Multi-Class Classification of Chest Pathologies Using Lightweight CNNs

---

1. Problem Statement

The early and accurate detection of chest pathologies from X-ray images is a critical component in clinical diagnosis. However, manual interpretation is time-consuming, prone to inter-observer variability, and requires radiological expertise. This project aims to develop a lightweight, multi-label image classification model that can automatically identify multiple thoracic diseases, including pneumonia, tuberculosis, and other abnormalities, from chest X-ray images using convolutional neural networks and transfer learning. By leveraging the NIH ChestX-ray14 dataset and lightweight architectures like MobileNetV2 and EfficientNet, the goal is to enable real-time, interpretable diagnosis support through a deployable, resource-efficient AI system.

---

2. Objectives

The primary goals of this project are to classify chest X-ray images into multiple disease categories using a multi-label classification approach, and to leverage lightweight CNN architectures such as MobileNetV2 and EfficientNetB0 to ensure faster inference and resource efficiency. Additionally, the project aims to enhance model interpretability using Grad-CAM visualizations and deliver an interactive, user-friendly deployment through a Streamlit-based web application.

---

3. Dataset Overview

This project utilizes the NIH ChestX-ray14 dataset available on Kaggle, comprising 112,120 frontal-view chest X-ray images collected from 30,805 patients. Each image is associated with one or more labels from a total of 14 thoracic disease categories. These labels include Atelectasis, Cardiomegaly, Effusion, Infiltration, Mass, Nodule, Pneumonia, Pneumothorax, Consolidation, Edema, Emphysema, Fibrosis, Pleural Thickening, and Hernia. The data is provided in PNG format with accompanying metadata such as patient age, gender, view position, and label annotations stored in a CSV file. Train-test split files are also available to facilitate standardized evaluation.

---

4. Benefits of using lightweight CNNs over heavy models

4.1. Faster Inference Time: Lightweight models (like MobileNet, EfficientNet-B0, or SqueezeNet) have fewer parameters and require less computation. This enables real-time or near-real-time diagnosis, which is crucial in emergency situations or point-of-care systems.

4.2. Lower Computational Requirements: Can run efficiently on resource-constrained devices like mobile phones, edge devices, or low-power hospital machines. Reduces dependency on expensive GPUs or cloud computing, making AI accessible even in rural or low-resource settings.
4.3. Better for Deployment: Easier to integrate into mobile health apps, embedded systems, or portable X-ray machines. This makes them ideal for telemedicine, where bandwidth and latency are issues.
4.4. Comparable Accuracy with Proper Training: When trained properly with transfer learning and fine-tuning, lightweight models can offer accuracy close to that of heavier models.
4.5. Smaller Model Size: Model files are much smaller in size, which makes them easier to store, transfer, and deploy. This plays an important role in systems with limited memory or storage capacity.
4.6. Reduced Risk of Overfitting: Lightweight models typically have fewer parameters, which can help reduce overfitting when training on smaller medical datasets, which are common in healthcare AI.

---

5. Exploratory Data Analysis (EDA)

5.1. Objective: To analyze the NIH ChestX-ray14 dataset in detail, understand the data distribution, identify potential data quality issues, and lay the groundwork for model training. EDA is crucial to assess class imbalance, correlations between pathologies, and image characteristics for efficient preprocessing and modeling.

5.2. Dataset Overview: Total Images: 112,120 frontal-view chest X-rays
Patients: 30,805
Labels: 14 thoracic disease categories (multilabel images)

5.3. EDA Components

Image and Label Distribution: Plotting the histogram of the number of samples per disease label and highlighting the class imbalance. Counting of how many images are labeled with each class and counting how many classes each image has (multi-label density).
Co-occurrence Matrix: Understanding which diseases co-occur frequently using Heatmaps (Seaborn/Matplotlib) by building a matrix of label co-occurrence.
Class Distribution Analysis: Generating visualisations to help in decisions on thresholding and performance metrics.
Image Metadata Analysis: Analysing columns in CSV and correlating findings with class labels.
Image Quality Check: Displaying images of different view positions and analysing image resolution.

5.4. Challenges Identified

Class imbalance: Will require augmentation, class weights, or threshold tuning.
Multi-label complexity: Affects choice of activation function and loss function.
Label noise: Labels generated via NLP may have inaccuracies (~90% accurate).
Rare classes: Hernia and fibrosis may be underrepresented and require special handling.

5.5. EDA Tools

Python libraries: Pandas, Seaborn, Matplotlib, NumPy, OpenCV, Plotly
Jupyter Notebook/Kaggle Notebook for Interactive Exploration.

5.6. Expected Outcomes of EDA

Comprehensive understanding of dataset structure and quality
Identification of preprocessing strategies
Baseline visuals and stats for report/paper
Groundwork for informed model selection and evaluation

---

6. Methodology

6.1 Data Preprocessing: The preprocessing pipeline consists of resizing all images to 224x224 pixels, normalizing pixel values to a [0, 1] range, and one-hot encoding the multi-label annotations. To address class imbalance, strategies such as class weighting or oversampling will be implemented.

