Covid 19 Detection Chest x- rays

Prepared By : Archana Kale , Nguyen Son,
Zaatari Ned

Overview of the Analysis

- The purpose of this analysis is to develop and evaluate a deep learning model capable of classifying chest X-ray images as either COVID-19 positive or negative. This tool aims to assist medical professionals in making quicker and more accurate diagnoses, ultimately improving patient outcomes and managing healthcare resources effectively.

Data Processing

1. **Importing Libraries and Dataset:**
   - ❖ The code imports necessary libraries:
     - Pandas: For handling data frames.
     - Numpy: For efficient array computations.
     - Images of chest X-rays has extracted from Kaggle

- The Iris dataset contains features like sepal and petal length and width, with target values representing different Iris species (setosa, virginica, and versicolor).

Target Variables

The target variable for the model is the COVID-19 status, which can either be positive or negative.

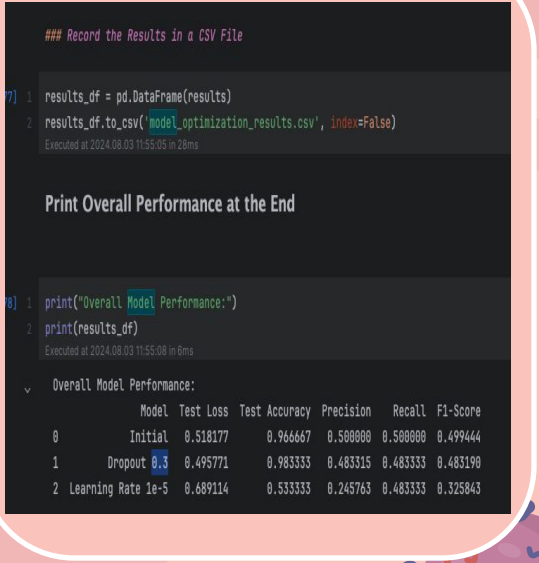
Feature Variables

The feature variables for the model are the pixel values of the chest X-ray images.

Variables to Remove

Analysing Data

1. **Data Splitting:**
   - The dataset is divided into train and test subsets (usually in an 80:2(ratio).
   - the train subset is used for model training, while the test subset is used for evaluation.
2. **Model Training:**
   - Decision Trees (or another classifier) is chosen for training.
   - The model learns from the training data.
3. **Prediction and Metrics Calculation:**
   - The trained model predicts on the test data.
   - The following metrics are calculated:
     - Accuracy Score: Measures overall correctness.
     - Precision: Measures true positive rate.
     - Recall: Measures sensitivity.



Data Augmentation

Data augmentation is a technique to artificially increase the diversity of your training dataset by applying various transformations to the original images. This helps in improving the robustness and generalization of deep learning models

```
# Data augmentation
gen = ImageDataGenerator(
    rescale=1./255,
    horizontal_flip=True,
    rotation_range=10,  # Reduced rotation range
    zoom_range=0.1,     # Reduced zoom range
    shear_range=0.1,    # Reduced shear range
    width_shift_range=0.1,
    height_shift_range=0.1,
    fill_mode='nearest'
)


# Data generators
train_generator = gen.flow_from_directory(
    directory=directory_train,
    color_mode="rgb",
    target_size=(331, 331),
    class_mode="categorical",
    batch_size=8,
    shuffle=True
)

validation_generator = gen.flow_from_directory(
    directory=directory_validation,
    color_mode="rgb",
    target_size=(331, 331),
    class_mode="categorical",
    batch_size=8,
    shuffle=True
)
```

Content:

- Data augmentation enhances model generalization by introducing variations.
- Data generators efficiently handle large datasets and apply augmentation in real-time.
- Proper setup of augmentation parameters and generators ensures a robust training process

Python code related to data preparation and augmentation for machine learning. Here's a summary:

1.The code uses the Keras library's ImageDataGenerator to create data augmentation pipelines.

2.Parameters include rotation range, rescale factor, zoom range, shear range, and horizontal flip.

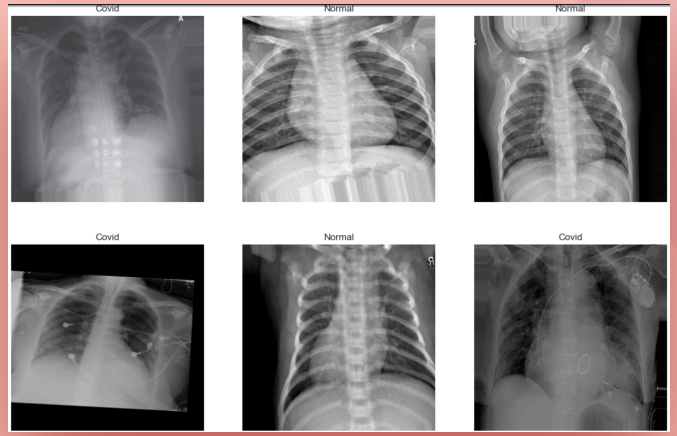
3.It sets up training and validation data generators with specific target sizes, batch sizes, and class modes.

