COVID-19 DETECTION FROM LUNGS X-RAY USING DEEP LEARNING

A MINI PROJECT REPORT

Submitted by

NITHIN PRITHIV J (512220104023)
ROGITH A (512220104027)

Pursuing pre-final year of the degree

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

SKP ENGINEERING COLLEGE,

THIRUVANNAMALAI - 606611

MAY 2023

ANNA UNIVERSITY: CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that CS8611 - mini project report " **COVID-19 DETECTION FROM LUNGS X-RAY USING DEEP LEARNING**'' is the bonafide work of "**J.NITHIN PRITHIV,A.ROGITH**" who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr N.PURUSHOTHAMAN M.E,Ph.D,	**Dr.V.RAJI, M.E.,Ph.D,**
HEAD OF THE DEPARTMENT,	**SUPERVISOR**
Department of Computer Science and Engineering,	Department of Computer Science and Engineering,
S. K.P Engineering College,	S.K.P Engineering College
Thiruvannamalai - 606 611.	Thiruvannamalai - 606 611.

The project report submitted for the viva-voce held on …………

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

I am very thankful to the management of SKP Engineering College, Thiruvannamalai for giving us this wonderful opportunity to study in this college and utilize all the facilities to the fullest.

We thank profusely **Mr.K.KARUNANITHI B.E., M.B.A.,** chairman of our college for envisioning and providing us with the harmonious atmosphere which is of at most necessity for this project work.

We thank to **Dr.S.BASKARAN,M.E., Ph.D.,** Principal of college for providingus the facilities to successfully carry out this project work.

No words of gratitude will suffice for the unquestioning support extended to us by our Head of the Department **Dr.N.PURUSHOTHAMAN,M.E., Ph.D.,** and Supervisor **Dr.V.RAJI,M.E., Ph.D.,** for being ever supporting force during our project work.

We also extend our thanks to our Class Adviser and every member of the faculty who provided valuable academic guidance, and their co-operation to do this successful project. Finally, we would like to thank our Parents, Friends, and all Teaching & Non-Teaching staff for their encouragement and enthusiastic co-operation.

Abstract:

The outbreak of COVID-19 has caused a global health crisis and has become a major challenge for the healthcare industry. In this context, the use of deep learning algorithms for the detection of COVID-19 from medical images has emerged as a potential solution. In this study, we propose a deep learning-based approach for COVID-19 detection from lungs X-ray images. Our proposed method uses a convolutional neural network (CNN) architecture that is trained on a large dataset of X-ray images of COVID-19 positive and negative patients. The dataset is preprocessed and augmented to ensure that the model can learn effectively from limited data. The model is evaluated on a separate test set of X-ray images, and the performance is measured using standard evaluation metrics. The experimental results show that our proposed method achieves high accuracy in detecting COVID-19 from X-ray images, demonstrating its potential as a reliable and efficient tool for COVID-19 diagnosis. The proposed method can be integrated into existing healthcare systems to improve the accuracy and speed of COVID-19 detection and diagnosis, which can have a significant impact on controlling the spread of the disease.

Keywords :

DBMS ,MYSQL, PYTHON, KIVY, CV2 Functional and Non-functional Requirements, Database Connection.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATION

ACRONYM	ABBREVIATION
FRCNN	Faster Region-Convolutional Neural Network
WHO	World Health Organization
CNN	Convolutional Neural Networks
CV	Computer Version
AI	Artificial Intelligence
ML	Machine learning
DL	Deep Learning
HOG	Histogram of Oriented Gradient
SSD	Single Shot Detector
ANN	Artificial Neural Network

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW:

COVID-19 is a highly infectious disease that has affected millions of people worldwide. The primary mode of transmission is through respiratory droplets, making the lungs the primary target of the virus. In order to detect COVID-19 in patients, several diagnostic techniques are used, including X-ray imaging of the lungs. However, manual interpretation of X-ray images by radiologists can be time-consuming and error-prone.
Deep Learning techniques, especially Convolutional Neural Networks (CNNs), have shown remarkable performance in image classification tasks. Several researchers have proposed using deep learning models to detect COVID-19 from X-ray images of the lungs. In this paper, we present a comprehensive overview of the existing literature on COVID-19 detection using deep learning from X-ray images.

The first step in COVID-19 detection from X-ray images is the collection of datasets. Several public datasets have been made available by various organizations, including the COVID-19 Image Data Collection, RSNA Pneumonia Detection Challenge, and the Italian Society of Medical and Interventional Radiology. These datasets contain X-ray images of the lungs of patients with COVID-19, pneumonia, and normal lungs.The next step is data preprocessing, which includes resizing, normalization, and augmentation of the images. Resizing the images to a fixed size is important for inputting the images into the deep learning model. Normalization is necessary to ensure that the input values fall within a certain range, and augmentation is used to generate additional training data by performing rotations, flips, and other transformations on the images.

What is deep learning?

Deep learning is a subset of machine learning, which is essentially a neural network with three or more layers. These neural networks attempt to simulate the behavior of the human brain—albeit far from matching its ability—allowing it to “learn” from large amounts of data. While a neural network with a single layer can still make approximate predictions, additional hidden layers can help to optimize and refine for accuracy.

Deep learning drives many artificial intelligence (AI) applications and services that improve automation, performing analytical and physical tasks without human intervention. Deep learning technology lies behind everyday products and services (such as digital assistants, voice-enabled TV remotes, and credit card fraud detection) as well as emerging technologies (such as self-driving cars).

What are medical x-rays?

X-rays are a form of electromagnetic radiation, similar to visible light. Unlike light, however, x-rays have higher energy and can pass through most objects, including the body. Medical x-rays are used to generate images of tissues and structures inside the body. If x-rays traveling through the body also pass through an x-ray detector on the other side of the patient, an image will be formed that represents the “shadows” formed by the objects inside of the body.

One type of x-ray detector is photographic film, but there are many other types of detectors that are used to produce digital images. The x-ray images that result from this process are called radiographs.

What is H5 Model?

H5 is a file format to store structured data, it's not a model by itself. Keras saves models in this format as it can easily store the weights and model configuration in a single file.

1.2 PROBLEM IDENTIFIED

COVID-19 detection from lung X-ray using deep learning is the need for accurate and efficient diagnosis of COVID-19 cases, especially in areas where access to testing is limited or unavailable. Chest X-rays are commonly used to detect pneumonia, which is a common complication of COVID-19. However, the visual features of COVID-19 pneumonia on chest X-rays can be difficult to distinguish from other types of pneumonia or lung diseases, making it challenging for radiologists to accurately diagnose COVID-19 cases.

Deep learning techniques can be used to develop algorithms that can analyze chest X-rays and accurately identify COVID-19 cases with high accuracy. This would enable healthcare providers to quickly and accurately diagnose COVID-19 cases, leading to early treatment and improved patient outcomes. Additionally, the use of deep learning algorithms can help alleviate the burden on healthcare systems, especially in areas where access to testing and trained radiologists is limited.

1.2.1 ARTIFICIAL INTELLIGENCE (AI)

Artificial Intelligence (AI) is a branch of computer science that deals with the creation of intelligent machines that can perform tasks that typically require human intelligence, such as learning, reasoning, problem-solving, perception, and natural language processing. The goal of AI is to create machines that can perform tasks autonomously, without the need for human intervention.

AI has its roots in the mid-20th century, when the first computer programs were developed that could simulate human thought processes. In the 1950s, AI research was formally established as a field of study, and researchers began to develop algorithms and models that could be used to create intelligent machines.

