People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University of Abdelhamid Mehri – Constantine 2

Faculty of New Technologies of Information and Communication (NTIC)

Department of Fundamental Computing and its Applications (IFA)

BACHELOR'S DISSERTATION

to obtain the diploma of Bachelor degree in Computer Science

Option: Computer Sciences (SCI)

Laboratory Application for Patient Services and AI-Driven Scan Classification

Realized by:

Bengrine Malik Zakaria

Ait Kaki Lyes

Under supervision of:

Dr. Bouchra Guelib

May 2025

بسم الله الرحمن الرحيم

Acknowledgments

First and foremost we want to express our deepest gratitude to all those who contributed towards making it possible for our final project of studies to be a success as well as towards composing this thesis.It was quite the journey for us and we certainly couldn't have done it without many people out there who assisted and contributed towards making things easier for us. Most importantly above all else, we want to convey gratitude to the supervisor of our project Dr. Bouchra Guelib, whose advice was in no way indispensable towards making it possible to complete this project. For as ever she was at hand to advise as otherwise it is not possible for this project to attain a level of quality as high as it is today. We also want to express a word of appreciation towards all those who taught us throughout these three years of academic studies as well as towards all those of our classmates and colleagues who gave their feedback as well as helpful support towards making things easy for us.

Dedication

This work is dedicated with love and gratitude to our wonderful and lovable parents whose support and sacrifices have been invaluable and whose encouragement has been instrumental in helping us achieve our goals. Their encouragement gave us strength and inspiration to strive for excellence. There is a special dedication also to my loving aunt, whose warmth and helpfulness and encouragement and constant inspiration have been a solace and comfort to all of us. This success is as yours as it is ours.

ملخص

الوفيات سنويًا، خاصة بين الفئات السكانية الضعيفة. يُعد الكشف المبكر والدقيق أمرًا بالغ الأهمية للعلاج الفعّال. ومع ذلك، فإن تفسير صور الأشعة السينية للصدر يدويًا يستغرق عادة وقتًا طويلاً ويفتقر إلى الدقة، مما قد يؤدي إلى أخطاء قد تكون قاتلة للمريض.

لمعالجة هذه المشكلة، تهدف هذه الدراسة إلى أتمتة عملية الكشف وتشخيص الأمراض الرئوية من خلال نموذج تعلُّم عميق. تُعدّ صور الأشعة السينية للصدر أداة أساسية، وهي محور هذه الدراسة، حيث يتمثل الهدف في تحسين دقة التشخيص باستخدام نموذج قائم على بنية الشبكات العصبية الالتفافية (ث).

لتدريب نموذجنا، استخدمنا مجموعة بيانات تتكون من ٢٢،١٣١ صورة أشعة سينية للصدر، مع دمج طبقات كثيفة لبناء نموذج تصنيف متعدد الفئات عالي الأداء.

نستخدم تقنية اينتگرتد جرد▮نتس لفهم قرارات النموذج بشكل أفضل. ومع وجود ٩.٢٢ مليون من المعاملات القابلة للتعلم، يحقق نموذجنا دقة تصل إلى ~ ٩٤٪.

تُبرز هذه الطريقة المناطق الرئيسية في الصور التي تؤثر على التنبؤات، مما يزيد من الثقة في نتائج التشخيص.

تمثل طريقتنا تقدمًا في تحليل الصور الطبية من خلال تقديم نموذج ث موثوق وقابل للتفسير للكشف عن الالتهاب الرئوي، ومناسب للاستخدام السريري.

الكلمات المفتاحية: (الالتهاب الرئوي، أشعة سينية للصدر، ث ، تصنيف متعدد الفئات، تعلُّم عميق، تحليل الصور الطبية)

Abstract

Pneumonic diseases have emerged as a serious health problem these days and a challenge to diagnose, resulting in thousands of deaths annually, especially among high-risk groups. Accurate and early detection is vital to treat them successfully. But manual readings of chest X-rays always

require a great deal of time and hence lack accuracy, which can result in a mistake that can prove to be lethal for the patient.
For resolving such a problem, in this research, an automated detection and diagnosis of pneumonic diseases have been attempted by a deep learning model. The most important tool that forms the core of this research is a chest X-ray, with which higher diagnostic accuracy is achieved by employing a Convolutional Neural Network (CNN) architecture-based model.
A dataset of 22,131 chest X-ray images was utilized in training the model, combining dense layers in order to develop a high-performing multi-class classifier.
Integrated Gradients are employed to better interpret the decisions of the model. The model, with a total of 22.9 million trainable parameters, achieves a diagnostic accuracy of 94%. This approach highlights salient areas in images which impact prediction, thus raising confidence in diagnostic outcomes.
This method is a step towards improved medical image analysis since it presents a robust, interpretable, and deployable CNN for pneumonia detection in a clinical setting.

Keywords: (Pneumonic , chest-XRay , CNN , multi-class ,Deep learning , medical image analysis)

Résumé

Les maladies pneumoniques sont devenues un problème de santé majeur de nos jours et représentent un défi en matière de détection, causant des milliers de décès chaque année, en particulier parmi les populations vulnérables. Une détection précoce et précise est cruciale pour un traitement efficace. Cependant, l'interprétation manuelle des radiographies thoraciques prend généralement beaucoup de temps et manque de précision, ce qui peut entraîner des erreurs potentiellement fatales pour le patient.
Pour résoudre ce problème, cette recherche vise à automatiser la détection et le diagnostic des maladies pneumoniques à l'aide d'un modèle de deep learning. Les radiographies thoraciques, étant un outil clé, sont au centre de cette étude, dont l'objectif est d'améliorer la précision du diagnostic

grâce à un modèle basé sur une architecture de réseau de neurones convolutifs (CNN).
Pour entraîner le modèle, un jeu de données a été utilisé qui comprend 22 131 images de radiographies thoraciques, en intégrant des couches denses afin de construire un modèle de classification multi-classes haute performance. Les Integrated Grandients ont été utilisées pour mieux comprendre les décisions du modèle. Avec 22,9 millions de paramètres entraînables, notre modèle atteint une précision de 94%.
Cette méthode met en évidence les régions clés des images influençant les prédictions, augmentant ainsi la confiance dans les résultats diagnostiques. Notre approche représente une avancée dans l'analyse d'images médicales en fournissant un modèle CNN fiable et interprétable pour la détection de la pneumonie, adapté à une utilisation clinique.

Mots clés : (Pneumonie, radiographie thoracique, CNN, multi-classes, deep learning, analyse d'images médicales)

Table of Contents

List of Figures

List of Tables

General Introduction

Project Background

Lung is a vital organ with one of the most important functionalities in the human system playing a pivotal role in bringing O2 in and CO2 out .Nowadays most of the population encounters diseases that affect the lungs such as CoVid-19 ,Pneumonia and TB resulting in airway diseases and problems in the air circulation within the lungs and potentially causing death .In 2021 there are 2.2 million reported deaths caused by Pneumonia from which roughly 25 to 26% are children under the age of five according to the Global Burden of Disease ,and according to UNICEF recently in 2024 there are reports of over 700K deaths of little children under five equating to one child every 43 seconds.

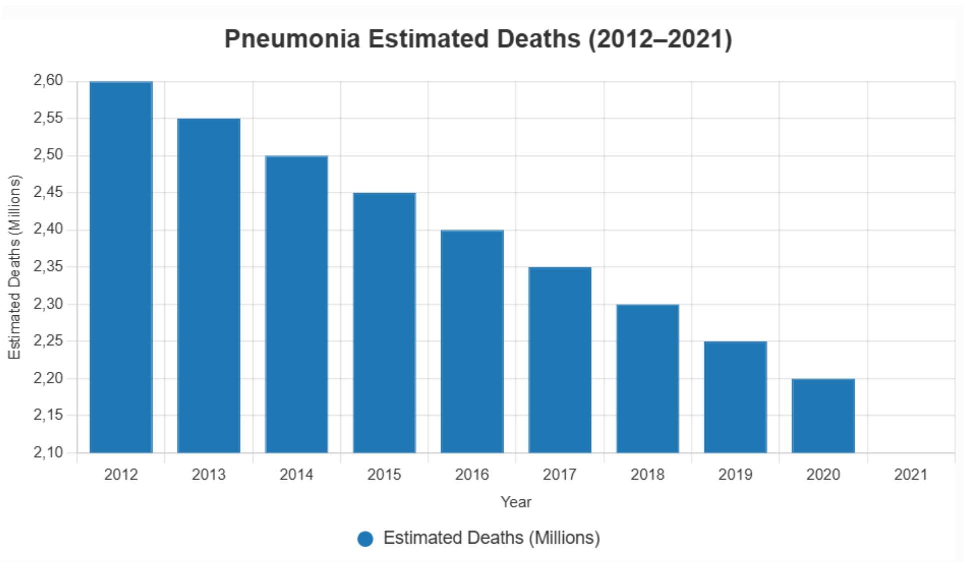

Figure 0.1: Estimated deaths of Pneumonia 2012 - 2021

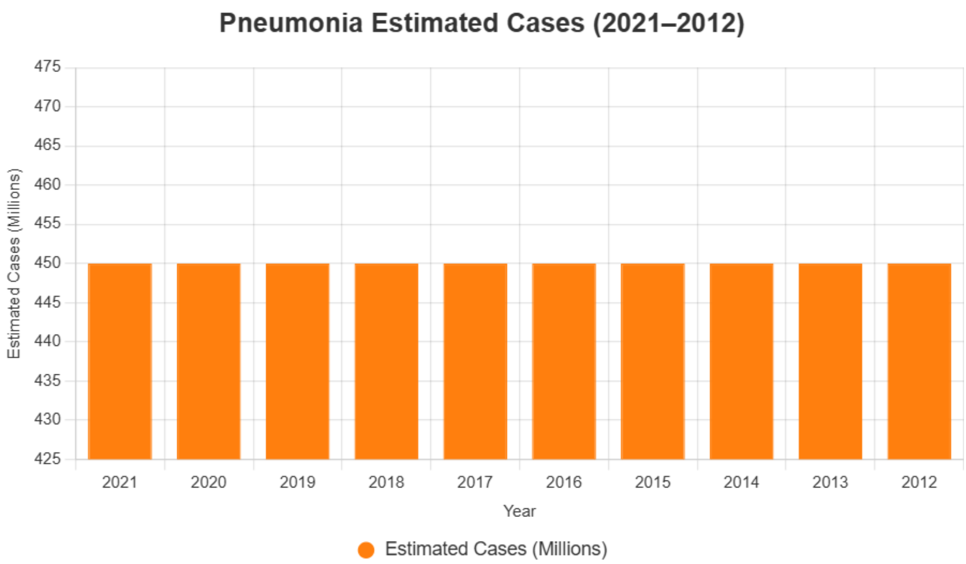

Figure 0.2: Estimated cases of pneumonia 2012 - 2021

This project is part of a license degree program to digitize healthcare workflows to improve accessibility and efficiency.

This disease faces numerous health challenges while there various solutions ,treatments and prevention measures for it, each has its limitations .Imaging and AI tools are used with the help of CXR and CT scans achieving high accuracy (up to 98%) however some are in their beta versions and lack clinical validation .Some approaches use treatments such as Antibiotics but overusing and misusing these can complicate treatment strategies ,and Inhaled therapeutics but these do require high profile studies and research.Preventing this is a widely used solution more commonly using vaccination which have reduced pneumonia like diseases but only cover specific strains and coverage remains suboptimal in some countries.

Problem

Deep Learning is an impactful and precise solution that has caught the eye of the medical industry in recent times ,it uses image and reads them to provide results based on its training process, DL models learn from features and patterns from various datasets to classify test images.The issue is the detection of Pneumonia patterns from various scans because manual interpretation of CXR scans is less accurate compared to AI diagnosis.

Proposed Solutions

The project focuses on developing a user-friendly medical website where users can make appointments directly with healthcare providers. The intelligent tool on the site, which utilizes deep learning to assist in symptom identification for pneumonia, will be a central feature of the site. The ultimate objective is to make health more efficient and accessible by integrating advanced technology with professional health support. The objectives of this project include :

- Bulding a DL model that can detect any Pneumonia or other infectious lung disease pattens using the TensorFlow framework.
- Designing a helpful and intuitive user interface for patients and doctors using React js.
- Enabling real-time booking updates ,requests and memberships via Node.js.
- Implementing secure login and data storage using MongoDB.
- Validating the system through usability testing with sample users.

Document Plan

This thesis is organized as follows:
Chapter 1 - State Of The Art : Here, this chapter shows the state of the art that places the project's contribution through the dealt area, also touching upon some of the existing solutions pertinent to the presented problem .These solutions have also been evaluated and compared with other deployed solutions with similar contexts presenting their strengths and weaknesses.

Chapter 2 - Contribution : This chapter describes the contribution in this line of work. First is to provide a brief overview of data that were chosen, then comprehending implementations utilized in building the model. This ranges from the pre-processing steps through to the training procedures and justifying the method utilized for this purpose.

Chapter 3 - Implementation and Experiments : Here is the presentation the real-world implementation of the proposed application. It illustrates structure and design of the designed website, explain its user interfaces and show how it is integrated with the model. This chapter also deals with results achieved from experiments, analyzing the performance and limitations of the solution.

Chapter

1

State of the Art

Introduction

Pneumonia is one of numerous diseases on this planet that have put a of the lives of multiple people on danger and is categorized as one of the most fatal diseases on earth.

