_accuracy did not improve from 0.91289
Epoch 126/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2635 -
accuracy: 0.8963 - recall_1: 0.8944 - val_loss: 0.2703 - val_accuracy: 0.9015 -
val_recall_1: 0.8988

Epoch 00126: val_accuracy did not improve from 0.91289
Epoch 127/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2442 -
accuracy: 0.9041 - recall_1: 0.9022 - val_loss: 0.3654 - val_accuracy: 0.8592 -
val_recall_1: 0.8553

Epoch 00127: val_accuracy did not improve from 0.91289
Epoch 128/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2573 -
accuracy: 0.9027 - recall_1: 0.8997 - val_loss: 0.2709 - val_accuracy: 0.9006 -
val_recall_1: 0.8984

Epoch 00128: val_accuracy did not improve from 0.91289
Epoch 129/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2566 -
accuracy: 0.9008 - recall_1: 0.8987 - val_loss: 0.4114 - val_accuracy: 0.8385 -
val_recall_1: 0.8355

Epoch 00129: val_accuracy did not improve from 0.91289
Epoch 130/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2551 -
accuracy: 0.9027 - recall_1: 0.9007 - val_loss: 0.2743 - val_accuracy: 0.8984 -
val_recall_1: 0.8966

Epoch 00130: val_accuracy did not improve from 0.91289
Epoch 131/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2482 -
accuracy: 0.9040 - recall_1: 0.9005 - val_loss: 0.2936 - val_accuracy: 0.8922 -
val_recall_1: 0.8900

Epoch 00131: val_accuracy did not improve from 0.91289
Epoch 132/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2652 -
accuracy: 0.8981 - recall_1: 0.8967 - val_loss: 0.3365 - val_accuracy: 0.8680 -
val_recall_1: 0.8667
```

```
Epoch 00132: val_accuracy did not improve from 0.91289
Epoch 133/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2456 -
accuracy: 0.9016 - recall_1: 0.8986 - val_loss: 0.2876 - val_accuracy: 0.8909 -
val_recall_1: 0.8887

Epoch 00133: val_accuracy did not improve from 0.91289
Epoch 134/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2472 -
accuracy: 0.9010 - recall_1: 0.8995 - val_loss: 0.3332 - val_accuracy: 0.8693 -
val_recall_1: 0.8671

Epoch 00134: val_accuracy did not improve from 0.91289
Epoch 135/400
331/331 [==============================] - 40s 121ms/step - loss: 0.2685 -
accuracy: 0.8952 - recall_1: 0.8920 - val_loss: 0.3380 - val_accuracy: 0.8711 -
val_recall_1: 0.8680