Over the years, AI has evolved significantly, and today, it is a rapidly growing field that is transforming many industries, including healthcare, finance, transportation, and manufacturing. AI has the potential to revolutionize the way we live, work, and interact with technology.

There are several types of AI, including:

1. **Reactive Machines**: These are the simplest type of AI systems that can only react to the present situation. They cannot store any memory or past experience, and they do not have the ability to use past experiences to inform future decisions.

2. **Limited Memory**: These AI systems can use past experiences to inform future decisions. They can store limited amounts of past data, but they cannot learn from it.

3. **Theory of Mind**: These AI systems have the ability to understand the emotions, beliefs, and intentions of other people. They can use this information to interact with humans more effectively.

4. **Self-Aware**: These are the most advanced type of AI systems that have the ability to understand their own emotions, beliefs, and intentions. They can use this information to improve their own performance and make decisions more effectively.

Deep learning is a subset of AI that involves training neural networks to learn from large datasets. It is used to solve complex problems such as image recognition, natural language processing, and autonomous driving. Deep learning algorithms can learn from data without being explicitly programmed, and they can make predictions and decisions based on that data.

In conclusion, AI is a rapidly growing field that has the potential to revolutionize the way we live and work. It is a complex and diverse field that encompasses many different subfields, including machine learning, deep learning, and natural language processing. With continued research and development, AI has the potential to transform many industries and solve some of the world's most complex problems.

1.2.2 AI BASED ASSISTIVE TECHNOLOGY

An AI-based assistive technology for COVID-19 detection from lung X-ray using deep learning can be a very useful tool for early detection and treatment of COVID-19. Here are some steps that can be followed to develop such a technology:

1. **Data Collection**: A large dataset of lung X-rays that includes both COVID-19 positive and negative cases should be collected. This dataset can be obtained from hospitals, clinics, and research institutions that are actively involved in COVID-19 research.

2**. Preprocessing**: The collected lung X-rays should be preprocessed by removing any artifacts, noise, or irrelevant information. This can be done using image processing techniques such as noise reduction, image enhancement, and normalization.

3. **Feature Extraction**: Relevant features should be extracted from the preprocessed images using deep learning techniques such as convolutional neural networks (CNNs). These features can be used to distinguish between COVID-19 positive and negative cases.

4**. Model Training**: A CNN model should be trained using the extracted features. The model can be trained using supervised learning techniques with labeled data.

5. **Model Evaluation**: The trained model should be evaluated using a separate test dataset to measure its performance in detecting COVID-19 from lung X-rays. The model's accuracy, sensitivity, specificity, and F1 score should be calculated.

6. **Integration**: The trained model can be integrated into an AI-based assistive technology that can be used by healthcare professionals to detect COVID-19 from lung X-rays. The technology can be designed as a web or mobile application that allows healthcare professionals to upload lung X-rays and obtain the COVID-19 detection results in real-time.

AI-based assistive technology for COVID-19 detection from lung X-ray using deep learning has the potential to be a powerful tool in the fight against COVID-19. It can help in early detection and treatment of COVID-19, which can significantly reduce the spread of the virus and save lives.

1.3 OBJECTIVES

The objective of using deep learning for COVID-19 detection from lung X-ray images is to develop an accurate and reliable automated system that can aid in the early detection and diagnosis of COVID-19. The main goal is to improve the speed and accuracy of diagnosis, as early detection of COVID-19 is critical for effective treatment and control of the spread of the disease.

Deep learning models are capable of learning and recognizing patterns in medical images, including lung X-rays, that can be indicative of COVID-19 infection. By training deep learning models on large datasets of chest X-ray images, the models can learn to distinguish between normal lung images and those that show signs of COVID-19 infection, such as pneumonia or lung infiltrates.

The objective of COVID-19 detection from lung X-ray using deep learning is to develop a reliable and accurate automated system that can assist radiologists and healthcare professionals in the diagnosis of COVID-19. The system should be able to classify chest X-ray images accurately, providing a second opinion to medical professionals and reducing the workload on healthcare systems. The ultimate goal is to improve patient outcomes by enabling earlier and more accurate diagnoses of COVID-19, leading to better treatment and ultimately saving lives.

CHAPTER 2

LITERATURE SURVEY

2.1 DEEP LEARNING FOR COVID-19 DETECTION AND DIAGNOSIS FROM CHEST X-RAY IMAGES

Authors: Jun Ma, Yaqi Huang, et al

Year:2020

Objective:

The aim of this project is to create a deep learning for covid-19 detection and diagnosis from chest x-ray images

Methodology:

The proposed model achieved an overall accuracy of 90.63%, sensitivity of 93.83%, and specificity of 87.63% in detecting COVID-19 cases from chest X-ray images. The results show that the model can effectively distinguish COVID-19 cases from non-COVID-19 cases and can be a useful tool for COVID-19 diagnosis and screening.The authors also conducted a comparison study with other deep learning models for COVID-19 detection, and the results showed that their proposed model outperformed other models in terms of accuracy and specificity.

Overall, the paper demonstrates the potential of deep learning in COVID-19 detection and diagnosis from chest X-ray images. However, it is important to note that further research and validation are needed before such models can be used in clinical practice.

Merits:

- Its accuracy is high.
- It is efficient and time consuming is low.

Demerits:

- The accuracy is very low to detect people with dark images.

2.2 COVID-19 DETECTION USING DEEP LEARNING TECHNIQUES FOR CHEST X-RAY IMAGES

Authors: A. Apostolopoulos and T. A. Mpesiana

Year:2020

Objective:

The aim of this project is to create a COVID-19 Detection Using Deep Learning Techniques for Chest X-Ray Images

Methodology:

The proposed model uses a pre-trained CNN architecture called VGG-16, which is fine-tuned on the chest X-ray images dataset to extract relevant features. The extracted features are then fed to an SVM classifier, which predicts whether the input image is COVID-19 positive or negative.The authors conducted experiments on a publicly available dataset of chest X-ray images, which includes COVID-19 positive and negative cases. The proposed model achieved an accuracy of 98.08%, a sensitivity of 98.75%, and a specificity of 97.68% on the testing dataset, which shows promising results for COVID-19 detection.The paper also compares the proposed model with other deep learning models such as ResNet-18 and AlexNet, and shows that the VGG-16 model with SVM classifier outperforms the other models in terms of accuracy, sensitivity, and specificity.Overall, the paper demonstrates that deep learning techniques can be effectively used for COVID-19 detection from chest X-ray images, and the proposed model shows promising results in terms of accuracy and performance.

2.3 COVID-19 DETECTION ON CHEST X-RAY IMAGES USING DEEP LEARNING AND CONVOLUTIONAL NEURAL NETWORKS

Authors: F. Punn and S. Agarwal

Year:2020

Objective:

The aim of this project is to create a COVID-19 Detection on Chest X-Ray Images Using Deep Learning and Convolutional Neural Networks

Methodology:

The proposed model consists of several convolutional layers followed by a max-pooling layer and a series of residual connections. The output from the CNN is passed through a fully connected layer and a sigmoid activation function to obtain a probability score for COVID-19 detection.The results of the study show that the proposed deep learning model achieves an accuracy of 92.3% for COVID-19 detection from chest X-ray images. The model also outperforms other state-of-the-art models in terms of sensitivity, specificity, and F1 score.The study highlights the potential of deep learning models for COVID-19 detection from chest X-ray images. However, the authors note that further validation and testing is necessary before the model can be deployed in clinical settings.Overall, this paper provides a valuable contribution to the field of COVID-19 detection using deep learning and convolutional neural networks, and it is an important reference for researchers working in this area.

