RESEARCH ARTICLE OPEN ACCESS

Pneumonia Detection Using Flask and CNN on Chest X-Rays

KHYATI SINGH , SHRAIYA GOYAL ,SONAL TYAGI

Department of Computer Applications
APJ Abdul Kalam University
Lucknow, Uttar Pradesh

ABSTRACT

The timely diagnosis of pneumonia is critical in preventing severe respiratory complications and improving patient outcomes. This research presents the design and development of a web-based pneumonia detection system that integrates deep learning with user-friendly frontend technologies. The system employs a Convolutional Neural Network (CNN) model, trained on chest X-ray images, to accurately classify input images as either "Pneumonia" or "Normal." The backend of the application is built using Python's Flask framework, while the frontend is developed using HTML, CSS, and JavaScript, ensuring a seamless and interactive user experience. Users can register, log in, and upload chest X-ray images for real-time analysis. Based on the model's prediction, the application displays result and provides helpful links to pulmonologists and nearby hospitals. This project demonstrates the potential of combining deep learning with web technologies to build accessible, efficient, and scalable diagnostic tools, especially for resource-constrained or remote environments. The system provides a foundation for future enhancements such as integrating larger datasets, improving model accuracy, and expanding detection capabilities to other lung diseases.

I. INTRODUCTION

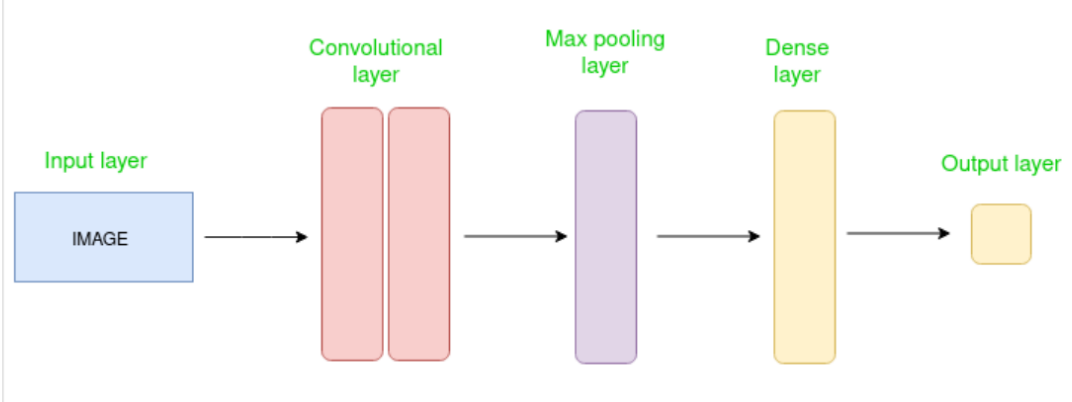
Pneumonia is an inflammatory state of the lung primarily impacting the tiny air sacs recognized as alveoli. Typical manifestations encompass a mixture of productive or dry cough, chest discomfort, fever, and respiratory distress. The seriousness of the ailment varies. Pneumonia commonly stems from viral or bacterial infections, with less frequent occurrences due to other microorganisms. Pinpointing the causative agent can pose challenges. Diagnosis frequently hinges on symptoms and physical evaluation. Chest X-rays, blood screenings, and sputum culture may aid in confirming the diagnosis. The ailment can be categorized based on its origin, whether community-acquired, hospital-acquired, or healthcare-associated pneumonia [1]. Every year, about 450 million people worldwide (making up 7% of the global population) suffer from pneumonia, leading to approximately 4 million deaths. Despite considerable progress made with the advent of antibiotics and vaccines in the 20th century, pneumonia remains a significant cause of mortality, especially in developing countries and among the elderly, infants, and individuals with chronic illnesses. Pneumonia often accelerates the demise of those already nearing death, earning it the nickname "the old man's friend." Infectious pneumonia typically presents with symptoms such as a productive cough, fever accompanied by severe chills, difficulty breathing, sharp chest pain worsened by deep breaths, and an increased respiratory rate. In older adults, confusion may be the primary indicator. In children under five, common signs include high body temperature, coughing, and rapid or labored breathing. While fever is common, it lacks specificity as it can occur in various other common illnesses and may be absent in individuals with severe diseases, malnutrition, or the elderly. Additionally, coughing may not always be present in infants under two months old. More severe symptoms in children may include cyanosis (bluish skin discoloration), reluctance to drink fluids, seizures, persistent vomiting, extreme temperature fluctuations, or decreased alertness. Treatment approaches vary depending on the underlying cause. Antibiotics are typically prescribed for suspected bacterial pneumonia, and hospitalization is often required for severe cases. Oxygen therapy may be administered if there are signs of low oxygen levels.