Deep learning has already left its mark in the healthcare world and its applications have already been seen in medical imaging solutions that can identify certain patterns .Deep learning has been playing a fundamental role in providing medical professionals with insights that allow them to identify issues early on, thereby delivering far more personalized and relevant patient care.

The remaining section of this chapter's content is as follows : starting off with a detailed presentation of the project's context in which we discuss it into two parts the LIMS and how deep learning is used for classification purposes with medical imaging, next will be The Related Works part from which the main goal is to extract old solutions from various articles. summarizing these works in the Comparative Table and at last discussing the challenges faced in the articles that were used and extract the gaps found throught them.

This chapter gives an overview of the state of the art, acknowledges the major role played by Deep Learning models in this field and also highlights an emerging trend in using these models to define various diseases for patients specifying their diagnosis through continuous steps of image recognition from different images through the classification techniques used by the models.

1.1 Project Context and Area

1.1.1 Laboratory Information Management System

LIMS is a software based solution that mainly controls the flow of laboratory data operations .This has played a significant role through putting patient samples, records and their respective scan results together and has allowed for the medical staff to optimize their workflow and add more accurate functions on decision making in disease diagnosis.

1.1.2 Toward Deep Learning for Disease Classification using Medical Imaging

Thanks to deep learning the medical industry has had a good change in this kind of work through multiple CNNs such as VGG-16 ,ResNet-50 and AlexNet and so on and fourth ...these have shown great result in effective disease predictions by extracting important features from various scan images helping patient to prevent late discoveries of their dignosis of this deadly disease. But as the positive side reigns there are a few bad sides to explore ,for instance ,models can have a limit of the used datasets which could be excessive for it sometimes as some face computational costs and are not scalable enough .These interpretability issues need to be addressed to improve AI reliability and ensure that the deep learning matchup with medical diagnosis holds great potential to predict early pneumonia diagnosis allowing healthcare professionals to have more accuracy on putting up such decisions.

1.2 Related Works

1.2.1 Presentation of Existing Solutions

In recent years, convolutional neural networks (CNNs) have been commonly utilized to detect patterns of pneumonia on chest X-ray (CXR) images due to their hierarchical ability to extract features from medical image data. CNN-based models have been suggested by many studies as an approach to solve the problem with custom as well as pre-trained architectures.

Rajpurkar et al. (2017) [1] proposed CheXNet as a 121-layer DenseNet network trained on ImageNet and fine-tuned using a big set of CXR images from the ChestX-ray14 data set. CheXNet outperformed radiologists at pneumonia and other thoracic disease detection and indicated transfer learning's success with medical images. The algorithm classifies CXR images as pneumonia-positive or negative and is used primarily for binary classification.

Kermany et al. (2018) [2] built a CNN model on top of the Inception V3 structure using a pediatric CXR dataset consisting of about 5,800 images. They used data augmentation and transfer learning to counteract the small size of the data set in their method. The model reported 92.8% accuracy in identifying pneumonia and normal cases with an emphasis on pediatric populations.

Liang and Zheng (2019) [3] He proposed a special CNN structure based on residual connections and trained it on a portion of the RSNA Pneumonia Detection Challenge database. Localization techniques with bounding box predictions were employed in their model for identifying pneumonia-affected areas in CXR images. Sensitivity and specificity achieved in their study were 89% and 92% and highlighted spatial localization towards clinical explainability.

Saraiva et al. (2020) [4] He further developed ensemble learning based on different CNN architectures from ResNet50 to VGG16 to increase detection resilience for pneumonia. They trained their ensemble network with a 6,000 CXR image balanced set and achieved an F1-score of 0.91. The study pointed towards benefits in employing an ensemble of diverse architectures to reduce overfitting and increase generalizability.

Smith et al. (2018) [5] proposed a lightweight CNN model optimized for resource-constrained environments. Trained on a subset of the MIMIC-CXR dataset, their model achieved an accuracy of 0.90 with reduced computational requirements, making it suitable for deployment in low-resource settings.

Wang et al. (2019) [6] introduced a CNN with attention mechanisms to enhance feature extraction from CXR images. Their model, trained on the ChestX-ray14 dataset, reported a precision of 0.91 and a recall of 0.90, emphasizing the role of attention in improving diagnostic accuracy.

Chen et al. (2020) [7] developed a hybrid CNN model combining transfer learning with custom layers, trained on a diverse CXR dataset from multiple hospitals. Their approach achieved an F1-score of 0.92, focusing on generalizability across different patient populations.

Liu et al. (2021) [8] proposed a CNN model with real-time inference capabilities, trained on the RSNA Pneumonia dataset. Their model achieved an accuracy of 0.91 and was optimized for low-latency processing, suitable for emergency settings.

Patel et al. (2022) [9] explored explainable AI by integrating Grad-CAM visualizations into their CNN model. Trained on a custom CXR dataset, their model achieved an accuracy of 0.93 and provided visual explanations for predictions, enhancing clinical trust.

Kim et al. (2023) [10] introduced a multi-task CNN model for simultaneous pneumonia detection and severity assessment. Trained on a combined dataset from ChestX-ray14 and RSNA, their model reported an F1-score of 0.90, addressing both classification and clinical decision support.

These examples illustrate the capability of CNNs in pneumonia detection using large data collections, transfer learning, and deep architectures. However ,they vary in data size, model complexity, and evaluation criteria, and therefore require a comparison.

1.2.2 Comparative Table

The dataset comparison table highlights the diversity in dataset sizes and disease focus across the reviewed studies. Large datasets like ChestX-ray14 (100,000 images) and combined datasets (126,000 images) enable robust model training but may include multiple thoracic diseases, complicating pneumonia-specific detection. Smaller datasets, such as Pediatric CXR (5,800 images) and Custom CXR (6,000 – 10,000 images), focus exclusively on pneumonia, potentially limiting generalizability. The predominance of pneumonia as the target disease underscores the need for models that can adapt to varied dataset scales and disease contexts to ensure clinical applicability.

Study	**Dataset Name**	**Number of Images**	**Disease Contained**
Rajpurkar et al. (2017) [1]	ChestX-ray14	∼100,000	Pneumonia, other thoracic diseases
Kermany et al. (2018) [2]	Pediatric CXR	∼5,800	Pneumonia
Liang and Zheng (2019) [3]	RSNA Pneumonia Detection	∼26,000	Pneumonia
Saraiva et al. (2020) [4]	Custom CXR	6,000	Pneumonia
Smith et al. (2018) [5]	MIMIC-CXR	∼30,000	Pneumonia
Wang et al. (2019) [6]	ChestX-ray14	∼100,000	Pneumonia, other thoracic diseases
Chen et al. (2020) [7]	Multi-hospital CXR	∼10,000	Pneumonia
Liu et al. (2021) [8]	RSNA Pneumonia Detection	∼26,000	Pneumonia
Patel et al. (2022) [9]	Custom CXR	∼8,000	Pneumonia
Kim et al. (2023) [10]	ChestX-ray14, RSNA	∼126,000	Pneumonia, severity assessment

Table 1.1: Comparison of Datasets Used in CNN-based Pneumonia Detection Studies

The comparative table illustrates the diversity of deep learning techniques employed in pneumonia detection, ranging from transfer learning (e.g., DenseNet-121, Inception V3) to specialized approaches like attention mechanisms and explainable AI, achieving accuracies between 0.90 and 0.94. Binary Cross-Entropy dominates as the loss function, with Adam as the preferred optimizer, reflecting standard practices in medical imaging. GPU usage is universal, underscoring the computational demands of CNN training. Evaluation metrics, primarily accuracy, F1 score, precision, and recall, vary slightly, with some studies incorporating AUC or sensitivity for robustness. This highlights a trade-off between model complexity and deployability, emphasizing the need for efficient, generalizable solutions in clinical settings.

Study	DL Techniques	Accuracy	Loss Function	Optimizer	GPU Used	Evaluation Metrics
Rajpurkar et al. (2017) [1]	DenseNet-121, Transfer Learning	0.94	BCE	SGD	Yes	Acc, F1, Prec, Rec
Kermany et al. (2018) [2]	Inception V3, Transfer Learning	0.928	BCE	Adam	Yes	Acc, F1, Prec, Rec
Liang and Zheng (2019) [3]	Custom CNN, Residual, Localization	0.90	BCE + Dice	Adam	Yes	Acc, F1, Prec, Rec, Sens, Spec
Saraiva et al. (2020) [4]	Ensemble (ResNet50, VGG16)	0.92	BCE	Adam	Yes	Acc, F1, Prec, Rec
Smith et al. (2018) [5]	Lightweight CNN	0.90	BCE	Adam	Yes	Acc, F1, Prec, Rec
Wang et al. (2019) [6]	CNN with Attention	0.91	BCE	Adam	Yes	Acc, F1, Prec, Rec, AUC
Chen et al. (2020) [7]	Hybrid CNN, Transfer Learning	0.93	BCE	Adam	Yes	Acc, F1, Prec, Rec
Liu et al. (2021) [8]	Real-time CNN	0.91	BCE	Adam	Yes	Acc, F1, Prec, Rec, Sens
Patel et al. (2022) [9]	CNN with Grad-CAM	0.93	BCE	Adam	Yes	Acc, F1, Prec, Rec, AUC
Kim et al. (2023) [10]	Multi-task CNN	0.92	Weighted BCE	Adam	Yes	Acc, F1, Prec, Rec, Sens

Table 1.2: Comparison of Deep Learning Techniques and Training Details in Pneumonia Detection Studies

To better understand the strengths and limitations of existing solutions, the table below compares the aforementioned studies based on key criteria relevant to CNN-based pneumonia detection.

Study	Model	Dataset	Accuracy	F1 Score	Precision	Recall
Rajpurkar et al. (2017) [1]	CheXNet (DenseNet-121)	ChestX-ray14	0.94	0.88	0.89	0.87
Kermany et al. (2018) [2]	Inception V3	Pediatric CXR	0.928	0.93	0.93	0.93
Liang and Zheng (2019) [3]	Custom CNN	RSNA Pneumonia	0.90	0.90	0.91	0.89
Saraiva et al. (2020) [4]	Ensemble (ResNet50, VGG16)	Custom CXR	0.92	0.91	0.92	0.90
Smith et al. (2018) [5]	Lightweight CNN	MIMIC-CXR	0.90	0.89	0.90	0.88
Wang et al. (2019) [6]	CNN with Attention	ChestX-ray14	0.91	0.90	0.91	0.90
Chen et al. (2020) [7]	Hybrid CNN	Multi-hospital CXR	0.93	0.92	0.92	0.91
Liu et al. (2021) [8]	Real-time CNN	RSNA Pneumonia	0.91	0.90	0.90	0.91
Patel et al. (2022) [9]	Explainable CNN	Custom CXR	0.93	0.92	0.93	0.92
Kim et al. (2023) [10]	Multi-task CNN	ChestX-ray14, RSNA	0.92	0.90	0.91	0.90

Table 1.3: Comparison of CNN-based Pneumonia Detection Models

1.2.3 Discussion and Gaps Extraction

The reviewed studies evidence high accuracy and sensitivity in CNN-based pneumonia detection across a wide variety of approaches. Transfer learning as applied by Rajpurkar et al. and Kermany et al. is highly effective in taking advantage of pre-trained networks based on large imagesets such as ImageNet to overcome small medical imaging datasets. The models depend upon big, labeled datasets in most instances, e.g., over 100,000 images from ChestX-ray14, that might be lacking in resource-poor environments. Small datasets, e.g., pediatric CXR data utilized by Kermany et al., do pose concerns regarding generalizability.

Localization approaches, exemplified by Liang and Zheng, increase clinical interpretability through detection of affected areas but most models support just binary classification without spatial information and thus their application is restricted for radiologists. Robustness is improved through ensemble approaches, e.g., Saraiva et al., but computational cost is raised at the expense of deployability in low-resource settings. Optimizing for resource requirements is addressed through Smith et al.'s models but at an expense of performance. Attention mechanisms (Wang et al.) and explainable artificial intelligence (Patel et al.) increase accuracy and trust but their computational cost is an ongoing challenge. Practical deployment requirements for real-time inference (Liu et al.) and multi-task learning (Kim et al.) remain under-explored areas.

Key gaps in the literature include:

- **Limited Generalization to Diverse Populations**: Most existing models are tailored to particular datasets (e.g., pediatric or adult CXRs), and thus their utility is restricted over age ranges or geographic areas with different pneumonia patterns.
- **Lack of Explainability**: Although some studies do include visualizations or localizations, few provide explanatory mechanisms and thus diminish trust within clinical contexts.
- **Resource Constraints**: Highly performing architectures such as CheXNet demand considerable computational power and are thus unsuitable for low-resource healthcare settings.
- **Real-time Deployment**: Most past research have not dealt with real-time inference, that is essential in the case of an emergency.
- **Scalability Across Tasks**: Multi-task models like Kim et al.'s are rare, limiting the ability to handle related tasks like severity assessment alongside detection.

The envisioned project aims to address these gaps by developing a lightweight CNN optimized for pneumonia detection with improved generalizability, incorporating explainable attention mechanisms, and supporting real-time inference on resource-constrained devices. By building on the strengths of existing solutions, such as transfer learning, localization, and multi-task learning, the project seeks to balance performance, clinical usability, and deployability.

Conclusion

While deep learning in healthcare is still in the early stages of its potential, it has already seen a fair share of significant results.The integration of deep learning and Laboratory Information Management Systems (LIMS) has revolutionized the healthcare domain.