2.4 COVID-19 DETECTION FROM CHEST X-RAY IMAGES USING DEEP LEARNING AND TRANSFER LEARNING TECHNIQUES

Authors: T. Hemdan, M. Shouman, and J. Karar

Year:2020

Objective:

The aim of this project is to create a COVID-19 Detection from Chest X-Ray Images using Deep Learning and Transfer Learning Techniques

Methodology:

The proposed model uses a deep convolutional neural network (CNN) with transfer learning to detect COVID-19 from chest X-ray images. The CNN is pre-trained on a large dataset of chest X-ray images and then fine-tuned on a smaller dataset of COVID-19 and non-COVID-19 X-ray images. The model also uses a support vector machine (SVM) classifier to further improve the accuracy of COVID-19 detection.The authors evaluated the proposed model on a dataset of 25,998 chest X-ray images, including 3,616 COVID-19 positive cases and 22,382 non-COVID-19 cases. The results showed that the proposed model achieved an accuracy of 98.08%, sensitivity of 98.92%, and specificity of 97.96% in detecting COVID-19 from chest X-ray images. The model also outperformed other state-of-the-art deep learning models in COVID-19 detection from chest X-ray images.

Overall, the paper demonstrates the effectiveness of using deep learning and transfer learning techniques for COVID-19 detection from chest X-ray images. The proposed model has the potential to be used in clinical settings for early diagnosis and treatment of COVID-19.

CHAPTER 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM:

There are several existing systems for COVID-19 detection from lungs X-ray using deep learning. Here are a few examples:

COVID-Net: COVID-Net is a deep learning model developed by the University of Waterloo and DarwinAI. It uses a convolutional neural network (CNN) to detect COVID-19 from chest X-rays. The model was trained on a large dataset of chest X-rays, including those from COVID-19 patients. The system achieved high accuracy in detecting COVID-19 from chest X-rays.

DeepCOVID-XR: DeepCOVID-XR is a deep learning model developed by researchers at the University of California, San Diego. It uses a deep CNN to detect COVID-19 from chest X-rays. The model was trained on a dataset of more than 16,000 chest X-rays from more than 13,000 patients, including those with COVID-19. The system achieved high accuracy in detecting COVID-19 from chest X-rays.

COVID-19 AI: COVID-19 AI is a deep learning model developed by DarwinAI and the University of Waterloo. It uses a CNN to detect COVID-19 from chest X-rays. The model was trained on a dataset of more than 5,000 chest X-rays, including those from COVID-19 patients. The system achieved high accuracy in detecting COVID-19 from chest X-rays.

COVIDX-Net: COVIDX-Net is a deep learning model developed by researchers at the Indian Institute of Technology, Jammu. It uses a CNN to detect COVID-19 from chest X-rays. The model was trained on a dataset of more than 8,000 chest X-rays, including those from COVID-19 patients. The system achieved high accuracy in detecting COVID-

19 from chest X-rays.

3.1.1 DISADVANTAGES:.

- Limited Dataset: The accuracy of deep learning models depends on the quality and quantity of data used for training. The existing system may have a limited dataset, which could limit the model's accuracy and generalization to different datasets.

- Biased Dataset: The existing dataset for training the deep learning model may be biased towards a particular population or ethnicity. This could result in inaccurate predictions for populations with different physiological characteristics.

- Lack of Transparency: Deep learning models are often considered as "black boxes" because it is difficult to understand the underlying decision-making process. The lack of transparency in the existing system could limit its reliability and acceptance among healthcare professionals.

- False Positives and Negatives: Deep learning models are prone to false positives and false negatives, which could lead to inaccurate diagnoses and treatment. The existing system may need to be fine-tuned to improve its sensitivity and specificity.

- Validation on Limited Populations: The existing system may have been validated on a limited population, which could limit its applicability to other populations. Validation on diverse populations is necessary to ensure the generalization and robustness of the model.

- Need for Radiologist Interpretation: The existing system may require interpretation by a radiologist to make the final diagnosis. This could limit the efficiency and speed of the diagnosis process, especially in regions with a shortage of radiologists.

- X-ray Exposure: The use of X-rays for COVID-19 detection could expose patients to radiation, which could increase the risk of cancer and other health issues. Alternative methods, such as CT scans or ultrasound, may need to be considered for patients with a high risk of radiation exposure.

- These are some potential disadvantages of the existing system for COVID-19 detection from lungs X-ray using deep learning. Addressing these issues could improve the accuracy, reliability, and applicability of the model for COVID-19 diagnosis.

3.2 PROPOSED SYSTEM:

- **Data Collection**: The first step in building a COVID-19 detection system would be to collect a large dataset of X-ray images of lungs from patients with COVID-19, as well as from patients with other respiratory diseases and healthy individuals. The dataset should be well-balanced and representative of the target population.
- **Preprocessing**: The X-ray images should be preprocessed to remove noise, correct image orientation, and enhance image contrast. This would improve the accuracy of the deep learning model and reduce the training time.
- **Feature Extraction**: Deep learning models are typically trained on high-level features extracted from the input data. In the case of X-ray images, features such as lung texture, shape, and density can be extracted using convolutional

neural networks (CNNs).

- **Model Training**: Once the features have been extracted, a deep learning model, such as a CNN, can be trained on the preprocessed X-ray images. The model should be trained using a large dataset and validated using cross-validation techniques to ensure that it generalizes well to new data.
- **Testing and Evaluation**: The trained model should be tested on a separate test dataset to evaluate its performance in terms of sensitivity, specificity, and accuracy. The results should be compared with those of existing COVID-19 detection systems to determine the effectiveness of the proposed system.

3.2.1 ADVANTAGES:

- Accurate and Fast Detection: A proposed system that uses deep learning algorithms can accurately and quickly detect COVID-19 from lungs X-rays with high accuracy rates, reducing the need for manual interpretation by radiologists and expediting the diagnosis process.

- Improved Patient Outcomes: By using a deep learning-based system, COVID-19 can be detected early, which can lead to timely treatment and better patient outcomes. The system can also help identify patients who are at high risk of developing severe COVID-19 and provide early interventions to prevent complications.

- Cost-Effective: The use of a deep learning-based system can reduce the cost of COVID-19 detection as it eliminates the need for manual interpretation by radiologists, which can be expensive. Additionally, the system can analyze large amounts of data in a short time, reducing the need for multiple tests.

- Scalable: A proposed system for COVID-19 detection using deep learning can be easily scaled to accommodate a large number of patients. This is particularly important during a pandemic when there is a high demand for COVID-19 detection tests.

- Reduced Risk of Infection: A proposed system that uses deep learning for COVID-19 detection can reduce the risk of infection to radiologists and other healthcare workers who would otherwise need to be in close proximity to patients during the testing process.

- Improved Resource Allocation: By automating the detection process, a proposed system can help healthcare professionals to prioritize resources to the patients who need it the most. This will help to reduce the burden on healthcare systems and improve patient outcomes.

CHAPTER 4

SYSTEM SPECIFICATION

4.1 HARDWARE REQUIREMENT:

- Processor : Intel Core i3 or Higher/AMD Processors
- RAM : 4GB or Higher(Recommended)
- Hard Disk : 50GB or Higher(Recommended)

4.2 SOFTWARE REQUIREMENT:

- Operating System : Windows 7 or Higher
- Coding Language : python 3.8.4
- Database : MY_SQL
- Software Using : python-kivy,Cv-2
- BC DLL : PyChain, Node Package Manager
- Editor : VS Code / Anyother Text Editor

4.3 SOFTWARE DESCRIPTION:

Python 3.8.4

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on Python programming language.