Symptoms	Frequency
Chest Pain	39-49%
Cough Sputum	79-91%
Fever	71-75%
Shortness of Breadth	67-75%
Fatigue	90%

Sputum	60-65%

Table 1: -Symptoms-Frequency Table

Radiology encompasses the utilization of medical imaging to diagnose ailments and direct their treatment within both human and animal bodies. Its origins lie in radiography, hence its nomenclature rooted in radiation. Presently, it encompasses a wide array of imaging techniques, including non-electromagnetic radiation-based modalities like ultrasonography and magnetic resonance imaging (MRI), alongside those utilizing electromagnetic radiation such as computed tomography (CT), fluoroscopy, and nuclear medicine, which includes positron emission tomography (PET). Interventional radiology entails conducting typically minimally invasive medical interventions with the assistance of imaging technologies mentioned earlier. [2]
CNNs consist of multiple layers like the input layer, Convolutional layer, pooling layer, and fully connected layers. Let's learn more about CNNs in detail.



Convolutional Neural Network (CNN) is an advanced version of **artificial neural networks (ANNs)**, primarily designed to extract features from grid-like matrix datasets. This is particularly useful for visual datasets such as images or videos, where data patterns play a crucial role. CNNs are widely used in **computer vision** applications due to their effectiveness in processing visual data.

II. LITERATURE REVIEW

In recent years, the intersection of healthcare and technology has led to the development of intelligent systems aimed at improving disease diagnosis and management. One of the most significant advancements has been the application of deep learning, particularly Convolutional Neural Networks (CNNs), in medical image analysis. Several studies have explored the effectiveness of CNNs in detecting pneumonia from chest X-rays, demonstrating promising accuracy and efficiency in clinical decision support systems.
Kermany et al. (2018) conducted a groundbreaking study using deep learning models trained on a large dataset of pediatric chest X-rays for pneumonia detection. Their model achieved high classification performance, reinforcing the potential of AI in assisting radiologists. Similarly, Rajpurkar et al. introduced **CheXNet**, a 121-layer DenseNet model that outperformed practicing radiologists in detecting pneumonia on the ChestX-ray14 dataset.
Various platforms have since attempted to bring these models to end-users through interactive applications. Researchers have developed mobile and web-based tools that allow users to upload medical images for instant analysis. For instance, studies by Lakhani and Sundaram (2017) showed how deep learning models can be embedded into software applications for real-time tuberculosis and pneumonia detection.
However, many existing solutions either lack accessibility, are limited to research settings, or require high computational resources. Additionally, most implementations do not provide a complete front-to-back user experience, including features like registration, image upload, and personalized results. This creates a gap for lightweight, user-friendly systems that can be deployed in resource-limited environments.

The current project builds upon this foundation by combining a CNN model with a web-based interface using Flask, HTML, CSS, and JavaScript. This approach enables users to interact with the system via a simple browser, eliminating the need for high-end systems or complex installations. By integrating healthcare intelligence with accessible technologies, this system aims to bridge the gap between AI research and real-world application in pneumonia detection.

The aim of this software is to streamline the identification of pneumonia utilizing a Convolutional Neural Network (CNN) architecture. Trained on a dataset comprising augmented images, the model becomes proficient in categorizing chest X-ray images as either 'Normal' or indicative of 'Pneumonia'. Users have the capability to register, sign in, and submit chest X-ray images for analysis. Backend routing and HTML template rendering are handled by Flask. Upon image submission, the CNN model generates a prediction regarding the presence of pneumonia. Based on this prediction, users are directed to relevant resources such as hospital and doctor contacts for further evaluation. Furthermore, users have the option to log out following the receipt of prediction results. The primary objective of this application is to offer an intuitive interface for pneumonia detection, benefiting both medical practitioners and individuals by aiding in the assessment of chest X-ray images for potential pneumonia cases. The overarching goal is to contribute to the automation of pneumonia diagnosis through image analysis.