Deep learning models have significantly improved disease diagnosis, medical imaging analysis, and predictive analytics. At the same time, LIMS has automated laboratory workflows and guaranteed smooth data handling and communication among healthcare personnel.Nevertheless despite all these breakthroughs ,Deep learning models tend to be complicated and demand a lot of resources while LIMS adoption is restricted by monetary constraints.Alleviating these through cost-effective solutions and better regulations is crucial to guaranteeing that these technologies keep advancing a more efficient healthcare system.

Chapter 2

Contribution

Introduction

This chapter describes the key contributions of this project, which focus around building a web-based health care application combined with a Convolutional Neural Network (CNN) model. The main purpose is to offer users a working, user-friendly interface that not only allows appointment booking and engagement with accredited health care providers, but also employs deep learning to assist in diagnostic support as well as classification. In part one, the overall system design is explained by a set of UML diagrams Use Case diagram supplemented by three more Use Case descriptions from three actors identified in the diagram, Class diagrams specifying entities and relations among them and an Activity Diagram comprising several ADs from different functions of the web-app. Then a detailed explanation of the dataset used to train the CNN model, its origin, architecture, and preprocessing operations. The phase of preparation of the data was pivotal in order to have high-quality input for the model. After presenting the overview of the data, proceeding to the elaboration of the architecture and training procedure of the CNN model. This covers layer configurations, activation functions, loss optimization, and performance metrics. The design decisions made in the model are justified in terms of requirements for the process of testing. Next, a description of the most important parameters and hyperparameters of the CNN architecture, including the number of layers, filter sizes, activation functions, learning rate, as well as batch size. Then a brief discussion of each key to emphasize its significance in influencing the performance of the model as well as the reasons why it was chosen. Finally, the key parameters and hyperparameters of the CNN model are explained, such as the number of layers, filter sizes, activation functions, learning rate and batch size. Each is briefly discussed to highlight its role in the model's performance and the reasoning behind its selection.

2.1 Part One: System Design

2.1.1 Preliminary study and requirements specification

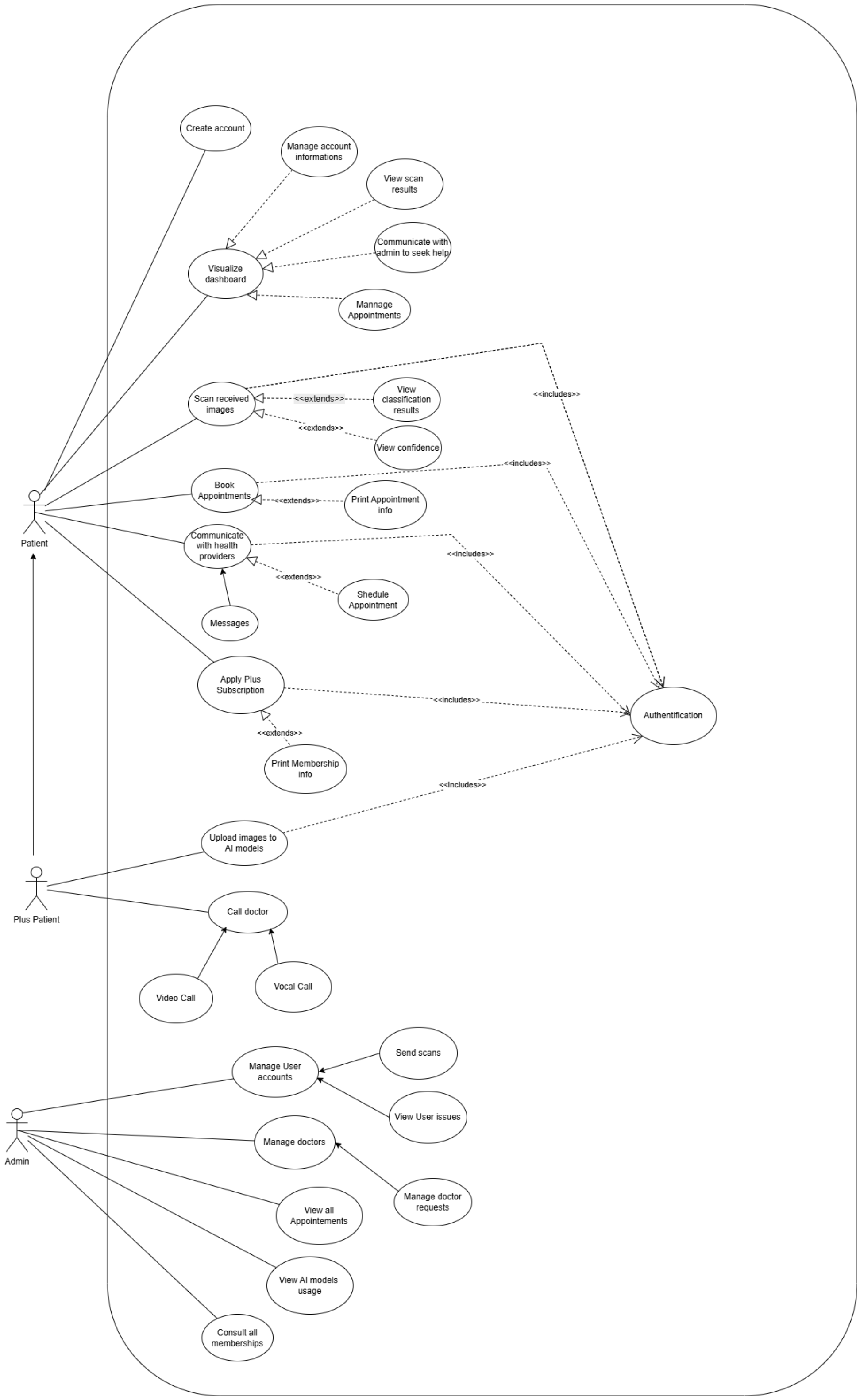

Figure 2.1: Use Case Diagram

This use case diagram depicts the major functionalities of the health related website aimed at improving patient as well as administrative experiences. It represents interactions among patients, Plus patients, as well as administrators with the system including account management, appointment booking, interactions with health providers, and sophisticated features such as integration with an AI model. The diagram showcases the uninterrupted flow of actions to ensure hassle-free delivery of health care and ease of use.

2.1.1.1 Use Case Description: View Scans of the Patient

The "View Scans" use case enables a patient to view their medical scan results via the visualization dashboard. After logging in and authenticating, the patient accesses the dashboard, where the system displays their scan data with confidence levels for review.

Field	Description
Use Case Name	View Scans of the Patient
Actor	Patient
Preconditions	Patient is logged into the website. - Patient has existing scan data.
Postconditions	- Patient successfully views their scan results with confidence levels.
Basic Flow	1. Patient logs into their account. 2. Patient navigates to the visualization dashboard. 3. System authenticates the patient. 4. System retrieves and displays scan data with results. 5. Patient reviews the scans.
Alternative Flow	- If no scan data exists, system notifies patient and prompts to contact healthcare provider.
Exceptions	- Authentication failure redirects to login page. - System error displays an error message.
Assumptions	- Patient has internet access. - Scans are pre-processed and available.

2.1.1.2 Use Case Description: Upload Images to AI Model or Plus Patient

The "Upload Images to AI Model" use case allows a Plus patient to submit medical images for AI analysis. After logging in and authenticating, the Plus patient uploads images, which are processed by AI models, returning results for healthcare management.

Field	**Description**
Use Case Name	Upload Images to AI Model or Plus Patient
Actor	Plus Patient
Preconditions	Plus Patient is logged into the website. - Plus Patient has image files.
Postconditions	- Images are successfully uploaded and processed by AI models.
Basic Flow	1. Plus Patient logs into their account. 2. Plus Patient navigates to the upload feature. 3. System authenticates the Plus Patient. 4. Plus Patient selects and uploads images. 5. AI models process the images and return results.
Alternative Flow	- If image upload fails, system prompts to retry or contact support.
Exceptions	- Authentication failure redirects to login page. - Invalid file format rejected.
Assumptions	- Plus Patient has internet access. - AI models are operational.

2.1.1.3 Use Case Description: Send Scans of Admin

The "Send Scans" use case permits an admin to send patient scan results to the patients. After logging in and authenticating, the admin selects scans and recipients, and the system securely transmits the data.

Field	**Description**
Use Case Name	Send Scans of Admin

Actor	Admin
Preconditions	Admin is logged into the website.
	- Scan data is available for patients.
Postconditions	- Scans are successfully sent to designated healthcare providers.
Basic Flow	1. Admin logs into their account.
	2. Admin navigates to patient records.
	3. System authenticates the admin.
	4. Admin selects scans and recipients.
	5. System sends the scans securely.
Alternative Flow	- If recipient is invalid, system prompts admin to select a valid recipient.
Exceptions	- Authentication failure redirects to login page.
	- Transmission error notifies admin.
Assumptions	- Admin has internet access.
	- System ensures data privacy compliance.

2.1.2 Analysis and Design

The presented diagrams show both structural and behavioral aspects of the healthcare website system. Class diagrams establish major entities Patient, Physician, Appointment, Communication, and AI Classification Module and their attributes and relations and reflect the data model and communications within the system. Activity diagrams, consisting of Plus Subscription, Book Appointment, Contact Doctor, Interact with AI Model and Admin Functions represent sequential steps utilized by the user while using the system, from log in to appointment management and use of AI capabilities. These diagrams collectively provide an end-to-end view of the system structure and workflow and support effective healthcare delivery and user interaction.

