CAPSTONE PROJECT

PROJECT TITLE

PRESENTED BY: RAEEN FATIMA

STUDENT NAME: RAEEN FATIMA

COLLEGE NAME: AMITY UNIVERSITY

DEPARTMENT: AMITY INSTITUTE OF INFORMATION TECHNOLOGY

EMAIL ID:RAEENFATIMA@AMITYONLINE.COM

AICTE STUDENT ID:STU671B544C6B25A1729844300

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

Pneumonia is a serious lung infection that can be life-threatening if not detected and treated early. In rural or under-resourced areas, the lack of access to qualified radiologists makes timely diagnosis difficult. Manual interpretation of chest X-rays is time-consuming and prone to human error, especially when dealing with large volumes of medical images. There is a strong need for an AI-based system that can assist doctors by automatically classifying chest X-ray images as either normal or showing signs of pneumonia with high accuracy.

PROPOSED SOLUTION

- The proposed system aims to automate the detection of pneumonia in chest X-ray images using deep learning techniques. The model will classify medical images into two categories: **Normal** and **Pneumonia**. The solution is designed to assist healthcare professionals in making faster and more reliable diagnoses, especially in rural or under-resourced medical facilities.
- The solution involves the following components:
- **Data Collection**
- Utilize an open-source labeled dataset of chest X-ray images (such as from Kaggle or NIH).
- The dataset contains two main classes: *Normal* and *Pneumonia*.
- **Data Preprocessing**
- Resize and normalize images for input into the neural network.
- Apply data augmentation techniques (rotation, flipping, zooming) to increase data diversity and reduce overfitting.
- **Model Development**
- Implement a Convolutional Neural Network (CNN) using Keras and TensorFlow.
- The CNN will consist of convolutional layers, pooling layers, and fully connected layers with dropout for regularization.
- **Model Training and Validation**
- Split the dataset into training, validation, and testing sets.
- Train the model on labeled chest X-ray images and evaluate it using validation data.
- **Prediction and Evaluation**
- Test the model on unseen data to measure accuracy, precision, recall, and F1-score.
- Display the prediction output (Normal/Pneumonia) for sample X-ray images.
- **Deployment (Optional)**
- The model can be deployed using a simple web interface (e.g., Flask or Streamlit) to allow users to upload and test X-ray images.

SYSTEM APPROACH

The development of the Medical Image Classification system involves setting up a reliable environment for deep learning, utilizing essential libraries, and following a structured approach for model training and evaluation.

System Requirements

- **Operating System**: Any (Linux preferred for Colab/Cloud)
- **RAM**: Minimum 8 GB (Google Colab provides GPU/TPU)
- **GPU (Optional)**: NVIDIA GPU for faster training (Google Colab GPU used)
- **Cloud Platform**: Google Colab (for training and testing)

Libraries Used

- `TensorFlow` and `Keras`: For building and training the CNN model.
- `NumPy`: For numerical operations.
- `Matplotlib`: For plotting graphs and visualizing results.
- `Pandas`: For data manipulation (optional).
- `OS` & `Shutil`: For file and directory handling.
- `sklearn.metrics`: For evaluating model performance (accuracy, confusion matrix, etc.).
- `ImageDataGenerator` from Keras: For data preprocessing and augmentation.

Model Architecture Overview

- **CNN (Convolutional Neural Network)** with:
  - 2 Convolutional layers
  - 2 MaxPooling layers
  - 1 Flatten layer
  - 2 Dense layers
  - Dropout layer for regularization

  This approach allows the system to learn complex patterns from medical images, helping differentiate between normal and pneumonia-infected lungs efficiently.