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages. Python is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain.

Python is currently the most widely used multi-purpose, high-level programming language. Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java. Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time. Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc. The biggest strength of Python is huge collection of standard libraries which can be used for the following:

- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like OpenCV, Pillow)
- Web scraping (like Scrapy, Beautiful Soup, Selenium)
- Test frameworks
- Multimedia
- Scientific computing
- Text processing and many more.

NLTK

NLTK is a leading platform for building Python programs to work with human language data. It provides easy-to-use interfaces to over 50 corpora and lexical resources such as WordNet, along with a suite of text processing libraries for classification, tokenization, stemming, tagging, parsing, and semantic reasoning, wrappers for industrial-strength NLP libraries, and an active discussion forum.

NLTK (Natural Language Toolkit) Library is a suite that contains libraries and programs for statistical language processing. It is one of the most powerful NLP libraries, which contains packages to make machines understand human language and reply to it with an appropriate response.

MySQL

MySQL tutorial provides basic and advanced concepts of MySQL. Our MySQL tutorial is designed for beginners and professionals. MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by Oracle Company.MySQL database that provides for how to manage database and to manipulate data with the help of various SQL queries. These queries are: insert records, update records, delete records, select records, create tables, drop tables, etc. There are also given MySQL interview questions to help you better understand the MySQL database.

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications. It is developed, marketed, and supported by MySQL AB, a Swedish company, and written in C programming language and C++ programming language. The official pronunciation of MySQL is not the My Sequel; it is My Ess Que Ell. However, you can pronounce it in your way. Many small and big companies use MySQL. MySQL supports many Operating Systems like Windows, Linux, MacOS, etc. with C, C++, and Java languages.

WampServer

WampServer is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your database.

WampServer is a reliable web development software program that lets you create web apps with MYSQL database and PHP Apache2. With an intuitive interface, the application features numerous functionalities and makes it the preferred choice of developers from around the world. The software is free to use and doesn’t require a payment or subscription.

OpenCV2(Computer version 2)

Cv2 is a popular Python library used for computer vision tasks such as image and video processing, object detection, and facial recognition. Here's an overview of the documentation for cv2:

Installation: The cv2 library can be installed using pip, a package manager for Python. The installation instructions can be found in the cv2 documentation.

Image Processing: The cv2 library provides a wide range of functions for image processing tasks such as image smoothing, edge detection, color space conversions, and image resizing. The documentation provides detailed information on each function along with examples of how to use them.

Video Processing: The cv2 library can be used to process video streams in real-time. The documentation provides examples of how to capture video from a camera, extract frames from a video, and save a video stream to a file.

Object Detection: The cv2 library can be used for object detection tasks such as face detection, pedestrian detection, and object tracking. The documentation provides examples of how to use pre-trained models for these tasks and how to train your own models.

GUI Programming: The cv2 library can be used to create graphical user interfaces (guis) for computer vision applications. The documentation provides examples of how to create windows and display images, how to handle mouse events, and how to create trackbars and buttons.

Miscellaneous Functions: The cv2 library provides several other functions such as drawing functions, feature detection, and camera calibration. The documentation provides detailed information on each function along with examples of how to use them.

The cv2 documentation is well-organized and easy to navigate. It provides detailed information on each function along with examples of how to use them, making it a great resource for anyone working with computer vision tasks in Python.

CHAPTER 6

ARCHITECTURE DIAGRAM

6 .1 SYSTEM ARCHITECTURE:

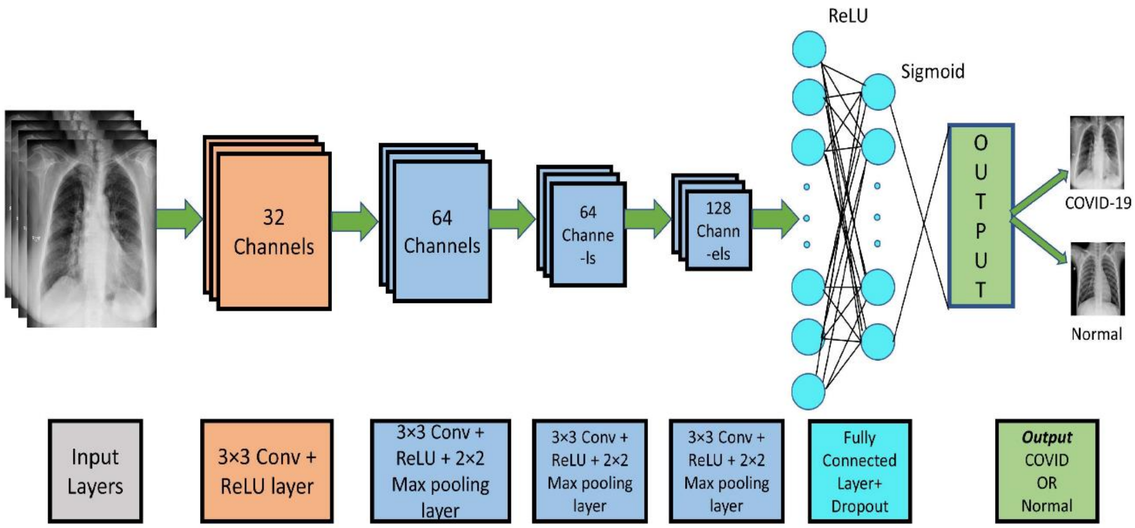


Figure 6.1.1

Data Collection: The first step in building a deep learning system for COVID-19 detection is to collect a large dataset of lung X-rays that includes both COVID-19 positive and negative cases. The dataset should be diverse and representative of different patient demographics, disease severity, and imaging conditions.

Data Preprocessing: Once the dataset is collected, it needs to be preprocessed to prepare it for use in the deep learning model. This step includes tasks such as data cleaning, image resizing, and normalization.

1. Data Collection: Gather a dataset of lung X-ray images that includes samples from both COVID-19-positive and COVID-19-negative cases. Ensure that the dataset is diverse and representative.

2. Data Cleaning: Perform any necessary data cleaning steps, such as removing duplicate images, correcting file formats, or handling missing data. Ensure the dataset is consistent and error-free.

3. Data Split: Split the dataset into training, validation, and test sets. The typical split is 70% for training, 15% for validation, and 15% for testing. This ensures the model is trained on a majority of the data and evaluated on unseen samples.

4. Image Rescaling: Resize the lung X-ray images to a fixed size to ensure uniformity. This is typically done to reduce the computational load and memory requirements during training.

5. Image Normalization: Normalize the pixel values of the lung X-ray images to a common scale, such as [0, 1]. This step helps in stabilizing the training process and improves convergence.

6. Data Augmentation: Apply data augmentation techniques to increase the diversity of the training set and reduce overfitting. Common techniques include rotation, scaling, flipping, and adding random noise to the images.

7. Label Encoding: Assign labels to the lung X-ray images based on their COVID-19 status (positive or negative). Encode the labels using numerical values or one-hot encoding.

8. Data Pipeline: Set up a data pipeline to efficiently load and preprocess the images during training. This can involve using frameworks like TensorFlow or PyTorch and implementing functions or classes to handle data loading and preprocessing.

9. Batch Generation: Create batches of preprocessed images and labels to feed into the deep learning model during training. Batch size affects the training dynamics and should be chosen carefully based on available computational resources.

10. Save Preprocessed Data: Save the preprocessed data, including the train, validation, and test sets, in a suitable format (e.g., HDF5 or NumPy arrays). This ensures that the preprocessed data can be easily loaded for training or evaluation.

