Classification CD

December 20, 2023

1 Binary Classification of Cats and Dogs Images

1.1 Preprocessing

[ ]:
```
import tensorflow as tf
import numpy as np
from tensorflow.keras.preprocessing import image_dataset_from_directory

# Importing data
ds_train = image_dataset_from_directory(
    'train',
    labels='inferred',
    label_mode = 'binary',
    interpolation='nearest',
    image_size = [128,128],
    batch_size=64,
)

ds_valid = image_dataset_from_directory(
    'test',
    labels='inferred',
    label_mode = 'binary',
    interpolation='nearest',
    image_size = [128,128],
    batch_size=64
)
```

```
Found 557 files belonging to 2 classes.
Found 140 files belonging to 2 classes.
```

1.1.1 Visualization

[ ]:
```
import matplotlib.pyplot as plt

# Selecting images and their respective labels
for images,labels in ds_train.take(1):

    # Creating figure of 10x10 (inches)
```

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

# Itering through images and plotting them
## uint8 ensures treatment as 8-bit integers
for i in range(9):
    ax = plt.subplot(3,3,i+1)
    plt.imshow(images[i].numpy().astype('uint8'))
    plt.title(f"Class: {labels[i].numpy()}")
    plt.axis('off')

plt.show()
```

1.2 Data Preparation

[ ]:
```
# Defining function for treating images as float data
def convert_to_float(image,label):
    image = tf.image.convert_image_dtype(image,dtype=tf.float32)
    return image,label

# Optimizing CPU usage
AUTOTUNE = tf.data.experimental.AUTOTUNE

# Converting images
## Storing data in cache memory
### Optimizing data availabilty
ds_train = (
    ds_train.map(convert_to_float).cache().prefetch(buffer_size = AUTOTUNE)
)

ds_valid = (
    ds_valid.map(convert_to_float).cache().prefetch(buffer_size = AUTOTUNE)
)
```

1.3 Architecture of the Convolutional Neural Network

[ ]:
```
from keras.applications import DenseNet121

base_model = DenseNet121(weights='imagenet', include_top=False,␣
 ↪input_shape=(128, 128, 3))
```

[ ]:
```
from tensorflow import keras
from keras.models import Sequential
from keras.layers import Dense, Dropout, Flatten, Activation
from keras.layers import Conv2D, MaxPooling2D, LeakyReLU
from keras.optimizers import Adam
from keras.callbacks import EarlyStopping, ReduceLROnPlateau,␣
 ↪LearningRateScheduler
from keras.optimizers.schedules import ExponentialDecay
from keras.applications import DenseNet121

early_stopping = EarlyStopping(
    min_delta=0.01,
    patience=100,
    restore_best_weights=True
)

reduce_learning_rate = ReduceLROnPlateau(
    monitor='binary_accuracy',
    patience=100,
```

```
    verbose=1,
    factor=0.5,
    min_lr=0.00001
)

learning_rate_schedule = ExponentialDecay(
    initial_learning_rate = 0.01,
    decay_steps = 1000,
    decay_rate=0.5
)

lr_callback = LearningRateScheduler(learning_rate_schedule)
callback = [lr_callback, reduce_learning_rate, early_stopping]

for layer in base_model.layers:
    layer.trainable = False

# Defining the model
model = Sequential()

model.add(base_model),

model.add(Flatten()),

model.add(Flatten())

model.add(Dense(256, activation='relu'))

model.add(Dense(128, activation='relu'))

model.add(Dense(64, activation='relu'))

model.add(Dropout(0.5))
model.add(Dense(1, activation='sigmoid'))

model.summary()
```

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 densenet121 (Functional)    (None, 4, 4, 1024)        7037504

 flatten_2 (Flatten)         (None, 16384)             0

 flatten_3 (Flatten)         (None, 16384)             0

 dense_4 (Dense)             (None, 256)               4194560
```

```
 dense_5 (Dense)             (None, 128)               32896
=================================================================
 Layer (type)                Output Shape              Param #
=================================================================
 densenet121 (Functional)    (None, 4, 4, 1024)        7037504

 flatten_2 (Flatten)         (None, 16384)             0

 flatten_3 (Flatten)         (None, 16384)             0

 dense_4 (Dense)             (None, 256)               4194560

 dense_5 (Dense)             (None, 128)               32896

 dense_6 (Dense)             (None, 64)                8256

 dropout_1 (Dropout)         (None, 64)                0

 dense_7 (Dense)             (None, 1)                 65

=================================================================
Total params: 11273281 (43.00 MB)
Trainable params: 4235777 (16.16 MB)
Non-trainable params: 7037504 (26.85 MB)
_________________________________________________________________
```