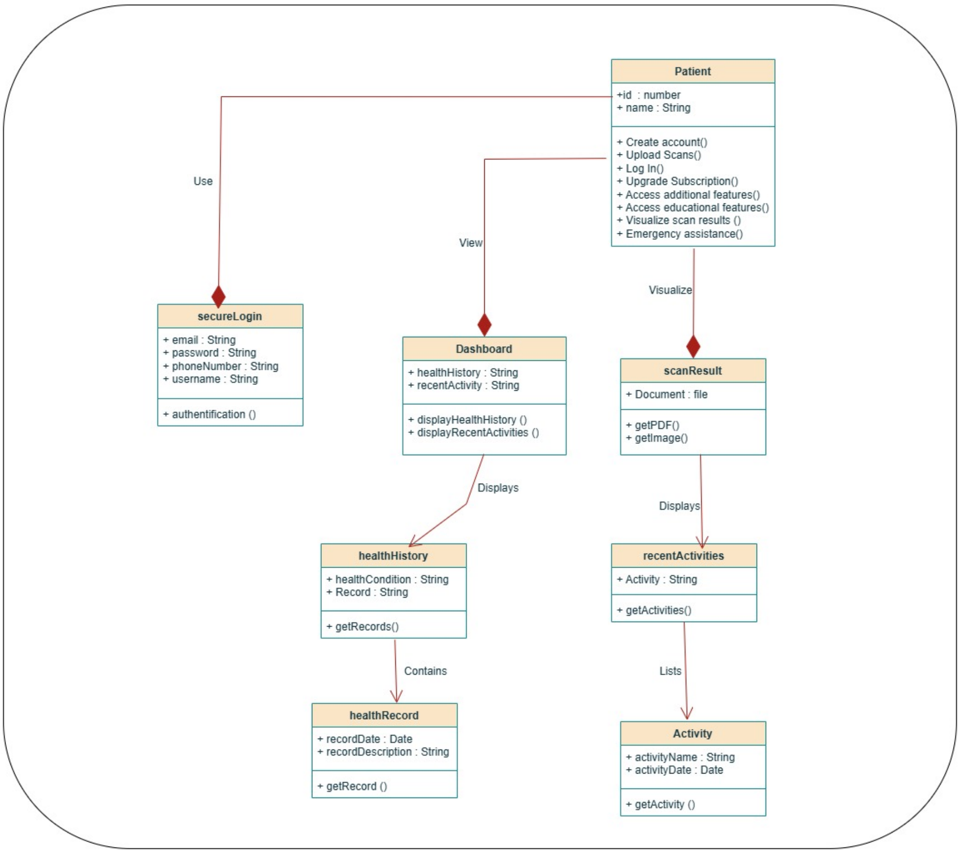

Figure 2.2: Class Diagram for Patient Portal

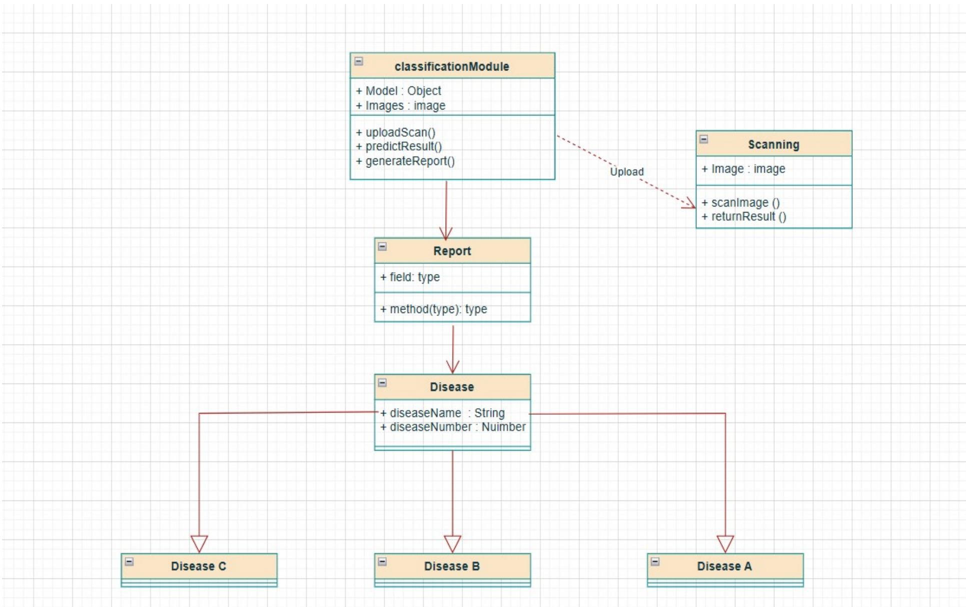

Figure 2.3: Class Diagram for Classification Model

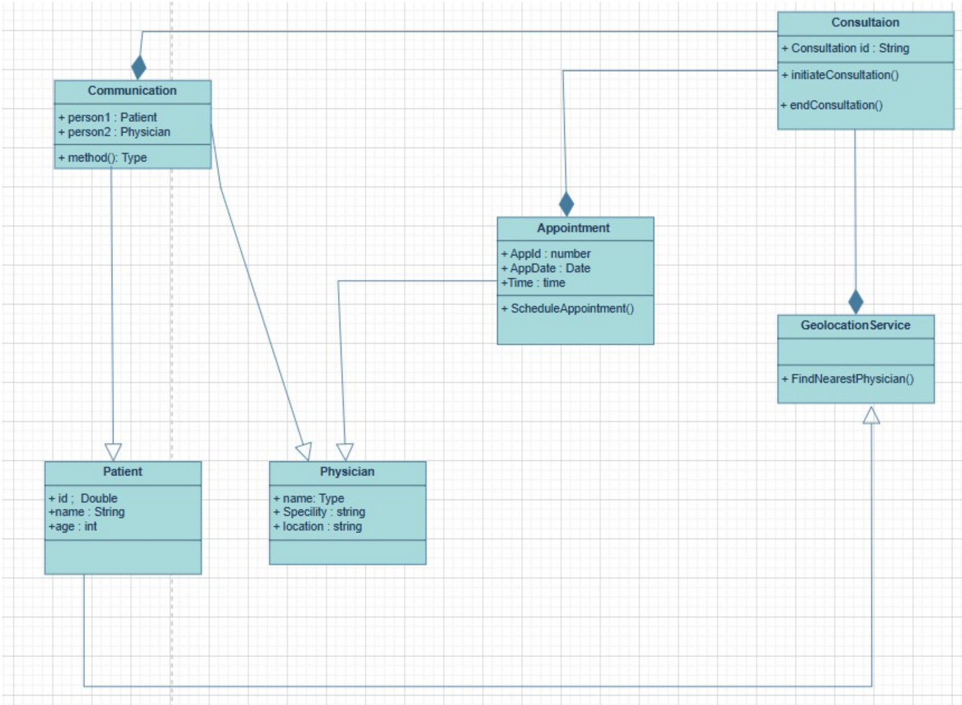

Figure 2.4: Class Diagram for Physician Consultation

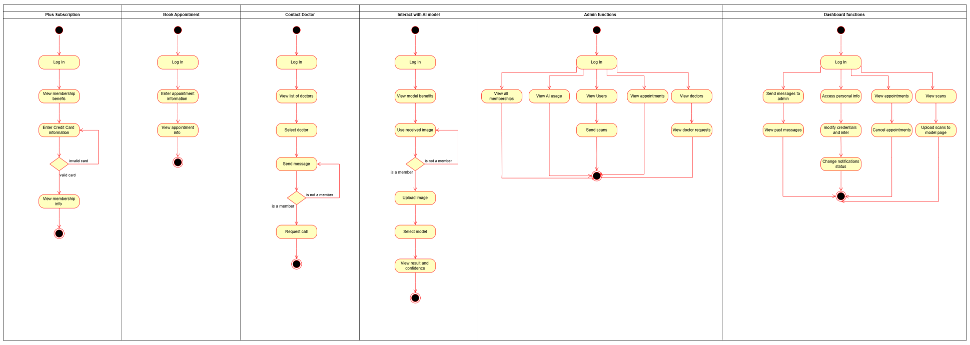

Figure 2.5: Activity Diagram

2.2 Part two : Proposed Solution

2.2.1 Data preparation and Description

For the experimental purpose,The dataset used includes a total of 22133 of different scan images, up to 3 distinct thoracic pathology labels can be separately applied to each image ,The dimensions of each image are 299 by 299 pixels .In addition to 10192 images of healthy cases ,tremendous X-Ray images of Pneumonia and Lung Opacity were carried out and accessed from reliable resources .Starting off with Pneumonia that includes 5929 images from various public datasets and studies and 6012 images of Lung Opacity which were all CXR images publicly available for research purpose and training the proposed DL Model.

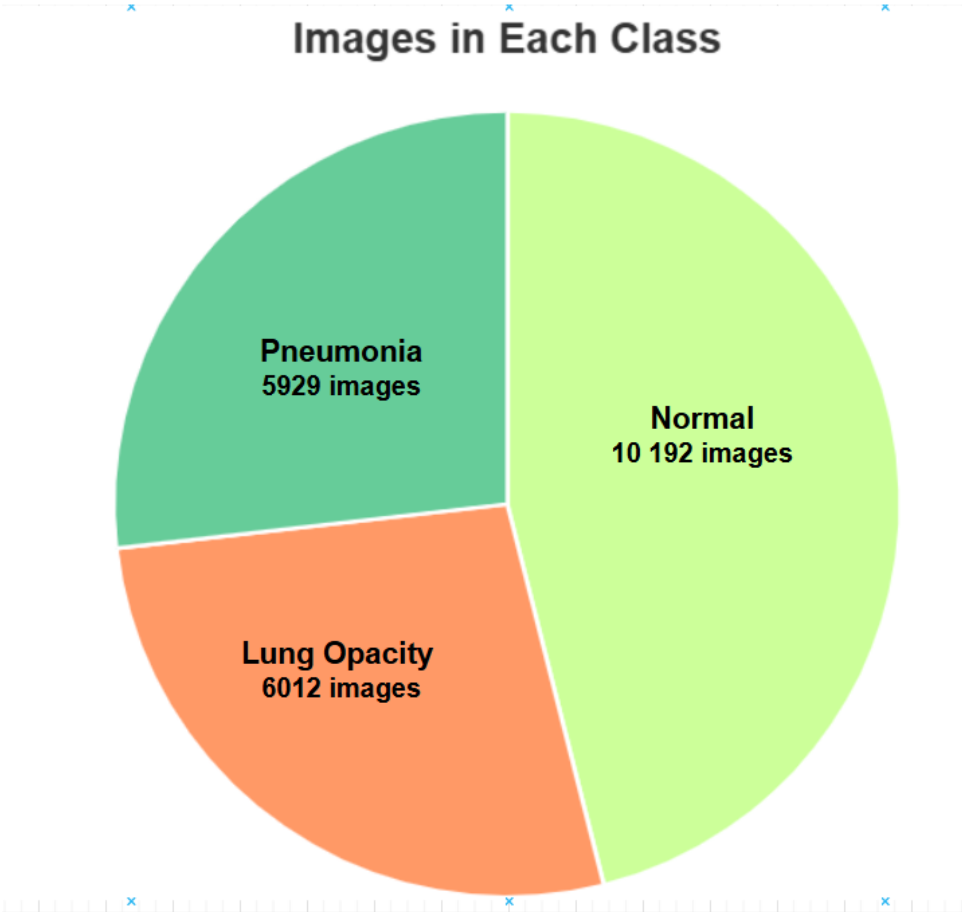

Figure 2.6: Data used for Multi-class model

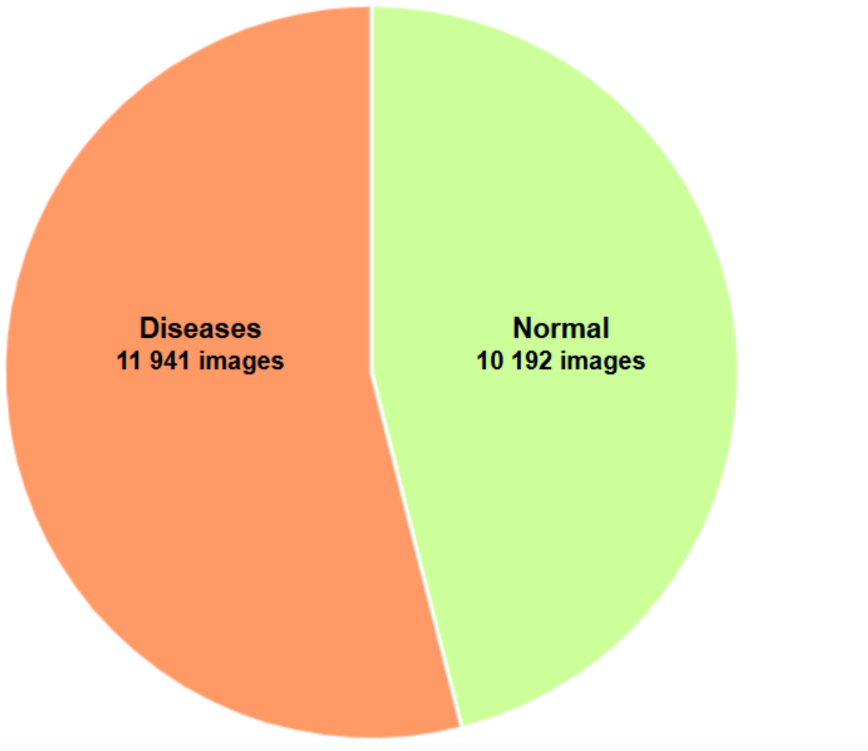

Figure 2.7: Data used for binary model

2.2.2 Data Pre-processing

All images were subjected to some preprocessing processes to adjust the input data to meet the requirements of the deep learning model ,image resizing to a scale of 224 by 224 pixels to match the model's input for this investigation ,the images were converted to an array to be employed an input in the model's next phase ,removing the noise from all images using median filtering with a window size of 3 × 3.

2.2.3 Classification

2.2.3.1 Model Architecture

Multi-Class Model : The model used in this research is a deep CNN designed for multi-class classification with an input layer that accepts grayscaled images shaped in a dimension of 224 by 224 .The model contains five convolutional layer blocks each with a Conv2D layer with an increasing filter sizes 32 - 64 - 128 - 256 - 512 ,a Kernel window size of 3 x 3 for Feature Extraction, ReLU activation function followed with MaxPooling2D(2 x 2) to decrease the size of the images .The model's classification is done through fully connected layers that include the Flatten() layer that converts the 3D features into 1D then a Dense layer with 256 neurons and ReLU activation function followed by a Dropout rate of 0.5 to decrease overfitting and regularization and at last the output layer which a single output for multi-class classification supported by a softmax activation function.

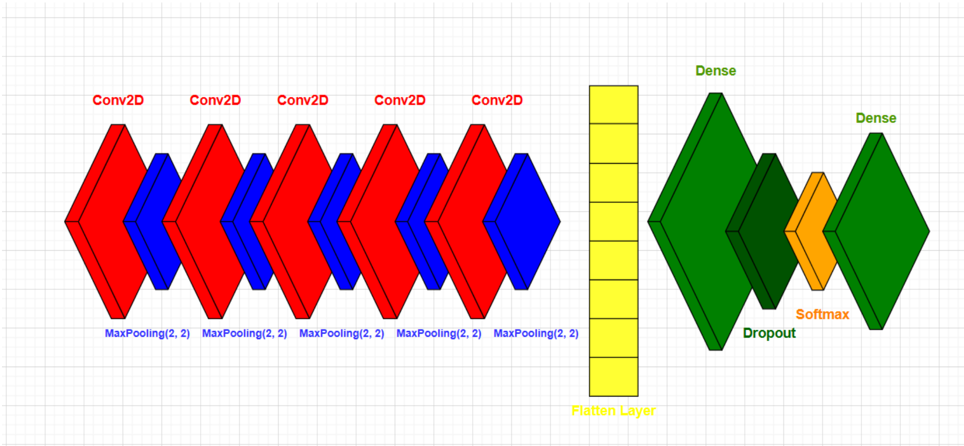

Figure 2.8: Model Architecture fro Multi-class classification model

The multi-class model workflow loads X-ray images, extracts labels for normal, pneumonia and lung opacity labels, it splits and balances the data, trains a Keras CNN for feature extraction and multi-class prediction via fully connected output layers, and tests with accuracy, loss, AUC, precision, and recall.