CNN ARCHITECTURE: -

The CNN (Convolutional Neural Network) architecture stands as a cornerstone in the realm of deep learning, particularly for tasks revolving around image recognition, classification, and computer vision. CNN models are structured as feed-forward networks, incorporating essential components such as convolutional layers, pooling layers, flattening layers, and fully connected layers, each employing suitable activation functions. Convolutional layers: Convolutional layers are the building blocks of Convolutional Neural Networks (CNNs). Rooted in mathematical convolution operations, which blend two functions [3], CNNs commence by transforming input images into matrix representations. These matrices undergo convolutional filtering, where a filter traverses the input matrix, performing element-wise multiplications and aggregating the outcomes to produce a feature map. Typically, a 3 × 3 filter is employed to generate 2D feature maps for grayscale images. In the case of coloured images, depicted as 3D matrices with RGB channels as the third 7 dimension, convolutions are carried out in 3D. Through the application of multiple feature detectors on the input matrix, a layer of feature maps is generated, thereby constituting the convolutional layer. Activation functions: In the models showcased within this project, two distinct activation functions are employed: the Rectified Linear Unit (ReLU) activation function and the Sigmoid activation function. The ReLU activation function, abbreviated for rectified linear unit, is a nonlinear function known for producing zero for negative inputs and retaining positive inputs. Widely utilized in Convolutional Neural Networks (CNNs), it effectively mitigates the problem of vanishing gradients while augmenting layer nonlinearity. Variants of ReLU, such as Noisy ReLUs, Leaky ReLUs, and Parametric ReLUs, introduce additional adaptability. Its superiority over alternative activation functions lies in its computational efficiency and sparse representation. [4]. ReLU Function: f(x) = max(0,x) The sigmoid activation function serves as a mathematical tool within convolutional neural networks (CNNs) to inject non-linearity into the network's output. It compresses the output of individual neurons into a bounded range spanning from 0 to 1, rendering it well-suited for tasks involving binary classification. Defined by its characteristic S-shaped curve, inputs tending toward negative infinity converge toward 0, while those approaching positive infinity converge toward 1. This function finds particular utility in the concluding layer of a CNN for binary classification endeavours, enabling it to depict the likelihood of a specific class. Sigmoid Function: 1 $(1+e-x)$ Pooling layer: After convolutional layers, pooling layers are employed. Across all four models, max-pooling layers are utilized. These layers, with a size of 2 × 2, extract the maximum pixel intensity values from the portion of the image encompassed by the kernel. Max-pooling serves to decrease the resolution of images, thereby diminishing their dimensionality and intricacy [5]. 8 The input image undergoes processing through the convolutional and pooling layers, which are succeeded by the flattening layer and fully connected layers. This process transforms the image into a single-column format, effectively reducing computational complexity [6]. Following this, the flattened data progresses into the fully connected layer, where each node establishes connections with every node in the subsequent layer, facilitating feature extraction for prediction [7,8]. This sequential flow constitutes forward propagation. Subsequent to forward propagation, the network assesses its performance by evaluating a cost function, such as categorical crossentropy. Backpropagation then iteratively adjusts the network's parameters until optimal performance is achieved. The Adam optimization algorithm is employed across all four models to accomplish this task.