1.4 Model Compilation

In this section, the model is compiled using the Adam optimizer for it being based on stochastic gradient descent algorithms, while its loss function is set to follow a binary crossentropy algorithm, which is optimal for binary classification task. Finally, in order to mensurate the model's accuracy, binary accuracy metric is implemented also for it being optimal for the present task.

[ ]:
```python
from keras.callbacks import ModelCheckpoint

# Compiling model
model.compile(
    optimizer = Adam(learning_rate=learning_rate_schedule),
    loss='binary_crossentropy',
    metrics = ['binary_accuracy']
)

# Fitting model
history = model.fit(
    ds_train,
    validation_data = ds_valid,
    epochs = 500,
```

```
    verbose=1,
    callbacks = [callback]
)
```

```
Epoch 1/500
9/9 [==============================] - 23s 2s/step - loss: 27.5138 -
binary_accuracy: 0.6517 - val_loss: 4.2007 - val_binary_accuracy: 0.8071 - lr:
0.0099
Epoch 2/500
9/9 [==============================] - 15s 2s/step - loss: 2.9691 -
binary_accuracy: 0.8779 - val_loss: 4.2927 - val_binary_accuracy: 0.7857 - lr:
0.0099
Epoch 3/500
9/9 [==============================] - 15s 2s/step - loss: 0.3855 -
binary_accuracy: 0.9120 - val_loss: 0.2490 - val_binary_accuracy: 0.8714 - lr:
0.0098
Epoch 4/500
9/9 [==============================] - 15s 2s/step - loss: 0.1821 -
binary_accuracy: 0.8995 - val_loss: 0.3186 - val_binary_accuracy: 0.8643 - lr:
0.0098
Epoch 5/500
9/9 [==============================] - 15s 2s/step - loss: 0.1406 -
binary_accuracy: 0.9318 - val_loss: 0.3958 - val_binary_accuracy: 0.8643 - lr:
0.0097
Epoch 6/500
9/9 [==============================] - 15s 2s/step - loss: 0.1351 -
binary_accuracy: 0.9336 - val_loss: 0.3717 - val_binary_accuracy: 0.8643 - lr:
0.0096
Epoch 7/500
9/9 [==============================] - 17s 2s/step - loss: 0.1091 -
binary_accuracy: 0.9479 - val_loss: 0.5644 - val_binary_accuracy: 0.8786 - lr:
0.0096
Epoch 8/500
9/9 [==============================] - 15s 2s/step - loss: 0.1082 -
binary_accuracy: 0.9551 - val_loss: 0.8362 - val_binary_accuracy: 0.8571 - lr:
0.0095
Epoch 9/500
9/9 [==============================] - 15s 2s/step - loss: 0.1371 -
binary_accuracy: 0.9461 - val_loss: 1.4883 - val_binary_accuracy: 0.8357 - lr:
0.0095
Epoch 10/500
9/9 [==============================] - 15s 2s/step - loss: 0.1483 -
binary_accuracy: 0.9731 - val_loss: 0.4772 - val_binary_accuracy: 0.8929 - lr:
0.0094
Epoch 11/500
9/9 [==============================] - 15s 2s/step - loss: 0.1166 -
binary_accuracy: 0.9587 - val_loss: 1.1433 - val_binary_accuracy: 0.8643 - lr:
0.0093
```

```
Epoch 12/500
9/9 [==============================] - 15s 2s/step - loss: 0.0776 -
binary_accuracy: 0.9713 - val_loss: 1.1572 - val_binary_accuracy: 0.8500 - lr:
0.0093
Epoch 13/500
9/9 [==============================] - 15s 2s/step - loss: 0.0452 -
binary_accuracy: 0.9820 - val_loss: 1.1240 - val_binary_accuracy: 0.8643 - lr:
0.0092
Epoch 14/500
9/9 [==============================] - 15s 2s/step - loss: 0.0419 -