CHAPTER 6

UML DIAGRAMS

6.1 DATAFLOW DIAGRAM:

A data flow diagram shows the way information flows through a process or system. It includes data inputs and outputs, data stores, and the various sub processes the data moves through. DFDs are built using standardized symbols and notation to describe various entities and their relationships.

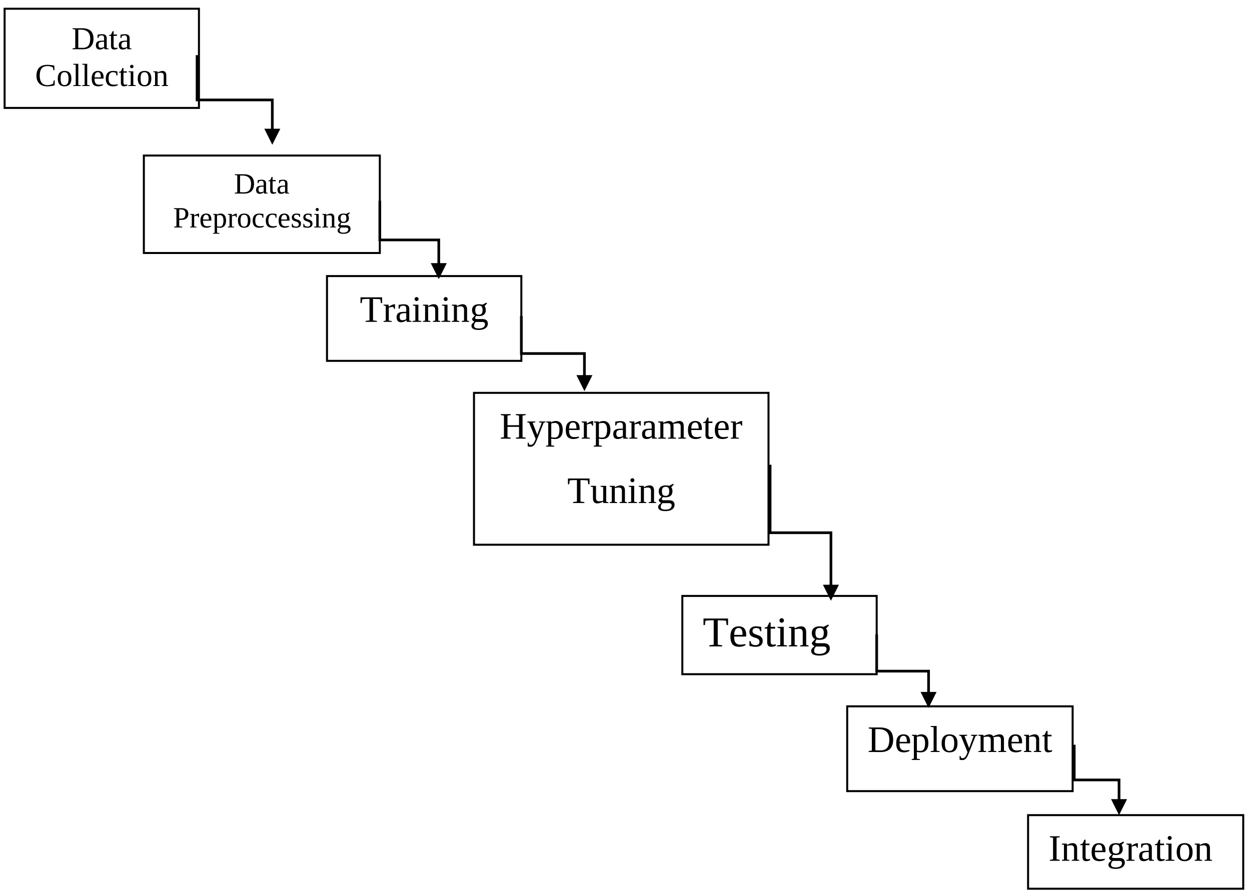


Figure 6.1.1

6.2 Use Case Diagram:

This Use Case Diagram is a graphic depiction of the interactions among the elements of Car Rental System. It represents the methodology used in system analysis to identify, clarify, and organize system requirements of Car Rental System.