Epoch 00135: val_accuracy did not improve from 0.91289
Epoch 00135: early stopping
```

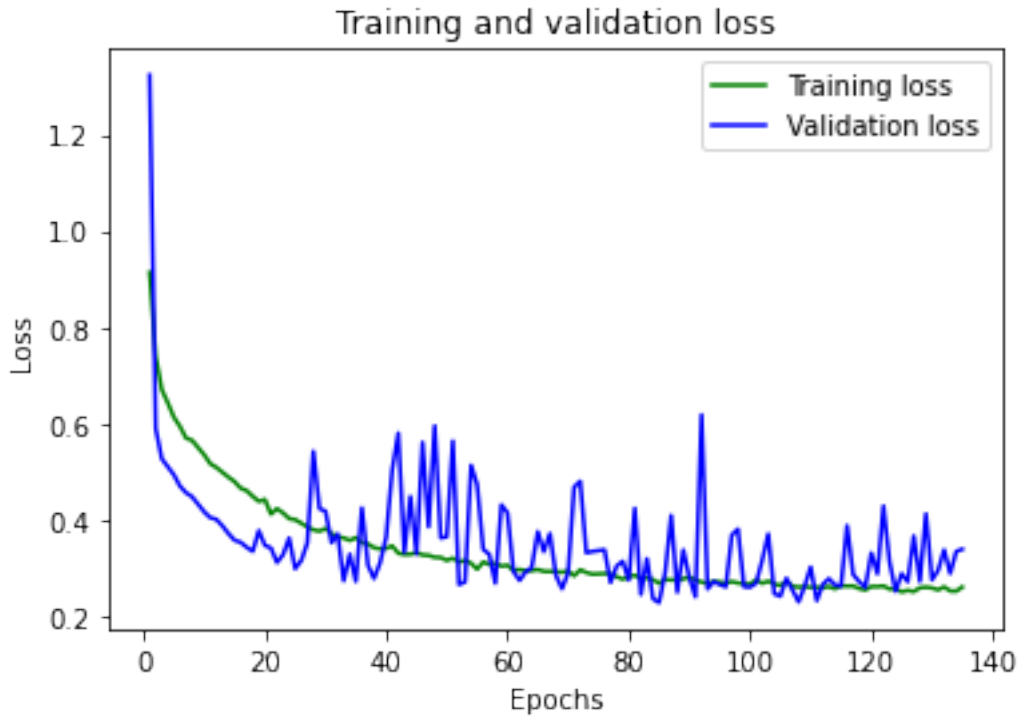
[ ]:
```
model.save(os.path.join(BASE_PATH, 'covid_classifier_result.h5'))
```

[ ]:
```
test_loss, test_acc, test_recall = model.evaluate(test_generator)
print("Loss on test set: ", test_loss)
print("Accuracy on test set: ", test_acc)
```

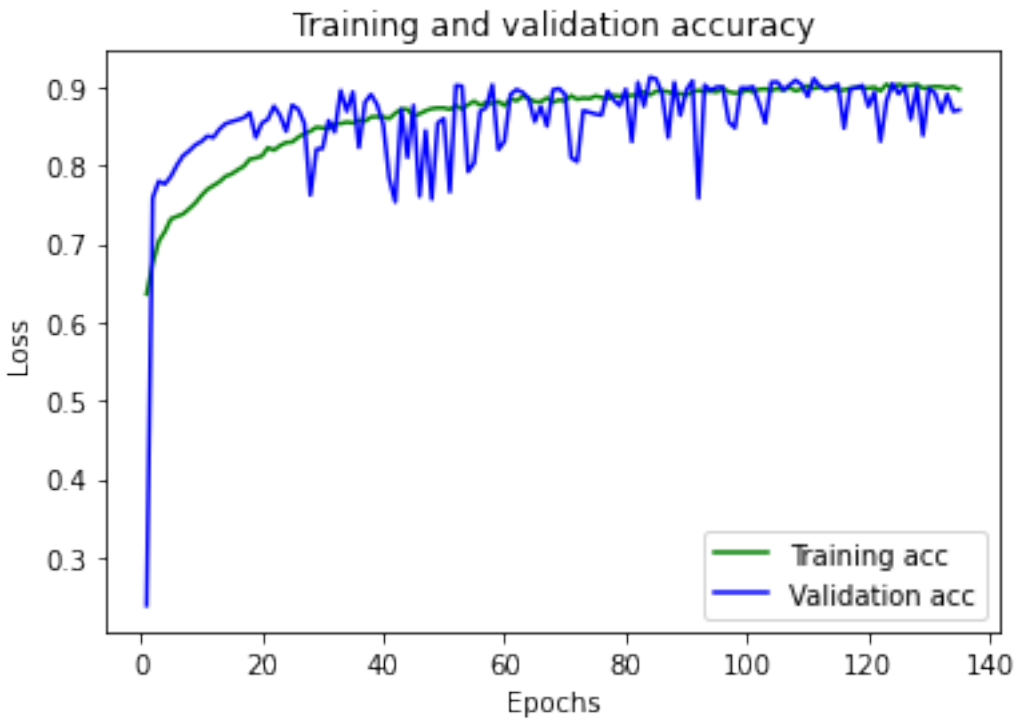
```
72/72 [==============================] - 5s 68ms/step - loss: 0.3459 - accuracy:
0.8606 - recall_1: 0.8566
Loss on test set:  0.34593790769577026
Accuracy on test set:  0.860598087310791
```

[ ]:
```
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(1, len(acc) + 1)
# bo is for blue dot.
plt.plot(epochs, loss, 'g', label='Training loss')
# b is for solid blue line
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('Epochs')
```

```
plt.ylabel('Loss')
plt.legend()

plt.show()
```



[ ]:
```
plt.clf()

plt.plot(epochs, acc, 'g', label='Training acc')
plt.plot(epochs, val_acc, 'b', label='Validation acc')
plt.title('Training and validation accuracy')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()

plt.show()
```


Training and validation accuracy
Loss
Epochs
Training acc
Validation acc
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0
20
40
60
80
100
120
140