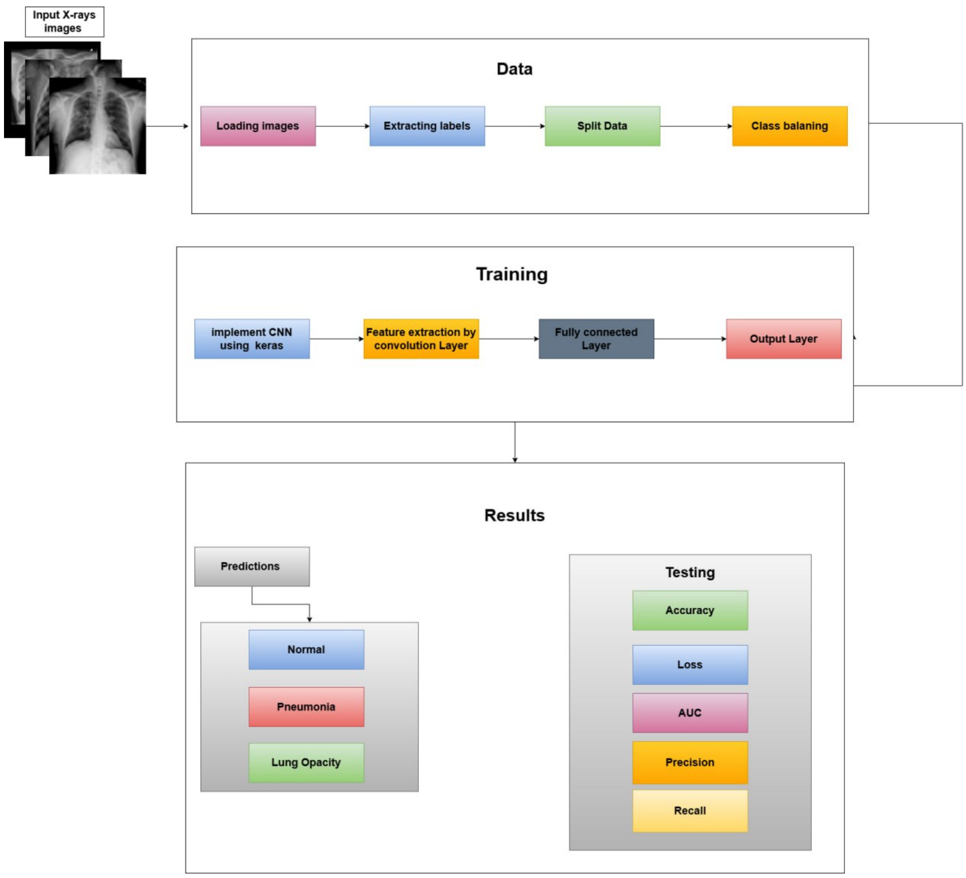

Figure 2.9: Global workflow for Multi-class classification model

Binary Model : The model used in this research is a deep CNN designed for binary classification of 0 and 1 with an input layer that accepts Gray scaled images shaped in a dimension of 224 by 224 .The model contains five convolutional layer blocks each with a Conv2D layer with an increasing filter sizes 32 - 64 - 128 - 256 - 512 ,a Kernel window size of 3 x 3 for Feature Extraction, ReLU activation function followed with MaxPooling2D(2 x 2) to decrease the size of the images .The model's classification is done through fully connected layers that include the Flatten() layer that converts the 3D features into 1D then a Dense layer with 256 neurons and ReLU activation function followed by a Dropout rate of 0.5 to decrease overfitting and regularization ,the use of dropout also helps with improvements of generalization and prevent overfitting and at last the output layer which a single output for Binary classification supported by a sigmoid activation function.

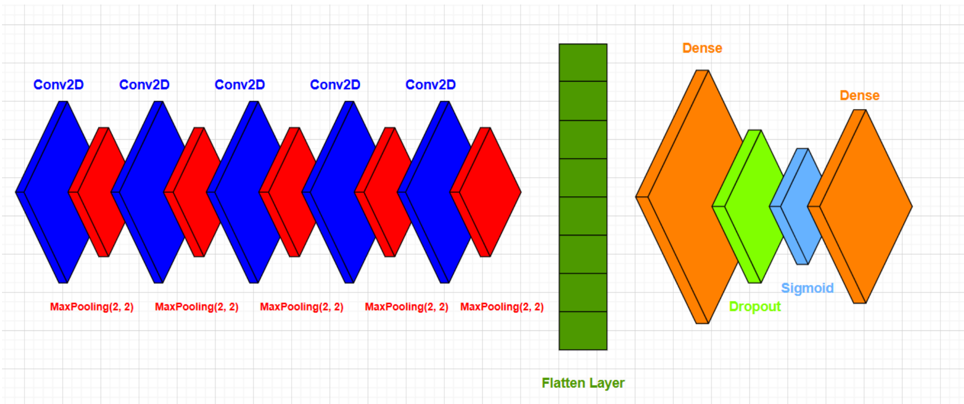

Figure 2.10: Model Architecture for Binary classification model

The binary model's workflow loads X-ray images, extracts normal or disease labels, splits and balances the data, then trains a Keras CNN to extract features and predict through fully connected output layers, finally testing with accuracy, loss, AUC, precision, and recall.

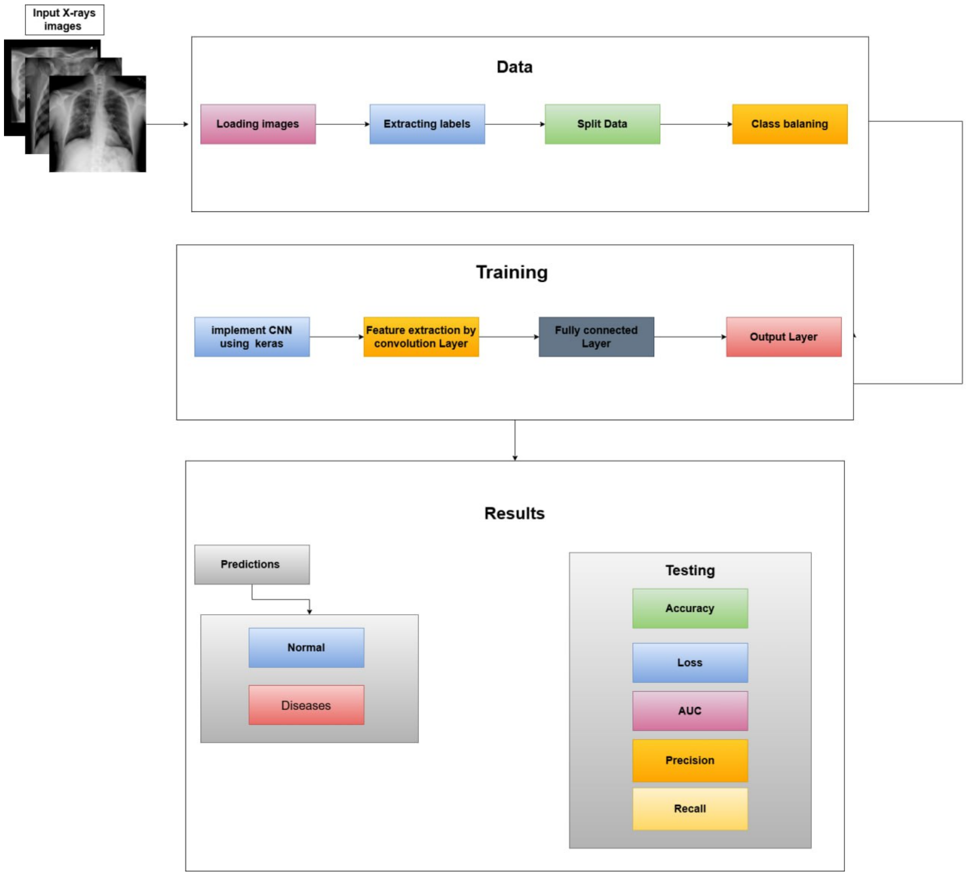

Figure 2.11: Global workflow for Binary classification model

2.2.3.2 Model Training

The model's training is as follows: the choice for a loss function is the Categorical Crossentropy function to calculate the model's loss because its output is a multiclass classification result, thus being suitable for the task, the Adaptive Moment Estimation **(ADAM)** optimizer was selected, a very common choice for Gradient Descent. Accuracy, Recall, Precision, AUC, with F1-Score metrics were adopted for measurement purposes as the evaluation metric for this task, The model also got a batch size of 32 with 10 epochs, thus dividing into 692 images for training.

2.2.3.3 Model Parameters

The model resides in a deep convolutional network which classifies images into three classes of outputs. It begins with a Conv2D layer which holds 32 filters as max pooling follows. The succeeding blocks of convolution increase in filter count progressively; they have 64, 128, 256, and 512 filters, respectively. There's a convolutional layer in each of these blocks, and max pooling scales down. Following the last convolutional block, dropout regularizes, flattens, then sends it to a dense layer with a count of 256 units then to an output layer of 3 outputs to classify. The field of parameters in total in the model stands at around >4.8 million, with more than 4.8 million of them as trainable.

Conclusion

As a whole, these deep CNN architectures provide a robust solution for both binary as well as multiclass image classification problems. With five convolutional blocks, robust feature extraction abilities, as well as a densely connected final layer, the design is perfectly positioned to learn from three distinct classes of image representations. Its train setup is optimized for multiple criteria like accuracy ,recall ,precision ,AUC as well as f1 score but still, parameter count depicts its ability. This model itself is a great remedy for recognizing uncommon patterns in chest X-rays as well as regulating its decision making.

Chapter 3

Implementation an Experiments

Introduction

This chapter describes the practical aspects of the project, in two broad sections: web application development, and experimentation with integrated CNN model. The first part highlights noticable user interfaces of the web application, describing their layout, behavior, and purpose in the complete system. Some of these include appointment booking views, points of interaction with the model, and administrative views, all with emphasis on usability as well as clarity.

The second part, titled Model Experimentation Discussion, focuses on the evaluation and analysis of the CNN model. A detailing of the development environment for the model is adopted alongside validation techniques and evaluation metrics used to assess performance. Both binary and multi-class classification results are presented, followed by comparisons with alternative models. Finally ,a brief discussion of the findings, analyzing the strengths and limitations of the proposed approach in light of the results.

3.1 Part One: Web Application

Hardware for web implementation includes a web server and a development machine. The development machine is an HP EliteBook with 8GB memory, an Intel i5 8th generation processor and 256GB SSD to meet website development performance demands. The web server is contained in a virtual private server (VPS), with a Windows 11 platform to ensure stable hosting capability. For software implementation, a frontend is developed using React.js, one of the top JavaScript libraries for executing dynamic user interfaces. The backend is executed based on Node.js and the Express framework to enable effective server-side programming. MongoDB is a No-SQL database utilized for flexible and expandable data storage. Version control is also achieved through Git and GitHub to enable collaborative development and management of code.

3.1.1 Web Application Presentation

The home page includes a navbar with shortcuts in the page and a sign in button

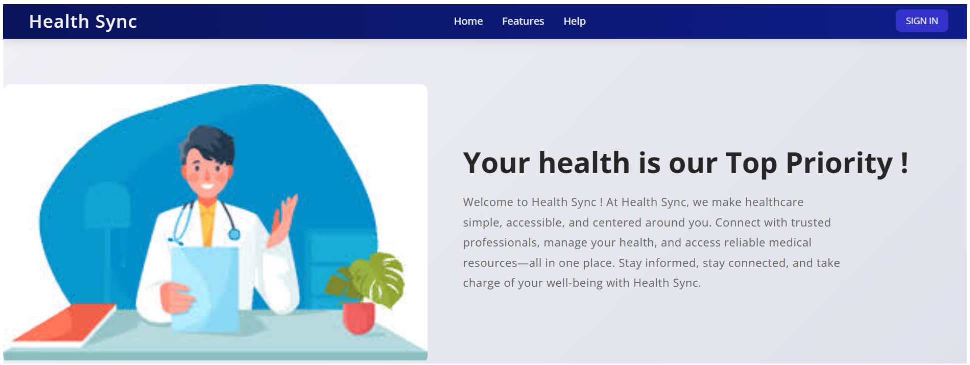

Figure 3.1: Main home page

The features can be accessed by pressing "features" on the navbar and viewing all available features