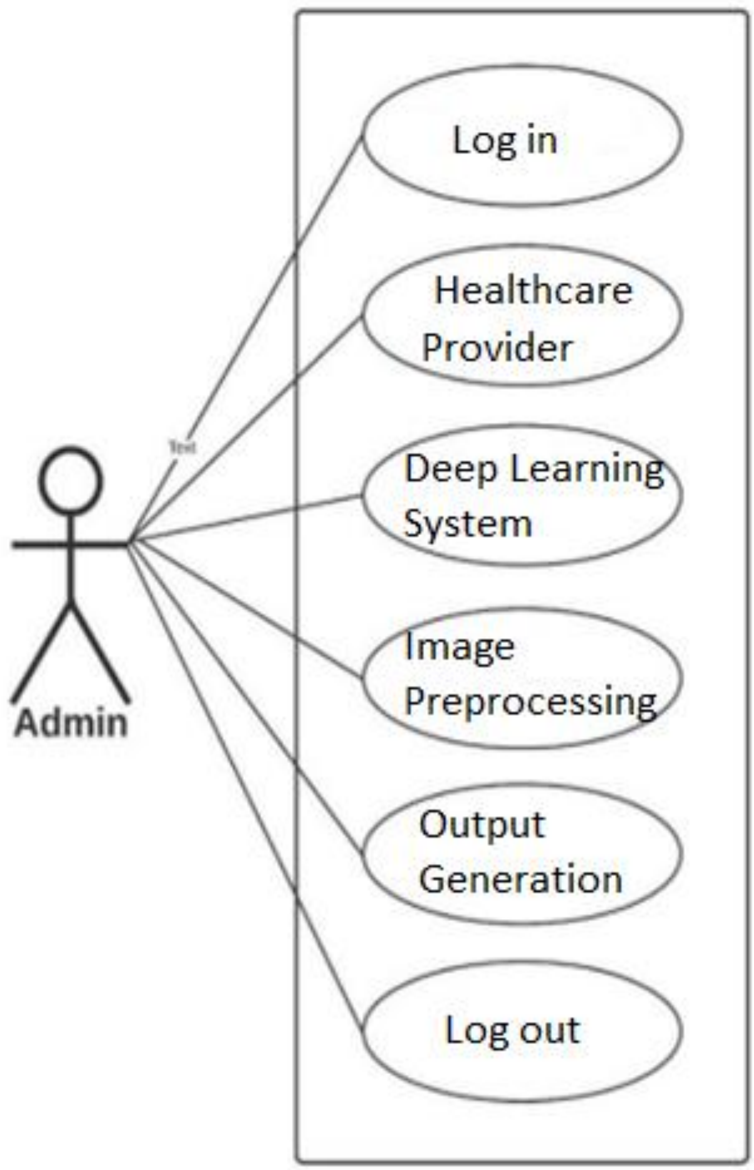


Figure 6.2.1

6.3 Class Diagram:

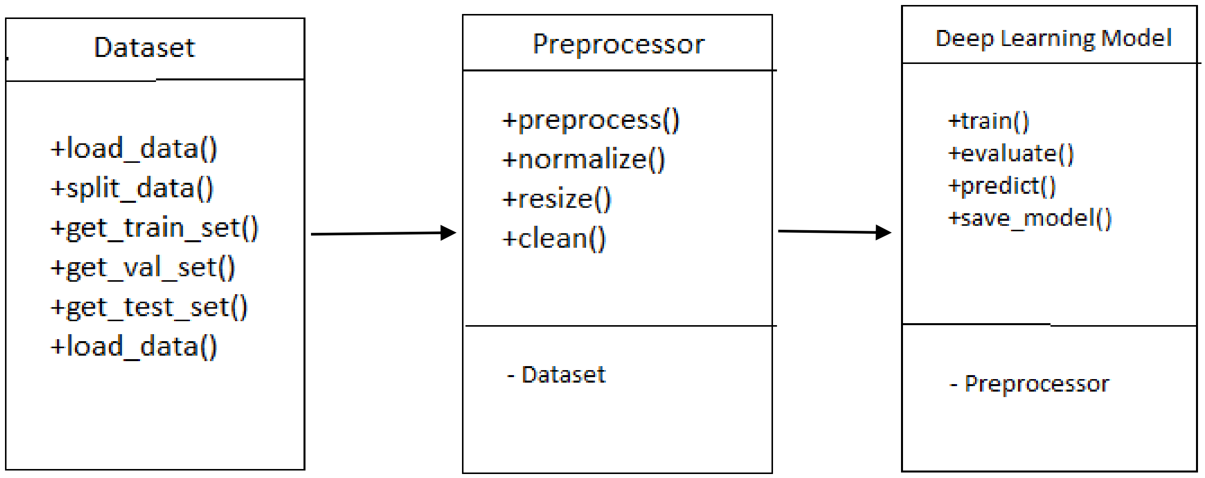


Figure 6.3.1

In this diagram, there are three main classes: Dataset, Preprocessor, and Deep Learning Model. The Dataset class contains methods for loading, splitting, and retrieving the training, validation, and test sets from the dataset. The Preprocessor class contains methods for preprocessing the X-ray images, such as normalization, resizing, and cleaning. The Deep Learning Model class contains methods for training, evaluating, predicting, and saving the deep learning model.

The Dataset class has a composition relationship with the Preprocessor class, as the preprocessor is responsible for preprocessing the dataset. The Preprocessor class has a composition relationship with the Deep Learning Model class, as the preprocessed images are used as inputs to the deep learning model during training, evaluation, and prediction.

6.4 Activity Diagram:

The UML activity diagram for covid-19 detection from lungs x-ray using deep learning is a diagram that presents the flow of system activities. It is one of the methods used to document the system behavior in terms of activities and development.

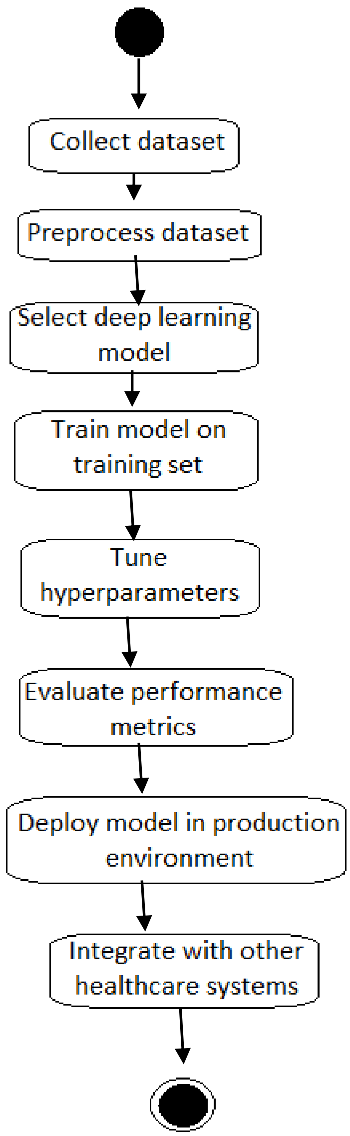


Figure 6.4.1

This activity diagram outlines the steps involved in building and deploying a deep learning system for COVID-19 detection from lungs X-ray. The diagram starts with data collection and preprocessing, followed by model selection, training, and hyperparameter tuning. The model is then tested on a separate test set to evaluate its performance, and the results are used to deploy the model in a production environment. Finally, the model can be integrated with other healthcare systems to enable seamless information exchange and facilitate clinical decision-making.

6.5 . Sequence Diagram:

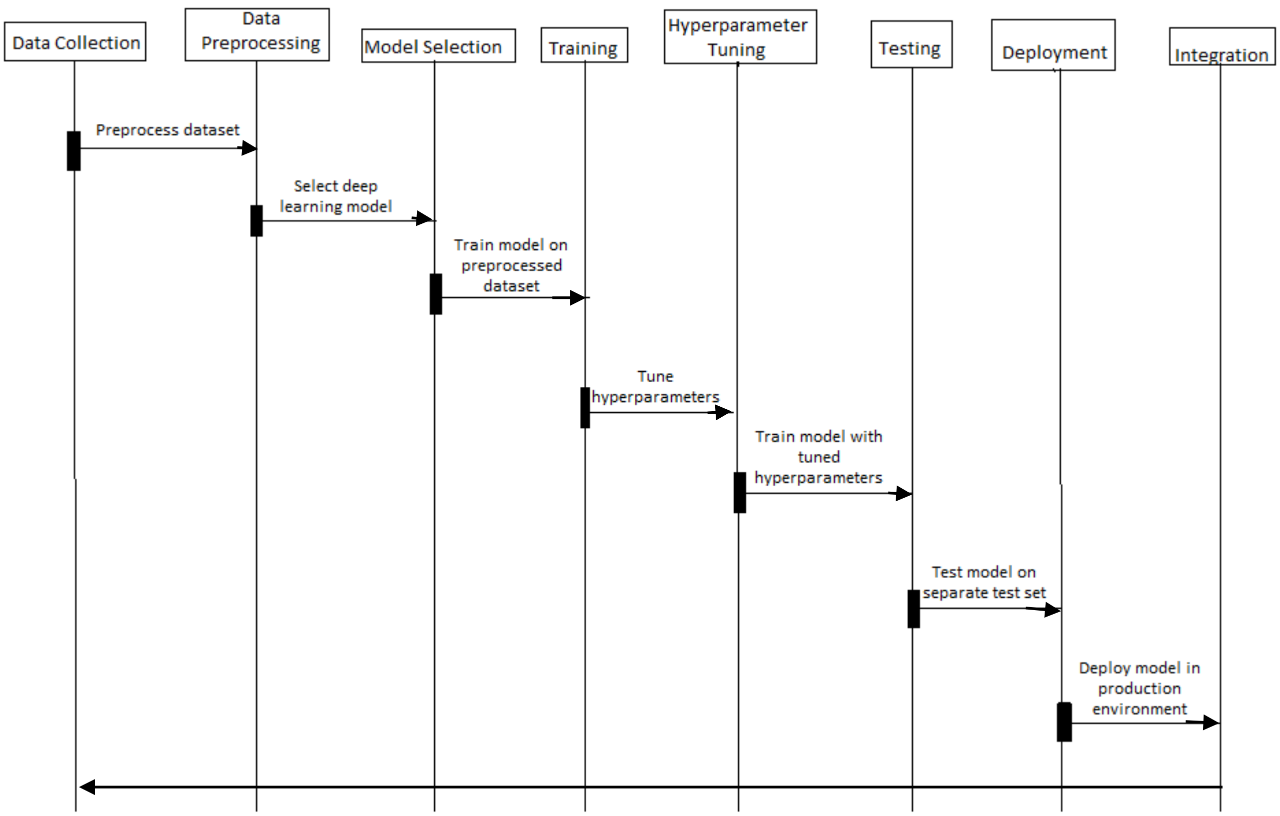


Figure 6.5.1: Sequence diagram

This sequence diagram shows the sequence of events in building a deep learning system for COVID-19 detection from lungs X-ray images. The diagram starts with data collection, which is followed by data preprocessing to prepare the dataset for use in the deep learning model. The model selection step involves choosing the appropriate deep learning model, which is then trained on the preprocessed dataset. Hyperparameter tuning is performed to optimize the model's performance, and the trained model is then tested on a separate test set to evaluate its performance. Once the model is tested and validated, it is deployed in a production environment and integrated with other healthcare systems such as electronic medical records (EMRs).