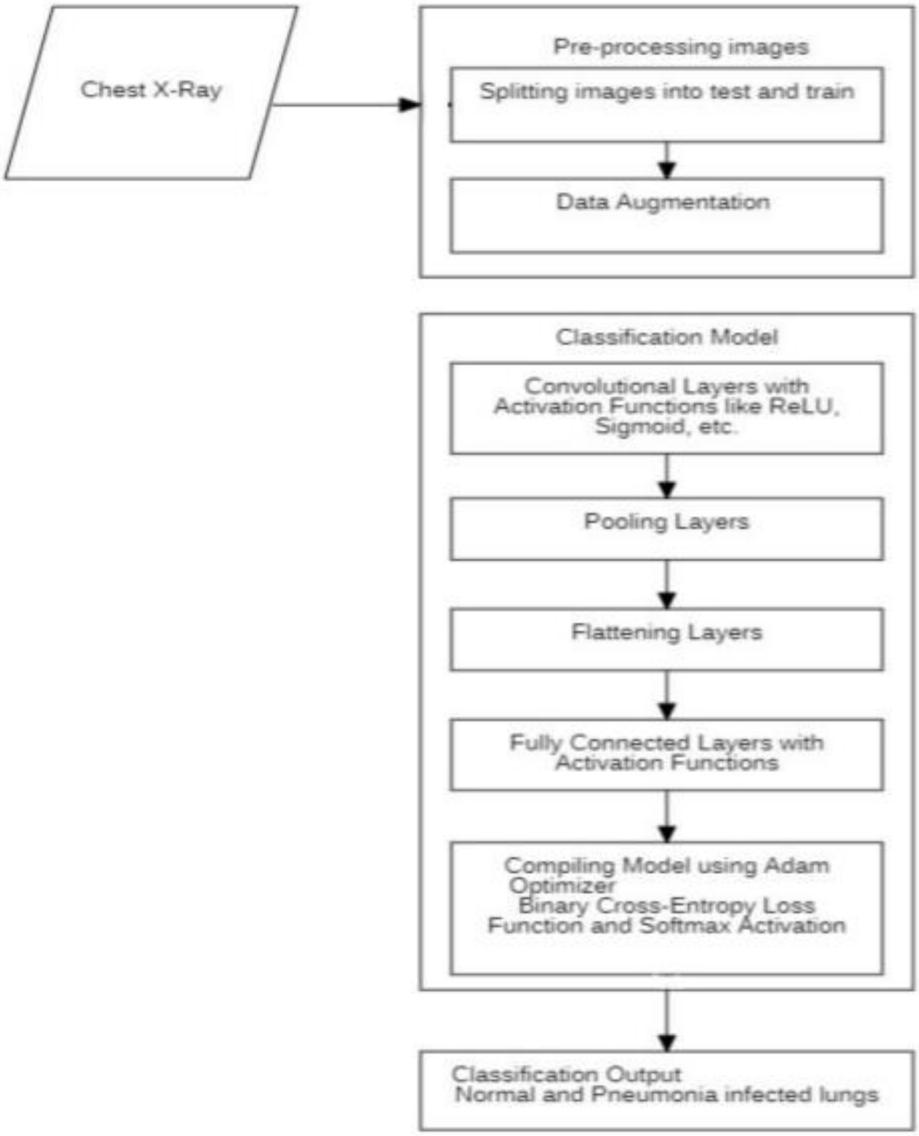


Figure 1: -CNN Architecture

CNN model needs dataset to be trained. Dataset: Chest X-Ray Images (Pneumonia) dataset of 2 GB size has been imported from Kaggle [9], with total of 5863 jpeg images split into Train, Test and Val folders each divided into category Pneumonia and Normal.

CHEST_XRAY	TEST	NORMAL
		PNEUMONIA
	TRAIN	NORMAL
		PNEUMONIA

Table 2: -Dataset Table

The following images are the sample images from the dataset used during the research.

