X-RAY CLASSIFICATION FOR COVID-19 DETECTION USING DEEP LEARNING

A Deep Learning Approach for Detecting Abnormalities in Chest X-rays

INTRODUCTION

Problem Statement:
Our main aim is to develop an automated system for classifying chest X-ray images into categories such as COVID-19, Pneumonia, Normal, and Viral Pneumonia. With the increasing global cases of lung diseases, particularly due to COVID-19, there is a critical need for rapid and accurate diagnostic tools.

- The project aims to classify chest X-ray images into four categories:
  - COVID
  - Lung Opacity (Pneumonia)
  - Normal
  - Viral Pneumonia

- The goal is to use Convolutional Neural Networks (CNNs) to accurately detect and classify lung conditions from X-ray images, specifically targeting diseases like COVID-19 and pneumonia.

METHODOLOGY

```
img_size = (128, 128)

# Load images and labels
for subset in subsets:
    image_dir = os.path.join(base_path, subset, "images")
    if os.path.exists(image_dir):
        image_files = os.listdir(image_dir)
        print(f"{subset}: Found {len(image_files)} images.")

        for image_name in image_files:
            image_path = os.path.join(image_dir, image_name)
            try:
                # Load and preprocess the image
                img = Image.open(image_path).convert("L")  # Convert to grayscale
                img = img.resize(img_size)
                img_array = img_to_array(img) / 255.0  # Normalize
                data.append(img_array)
                labels.append(subset)
            except Exception as e:
                print(f"Error loading image {image_path}: {e}")
    else:
        print(f"{subset}: No directory found.")
```

Dataset:

- The dataset used is the COVID-19 Radiography Dataset from Kaggle, which contains labeled chest X-ray images.
- The images are categorized into four classes:
  - COVID
  - Lung_Opacity (Pneumonia)
  - Normal
  - Viral Pneumonia
- Image Dimensions: All images are resized to 128x128 pixels for consistency and efficiency in training.

```
# Convert data and labels to numpy arrays
data = np.array(data, dtype="float32")
labels = np.array(labels)
```

```
# Encode labels
label_encoder = LabelEncoder()
labels_encoded = label_encoder.fit_transform(labels)
labels_categorical = to_categorical(labels_encoded)
```

```
# Split into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(da

```

Preprocessing:

- Grayscale Conversion: Images are converted to a single-channel (grayscale) format to simplify the data and reduce computational complexity.
- Normalization: Pixel values are normalized to the range [0, 1] to improve model convergence during training.
- 80% of the dataset is used for training, while 20% is set aside for testing the model's performance.
- Label encoding converts the four categories into numerical values, which are further transformed into one-hot encoded vectors for multi-class classification.

MODEL ARCHITECTURE

CNN VARIATION - 1:

```
model = Sequential([
    Conv2D(32, (3, 3), activation="relu", input_shape=(img_size[0], img_size[1], 1)),
    MaxPooling2D(pool_size=(2, 2)),
    Conv2D(64, (3, 3), activation="relu"),
    MaxPooling2D(pool_size=(2, 2)),
    Flatten(),
    Dense(128, activation="relu"),
    Dropout(0.5),
    Dense(len(subsets), activation="softmax")
```

1. Convolutional Layers:

- Conv2D (32 filters) with ReLU activation.
- Conv2D (64 filters) with ReLU activation.

2. Max-Pooling: Reduces spatial dimensions after each convolutional layer.

3. Flatten: Converts 2D feature maps into a 1D vector.

Dense Layers:

- 128 neurons in the dense layer.
- Dropout rate of 50% to avoid overfitting.

4. Output Layer: Softmax with 4 neurons for multi-class classification.

CNN Variation - 2

1. Convolutional Layers:
- Conv2D (16 filters), Conv2D (64 filters), Conv2D (128 filters) with ReLU activation and 'same' padding.

2. Max-Pooling: Reduces spatial dimensions after each convolution.

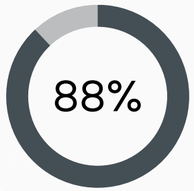
3. Dropout: Applied at varying rates (25%, 30%, 40%) to combat overfitting.