binary_accuracy: 0.9856 - val_loss: 1.7421 - val_binary_accuracy: 0.8571 - lr:
0.0092
Epoch 15/500
9/9 [==============================] - 15s 2s/step - loss: 0.0409 -
binary_accuracy: 0.9820 - val_loss: 1.1774 - val_binary_accuracy: 0.8714 - lr:
0.0091
Epoch 16/500
9/9 [==============================] - 15s 2s/step - loss: 0.0265 -
binary_accuracy: 0.9892 - val_loss: 1.8678 - val_binary_accuracy: 0.8786 - lr:
0.0091
Epoch 17/500
9/9 [==============================] - 15s 2s/step - loss: 0.0178 -
binary_accuracy: 0.9964 - val_loss: 2.4453 - val_binary_accuracy: 0.8643 - lr:
0.0090
Epoch 18/500
9/9 [==============================] - 15s 2s/step - loss: 0.0127 -
binary_accuracy: 0.9964 - val_loss: 1.7314 - val_binary_accuracy: 0.8714 - lr:
0.0089
Epoch 19/500
9/9 [==============================] - 14s 2s/step - loss: 0.0089 -
binary_accuracy: 0.9982 - val_loss: 2.7436 - val_binary_accuracy: 0.8643 - lr:
0.0089
Epoch 20/500
9/9 [==============================] - 15s 2s/step - loss: 0.0233 -
binary_accuracy: 0.9946 - val_loss: 2.1677 - val_binary_accuracy: 0.8714 - lr:
0.0088
Epoch 21/500
9/9 [==============================] - 15s 2s/step - loss: 0.0381 -
binary_accuracy: 0.9856 - val_loss: 2.6391 - val_binary_accuracy: 0.8643 - lr:
0.0088
Epoch 22/500
9/9 [==============================] - 15s 2s/step - loss: 0.0188 -
binary_accuracy: 0.9964 - val_loss: 1.6574 - val_binary_accuracy: 0.8786 - lr:
0.0087
Epoch 23/500
9/9 [==============================] - 15s 2s/step - loss: 0.0422 -
binary_accuracy: 0.9928 - val_loss: 3.2221 - val_binary_accuracy: 0.8643 - lr:
0.0087
```

```
Epoch 24/500
9/9 [==============================] - 15s 2s/step - loss: 0.0109 -
binary_accuracy: 0.9982 - val_loss: 2.4559 - val_binary_accuracy: 0.8714 - lr:
0.0086
Epoch 25/500
9/9 [==============================] - 15s 2s/step - loss: 0.0496 -
binary_accuracy: 0.9910 - val_loss: 3.0288 - val_binary_accuracy: 0.8643 - lr:
0.0086
Epoch 26/500
9/9 [==============================] - 15s 2s/step - loss: 0.0138 -
binary_accuracy: 0.9946 - val_loss: 2.8882 - val_binary_accuracy: 0.8429 - lr:
0.0085
Epoch 27/500
9/9 [==============================] - 15s 2s/step - loss: 0.0275 -
binary_accuracy: 0.9982 - val_loss: 1.6839 - val_binary_accuracy: 0.8714 - lr:
0.0085
Epoch 28/500
9/9 [==============================] - 15s 2s/step - loss: 0.0662 -
binary_accuracy: 0.9838 - val_loss: 1.8664 - val_binary_accuracy: 0.8571 - lr:
0.0084
Epoch 29/500
9/9 [==============================] - 15s 2s/step - loss: 0.0242 -
binary_accuracy: 0.9874 - val_loss: 4.0791 - val_binary_accuracy: 0.8429 - lr:
0.0084
Epoch 30/500
9/9 [==============================] - 15s 2s/step - loss: 0.0212 -
binary_accuracy: 0.9946 - val_loss: 2.1068 - val_binary_accuracy: 0.8643 - lr:
0.0083
Epoch 31/500
9/9 [==============================] - 15s 2s/step - loss: 0.0126 -
binary_accuracy: 0.9946 - val_loss: 2.6316 - val_binary_accuracy: 0.8714 - lr:
0.0082
Epoch 32/500
9/9 [==============================] - 15s 2s/step - loss: 0.0088 -