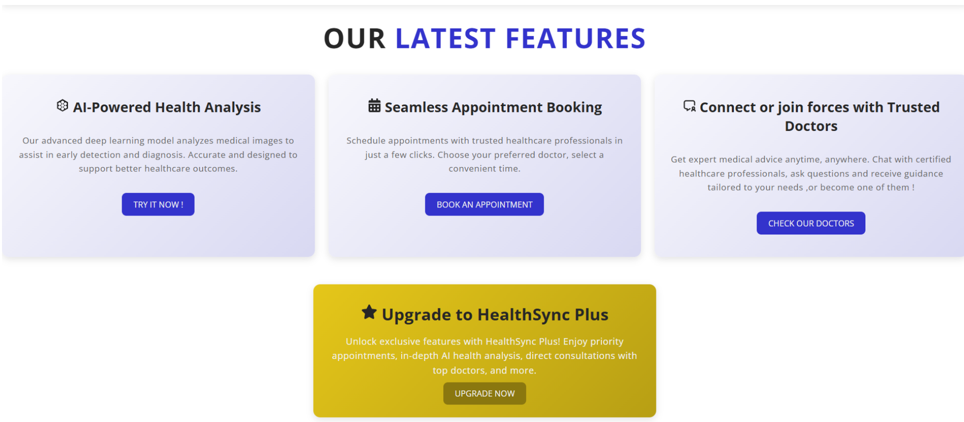

Figure 3.2: Features section

The sign in page includes a sign up component next to it for better UX

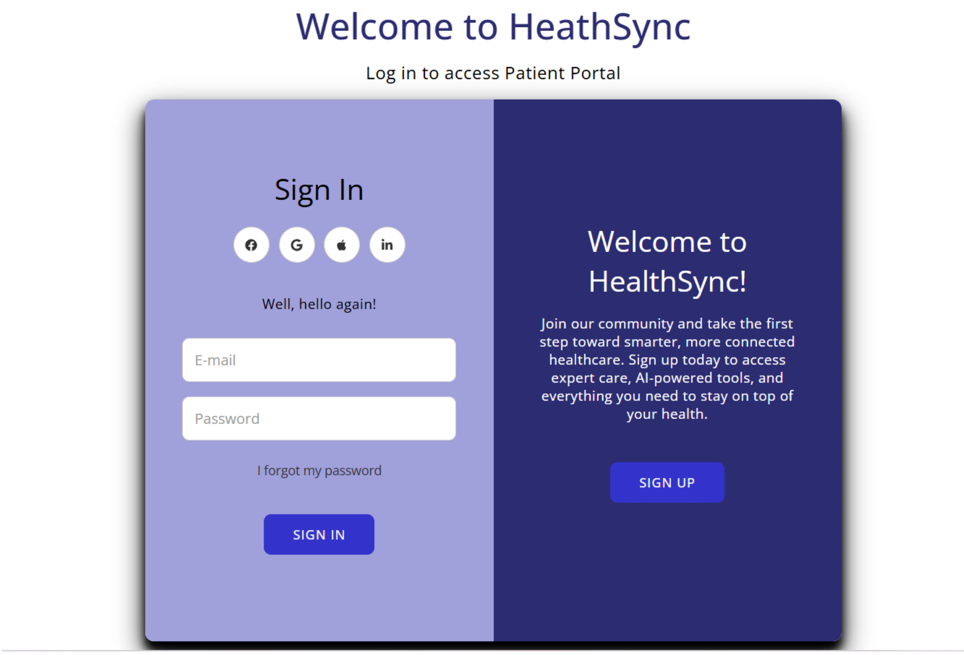

Figure 3.3: Sign in

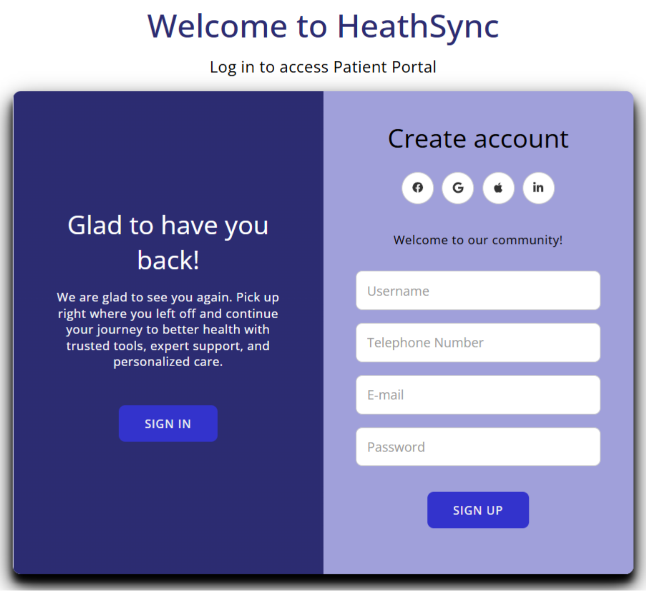

Figure 3.4: Sign up page

The booking can be done by filling this small form which later on displays all info and the option to print it

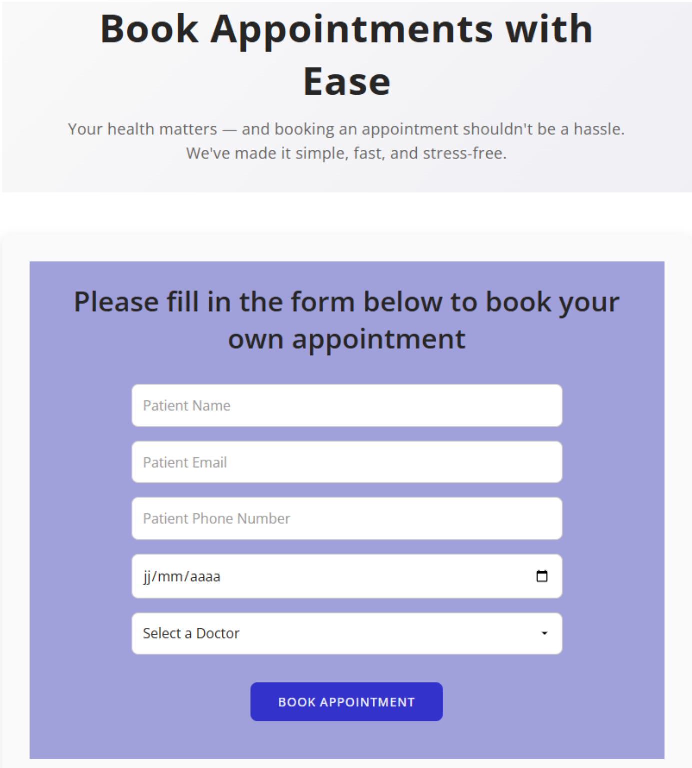

Figure 3.5: Booking page

The doctors' page includes a list of all doctors and communications with messages as default and calls for Plus members

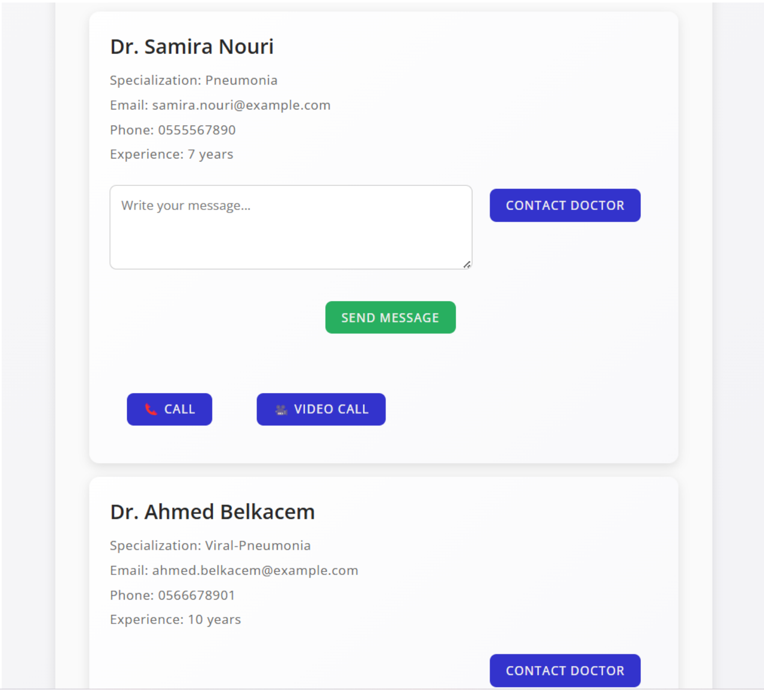

Figure 3.6: Doctors' page

The membership page includes an upper interface to display membership status and membership benefits

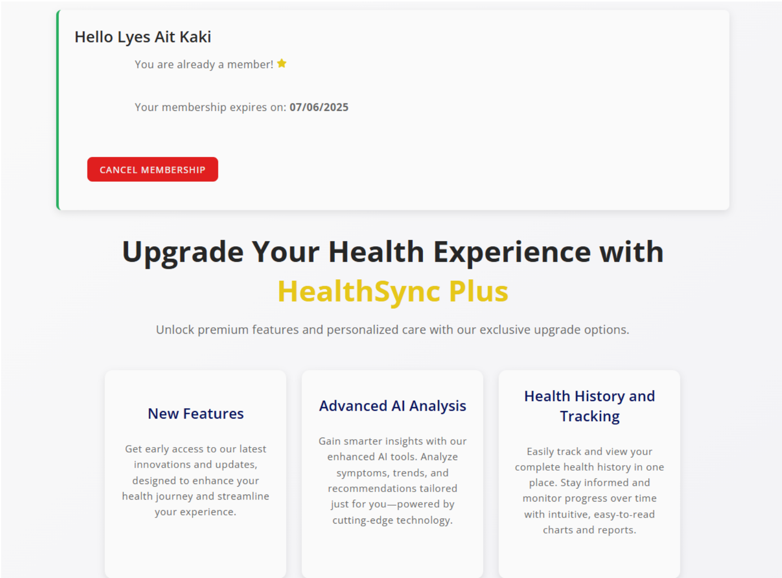

Figure 3.7: Membership page upper interface

Membership form to be filled then displays all membership info with the option to print

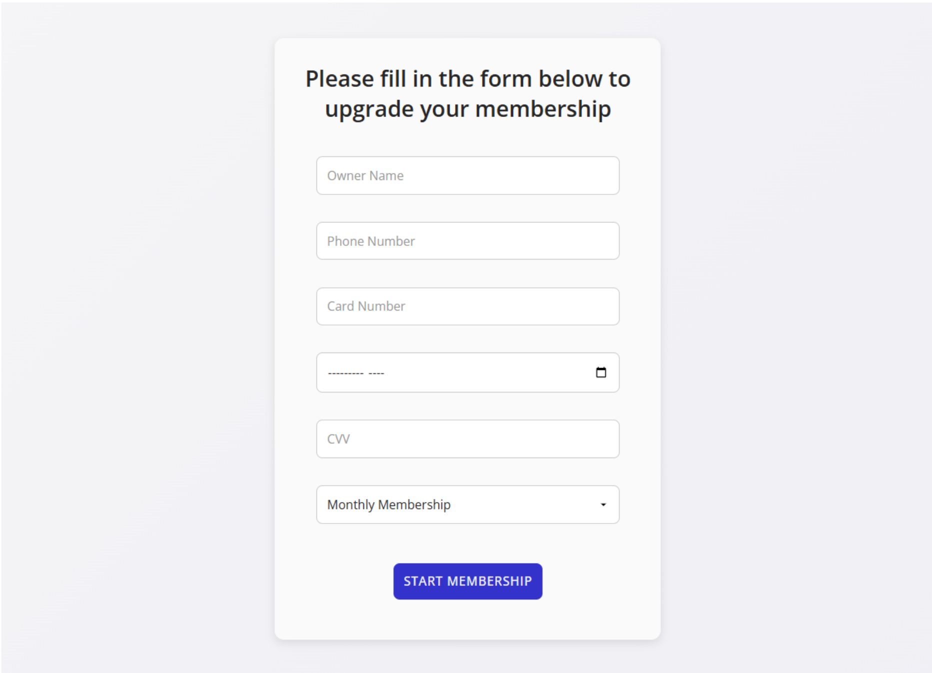

Figure 3.8: Membership form

The model form to be filled as an image can be uploaded directly from the dashboard or from the device for membership members only and by selecting the prefered AI model

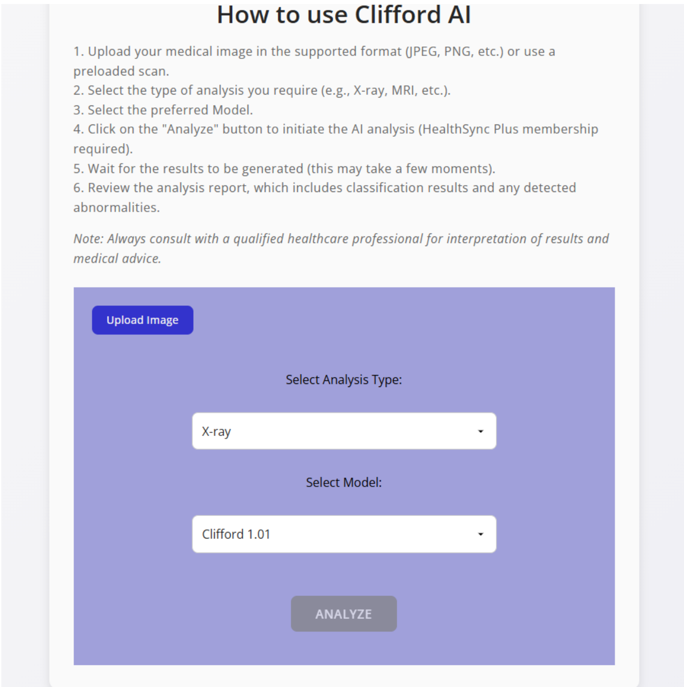

Figure 3.9: Model form

The dashboard contains all personal info which can be customized and saved at any time with the ability to disable or enable notifications

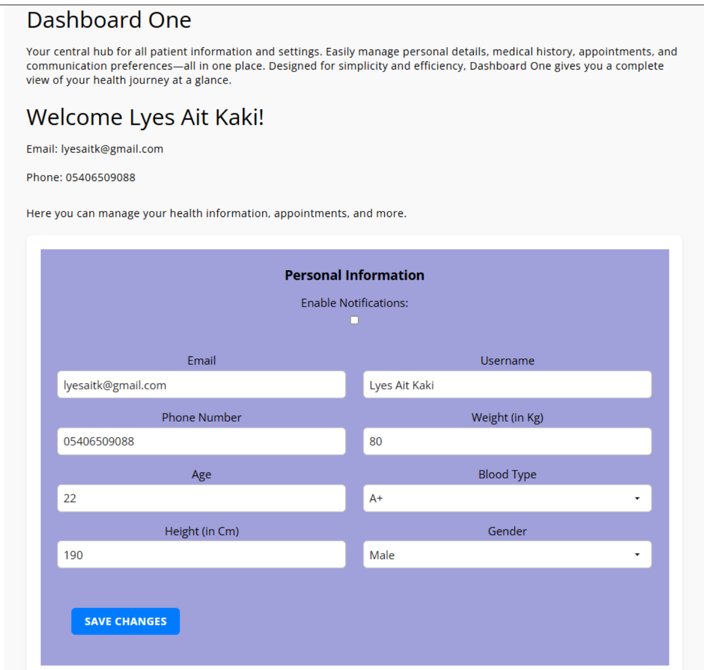

Figure 3.10: dashboard form

other dashboard features include a help center for communicating with admin ,an appointment section to view appointments or cancel them and a medical history section to view any received scans from the admin