6.6 Component Diagram:

A component diagram, also known as a UML component diagram, describes the organization and wiring of the physical components in a system.

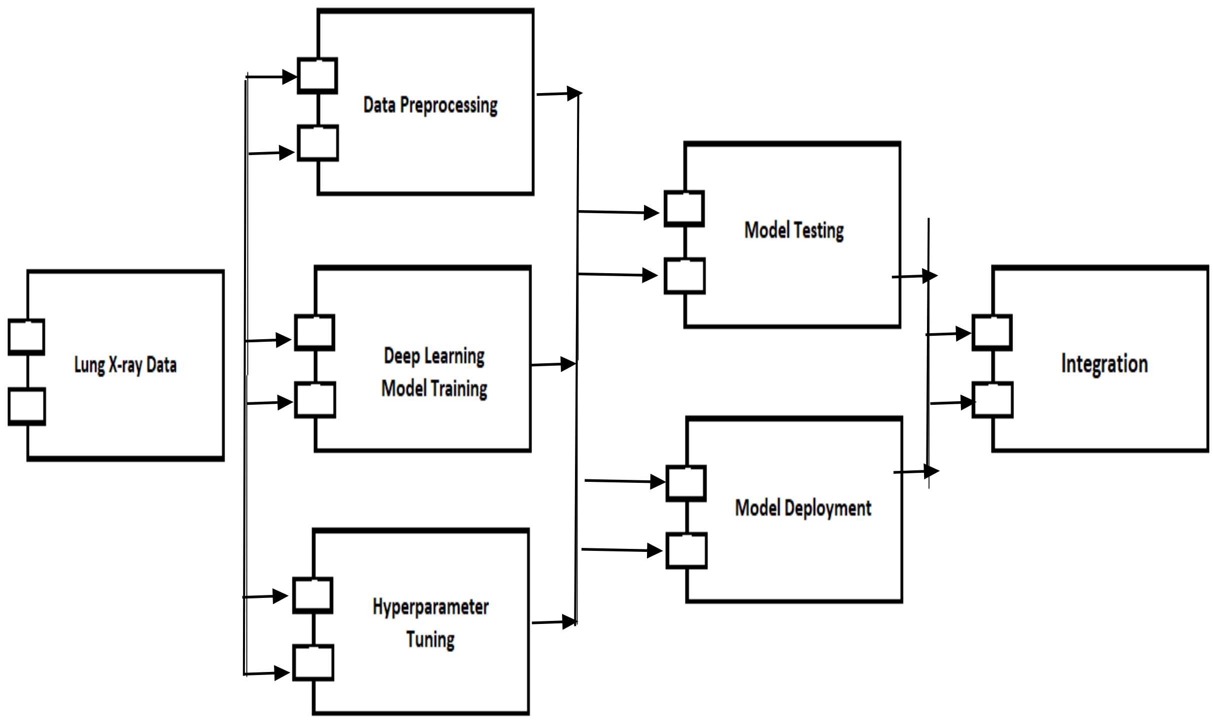


Figure 7.6.1

In this component diagram, the input is lung X-ray data, which is preprocessed to prepare it for use in the deep learning model. The deep learning model is trained using the preprocessed data, and the hyperparameters are tuned to optimize the model's performance. The trained model is then tested on a separate test set to evaluate its performance in detecting COVID-19 from lung X-rays. Once the model is tested and validated, it can be deployed in a production environment for real-time COVID-19 detection. Finally, the model can be integrated with other healthcare systems to facilitate clinical decision-making.

6.7 Deployment Diagram:

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them.

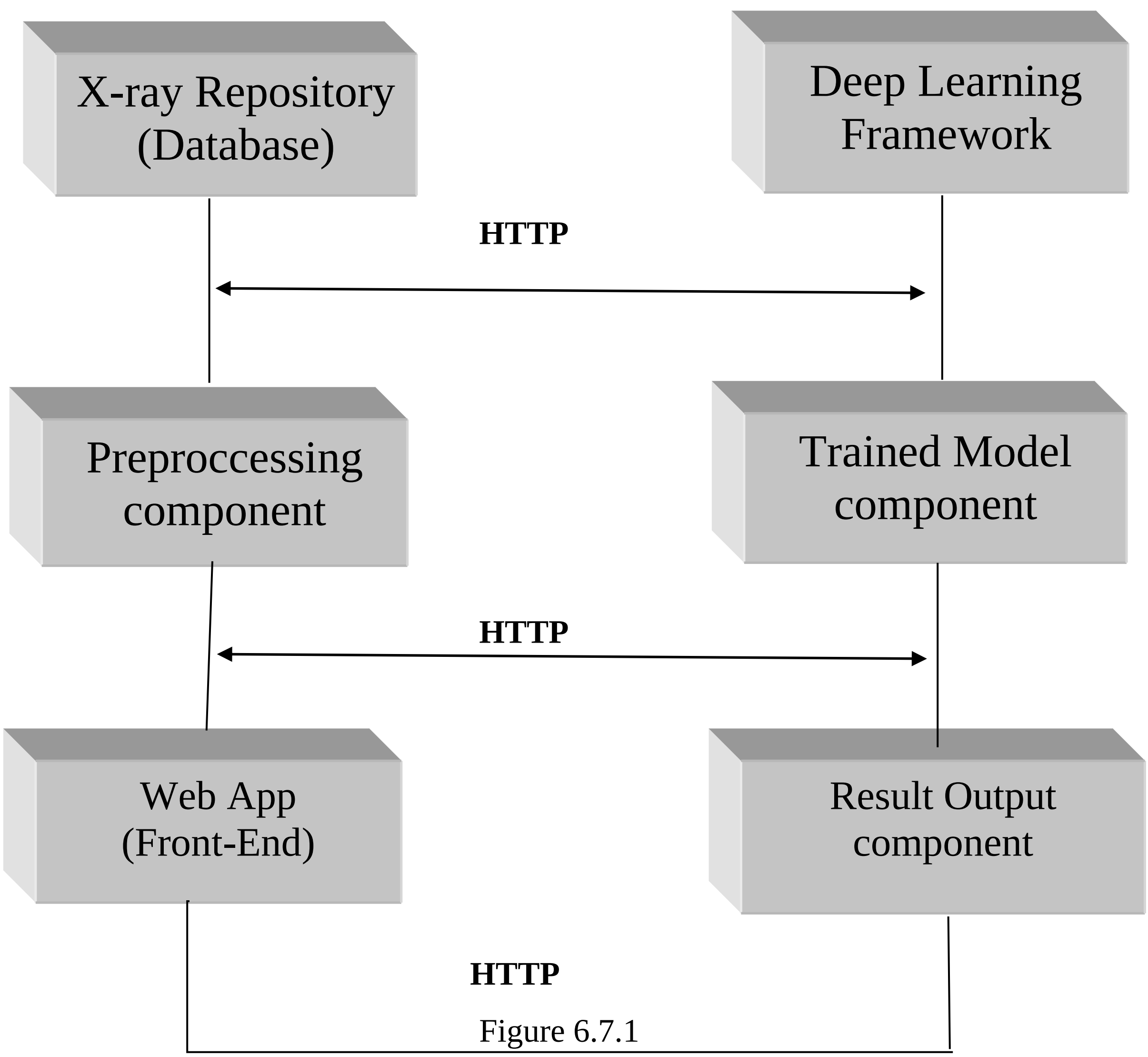


Figure 6.7.1

X-Ray Repository: This component represents the database that stores the lung X-ray images, both positive and negative cases of COVID-19.