binary_accuracy: 1.0000 - val_loss: 3.2836 - val_binary_accuracy: 0.8714 - lr:
0.0082
Epoch 33/500
9/9 [==============================] - 15s 2s/step - loss: 0.0065 -
binary_accuracy: 1.0000 - val_loss: 8.2647 - val_binary_accuracy: 0.8571 - lr:
0.0081
Epoch 34/500
9/9 [==============================] - 15s 2s/step - loss: 0.0087 -
binary_accuracy: 0.9982 - val_loss: 5.8173 - val_binary_accuracy: 0.8786 - lr:
0.0081
Epoch 35/500
9/9 [==============================] - 15s 2s/step - loss: 0.0086 -
binary_accuracy: 0.9964 - val_loss: 6.2951 - val_binary_accuracy: 0.8786 - lr:
0.0080
```

```
Epoch 36/500
9/9 [==============================] - 15s 2s/step - loss: 0.0011 -
binary_accuracy: 1.0000 - val_loss: 7.6104 - val_binary_accuracy: 0.8786 - lr:
0.0080
Epoch 37/500
9/9 [==============================] - 15s 2s/step - loss: 0.0030 -
binary_accuracy: 0.9982 - val_loss: 8.5429 - val_binary_accuracy: 0.8714 - lr:
0.0079
Epoch 38/500
9/9 [==============================] - 15s 2s/step - loss: 0.0030 -
binary_accuracy: 0.9982 - val_loss: 8.9886 - val_binary_accuracy: 0.8571 - lr:
0.0079
Epoch 39/500
9/9 [==============================] - 15s 2s/step - loss: 0.0062 -
binary_accuracy: 0.9982 - val_loss: 6.9833 - val_binary_accuracy: 0.8571 - lr:
0.0078
Epoch 40/500
9/9 [==============================] - 15s 2s/step - loss: 0.0167 -
binary_accuracy: 0.9964 - val_loss: 9.4395 - val_binary_accuracy: 0.8786 - lr:
0.0078
Epoch 41/500
9/9 [==============================] - 15s 2s/step - loss: 0.0745 -
binary_accuracy: 0.9856 - val_loss: 1.3509 - val_binary_accuracy: 0.8714 - lr:
0.0077
Epoch 42/500
9/9 [==============================] - 15s 2s/step - loss: 0.0335 -
binary_accuracy: 0.9785 - val_loss: 1.9910 - val_binary_accuracy: 0.8714 - lr:
0.0077
Epoch 43/500
9/9 [==============================] - 15s 2s/step - loss: 0.0245 -
binary_accuracy: 0.9946 - val_loss: 2.7087 - val_binary_accuracy: 0.8714 - lr:
0.0077
Epoch 44/500
9/9 [==============================] - 15s 2s/step - loss: 0.0110 -
binary_accuracy: 1.0000 - val_loss: 2.3919 - val_binary_accuracy: 0.8714 - lr:
0.0076
Epoch 45/500
9/9 [==============================] - 15s 2s/step - loss: 0.0074 -
binary_accuracy: 1.0000 - val_loss: 2.5099 - val_binary_accuracy: 0.8714 - lr:
0.0076
Epoch 46/500
9/9 [==============================] - 15s 2s/step - loss: 0.0064 -
binary_accuracy: 0.9982 - val_loss: 4.9619 - val_binary_accuracy: 0.8714 - lr:
0.0075
Epoch 47/500
9/9 [==============================] - 15s 2s/step - loss: 0.0033 -
binary_accuracy: 1.0000 - val_loss: 7.6789 - val_binary_accuracy: 0.8714 - lr:
0.0075
```

```
Epoch 48/500
9/9 [==============================] - 15s 2s/step - loss: 0.0052 -
binary_accuracy: 0.9982 - val_loss: 8.6352 - val_binary_accuracy: 0.8714 - lr:
0.0074
Epoch 49/500
9/9 [==============================] - 15s 2s/step - loss: 0.0010 -
binary_accuracy: 1.0000 - val_loss: 8.7008 - val_binary_accuracy: 0.8714 - lr:
0.0074
Epoch 50/500
