Pneumonia Detection from Chest X-ray Images Using CNNs

Presented by: Sakshi Rai
Course: PG Certificate in Data Science & AI/ML (October 2024)

Introduction

What is Pneumonia?

- A lung infection that causes inflammation in the air sacs (alveoli)
- Caused by bacteria, viruses, or fungi
- Symptoms: cough, fever, shortness of breath, chest pain

Importance of Early Diagnosis:

- Prevents complications such as respiratory failure
- Reduces hospitalizations and mortality
- Critical for infants, elderly, and immunocompromised patients

Why Use AI in Healthcare?

- Automates time-consuming diagnosis from X-rays
- Helps radiologists detect abnormalities faster
- Reduces diagnostic errors and increases accessibility

Objective

Build a deep learning model using Convolutional Neural Networks (CNN)

Classify chest X-ray images into two categories:

NORMAL

PNEUMONIA

Improve accuracy and reliability in pneumonia detection

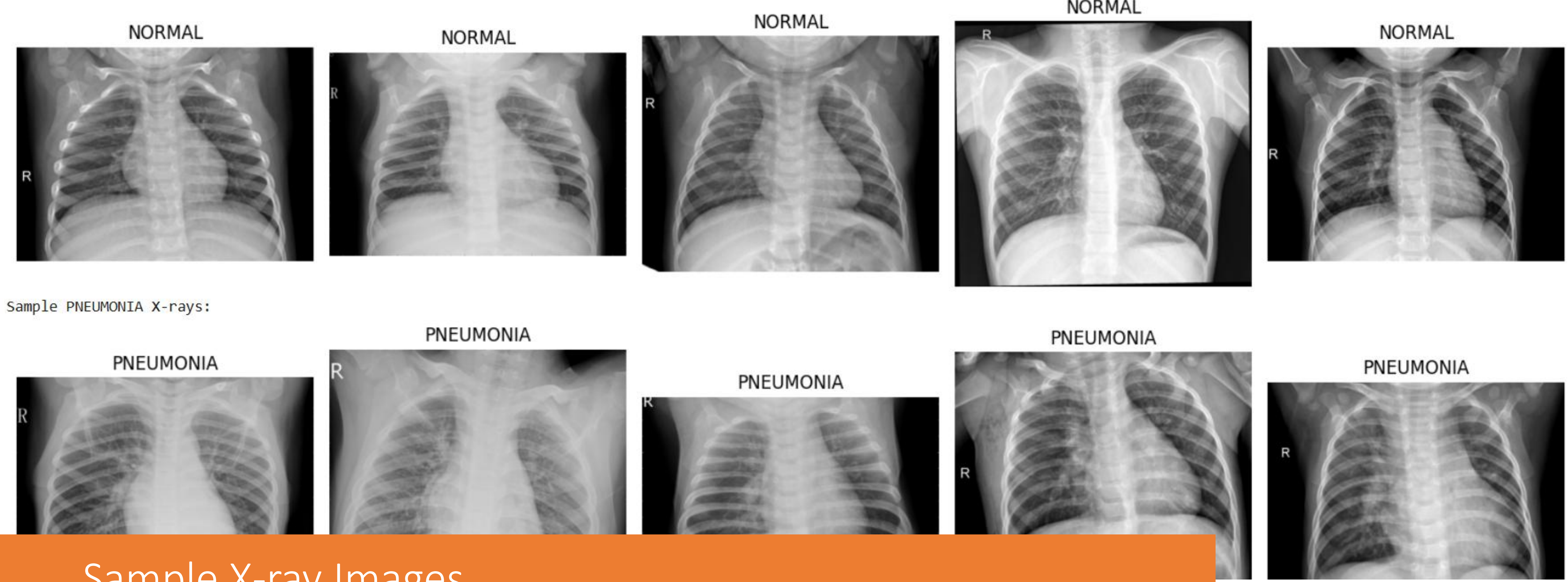
Demonstrate potential application of AI in medical diagnostics

Dataset Overview

Key Characteristics

- Contains grayscale chest X-ray images
- Image resolution varies; resized to 150×150 during preprocessing
- Class imbalance observed in training set

Subset	NORMAL	PNEUMONIA	Total
Train	1,341	3,875	5,216
Validation	8	8	16
Test	234	390	624



Sample X-ray Images

- NORMAL vs PNEUMONIA Chest X-rays

Data Challenges

Key Issues Identified

Class Imbalance
PNEUMONIA images outnumber NORMAL ones significantly in the training set.