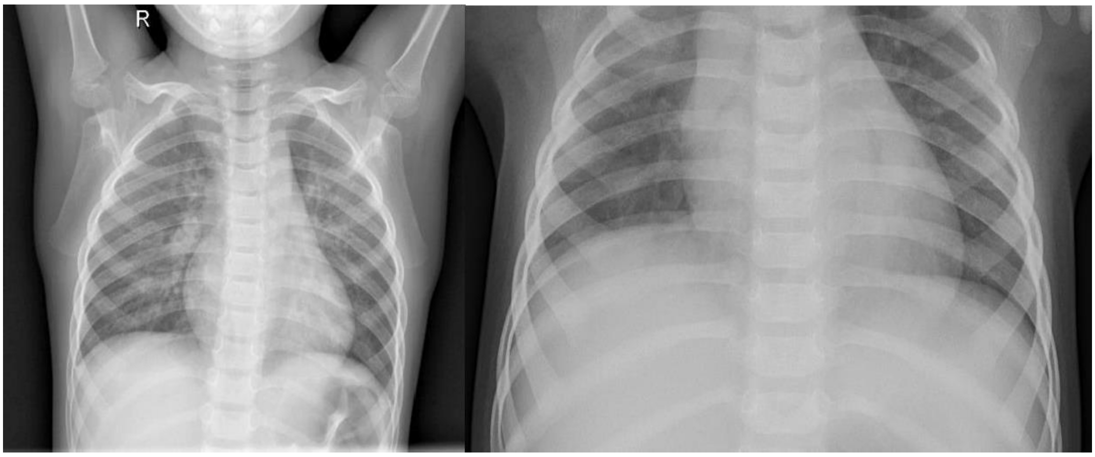


Figure 2:-X-Ray Samples Images

III. METHODOLOGY

- Dataset Collection and Preprocessing:
    - Gather a labelled dataset of chest X-ray images containing both pneumonia and normal cases.
    - Preprocess the images by resizing them to a consistent size (e.g., 64x64 pixels) and normalizing pixel values.
- Convolutional Neural Network (CNN) Model Training:
    - Design a CNN architecture for pneumonia detection using frameworks like TensorFlow and Keras.
    - Train the CNN model on the training dataset.
    - Save the trained model for future use (e.g., as "Pneumonia_Model.h5").
- Flask Web Application:
    - Install and import flask, TensorFlow, keras and NumPy
    - Develop a Flask web application for pneumonia prediction.
    - Load the trained CNN model ("Pneumonia_Model.h5") using TensorFlow and Keras.
    - Preprocess uploaded image and predicted result.
    - Create routes for:
        - Home page ("/").
        - Login page ("/login").
        - Registration page ("/register").
        - Prediction Page ("/predict").
        - Log Out ("/logout")
- HTML Templates:
    - Create HTML templates for the web application:

        - login.html: Users log in if their account does not exist.
        - register.html: New users can create their account, and then they can log in.
        - imageuploading.html: Upload a chest X-ray image for prediction.
        - resultforPneumonia.html: Display the prediction result for the pneumonia case, which will provide links for appointments with doctors.
        - resultforNormal.html: Display the prediction result for the normal case.

- CSS Styling:
    - Style the web application for a visually appealing and user-friendly interface.
    - Use CSS for layout, formatting, and animation.
- JavaScript:
    - Used for reset image function.
- Image Processing in Flask:
    - Handle image uploads in the Flask application using the "/predict" route.
    - Save the uploaded image temporarily and preprocess it for model input.
    - Use the trained CNN model to make predictions on the processed image.
    - Delete the image after prediction.
- Result Presentation:
    - Display the prediction result on either "resultforNormal.html" page or “resultforPneumonia.html” accordingly.
    - Provide a clear indication if pneumonia is detected or if the X-ray appears normal.

- Run the Flask Application:
    - Ensure the necessary dependencies are installed (NumPy, Flask, TensorFlow, Keras).
    - Run the Flask application using the "main" block in "app.py".
    - Access the web application through a web browser and test its functionality.
- User Interface Styling:
    - Enhance the user interface using CSS styles, including form layout, button design, and overall aesthetics.
- Deployment:

Deploy the Flask application on a web server for broader accessibility on http://127.0.0.1:5000.

III. IDENTIFIED GAPS

While pneumonia detection using deep learning has gained considerable momentum in recent years, several **research and implementation-level gaps** still exist. Most academic work and real-world applications in this area focus on model performance and technical accuracy, often neglecting **accessibility**, **user-friendliness**, and **deployability**. The following are the major gaps identified in the current body of work and technology implementations:

1. Lack of Practical Deployment and Accessibility

Most research studies on pneumonia detection use large, complex deep learning models trained on datasets like ChestX-ray14, LUNA16, or MIMIC-CXR. While these models demonstrate high classification accuracy, they are often deployed in restricted environments, such as high-performance computing labs or dedicated AI platforms. Very few systems make these models accessible through interactive, real-time web applications for everyday users, healthcare professionals, or rural clinics. This limits their practical utility in real-world healthcare settings.