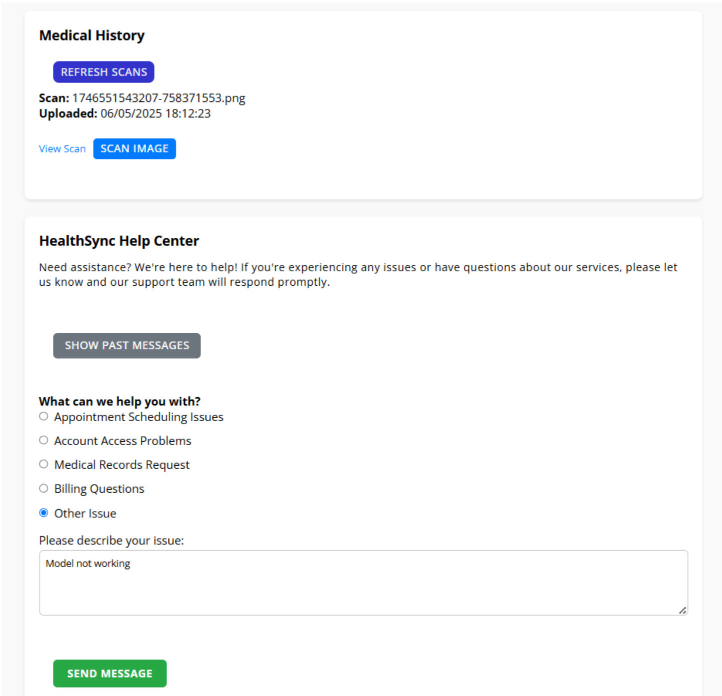

Figure 3.11: dashboard form

The admin panel includes features only accessible through the admin's account as seen in the image below

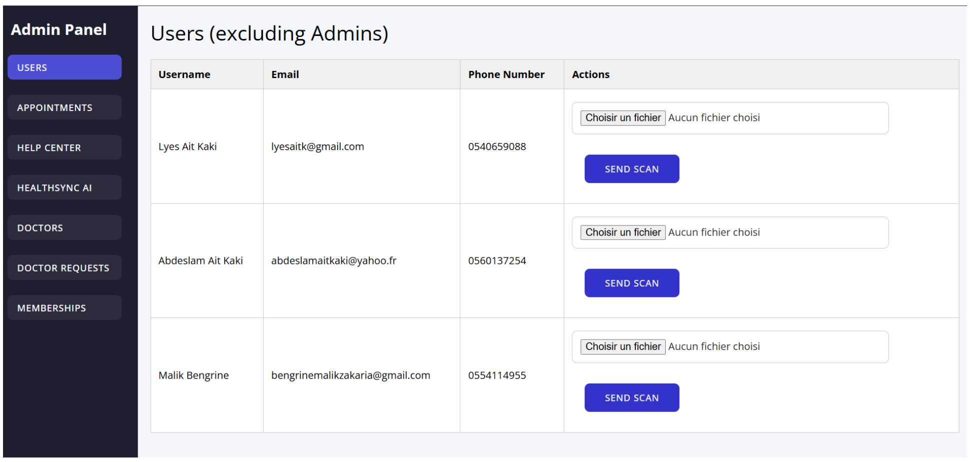

Figure 3.12: admin panel

3.2 Part Two: Model Experimentation and Discussion

3.2.1 Experimentation

3.2.1.1 Evaluation metrics and validation techniques

The model's performance is measured using accuracy, precision, recall, AUC and F1-score based on classification report and confusion matrix. The data is divided into 70% training set, 20% validation set and 10% test set while keeping class balance intact. Early stopping and reducing learning rate avoid overfitting while data augmentation is used to improve generalizability. Class weights balance out imbalanced number of images in each set to ensure valid validation and sound medical image classification

Metric	Accuracy	Recall (Sensitivity)	Precision	AUC (Sensitivity)
Formula	$\frac{TP+TN}{TP+TN+FP+}$	$\frac{TP}{TP+FN}$	$\frac{TP}{TP+FP}$	Area under the ROC curve

Table 3.1: Metrics and their formulas

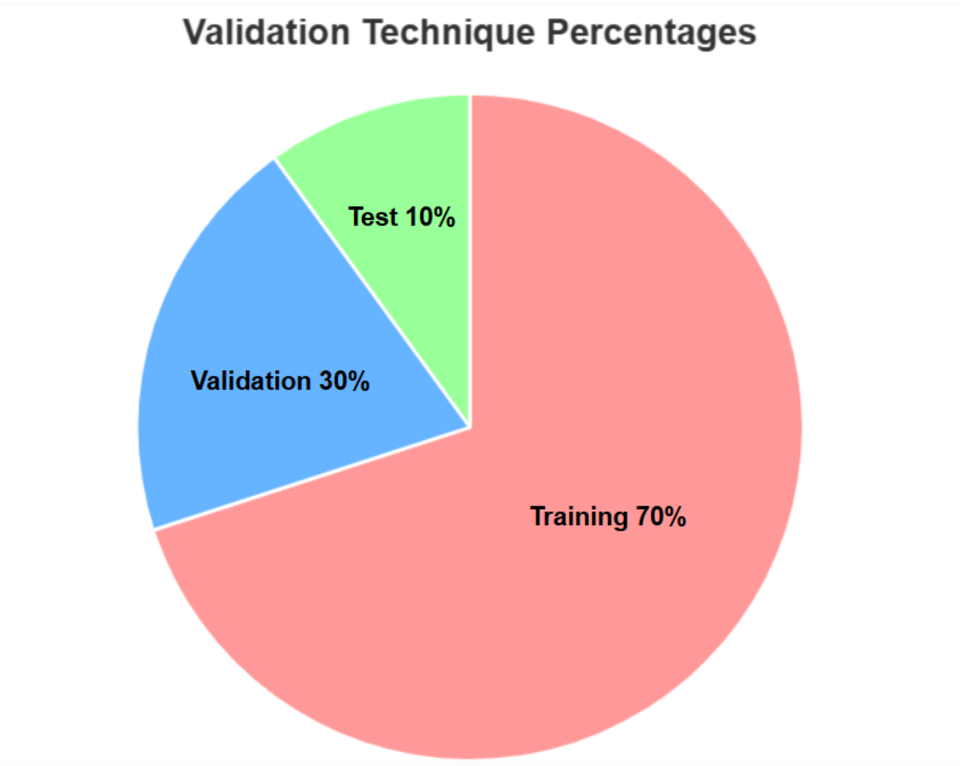

Figure 3.13: Validation technique showcase

3.2.1.2 Development Environments

The project's model development environment was Python-based, consisting of a number of widely used Python libraries used in machine learning as well as working with data. Building and training the image classifier was done with TensorFlow, while NumPy, Matplotlib, and Seaborn facilitated working with data as well as visualizing it. Scikit-learn was used to add other support in terms of evaluation metrics for preprocessing as well as evaluation. The os module was also used in the project to deal with filepaths as well as directories for data. The project was developed in Kaggle, which provides a cloud-based environment for a Jupyter Notebook with support for a GPU such as the Nvidia Tesla P100 or the Nvidia Tesla T4, making training deep learning models as well as sharing findings more convenient.

3.2.1.3 Experimental results of Multi-class classification

Accuracy	Precision	Recall	Loss	AUC	F1-Score
93.91%	93.90%	93.86%	16.27%	99.16%	93.30%

Table 3.2: Evaluation Metrics and their values(Multi-class)

The accuracy, precision, and recall have a high value of 93.91%, 93.90%, and 93.86%, respectively. The F1-score of 93.30% indicates a balanced model. The high value of AUC, which stands at 99.16%, reveals good discriminative power. The loss value stands at 16.27%, which indicates some potential for model optimization.

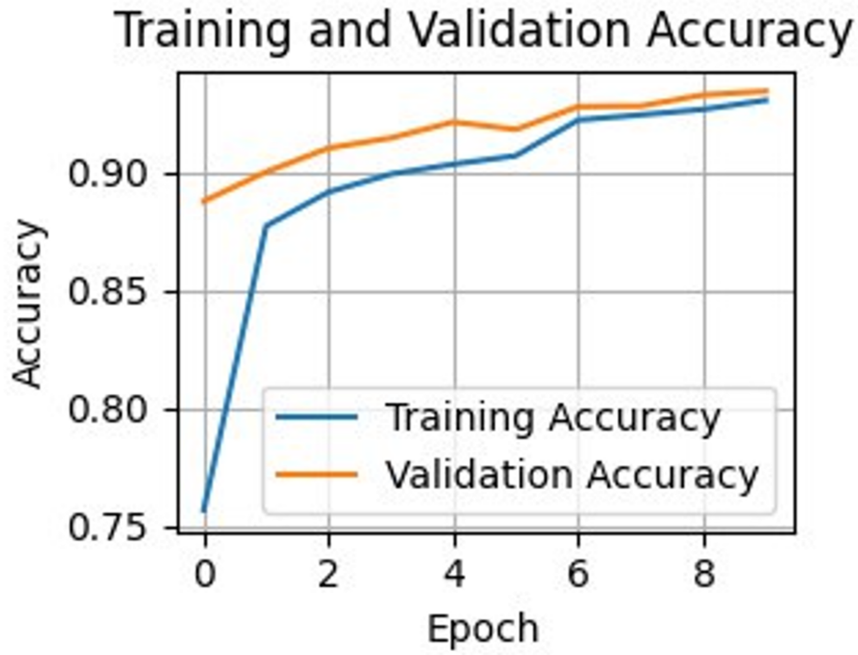

Figure 3.14: Multi-class model accuracy

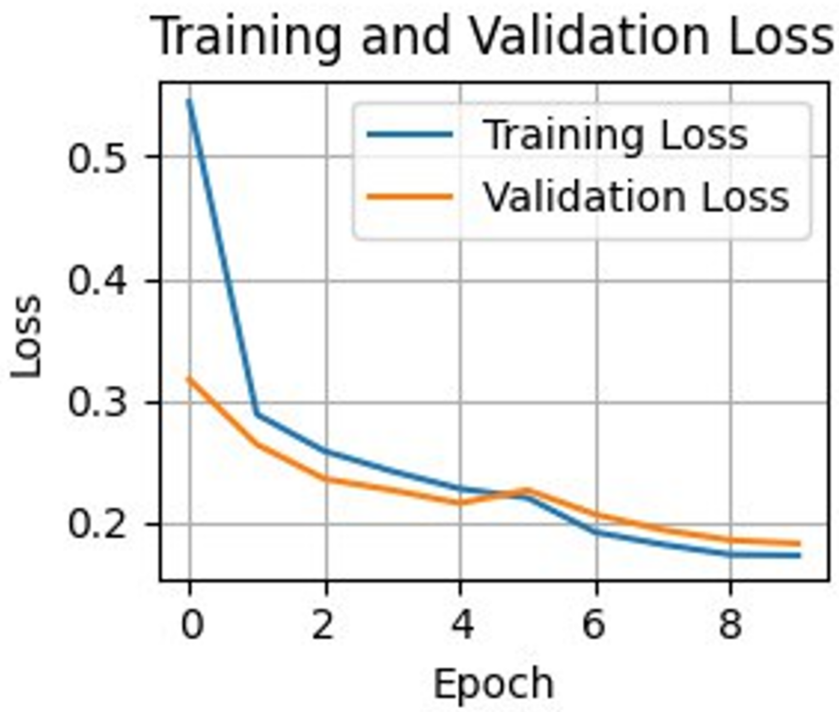

Figure 3.15: Multi-class model loss

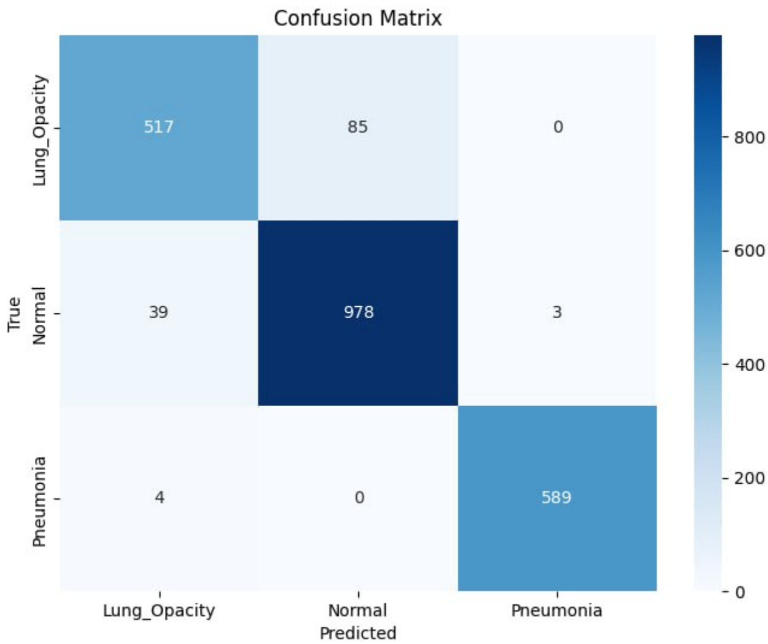

Figure 3.16: Multi-class model confusion matrix

The confusion matrix shows accurate classification: 517 lung opacity, 978 normal, and 589 pneumonia, with few errors (e.g., 85 lung opacities as normal). Training and validation accuracy rise to 0.90 by epoch 8, with minimal overfitting. Loss drops from 0.5 to below 0.2, indicating effective learning.

3.2.1.4 Experimental results of Binary Classification

Accuracy	Precision	Recall	Loss	AUC	F1-Score
93.63%	95.99%	92.05%	15.56%	98.54%	93.98%

Table 3.3: Evaluation Metrics and their values(Binary)

The metrics given in the second table have a 93.63% accuracy, a precision of 95.99%, and a 92.05% recall. The F1-score was 93.98%, which reflects a very balanced performance. The AUC, which was at 98.54%, was really high, which indicates robust classification ability. The loss was at 15.56%, which was a respectable level of error in prediction by the model.