9/9 [==============================] - 15s 2s/step - loss: 0.0040 -
binary_accuracy: 1.0000 - val_loss: 8.2241 - val_binary_accuracy: 0.8643 - lr:
0.0073
Epoch 51/500
9/9 [==============================] - 14s 2s/step - loss: 0.0036 -
binary_accuracy: 1.0000 - val_loss: 7.7800 - val_binary_accuracy: 0.8571 - lr:
0.0073
Epoch 52/500
9/9 [==============================] - 14s 2s/step - loss: 0.0030 -
binary_accuracy: 1.0000 - val_loss: 7.5644 - val_binary_accuracy: 0.8643 - lr:
0.0072
Epoch 53/500
9/9 [==============================] - 14s 2s/step - loss: 0.0025 -
binary_accuracy: 0.9982 - val_loss: 7.4957 - val_binary_accuracy: 0.8643 - lr:
0.0072
Epoch 54/500
9/9 [==============================] - 15s 2s/step - loss: 0.0010 -
binary_accuracy: 1.0000 - val_loss: 7.4708 - val_binary_accuracy: 0.8643 - lr:
0.0071
Epoch 55/500
9/9 [==============================] - 14s 2s/step - loss: 0.0816 -
binary_accuracy: 0.9982 - val_loss: 8.0182 - val_binary_accuracy: 0.8643 - lr:
0.0071
Epoch 56/500
9/9 [==============================] - 15s 2s/step - loss: 0.0025 -
binary_accuracy: 1.0000 - val_loss: 4.5288 - val_binary_accuracy: 0.8643 - lr:
0.0071
Epoch 57/500
9/9 [==============================] - 15s 2s/step - loss: 0.0023 -
binary_accuracy: 1.0000 - val_loss: 4.1988 - val_binary_accuracy: 0.8643 - lr:
0.0070
Epoch 58/500
9/9 [==============================] - 15s 2s/step - loss: 0.0022 -
binary_accuracy: 1.0000 - val_loss: 4.8575 - val_binary_accuracy: 0.8643 - lr:
0.0070
Epoch 59/500
9/9 [==============================] - 15s 2s/step - loss: 0.0013 -
binary_accuracy: 1.0000 - val_loss: 5.2846 - val_binary_accuracy: 0.8714 - lr:
0.0069
```

```
Epoch 60/500
9/9 [==============================] - 15s 2s/step - loss: 0.0044 -
binary_accuracy: 1.0000 - val_loss: 5.3845 - val_binary_accuracy: 0.8714 - lr:
0.0069
Epoch 61/500
9/9 [==============================] - 15s 2s/step - loss: 0.0030 -
binary_accuracy: 1.0000 - val_loss: 5.3317 - val_binary_accuracy: 0.8643 - lr:
0.0068
Epoch 62/500
9/9 [==============================] - 15s 2s/step - loss: 0.0021 -
binary_accuracy: 1.0000 - val_loss: 5.2692 - val_binary_accuracy: 0.8643 - lr:
0.0068
Epoch 63/500
9/9 [==============================] - 15s 2s/step - loss: 0.0038 -
binary_accuracy: 0.9982 - val_loss: 4.7727 - val_binary_accuracy: 0.8643 - lr:
0.0068
Epoch 64/500
9/9 [==============================] - 15s 2s/step - loss: 0.0041 -
binary_accuracy: 1.0000 - val_loss: 4.6460 - val_binary_accuracy: 0.8714 - lr:
0.0067
Epoch 65/500
9/9 [==============================] - 15s 2s/step - loss: 0.0094 -
binary_accuracy: 0.9982 - val_loss: 5.1330 - val_binary_accuracy: 0.8714 - lr:
0.0067
Epoch 66/500
9/9 [==============================] - 15s 2s/step - loss: 0.0011 -
binary_accuracy: 1.0000 - val_loss: 3.8500 - val_binary_accuracy: 0.8714 - lr:
0.0066
Epoch 67/500
9/9 [==============================] - 15s 2s/step - loss: 0.0021 -
binary_accuracy: 1.0000 - val_loss: 3.6132 - val_binary_accuracy: 0.8500 - lr:
0.0066
Epoch 68/500