2. Absence of Complete End-to-End User Flow

Existing solutions frequently focus on the core classification task and fail to incorporate **user flow elements**, such as user authentication, personalized experience, and intuitive interfaces for uploading and processing images. In real-world scenarios, end-users need more than a prediction—they require guidance, interaction, and smooth usability. This gap becomes even more critical when considering non-technical users who cannot operate Python notebooks or command-line tools. A complete user-centric system should include login/registration, image upload, prediction display, and next-step recommendations, all of which are often missing in many implementations.

3. Limited Integration with Frontend and Web Technologies

Most existing pneumonia detection projects are built as isolated Python scripts or Jupyter notebooks, without proper frontend integration. As a result, these tools are not scalable, cannot be hosted easily, and are not usable through a browser interface. There is a lack of systems that combine AI with HTML, CSS, and JavaScript to offer interactive and responsive interfaces. This significantly reduces the reach and adoption of these models outside of research institutions.

4. No Post-Prediction Support or Healthcare Pathways

Another critical gap in most applications is the **absence of healthcare connectivity**. Current systems stop at predicting whether the input X-ray indicates pneumonia or not. They do not guide the user with meaningful follow-up steps, such as consulting a pulmonologist, accessing nearby hospitals, or understanding treatment options. This disconnect between diagnosis and next actions undermines the effectiveness of such tools in assisting real patients or supporting doctors in triage situations.

5. Ignoring Lightweight and Resource-Conscious Design

Deep learning models are often deployed using heavy frameworks that require high-end GPU-enabled systems or cloud platforms. However, users in low-resource settings may only have access to basic computers or mobile devices. Many existing systems are not optimized for such environments. There is a lack of solutions that use minimal backend requirements (e.g., Flask), lightweight preprocessing, and locally executable models to ensure that even resource-constrained users can benefit.

6. Insufficient Consideration of Data Privacy and Security

As medical images are sensitive data, any application dealing with X-rays should consider privacy, secure storage, and temporary data handling. Many experimental projects overlook these aspects, saving data permanently or exposing user information. There is a clear gap in providing privacy-aware, session-based systems that remove images post-analysis and don't store personally identifiable information (PII), especially without encryption or authentication.

7. Lack of Multi-Device and Cross-Platform Compatibility

A majority of the existing implementations are not responsive or tested across multiple devices and screen sizes. With the increasing use of smartphones and tablets in healthcare access, especially in developing regions, there is a need for responsive web applications that adapt to various devices. Most research ignores frontend responsiveness or mobile usability.

Bridging the Gap

This project aims to bridge these gaps by creating a **fully integrated web-based pneumonia detection system** that combines a pre-trained CNN model with Flask (Python backend) and modern frontend technologies (HTML, CSS, JavaScript). The system includes user authentication, real-time image prediction, hospital/doctor suggestions post-diagnosis, and temporary file handling to maintain privacy. Its lightweight architecture ensures that it can run on local machines without needing extensive hardware or cloud subscriptions. Thus, the proposed system stands out by offering a **complete, accessible, and practical** solution that addresses both technical and usability challenges found in existing tools.

Despite significant advancements in deep learning for medical image classification, several limitations persist in existing pneumonia detection systems. Most state-of-the-art models are developed and tested in controlled research environments and often lack deployment in accessible, user-friendly platforms. These models, although accurate, are typically not integrated into web-based systems that can be used by patients or general practitioners without technical expertise.

Furthermore, many existing solutions do not provide a seamless end-to-end user experience. Features such as user registration, image uploading, real-time predictions, and personalized output are often missing or underdeveloped. As a result, users are required to interact with command-line interfaces or standalone applications, which are not ideal for practical usage in healthcare settings, especially in rural or resource-constrained regions.

Another major gap lies in the lack of post-diagnostic support. While many models can detect pneumonia, they do not assist users in the next steps after diagnosis, such as connecting with healthcare providers or accessing hospital directories. Moreover, issues like data privacy, temporary file handling, and lightweight system requirements are often overlooked in AI-based healthcare tools.

Lastly, existing literature largely focuses on the model accuracy and performance metrics, but not enough attention is given to **deployability**, **user accessibility**, and **real-world usability** of such systems. This research addresses these gaps by developing a lightweight, user-interactive, web-based pneumonia detection system that integrates a CNN model with a complete frontend-backend workflow, offering practical value and user convenience.