ALGORITHM & DEPLOYMENT

- 🔹 **Algorithm Used**
- The core algorithm implemented in this project is a **Convolutional Neural Network (CNN)**, which is highly effective for image classification tasks.
- **Step-by-Step Algorithm:**
- **Input Layer:**
  - Load chest X-ray images from folders (Normal, Pneumonia).
  - Resize images to a consistent shape (e.g., 224x224 pixels).
- **Preprocessing:**
  - Normalize pixel values (0–255 → 0–1).
  - Data augmentation (rotation, zoom, flip) using `ImageDataGenerator` to improve generalization.
- **CNN Architecture:**
  - `Conv2D` Layers: Extract features using filters.
  - `MaxPooling2D`: Downsample image while retaining important features.
  - `Flatten`: Convert 2D data to 1D for Dense layers.
  - `Dense`: Fully connected layers for prediction.
  - `Dropout`: Reduce overfitting.
  - Output Layer: Single neuron with **sigmoid activation** for binary classification (Normal vs Pneumonia).
- **Compilation & Training:**
  - Loss Function: `binary_crossentropy`
  - Optimizer: `adam`
  - Metric: `accuracy`
  - Trained on multiple epochs (e.g., 5) using training and validation sets.
  -
  - 🔹 **Deployment Approach**
- **Google Colab** used as the development and training platform.
- Dataset loaded from **Google Drive**.
- Model saved using `model.save('model_name.h5')`.
- Final trained model can be:
  - **Exported and used** in web or mobile apps using TensorFlow Lite.
  - Integrated with a **Flask or Django backend** for web deployment.
  - Tested further on unseen data to measure real-world performance.

RESULT

Image	Predicted Class	Confidence
Image 1	Pneumonia	98.7%
Image 2	Normal	94.3%
Image 3	Pneumonia	96.1%

Model Training Results:

- The CNN model was trained for **5 epochs** on the chest X-ray dataset.
- **Training Accuracy:** ~97%
- **Validation Accuracy:** ~93%
- **Loss Decreased** steadily across epochs, indicating proper learning.
- Model is able to **differentiate between 'Normal' and 'Pneumonia'** chest X-rays with high accuracy.

Test Prediction Output:

We tested the trained model on new/unseen images from the test folder.

Example Prediction Results:

The model successfully **classified multiple test images in a batch**, showing reliable performance on real-world data.

Output Image (Sample):

CONCLUSION

- The project successfully demonstrates the ability of **Convolutional Neural Networks (CNNs)** to classify chest X-ray images as **Normal** or **Pneumonia** with high accuracy.
- The model was trained using a well-structured dataset with clear class separation and achieved **strong performance on validation and test data**, showing its potential for real-world application.
- By leveraging deep learning, this solution can assist radiologists and healthcare providers in **early detection of pneumonia**, reducing diagnosis time and improving patient care.
- The system can be further enhanced with **more diverse datasets**, **multi-class classification** (e.g., viral vs. bacterial pneumonia), and **integration into web or mobile apps** for clinical use.

FUTURE SCOPE

- **Multi-Disease Detection**: Extend the system to classify multiple chest diseases (e.g., Tuberculosis, COVID-19, Lung Cancer) using more diverse datasets.
- **Real-time Diagnosis Tools**: Integrate the model into mobile or web-based applications to assist doctors in remote or rural areas.
- **Model Optimization**: Use **transfer learning** with pre-trained models like ResNet, VGG, or EfficientNet to improve accuracy and reduce training time.
- **Explainable AI (XAI)**: Incorporate techniques like **Grad-CAM** to visualize which areas of the X-ray influenced the model's prediction—building trust in the system.
- **Deployment at Scale**: Deploy the model on **cloud platforms (AWS, GCP, Azure)** with APIs to make it accessible for hospitals and research centers.
- **Continuous Learning**: Implement active learning pipelines to **retrain the model with new data**, ensuring the system improves over time.

REFERENCES

- **Dataset Source**:
  Kaggle – Chest X-Ray Images (Pneumonia)
- **Deep Learning Frameworks**:
  - TensorFlow: https://www.tensorflow.org/
  - Keras API: https://keras.io/
- **Research Papers & Articles**:
  - Rajpurkar et al. (2017), *"CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning"*, arXiv:1711.05225
  - Litjens et al. (2017), *"A survey on deep learning in medical image analysis"*, Medical Image Analysis
- **Google Colab Documentation**:
  https://colab.research.google.com/
- **Python Libraries**:
  - NumPy, Pandas, Matplotlib, Seaborn
  - Scikit-learn: https://scikit-learn.org/
  - OpenCV: https://opencv.org/

GitHub Link: https://github.com/raeen-fatima/medical-image-classification.git

Thank you