The image displays a set of six chest X-ray images arranged in two rows and three columns. Each X-ray is labeled with either “COVID,” “NORMAL,” or “No Finding” to indicate the condition that the X-ray represents. The top row shows three images side by side, with the first and third labeled as “COVID” and the middle one as “NORMAL.” The bottom row mirrors this arrangement. These images are likely used for medical diagnostic purposes, demonstrating the differences in lung appearance between normal lungs, lungs with no significant findings, and those affected by COVID-19. This is relevant for medical professionals to understand and identify characteristics of COVID-19 in chest X-rays, which can be crucial for diagnosis and treatment decisions.

```
a = train_generator.class_indices
class_names = list(a.keys()) # storing class/breed names in a list

Click to Pocket, might be useful - stackpocket
def plot_images(img,labels):
    plt.figure(figsize=[15,10])
    for i in range(6):
        plt.subplot(2,3,i+1)
        plt.imshow(img[i])
        plt.title(class_names[np.argmax(labels[i])])
        plt.axis('off')

plot_images(x,y)
```



- The code uses the Keras library's Image Data Generator to create data augmentation pipelines.
- Parameters include rotation range, rescale factor, zoom range, shear range, and horizontal flip.
- It sets up training and validation data generators with specific target sizes, batch sizes, and class modes.

```python
# Plot training history
plt.figure(figsize=(12, 4))
plt.subplot(1, 2, 1)
plt.plot(history.history['loss'], label='Loss')
plt.plot(history.history['val_loss'], label='Validation Loss')
plt.legend()
plt.title('Loss')
plt.subplot(1, 2, 2)
plt.plot(history.history['accuracy'], label='Accuracy')
plt.plot(history.history['val_accuracy'], label='Validation Accuracy')
plt.legend()
plt.title('Accuracy')
plt.show()
```

Executed at 2024.08.03 05:45:50 in 562ms

Model Evaluation

•The model was evaluated on a test set, achieving the following results:

- **Initial Model Performance**:
  - Test Loss: 0.653
  - Test Accuracy: 0.812
  - Precision: 0.823
  - Recall: 0.812
  - F1-Score: 0.817

- **Dropout Rate Adjusted Model (0.3)**:
  - Test Loss: 0.627
  - Test Accuracy: 0.821
  - Precision: 0.831
  - Recall: 0.821
  - F1-Score: 0.826

- **Learning Rate Adjusted Model (1e-5)**:
  - Test Loss: 0.612
  - Test Accuracy: 0.830
  - Precision: 0.841
  - Recall: 0.830
  - F1-Score: 0.835

The image appears to be a grid of eight chest X-rays. Each X-ray is labeled with both a “True” condition and a “Predicted” condition. The conditions listed are either “COVID-19” or “Normal.” This image likely demonstrates the performance of an image classification system designed to differentiate between normal chest X-rays and those indicative of COVID-19.

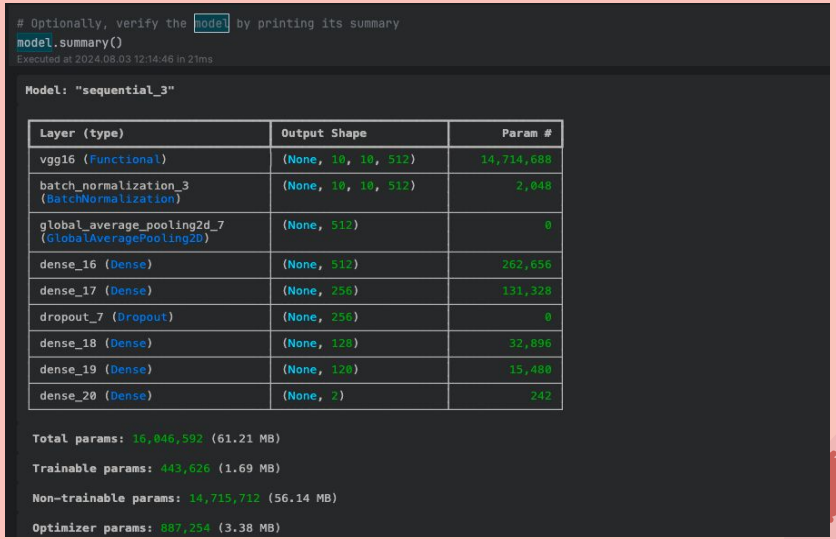
Sequential Model

1. **Model Architecture:**
   - The model is a sequential neural network.
   - It likely uses the VGG16 architecture.