IV. FUTURE POSSIBILITIES OF DEEP LEARNING

The prospective advancements for the "Pneumonia Detection Using CNN" project encompass various avenues to enhance functionality, usability, and performance. Here's a breakdown of potential future developments:

• Augmented User Authentication and Security: o Implement robust password hashing methods to securely store user credentials. o Introduce supplementary authentication layers like two-factor authentication (2FA) for heightened security measures. o Regularly update security protocols to pre-emptively address potential vulnerabilities and fortify against cyber threats.

• Enhanced User Experience: o Revamp the user interface (UI) and user experience (UX) design for an aesthetically pleasing and intuitive application interface. o Incorporate real-time feedback mechanisms during image upload and prediction processes to enhance user engagement. o Integrate features such as progress indicators and loading animations to provide visual cues for ongoing processes.

• Scalability and Performance Optimization: o Fine-tune the CNN model to enhance performance and efficiency, particularly focusing on reducing inference time. o Explore optimization techniques such as model quantization or pruning to minimize model size and accelerate inference speed. o Implement caching mechanisms to store frequently accessed data, reducing computational overhead and enhancing system responsiveness.

• Integration with Cloud Services: o Investigate integration with cloud storage services like AWS S3 or Google Cloud Storage to securely store uploaded images. o Leverage cloud-based machine learning services for deploying and serving the CNN model, offering scalability and reliability advantages.

• Multi-Model Ensemble: o Explore ensemble learning strategies by aggregating predictions from multiple CNN models trained on diverse architectures or datasets. 43 o Implement a voting mechanism or weighted averaging approach to consolidate individual model predictions, bolstering accuracy and resilience.

• Continuous Model Training and Updating: o Establish a pipeline for continuous model training using newly available data to ensure the pneumonia detection model remains current. o Introduce automated model retraining protocols based on predefined criteria such as performance degradation or the availability of new labelled data.

• Internationalization and Localization: o Facilitate support for multiple languages to accommodate a diverse user base. o Incorporate localization features to tailor the application interface and content based on users' geographic location or language preferences.

• Comprehensive Reporting and Analytics: o Integrate analytics tools to monitor user interactions, usage patterns, and diagnostic outcomes, providing actionable insights. o Generate comprehensive reports and visualizations to offer stakeholders insights into the system's performance and efficacy.

• Collaboration with Healthcare Providers: o Forge partnerships with healthcare institutions to seamlessly integrate the application with existing systems, enabling streamlined patient data exchange and inter-professional collaboration. o Ensure adherence to pertinent healthcare regulations and standards like HIPAA to safeguard patient privacy and uphold data security.

• Mobile Application Development: o Develop mobile applications for iOS and Android platforms, enabling users to access the pneumonia detection system from their smartphones or tablets.

Optimize the application interface and functionality for mobile devices, ensuring a consistent user experience across diverse platforms. By pursuing these potential enhancements and expansions, the project can evolve into a more resilient, feature-rich, and impactful solution for pneumonia detection and diagnosis.

Improved Medical Diagnostics: -

In the future, deep learning models will not only detect diseases like pneumonia but also **predict disease progression**, **recommend treatments**, and even identify early-stage illnesses before symptoms appear. With enhanced access to diverse and high-quality medical datasets, models can be trained to handle complex conditions like cancer, Alzheimer's, and rare genetic disorders with greater accuracy and fewer false positives.

Multi-Disease Detection Systems: -

Rather than focusing on one specific disease per model (e.g., only pneumonia), future systems may use **multi-label classification** to detect a variety of conditions from a single image or dataset. A single chest X-ray, for instance, could be analyzed for pneumonia, tuberculosis, lung cancer, and COVID-19—improving efficiency in diagnosis.

Integration with Wearables and IoT Devices: -

Deep learning can be extended beyond static image analysis by integrating with **real-time data from wearables**, such as heart rate monitors, oxygen saturation sensors, or smartwatches. This would allow for **continuous health monitoring** and predictive alerts, enabling proactive healthcare and reducing emergency cases.