9/9 [==============================] - 15s 2s/step - loss: 0.0016 -
binary_accuracy: 1.0000 - val_loss: 3.8220 - val_binary_accuracy: 0.8643 - lr:
0.0065
Epoch 69/500
9/9 [==============================] - 15s 2s/step - loss: 0.0032 -
binary_accuracy: 1.0000 - val_loss: 3.7776 - val_binary_accuracy: 0.8571 - lr:
0.0065
Epoch 70/500
9/9 [==============================] - 15s 2s/step - loss: 0.0030 -
binary_accuracy: 1.0000 - val_loss: 3.7905 - val_binary_accuracy: 0.8500 - lr:
0.0065
Epoch 71/500
9/9 [==============================] - 15s 2s/step - loss: 0.0020 -
binary_accuracy: 1.0000 - val_loss: 3.9374 - val_binary_accuracy: 0.8571 - lr:
0.0064
```

```
Epoch 72/500
9/9 [==============================] - 15s 2s/step - loss: 2.7534e-05 -
binary_accuracy: 1.0000 - val_loss: 4.0534 - val_binary_accuracy: 0.8571 - lr:
0.0064
Epoch 73/500
9/9 [==============================] - 15s 2s/step - loss: 0.0015 -
binary_accuracy: 1.0000 - val_loss: 4.3104 - val_binary_accuracy: 0.8643 - lr:
0.0063
Epoch 74/500
9/9 [==============================] - 15s 2s/step - loss: 0.0026 -
binary_accuracy: 0.9982 - val_loss: 4.6393 - val_binary_accuracy: 0.8714 - lr:
0.0063
Epoch 75/500
9/9 [==============================] - 15s 2s/step - loss: 0.0029 -
binary_accuracy: 1.0000 - val_loss: 4.7860 - val_binary_accuracy: 0.8786 - lr:
0.0063
Epoch 76/500
9/9 [==============================] - 15s 2s/step - loss: 0.0073 -
binary_accuracy: 0.9982 - val_loss: 4.8435 - val_binary_accuracy: 0.8786 - lr:
0.0062
Epoch 77/500
9/9 [==============================] - 15s 2s/step - loss: 0.0048 -
binary_accuracy: 0.9982 - val_loss: 4.7151 - val_binary_accuracy: 0.8643 - lr:
0.0062
Epoch 78/500
9/9 [==============================] - 15s 2s/step - loss: 0.0099 -
binary_accuracy: 0.9982 - val_loss: 9.3500 - val_binary_accuracy: 0.8714 - lr:
0.0062
Epoch 79/500
9/9 [==============================] - 15s 2s/step - loss: 0.0128 -
binary_accuracy: 0.9964 - val_loss: 6.4977 - val_binary_accuracy: 0.8643 - lr:
0.0061
Epoch 80/500
9/9 [==============================] - 15s 2s/step - loss: 8.9750e-05 -
binary_accuracy: 1.0000 - val_loss: 5.6289 - val_binary_accuracy: 0.8643 - lr:
0.0061
Epoch 81/500
9/9 [==============================] - 15s 2s/step - loss: 0.1136 -
binary_accuracy: 0.9874 - val_loss: 16.7078 - val_binary_accuracy: 0.8214 - lr:
0.0060
Epoch 82/500
9/9 [==============================] - 15s 2s/step - loss: 0.0817 -
binary_accuracy: 0.9713 - val_loss: 1.0177 - val_binary_accuracy: 0.8571 - lr:
0.0060
Epoch 83/500
9/9 [==============================] - 15s 2s/step - loss: 0.0714 -
binary_accuracy: 0.9551 - val_loss: 4.2829 - val_binary_accuracy: 0.8500 - lr:
0.0060
```

```
Epoch 84/500
9/9 [==============================] - 15s 2s/step - loss: 0.0239 -
binary_accuracy: 0.9928 - val_loss: 4.0859 - val_binary_accuracy: 0.8500 - lr:
0.0059
Epoch 85/500
9/9 [==============================] - 15s 2s/step - loss: 0.0157 -
binary_accuracy: 0.9964 - val_loss: 4.1838 - val_binary_accuracy: 0.8500 - lr:
0.0059
Epoch 86/500