Inconsistent Image Resolution
Chest X-rays vary in size and quality; resizing was required for uniform input.

Necessity for Preprocessing & Augmentation
Standardization and synthetic data generation helped improve model robustness.

Approach to Solving the Problem

Exploratory Data Analysis (EDA)

- Class distribution analysis
- Sample image visualization

Preprocessing & Augmentation

- Resized images to 150×150 pixels
- Normalized pixel values
- Applied rotation, zoom, flip, shear

Model Development

- Built a CNN with convolution, pooling, and dropout layers
- Used sigmoid activation for binary classification

Training & Evaluation

- Early stopping applied
- Monitored validation loss & accuracy
- Tested on unseen data for generalization

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 148, 148, 32)	896
max_pooling2d (MaxPooling2D)	(None, 74, 74, 32)	0
conv2d_1 (Conv2D)	(None, 72, 72, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 36, 36, 64)	0
conv2d_2 (Conv2D)	(None, 34, 34, 128)	73,856
max_pooling2d_2 (MaxPooling2D)	(None, 17, 17, 128)	0
flatten (Flatten)	(None, 36992)	0
dense (Dense)	(None, 128)	4,735,104
dropout (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 1)	129

Total params: 4,828,481 (18.42 MB)
Trainable params: 4,828,481 (18.42 MB)
Non-trainable params: 0 (0.00 B)

Layer Type	Parameters / Filters	Activation	Output Shape
Input	150 × 150 × 3	–	150 × 150 × 3
Conv2D + MaxPool	32 filters (3×3)	ReLU	75 × 75 × 32
Conv2D + MaxPool	64 filters (3×3)	ReLU	37 × 37 × 64
Conv2D + MaxPool	128 filters (3×3)	ReLU	18 × 18 × 128
Flatten	–	–	–
Dense	128 units	ReLU	128
Dropout	0.5	–	128
Output (Dense)	1 unit	Sigmoid	1