Personalized Healthcare Systems: -

Using patient history, genetics, lifestyle data, and medical images, deep learning can drive **personalized medicine**, where treatments and diagnoses are tailored for each individual. Future systems may suggest highly specific care plans that adapt in real-time based on a patient's response to treatment.

Automated Medical Reporting and Explanation: -

Future deep learning systems may not only predict a condition but also generate **comprehensive medical reports** in human language, explaining the rationale behind the prediction. These explanations would increase **trust** and **transparency** in AI-driven systems, making them more acceptable to clinicians and patients.

Cross-Modal Learning and Fusion of Data Types: -

Deep learning models may evolve to combine **multiple data types**—such as X-rays, lab test results, doctor's notes, and MRI scans—into a unified diagnosis model. This fusion of textual, visual, and numerical data could significantly improve diagnostic accuracy and provide a more holistic view of the patient's health.

Edge Deployment and Offline Capability: -

To support rural and resource-constrained areas, future deep learning systems will increasingly run on **edge devices**, allowing predictions without needing continuous internet access. These systems will be lightweight, fast, and cost-effective, enabling **inclusive healthcare access** around the globe.

Ethical, Transparent, and Bias-Free AI: -

Ongoing research is addressing the challenges of bias, fairness, and explainability in AI models. The future of deep learning will see the rise of models that are **auditable, ethically trained**, and free from racial, gender, or socioeconomic bias, making them safer for use in global healthcare.

IV. CONCLUSION

In this project, we have successfully developed a Pneumonia Detection system using Convolutional Neural Networks (CNN) integrated with a Flask web application. The system allows users to register, login, upload chest X-ray images, and receive predictions on whether the images indicate pneumonia or not. The CNN model used for pneumonia detection has been trained on a dataset of chest X-ray images, enabling it to effectively classify images into pneumonia-positive or pneumonianegative categories. Through the utilization of TensorFlow and Keras libraries, we were able to load and utilize the pre-trained CNN model within the Flask framework. Moreover, the integration of local storage (within Python) for user data management ensures a seamless experience for users, allowing them to securely register, login, and access the prediction service. The web interface provides a user-friendly experience, allowing users to easily interact with the system by uploading their chest X-ray images and receiving prompt predictions. Furthermore, for cases where pneumonia is detected, the system provides additional resources such as links to best hospitals and doctors specializing in pulmonology, thereby facilitating further medical assistance for users. Overall, this project demonstrates the practical application of deep learning techniques in healthcare, particularly in the early detection of pneumonia, which can significantly aid medical professionals in timely diagnosis and treatment. Additionally, the incorporation of web technologies enhances accessibility and usability, making the system beneficial for both healthcare professionals and individuals seeking medical guidance.

IV. REFERENCES

1. From Wikipedia - https://en.wikipedia.org/wiki/Radiology#:~:text=Radiology%20(/,those%20mentioned%20above.

2. From Wikipedia - https://en.wikipedia.org/wiki/Pneumonia.

3. Bi, X., Li, S., Xiao, B., Li, Y., Wang, G., Ma, X.: Computer aided Alzheimer's disease diagnosis by an unsupervised deep learning technology. Neurocomputing (2019).

4. Eckle, K., Schmidt-Hieber, J.: A comparison of deep networks with ReLU activation function and linear spline-type methods. Neural Netw. 110, 232–242 (2019).

5. Zheng, Y., Iwana, B.K., Uchida, S.: Mining the displacement of max-pooling for text recognition. Pattern Recogn. 93, 558–569 (2019).

6. Bhumika, P.S.S.S., Nayyar, P.A.: A review paper on algorithms used for text classification. Int. J. Appl. Innov. Eng. Manage. 3(2), 90–99 (2013).

7. Kumar, A., Sangwan, S.R., Arora, A., Nayyar, A., Abdel-Basset, M.: Sarcasm detection using soft attention-based bidirectional long short-term memory model with convolution network. IEEE Access 7, 23319–23328 (2019).

8. Saeed, F., Paul, A., Karthigaikumar, P., Nayyar, A.: Convolutional neural network based early fire detection. In: Multimedia Tools and Applications, pp. 1–17 (2019).

9. From Kaggle - https://www.kaggle.com/datasets/paultimothymooney/chest-xraypneumonia.