9/9 [==============================] - 15s 2s/step - loss: 0.0066 -
binary_accuracy: 0.9964 - val_loss: 4.2864 - val_binary_accuracy: 0.8500 - lr:
0.0059
Epoch 87/500
9/9 [==============================] - 15s 2s/step - loss: 0.0070 -
binary_accuracy: 0.9964 - val_loss: 5.4502 - val_binary_accuracy: 0.8571 - lr:
0.0058
Epoch 88/500
9/9 [==============================] - 15s 2s/step - loss: 0.0027 -
binary_accuracy: 1.0000 - val_loss: 6.2721 - val_binary_accuracy: 0.8643 - lr:
0.0058
Epoch 89/500
9/9 [==============================] - 15s 2s/step - loss: 0.0020 -
binary_accuracy: 1.0000 - val_loss: 6.3323 - val_binary_accuracy: 0.8643 - lr:
0.0057
Epoch 90/500
9/9 [==============================] - 15s 2s/step - loss: 0.0044 -
binary_accuracy: 1.0000 - val_loss: 6.1982 - val_binary_accuracy: 0.8571 - lr:
0.0057
Epoch 91/500
9/9 [==============================] - 15s 2s/step - loss: 0.0037 -
binary_accuracy: 1.0000 - val_loss: 5.9898 - val_binary_accuracy: 0.8571 - lr:
0.0057
Epoch 92/500
9/9 [==============================] - 15s 2s/step - loss: 6.9680e-04 -
binary_accuracy: 1.0000 - val_loss: 5.9117 - val_binary_accuracy: 0.8571 - lr:
0.0056
Epoch 93/500
9/9 [==============================] - 15s 2s/step - loss: 0.0028 -
binary_accuracy: 1.0000 - val_loss: 5.8163 - val_binary_accuracy: 0.8571 - lr:
0.0056
Epoch 94/500
9/9 [==============================] - 15s 2s/step - loss: 9.9694e-04 -
binary_accuracy: 1.0000 - val_loss: 5.5803 - val_binary_accuracy: 0.8643 - lr:
0.0056
Epoch 95/500
9/9 [==============================] - 15s 2s/step - loss: 0.0049 -
binary_accuracy: 0.9982 - val_loss: 5.7080 - val_binary_accuracy: 0.8643 - lr:
0.0055
```

```
Epoch 96/500
9/9 [==============================] - 15s 2s/step - loss: 2.8849e-04 -
binary_accuracy: 1.0000 - val_loss: 6.0265 - val_binary_accuracy: 0.8571 - lr:
0.0055
Epoch 97/500
9/9 [==============================] - 15s 2s/step - loss: 0.0027 -
binary_accuracy: 1.0000 - val_loss: 6.2621 - val_binary_accuracy: 0.8500 - lr:
0.0055
Epoch 98/500
9/9 [==============================] - 15s 2s/step - loss: 7.9439e-05 -
binary_accuracy: 1.0000 - val_loss: 6.3534 - val_binary_accuracy: 0.8500 - lr:
0.0054
Epoch 99/500
9/9 [==============================] - 15s 2s/step - loss: 0.0029 -
binary_accuracy: 1.0000 - val_loss: 6.3739 - val_binary_accuracy: 0.8500 - lr:
0.0054
Epoch 100/500
9/9 [==============================] - 15s 2s/step - loss: 0.0049 -
binary_accuracy: 0.9982 - val_loss: 6.3476 - val_binary_accuracy: 0.8500 - lr:
0.0054
Epoch 101/500
9/9 [==============================] - 15s 2s/step - loss: 0.0016 -
binary_accuracy: 1.0000 - val_loss: 6.2989 - val_binary_accuracy: 0.8500 - lr:
0.0053
Epoch 102/500
9/9 [==============================] - 15s 2s/step - loss: 8.2982e-04 -
binary_accuracy: 1.0000 - val_loss: 6.2271 - val_binary_accuracy: 0.8500 - lr:
0.0053
Epoch 103/500
9/9 [==============================] - 15s 2s/step - loss: 0.0010 -
binary_accuracy: 1.0000 - val_loss: 6.2363 - val_binary_accuracy: 0.8643 - lr:
0.0053
```