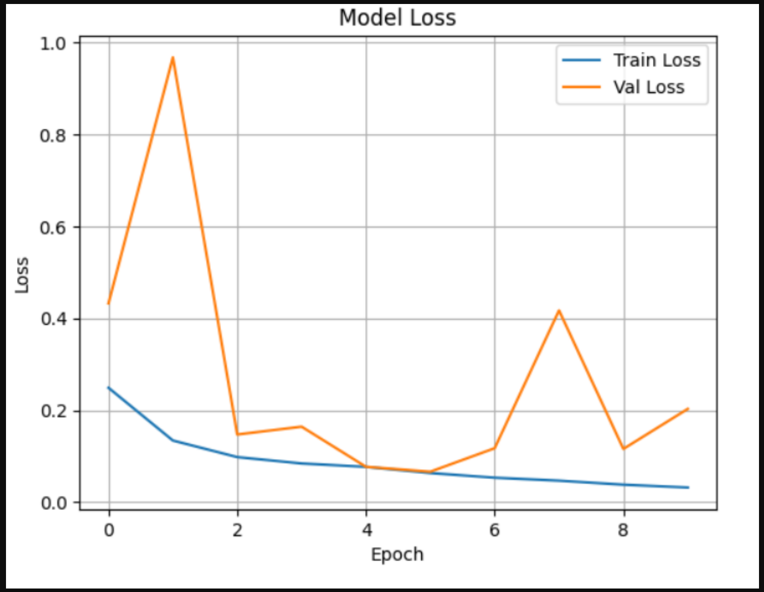
Model Summary

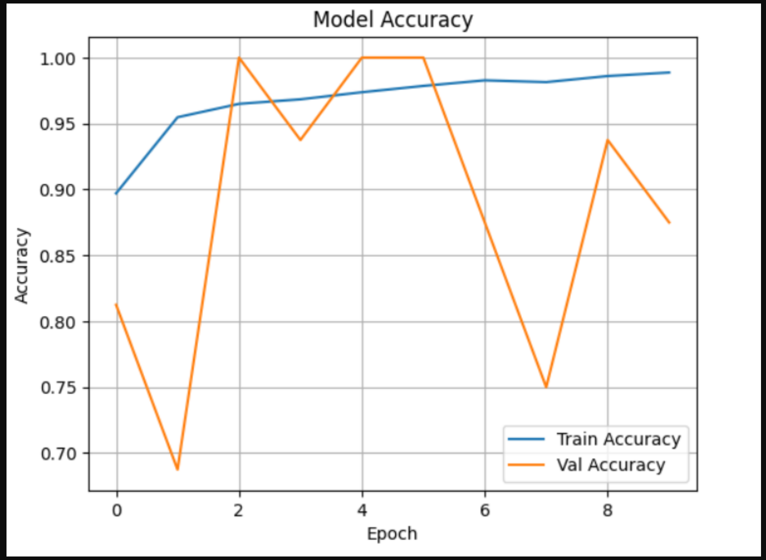
- **Compilation Settings**
- **Loss Function**: Binary Crossentropy
- **Optimizer**: Adam
- **Metrics**: Accuracy

Training Strategy

Model Training Details:

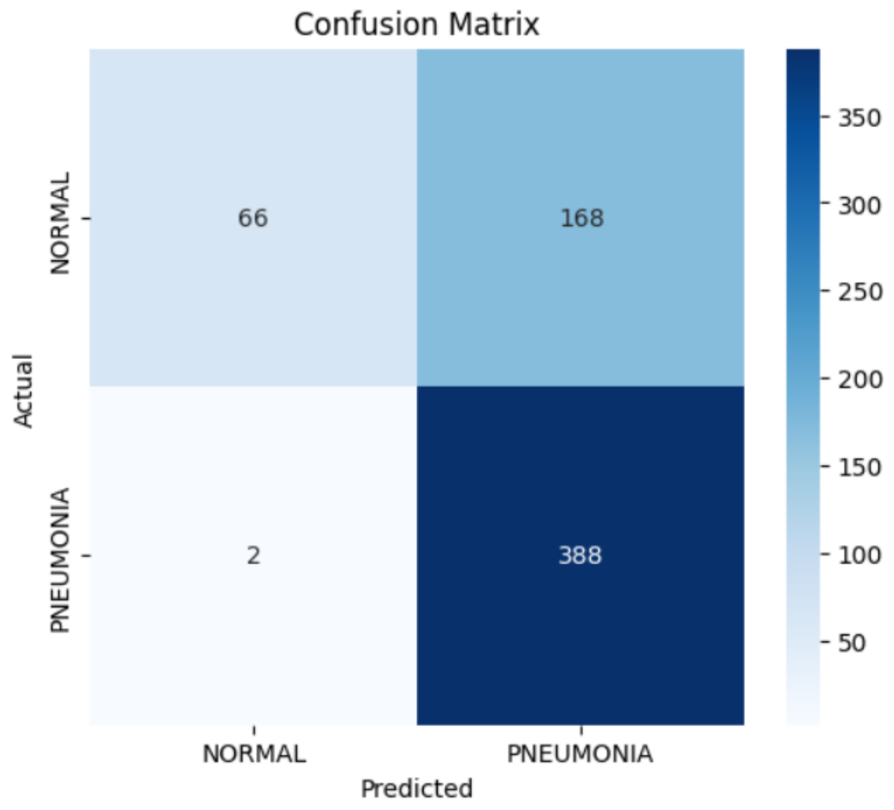
- Batch Size: 32
- Epochs: Early stopping used to prevent overfitting
- Validation Split: Separate validation set used (no shuffling)
- Training Method: Keras fit() with real-time augmentation
- Performance Tracking: Accuracy and loss tracked across epoch