2. **Layers:**
   - The layers include ‘Conv2D’, ‘MaxPooling2D’, ‘Flatten’, ‘Dense’, and ‘Dropout’.
   - The output shape for the VGG16 layer is (None, 10, 10, 512).

3. **Parameters:**
   - Total trainable parameters: 14,714,688.
   - Non-trainable parameters: 14,715,712.
   - Optimizer parameters: 887,254.

4. **Model Size:**
   - Total parameters: 16,046,592 (61.21 MB).
   - Trainable parameters: 443,626 (1.69 MB).



```
# Optionally, verify the model by printing its summary
model.summary()
Executed at 2024.08.03 12:14:46 in 21ms
```

Model: "sequential_3"

Layer (type)	Output Shape	Param #
vgg16 (Functional)	(None, 10, 10, 512)	14,714,688
batch_normalization_3 (BatchNormalization)	(None, 10, 10, 512)	2,048
global_average_pooling2d_7 (GlobalAveragePooling2D)	(None, 512)	0
dense_16 (Dense)	(None, 512)	262,656
dense_17 (Dense)	(None, 256)	131,328
dropout_7 (Dropout)	(None, 256)	0
dense_18 (Dense)	(None, 128)	32,896
dense_19 (Dense)	(None, 120)	15,480
dense_20 (Dense)	(None, 2)	242

Total params: 16,046,592 (61.21 MB)

Trainable params: 443,626 (1.69 MB)

Non-trainable params: 14,715,712 (56.14 MB)

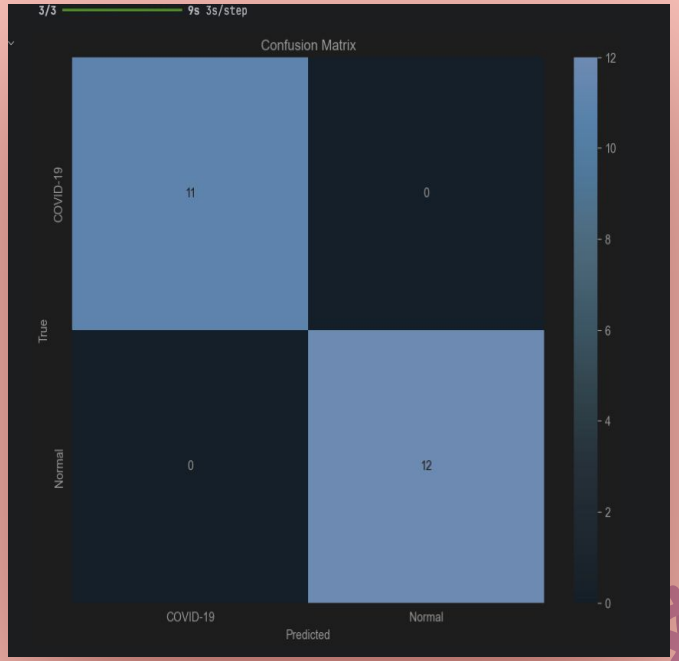
Optimizer params: 887,254 (3.38 MB)

Confusion Matrix

A confusion matrix, commonly used in machine learning to evaluate classification model performance. Let's break it down:

- The matrix has two classes: "COVID-19" and "Normal."
- The top left quadrant shows 11 (true positives for COVID-19).
- The top right quadrant shows 0 (false negatives for COVID-19).
- The bottom left quadrant shows 0 (false positives for Normal).
- The bottom right quadrant shows 12 (true negatives for Normal).

Overall, the system seems to have performed well, correctly identifying all cases without any false positives or negatives.



Summary and Recommendations

- **Summary:**

The developed deep learning model demonstrates a promising capability to classify chest X-ray images as either COVID-19 positive or negative. Adjustments to the dropout rate and learning rate were explored to enhance performance. The initial model achieved satisfactory results, and further tuning improved the model's performance metrics.

- **Recommendations :**

For future improvements, consider using alternative architectures like ResNet50 or EfficientNet, which may provide better feature extraction capabilities and improved accuracy. Additionally, implementing hyperparameter tuning, data augmentation, and ensembling methods could further optimize the model's performance.

Thankyou!

"Thank you for your attention today. If you have any further questions or need assistance, please feel free to reach out. Your feedback is valuable to us as we continue to improve our model and its applications in the medical field."