**Title:** Chest X-Ray Multi-Disease Classification using Deep Learning

**1. Introduction**

The detection of respiratory diseases using **chest radiography (CXR)** is a crucial task in medical diagnostics. Traditionally, diagnosis has relied heavily on radiologists’ expertise, which is time-consuming and prone to inter-observer variability.

This project leverages **Deep Learning** to automate the classification of chest X-rays into three categories:

* **Normal**
* **Pneumonia**
* **Tuberculosis**

By combining state-of-the-art **Convolutional Neural Networks (CNNs)** with curated datasets, the system achieves reliable predictions. The solution is also deployed as a **Flask-based web application**, providing a user-friendly interface for uploading X-ray images and receiving predictions.

**2. Objectives**

* To design a deep learning pipeline capable of classifying CXR images into **Normal**, **Pneumonia**, and **Tuberculosis**.
* To integrate multiple datasets into a unified training and testing framework.
* To implement a robust model using **transfer learning** with **ResNet-18**.
* To build a **web application** that allows users to upload X-rays and view automated predictions.
* To provide an extensible framework for future multi-disease classification.

**3. Datasets Used**

**1. Chest X-ray Pneumonia Dataset**

* Source: Kaggle – Chest X-Ray Images (Pneumonia)
* Contents: X-ray images categorized into **Normal** and **Pneumonia**.
* Contribution: Provides samples for pneumonia detection.

**2. Tuberculosis Chest X-ray Dataset**

* Source: Kaggle – TB Chest Radiography Database
* Contents: X-rays of patients with **Tuberculosis (TB)**.
* Contribution: Introduces an additional disease class for multi-class classification.

**Final Dataset Summary (after preprocessing)**

* **Normal:** 1341 (train), 8 (val), 234 (test)
* **Pneumonia:** 3875 (train), 8 (val), 390 (test)
* **Tuberculosis:** 560 (train), 70 (val), 70 (test)

**4. Methodology**

**Step 1: Data Preprocessing**

* Images resized to **224 × 224** pixels.
* Applied **data augmentation** (RandomHorizontalFlip, Normalization).
* Constructed CSV-based metadata linking file paths and labels.
* Split into **Train (70%)**, **Validation (15%)**, and **Test (15%)** sets.

**Step 2: Model Selection**

We employed **ResNet-18**, a deep convolutional neural network introduced by Microsoft Research, which is well-suited for image classification tasks.

* **Transfer Learning:**
  + Pre-trained on **ImageNet**.
  + Final fully-connected layer modified to output **3 classes**.
* **Loss Function:** Weighted Cross-Entropy (to handle class imbalance).
* **Optimizer:** Adam with learning rate = 0.0001.
* **Training Epochs:** 10 (can be extended).

**Step 3: Training**

* Model trained on the combined dataset.
* Validation accuracy tracked after each epoch.
* Best model checkpoint stored as model\_multiclass.pth.

**Step 4: Web Application Integration**

* Framework: **Flask (Python)**.
* Frontend: **HTML, Bootstrap 5** for styling.
* User Workflow:
  1. Upload chest X-ray.
  2. Server preprocesses image and sends it to trained model.
  3. Model predicts disease class.
  4. Result displayed with uploaded image.

**5. System Architecture**

**Workflow Diagram:**

User Upload (Web App)

↓

Flask Backend (app.py)

↓

Preprocessing (Resize, Normalize)

↓

Deep Learning Model

(ResNet-18, Transfer Learning)

↓

Output Prediction (Normal /

Pneumonia / Tuberculosis)

↓

Web Interface Result Page

**6. Results & Observations**

* **Training Accuracy:** Improved steadily across epochs.
* **Validation Accuracy:** ~85–90% depending on hyperparameters.
* **Class-wise Performance:**
  + **Normal:** High accuracy due to clear radiographic patterns.
  + **Pneumonia:** Good detection but sometimes confused with TB.
  + **Tuberculosis:** Slightly lower accuracy due to fewer samples.

**7. Key Features**

* ✅ **Multi-class classification (Normal, Pneumonia, TB)**
* ✅ **Transfer learning using ResNet-18**
* ✅ **Class imbalance handling with weighted loss**
* ✅ **Web application deployment with Flask**
* ✅ **Bootstrap-based UI for professional appearance**

**8. Limitations**

* Limited to 3 disease categories; does not generalize to all thoracic diseases.
* Performance dependent on dataset size and quality.
* May struggle with unseen image variations (different machines, noise, low resolution).

**9. Future Enhancements**

* Extend to additional diseases (e.g., COVID-19, Lung Cancer).
* Deploy model as a **REST API** or **cloud service**.
* Integrate **Grad-CAM visualizations** to show which lung areas influenced predictions.
* Improve performance using larger networks (e.g., DenseNet, EfficientNet).

**10. Conclusion**

This project demonstrates the successful application of **deep learning and transfer learning** for **chest X-ray disease classification**. By integrating multiple datasets and building a user-friendly web interface, we provide a robust and scalable solution for medical image analysis.

Such AI-based systems have the potential to assist radiologists, reduce diagnostic workload, and provide **early detection** of respiratory diseases.

**11. References**

1. Kaggle Dataset – Chest X-Ray Pneumonia: https://www.kaggle.com/paultimothymooney/chest-xray-pneumonia
2. Kaggle Dataset – Tuberculosis CXR Database: https://www.kaggle.com/tawsifurrahman/tuberculosis-chest-xray-dataset
3. Kaiming He et al., "Deep Residual Learning for Image Recognition," CVPR 2016.
4. PyTorch Documentation – https://pytorch.org/docs/stable/
5. Flask Documentation – https://flask.palletsprojects.com/