4. Dense Layers:
- 128 neurons in the first dense layer.
- 64 neurons in the second dense layer.

5. Output Layer: Softmax with 4 neurons for classification.

```
model = Sequential([
    Conv2D(16, (3, 3), activation='relu', input_shape=(128, 128, 1)),
    MaxPooling2D(pool_size=(2, 2)),

    Conv2D(64, (3, 3), activation='relu', padding='same'),
    MaxPooling2D(pool_size=(2, 2)),
    Dropout(0.25),

    Conv2D(128, (3, 3), activation='relu', padding='same'),
    MaxPooling2D(pool_size=(2, 2)),
    Dropout(0.3),

    Conv2D(128, (3, 3), activation='relu', padding='same'),
    MaxPooling2D(pool_size=(2, 2)),
    Dropout(0.4),

    Flatten(),

    Dense(128, activation='relu'),
    Dropout(0.25),

    Dense(64, activation='relu'),

    Dense(4, activation='softmax')
])
```

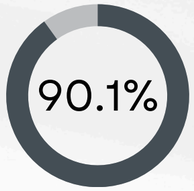
ANALYSIS
OVERVIEW



Variation 01

Test Accuracy: 87.78%

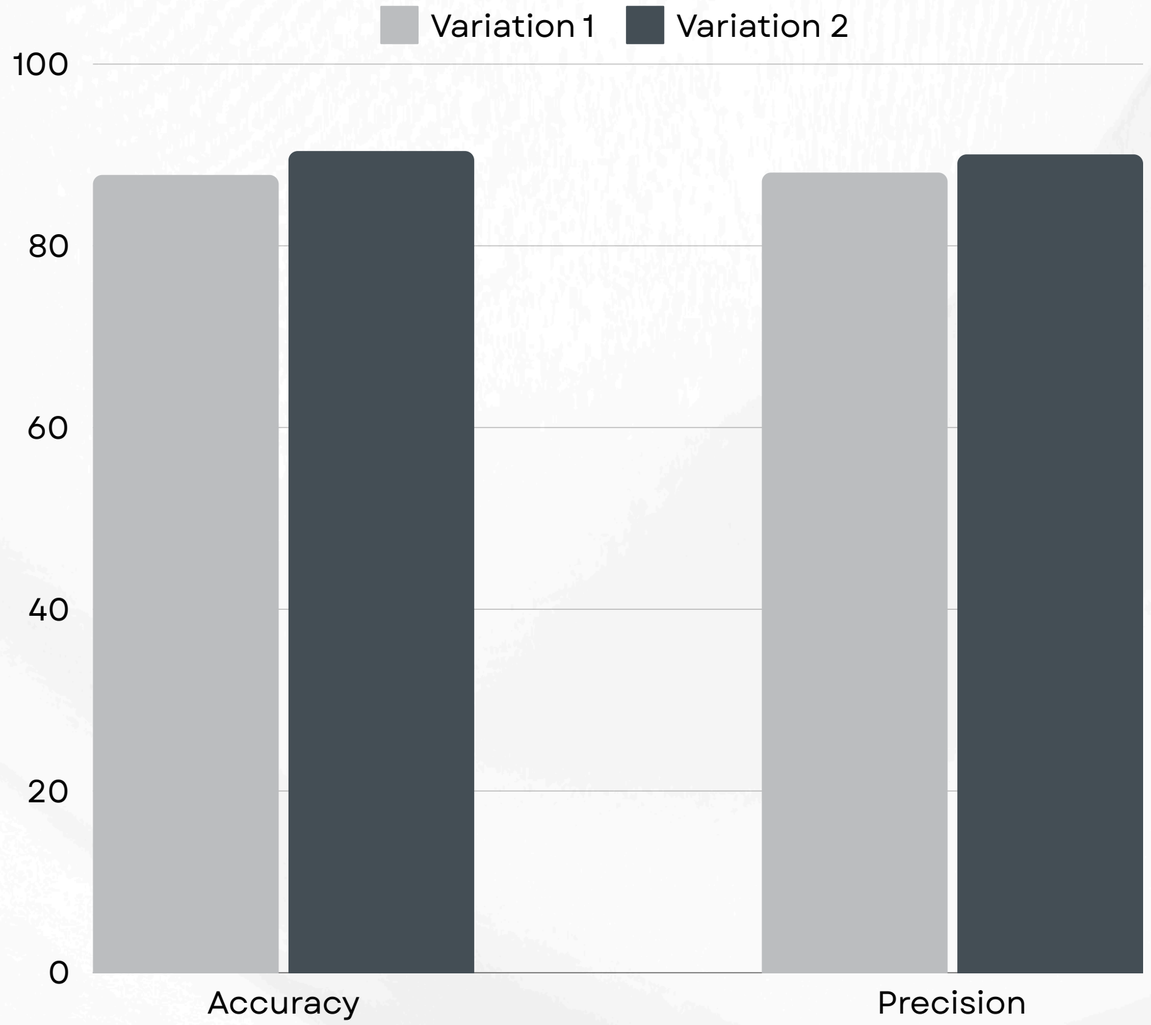
Test Precision:88.06%



Variation 02

Test Accuracy: 90.43%

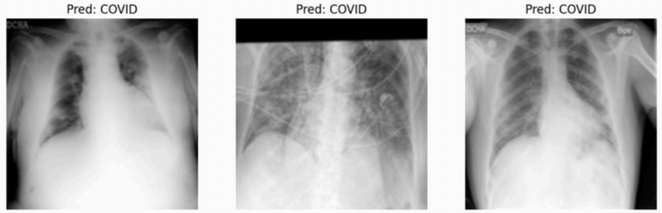
Test Precision:90.06%



Variation 2 outperforms Variation 1 with a higher test accuracy and precision across all categories.

RESULTS

```
and processing 3 images in the 'COVID' directory:
1s 584ms/step
0s 19ms/step
0s 18ms/step
ions for /root/.cache/kagglehub/datasets/tawsifurrahman/covid19-radiography-database/versions/5/COVID-19_Radiography_Dataset/COVID/images Di