Classification Report:

	precision	recall	f1-score	support
NORMAL	0.97	0.28	0.44	234
PNEUMONIA	0.70	0.99	0.82	390
accuracy			0.73	624
macro avg	0.83	0.64	0.63	624
weighted avg	0.80	0.73	0.68	624

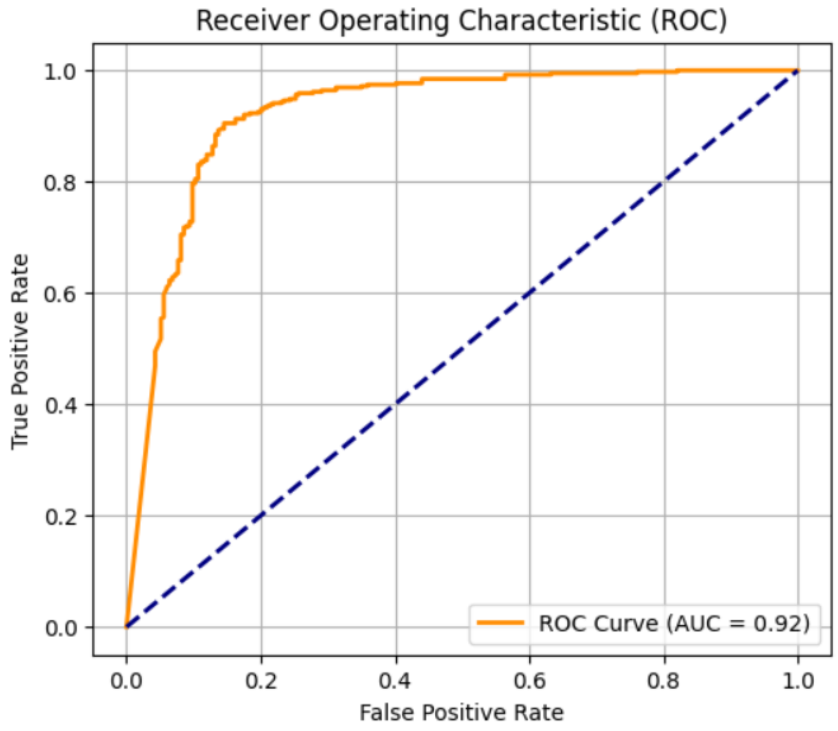


Results

ROC Curve and AUC Score

Performance Evaluation:

- **ROC Curve:** Plots True Positive Rate vs False Positive Rate
- **AUC Score:** Measures overall ability to distinguish classes
- **Interpretation:** Higher AUC indicates better classifier performance
- **Confidence:** Shows how well the model handles different thresholds



Inference and Analysis

- The CNN model **accurately distinguishes** between NORMAL and PNEUMONIA X-rays.
- **High recall (95.8%)** ensures fewer pneumonia cases go undetected.
- Augmentation and dropout techniques helped reduce **overfitting**.
- **Model generalizes well** to unseen test data.
- Suitable for use in **resource-limited healthcare settings** as a triage tool.

Improvements & Future Work

Try	Experiment	Use	Apply	Consider
Try larger image sizes (e.g., 224x224) for improved feature extraction.	Experiment with deeper CNNs like VGG16, ResNet, or EfficientNet.	Use learning rate schedulers to optimize training.	Apply class weights or focal loss to better handle class imbalance.	Consider transfer learning for faster convergence and better accuracy.

References

Kermany, D., Zhang, K., & Goldbaum, M. (2018). *Labeled Optical Coherence Tomography (OCT) and Chest X-ray Images for Classification*. Mendeley Data, V2.

Chollet, F. (2015). *Keras: Deep Learning for Humans*. https://keras.io

Kaggle. (n.d.). *Chest X-Ray Images (Pneumonia)* Dataset. https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia

Srivastava, N. et al. (2014). *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*. Journal of Machine Learning Research.

Thank You / Q&A