6.2 Model Architecture: The model architecture will employ transfer learning using MobileNetV2 and EfficientNetB0 as base models. A custom classification head with GlobalAveragePooling2D followed by a Dense layer with 14 sigmoid-activated units will be appended to each base model.

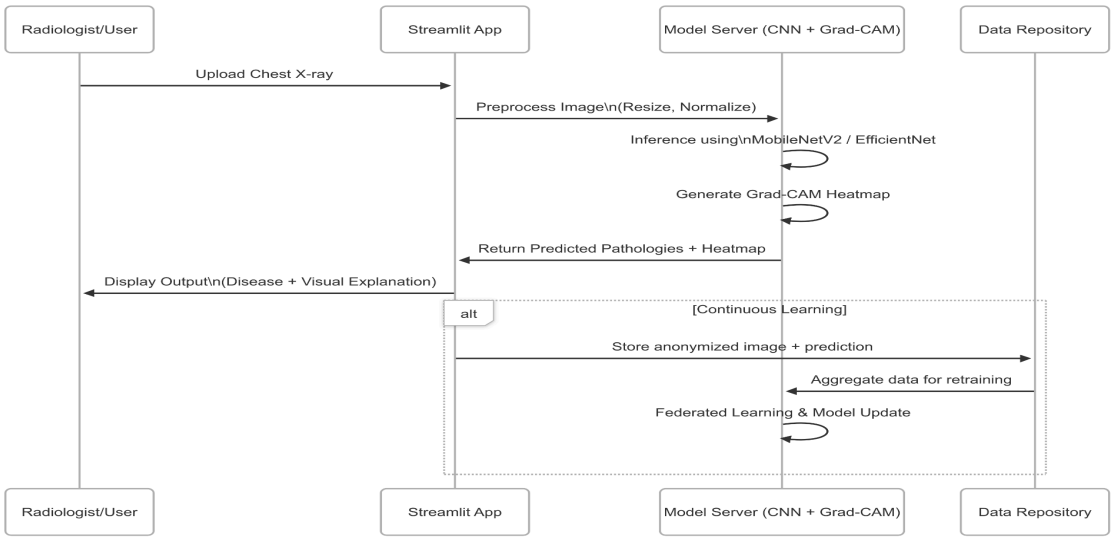
6.3 Training Strategy: Training will use the Binary Crossentropy loss function and the Adam optimizer. Performance will be evaluated using metrics including Area Under the Curve (AUC), precision, recall, and F1-score calculated per class.

6.4 Model Explainability: To enhance interpretability, Grad-CAM will be used to generate heatmaps that visually highlight the regions of input X-rays that contributed most to the model's predictions.

6.5 Model Deployment: The trained model will be deployed using a Streamlit dashboard where users can upload X-ray images, view predicted diseases, and see the associated Grad-CAM visualizations.

---

7. Technical Implementation Details



8. Expected Outcomes

At the conclusion of this project, we expect to deliver a trained, lightweight, multi-label classification model capable of identifying various chest pathologies with competitive accuracy. The model will be supported with Grad-CAM visualizations for interpretability. A fully functional web application built with Streamlit will allow users to interactively upload X-rays and receive disease predictions and attention heatmaps. Additionally, a comprehensive report and a well-documented codebase will be made available for further development and research.

---

9. Tools & Technologies

The project will be developed using Python. TensorFlow and Keras will be used for model development, while OpenCV, Matplotlib, and Seaborn will support data visualization and image processing. Grad-CAM will facilitate model interpretability. For deployment, Streamlit will be used to create an interactive web dashboard. The entire project will be executed in Jupyter Notebooks or Google Colab environments

---

10. References

- NIH ChestX-ray14 Dataset: https://www.kaggle.com/datasets/nih-chest-xrays/data

- Lakshita Agarwal, Bindu Verma. "Advanced Chest X-Ray Analysis via Transformer-Based Image Descriptors and Cross-Model Attention Mechanism." https://arxiv.org/abs/2504.16774

- Wang, Xiaosong et al. “ChestX-ray8: Hospital-scale Chest X-ray Database and Benchmarks on Weakly-Supervised Classification and Localization of Common Thorax Diseases.” CVPR 2017. https://arxiv.org/abs/1705.02315

- Zeiser, F.A., et al. “CheXReport: A transformer-based architecture to generate chest X-ray report suggestions.” https://doi.org/10.1016/j.eswa.2024.124644

- Yan, B. and Pei, M. “Clinical-BERT: Vision-language pre-training for radiograph diagnosis and reports generation.” AAAI 2022. https://arxiv.org/abs/2110.08250