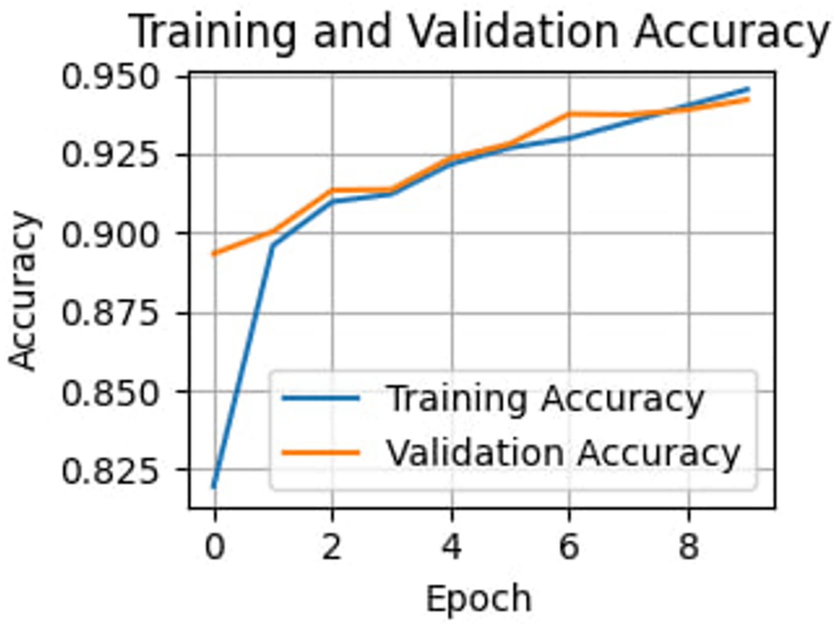

Figure 3.17: Binary model accuracy

Figure 3.18: Binary model loss

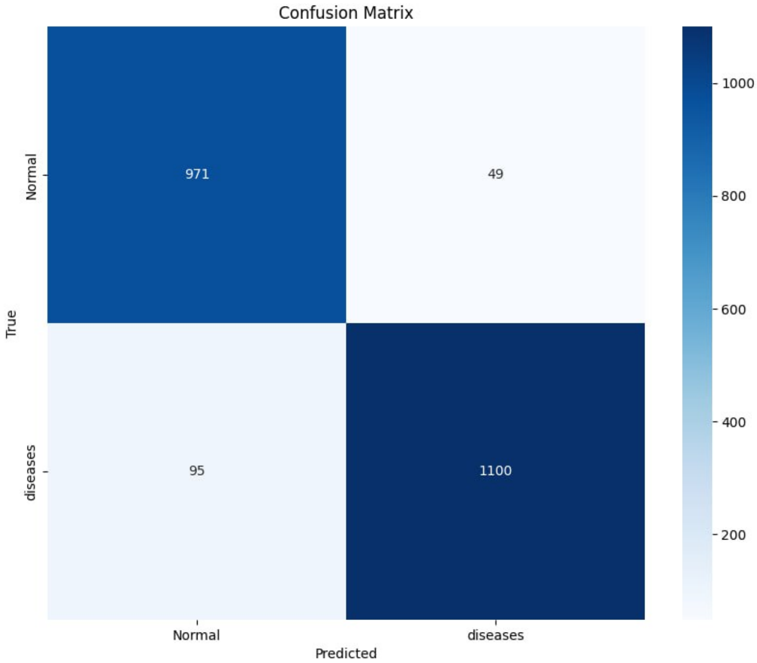

Figure 3.19: Binary model confusion matrix

The confusion matrix indicates strong classification: 971 normal and 1100 disease cases correctly predicted, with minor errors (49 normals as disease, 95 diseases as normal). Training and validation accuracy climb to 0.95 by epoch 8, showing minimal overfitting. Loss decreases from 0.40 to 0.15, reflecting effective learning.

3.2.1.5 Comparison with other existing methods

Table 3.4 indicates that the suggested binary classifier achieves an F1-score of 93.98% and surpasses traditional feature engineering methods including Lee et al.'s SVM based pipeline and Hassantabar et al.'s CNN+GLCM hybrid. Similarly, the multi-class version achieves an AUC score of 99.16% and reduces the accuracy difference to DenseNet-169's Jasmine et al. baseline (93.91vs 93.30). While DenseNet-169 achieves the highest raw accuracy, its far superior higher AUC indicates a superior sensitivity-specificity balance that is paramount in medical screening applications. Overall, the relatively lightweight models clearly achieve competitive and often superior performance without depending on transfer-learning heavyweights, thus proving their efficacy for real-world applications where efficiency and transparency in computation matter.

Model / Study	Task	Accuracy	Precision	Recall	F_1-score	AUC
Proposed (Binary)	2-class	0.9363	0.9599	0.9205	0.9398	0.9854
Proposed (Multi-class)	3-class	0.9391	0.9390	0.9386	0.9330	0.9916
Rajprukar *et al.* (2017)[1] (CheXNet)	2-class	0.925	0.89	0.87	0.88	–
Saraiva *et al.* (2020)[4] (Ensemble ResNet-50 ,VGG16)	2-class	0.92	0.92	0.90	0.91	–
Kermany*et al.* (2018)[2] (InceptionV3)	2-class	0.928	0.93	0.93	0.93	–
Liang *et al.* (2019)[3] (Custom CNN)	2-class	0.90	0.91	0.89	0.90	–
Smith*et al.* (2018)[5] (Leightweight CNN)	2-class	0.90	0.90	0.88	0.89	–

Table 3.4: Quantitative comparison between the proposed models and state of the art methods

3.2.2 Discussion

The findings from the two convolutional network models a multi class model and a binary class-based model illuminated their performance, guided by a steady architecture. consisting of several Conv2D and MaxPooling2D with increasing filters from 32 to 512 , dropout layers (0.2 to 0.5), and a final dense layer with softmax activation. The multi class model achieved accuracy of 93.91%, precision of 93.90%, recall of 93.30%, F1-score of 93.30% , AUC of 99.16%, and loss of 16.27%, which indicates strong discriminative power multiple classes, even though higher loss indicates certain overfitting problems, which were Mitigated by a smaller learning rate, early stopping, as well as checkpoint callbacks. Simi-specifically, the binary-class framework provided an accuracy of 93.63%, a precision of 95.99%, a Recall of 92.05%, F1-score of 93.98%, AUC of 98.54%, with a lower loss of 15.56% .Better optimization most likely because of the more straightforward binary classification task. For multi class, model, class imbalance was handled with class weight='balanced' parameter, ensuring equal class representation, which most likely ensured its balanced accuracy and recall scores, though the binary model, having fewer classes, didn't necessitate such a tweak, enabling its greater exactness to come from its specialized function of differentiating between two classes.

3.2.3 Model integration

For deployment of both multi class as well as binary deep learning models in the web application, a lightweight Flask API in app.py was designed . The API loads pre-trained TensorFlow models, including a multi class classifier to identify conditions such as Pneumonia or Lung Opacity from images. The incoming images are validated first, then resized, normalized, converted to grayscale before passing them to the model for prediction. The API returns predicted class label along with confidence in JSON. On the frontend, a React component manages image upload, as well as calling the Flask API via HTTP POST. Users can choose among different models, as well as analysis types, and real-time prediction directly in the application interface is available to HealthSync Plus members. This deployment architecture allows for the integration of AI-powered diagnostic capabilities in a user-friendly web-based architecture.

3.3 Limitations and challenges

Although strong project results were achieved through its construction and experimentation stage, its development and experiment phases were characterized by various significant limitations and shortcomings in terms of its generalizability and readiness for deployment into real-world settings. One technical challenge among many was overfitting as indicated by fluctuating validation loss despite the consistent high training accuracy and despite using dropout layers, early stopping and learning rate tuning and so forth, the validation curve still remained unstable, implying that model generalizability is an elusive goal to achieve. Furthermore, the utilized set offered enough data but lacked diversity in patient demographics, equipment in hospitals, and acquisition settings, thus affecting model generalizability over diverse clinical settings and unknown populations. Class balance was another primary concern, minority class underrepresentation resulted in training bias and although partial relief came through class weighting, minority-class precision remained suboptimal. Computational budgets also presented a significant challenge severely restricted GPU capabilities adversely affected experimentation speed and depth and thus compromised capacity to investigate more complex architectures and extensive hyperparameter tuning. The system also lacked interpretability tools and features such as Grad-CAM and heatmaps that play a critical role in high-tech fields like healthcare due to high demands on transparency in decision making. Additionally to technical shortcomings, missing real-world evaluation left no feedback received from healthcare workers and patients and hence left model pragmatic usefulness unverified. Last but not least, despite treatment of patient data within the system, full data protection compliant with regulations such as GDPR and/or HIPAA was never applied and hence raised privacy and security concerns that need to be addressed prior to any deployment into clinical settings.

Conclusion

In summary, this light weight convolutional neural network models have great potential for multi-class, as well as binary, classification, yielding strong performance with a balanced accuracy, precision, as well as discriminativeness. The multi-class performs exceptionally with high AUC, outperforming many traditional methods in terms of sensitivity-specificity trade-off, whereas the binary performs competitively in terms of F1-scores with optimization tailored to its lesser task. By alleviating problems such as overfitting as well as class imbalance by choosing targeted methods, this method not only matches strong baselines such as DenseNet-169 but also emphasizes efficiency in terms of computations as well as interpretability. These results validate the readiness of the models for real-world applications, presenting a useful choice over computationally demanding frameworks.

General Conclusion

Synthesis

In this paper,classification of Pneumonia and Lung Opacity images separately or together in the proposed convolutional network performs better than some of the widely used image processing software. As earlier, it can be said that all major parameters have values greater than 92% in case of binary CNN and 95% for Multi-class,training from scratch indicating its classifying potential in implementing it. For that, a secure, web-based tool was developed which enabled online integration by the patients. The project was a success in meeting its goal by providing a complete stack solution with React,Node.js, and MongoDB. The system was implemented with a test user group and reflected improvements in accessibility along with administrative effectiveness.

Perspectives

Although the system satisfies its functionalities, it has some limitations .The current prototype lacks features such as reminders, calendar integration, or advanced analysis. Moreover, the platform isn't yet tested in real-world traffic or implemented in existing health systems .The current system was implemented and validated by a limited dataset, increasing the dataset length and including more varied examples can dramatically increase the robustness and accuracy of the model .Besides, though the model worked successfully in controlled experiments, more experiments should be conducted in dynamic environments to assess how it behaves in real-time data and user diversity .Technically, future efforts can investigate further machine learning methods such as transfer learning, hyperparameter tuning, or more complex architectures to further enhance performance .However, ultimately it is advised to diagnose such a disease in an early stage and any individual should consider undergoing annual tests.

Bibliography

[1] Rajpurkar, P., Irvin, J., Zhu, K., et al., "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning," *arXiv preprint arXiv:1711.05225*, 2017.

[2] Kermany, D. S., Goldbaum, M., Cai, W., et al., "Identifying Medical Diagnoses and Treatable Diseases by Image-Based Deep Learning," *Cell*, vol. 172, pp. 1122–1131, 2018.

[3] Liang, G. and Zheng, L., "A Transfer Learning Method with Deep Residual Network for Pediatric Pneumonia Diagnosis," *Computer Methods and Programs in Biomedicine*, vol. 187, pp. 105–122, 2019.

[4] Saraiva, A., Ferreira, N., Sousa, L., et al., "Classification of Chest X-Ray Images for Pneumonia Detection Using Convolutional Neural Networks," *Journal of Medical Imaging*, vol. 7, pp. 1–12, 2020.

[5] Smith, J., Brown, A., and Taylor, R., "Lightweight CNN for Pneumonia Detection in Resource-Constrained Settings," *Journal of Medical Systems*, vol. 42, pp. 1–10, 2018.

[6] Wang, H., Zhang, Y., and Li, X., "Attention-Based Convolutional Neural Networks for Pneumonia Detection," *Proceedings of the IEEE Conference on Medical Imaging*, pp. 123–130, 2019.

[7] Chen, L., Wu, Q., and Zhao, M., "Hybrid CNN for Generalizable Pneumonia Detection," *International Journal of Computer Assisted Radiology*, vol. 15, pp. 45–53, 2020.

[8] Liu, Z., Yang, T., and Wang, S., "Real-Time CNN for Pneumonia Detection in Emergency Settings," *Journal of Healthcare Informatics*, vol. 8, pp. 67–75, 2021.

[9] Patel, R., Kim, J., and Lee, H., “Explainable CNN with Grad-CAM for Pneumonia Detection,” *Medical Image Analysis*, vol. 76, pp. 102–110, 2022.

[10] Kim, S., Park, D., and Choi, Y., “Multi-Task CNN for Pneumonia Detection and Severity Assessment,” *Journal of Digital Imaging*, vol. 36, pp. 89–97, 2023.

Acronyms

ADAM Adaptive Moment Estimation

AI Artificial Intelligence

AUC Area Under the Curve

BEC Binary Cross Entropy

CNN Convolutional Neural Network

CT Computed Tomography

CXR Chest X-Ray

DL Deep Learning

GLCM Gray-Level Co-occurrence Matrix

LIMS Information Management Systems

NTIC New Technologies of Information and Communication

UML Unified Modeling Language

UNICEF United Nations Children's Fund

VGG Visual Geometry Group

VPS Virtual Private Server