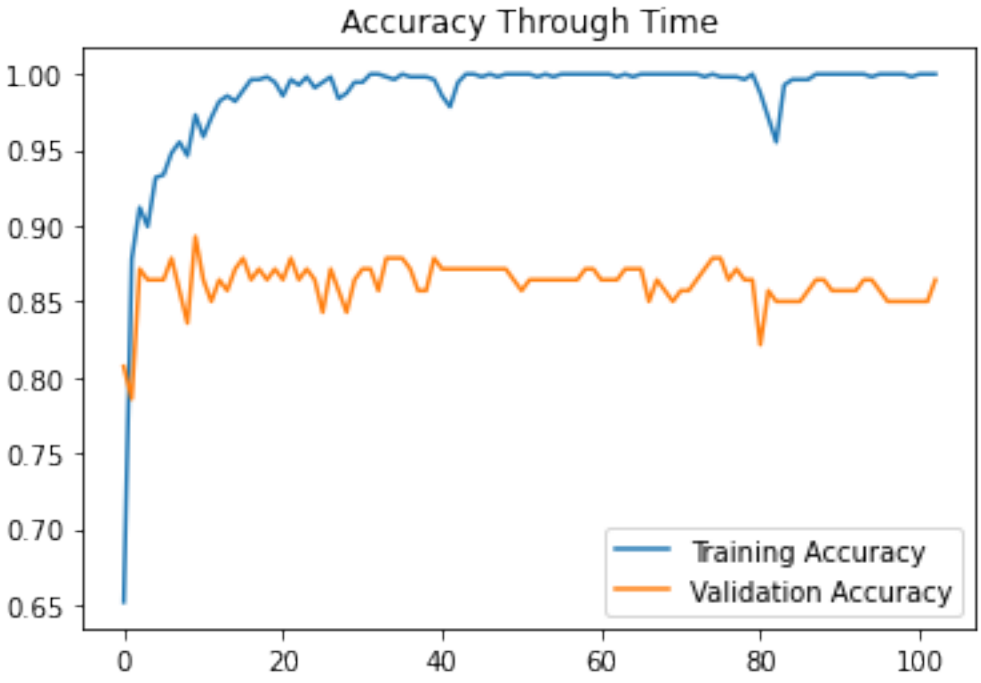
[ ]:
```
# checkpoint = ModelCheckpoint(
#     'best_model_weights.h5',
#     monitor='binary_accuracy',
#     save_best_only=True,
#     mode='max',
#     verbose=1
# )

# # Fitting model
# history = model.fit(
#     ds_train,
#     validation_data = ds_valid,
#     epochs = 10,
#     verbose=1,
```

```
#     callbacks = [checkpoint]
# )
```

1.5 Visualizing Results

[ ]:
```
# Gráfico para visualizar os erros e accuracy
history.history.keys()
#evolução do erro, azul
plt.plot(history.history['binary_accuracy'],label='Training Accuracy')
#performance da rede
plt.plot(history.history['val_binary_accuracy'],label='Validation Accuracy')
plt.title('Accuracy Through Time')
plt.legend()
```

[ ]:
```
<matplotlib.legend.Legend at 0x7fa4c84e96d0>
```



[ ]:
```
test_loss, test_acc = model.evaluate(ds_valid, verbose=1)
```

```
3/3 [==============================] - 3s 802ms/step - loss: 0.2490 -
binary_accuracy: 0.8714
```

[ ]:
```
print(f'Acurácia: {test_acc} - Perda: {test_loss}')
```

```
Acurácia: 0.8714285492897034 - Perda: 0.2490101456642151
```