```




```
and processing 3 images in the 'Lung_Opacity' directory:
0s 18ms/step
0s 17ms/step
0s 17ms/step
ons for /root/.cache/kagglehub/datasets/tawsifurrahman/covid19-radiography-database/versions/5/COVID-19_Radiography_Dataset/Lung_Opacity/images Di
```

	precision	recall	f1-score
COVID	0.93	0.93	0.93
Lung_Opacity	0.90	0.81	0.85
Normal	0.89	0.94	0.91
ral Pneumonia	0.96	0.95	0.96
accuracy			0.90
macro avg	0.92	0.91	0.91
weighted avg	0.90	0.90	0.90

```
rocessing 3 images in the 'Normal' directory:
0s 32ms/step
0s 32ms/step
0s 25ms/step
r /root/.cache/kagglehub/datasets/tawsifurrahman/covid19-radiography-database/versions/5/COVID-19_Radiography_Dataset/Normal/ima
```


```
rocessing 3 images in the 'Viral Pneumonia' directory:
0s 27ms/step
0s 31ms/step
0s 28ms/step
r /root/.cache/kagglehub/datasets/tawsifurrahman/covid19-radiography-database/versions/5/COVID-19_Radiography_Dataset/Viral Pneumonia/ima
```

	precision	recall	f1-score
COVID	0.87	0.88	0.88
Lung_Opacity	0.81	0.82	0.82
Normal	0.89	0.89	0.89
Viral Pneumonia	0.97	0.88	0.92
accuracy			0.87
macro avg	0.88	0.87	0.87
weighted avg	0.87	0.87	0.87

THANK YOU

TEAM MEMBERS:

P CHAITRA SE22UCSE193

P SRIROOP SE22UCSE190

P MADHUKAR SE22UCSE191

P AKSHAY SE22UCSE192

P TEJA REDDY SE22UCSE189