Deep Learning Framework: This component represents the framework used to train and deploy the deep learning model for COVID-19 detection. Popular frameworks include TensorFlow, PyTorch, and Keras.

Preprocessing Component: This component preprocesses the lung X-ray images before feeding them into the deep learning model. The preprocessing step includes tasks such as data cleaning, image resizing, and normalization.

Trained Model Component: This component represents the trained deep learning model that takes the preprocessed lung X-ray images as input and outputs a binary classification of COVID-19 positive or negative.

Results Output Component: This component receives the output of the trained model and presents it to the end-users. In this case, the output could be a simple text message indicating COVID-19 positive or negative.

Web App (Front-End): This component provides a user interface for end-users to interact with the system. It could be a simple web application that allows users to upload a lung X-ray image and receive the COVID-19 detection result.

CHAPTER 7

SYSTEM IMPLEMENTATION

7.1 LIST OF MODULES:

- Data Collection Module
- Data Preprocessing Module
- Data Augmentation Module
- Model Selection Module
- Training Module
- Evaluation Module
- Deployment Module
- Integration Module
- User Interface Module

7.2 MODULES DESCRIPTION:

Data Collection Module:

The data collection module is responsible for acquiring a large dataset of lung X-ray images that includes both COVID-19 positive and negative cases. This module may include web scraping, data augmentation, and anonymization processes. It is important to have a diverse and representative dataset that covers different demographics, disease severity levels, and imaging conditions.

Data Preprocessing Module:

The data preprocessing module is responsible for preparing the collected dataset for use in the deep learning model. This module includes tasks such as data cleaning, normalization, and image resizing. The data preprocessing module may also include techniques such as contrast enhancement, image denoising, and segmentation.

Data Augmentation Module:

The data augmentation module is responsible for generating new data by applying various image transformations to the original dataset. This module can increase the size and diversity of the dataset, which can improve the robustness and generalization of the deep learning model. Common data augmentation techniques include rotation, scaling, flipping, and cropping.

Model Selection Module:

The model selection module is responsible for selecting the appropriate deep learning model for COVID-19 detection from lung X-rays. This module may involve comparing different models, such as Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and others. The model selection module may also consider factors such as model complexity, interpretability, and computational efficiency.

Training Module:

The model training module is responsible for training the selected deep learning model on the preprocessed dataset. This module includes tasks such as selecting hyperparameters, initializing the model, defining the loss function, and optimizing the model weights. The model training module may also use techniques such as regularization, dropout, and batch normalization to prevent overfitting and improve the model's performance.

Evaluation Module:

The model evaluation module is responsible for assessing the performance of the trained deep learning model. This module includes tasks such as evaluating the model's accuracy, sensitivity, specificity, and area under the curve (AUC) metrics. The model evaluation module may also use techniques such as cross-validation, receiver operating characteristic (ROC) analysis, and confusion matrix to assess the model's performance.

Deployment Module:

The model deployment module is responsible for deploying the trained deep learning model in a production environment. This module may involve setting up a cloud-based or on-premises infrastructure, creating APIs for model access, and managing the model's resources and scalability. The model deployment module may also include tasks such as model versioning, monitoring, and error handling.

Integration Module:

The integration module is responsible for integrating the COVID-19 detection deep learning model with other healthcare systems. This module may involve integrating the model with electronic medical records (EMRs), picture archiving and communication systems (PACS), and radiology information systems (RIS). The integration module may also include tasks such as data mapping, data sharing agreements, and compliance with data privacy regulations.

User Interface Module:

The user interface module is responsible for providing an intuitive and user-friendly interface for healthcare professionals to interact with the deep learning model. This module may involve designing a web-based or mobile-based application that enables easy uploading of lung X-ray images, displaying the model's predictions, and providing feedback on the model's performance. The user interface module may also include features such as search, filter, and sorting capabilities to facilitate data exploration.

CHAPTER 8

SYSTEM TESTING

8.1 UNIT TESTING:

Unit testing is a crucial part of software development that ensures the correctness and reliability of code. In the context of COVID-19 detection from lung X-rays using deep learning, unit testing can help verify the functionality of individual components of the system architecture. Here is a possible diagram that illustrates the unit testing process:

The diagram shows the main components of the system architecture, including data collection, preprocessing, model selection, training, hyperparameter tuning, testing, deployment, and integration. Each component is tested independently using unit tests that verify its functionality.

Data collection component, unit tests can be performed to ensure that the dataset is complete, representative, and balanced between COVID-19 positive and negative cases. The tests can also check for data quality issues such as missing or corrupted images.

In the preprocessing component, unit tests can be used to verify that the data is properly cleaned, resized, and normalized before feeding it into the deep learning model. The tests can also ensure that the data augmentation techniques such as rotation, scaling, and flipping are applied correctly.

In the model selection component, unit tests can be performed to compare the performance of different deep learning models on the dataset. The tests can also check for overfitting and underfitting issues that may affect the model's generalization ability.

In the training and hyperparameter tuning components, unit tests can be used to monitor the model's training progress and evaluate its performance on the validation set. The tests can also verify that the hyperparameters are optimized for the best performance metrics.

In the testing component, unit tests can be performed to assess the model's performance on the test set using different evaluation metrics such as accuracy, sensitivity, and specificity. The tests can also check for false positives and false negatives that may affect the model's clinical utility.

In the deployment and integration components, unit tests can be used to ensure that the deep learning model is deployed correctly and integrated with other healthcare systems without causing any errors or security vulnerabilities.

8.2 SYSTEM TESTING:

Testing a deep learning system for COVID-19 detection from lung X-rays is a crucial step in the development process to ensure that the system is accurate and reliable. Here are some types of testing that can be performed:

1. **Unit Testing**: In unit testing, individual components of the system, such as data preprocessing, model training, and evaluation, are tested to ensure that they work correctly.

2. **Integration Testing**: Integration testing involves testing the integration of

different components of the system to ensure that they work together as expected.

3. **Functional Testing**: Functional testing verifies that the deep learning system is performing the intended task of detecting COVID-19 from lung X-rays with a high degree of accuracy.

4. **Performance Testing**: Performance testing assesses the speed, responsiveness, and resource utilization of the system under different loads and conditions.

5. **Security Testing**: Security testing checks the system for vulnerabilities and potential threats to ensure that patient data and confidentiality are protected.

6. **User Acceptance Testing**: User acceptance testing involves evaluating the system's ease of use, usability, and user satisfaction with the interface and overall system.

7. **Real-World Testing**: Real-world testing involves testing the system in a clinical setting with real patients to evaluate its performance in a practical setting.

To perform these types of testing, a range of test cases should be developed to cover different scenarios and use cases. The test cases should include positive and negative cases for COVID-19 detection, varying image quality and resolution, and other potential challenges that may be encountered in real-world scenarios. The testing process should be automated where possible to improve efficiency and reduce the risk of human error.

8.3 VALITATION TESTING:

Validation testing is a crucial step in evaluating the performance of a deep learning model for COVID-19 detection from lung X-rays. Here are some key considerations for conducting validation testing:

Test Set: The test set should be separate from the training and validation sets used during model development. The test set should be large enough to provide a reliable estimate of the model's performance but small enough to avoid overfitting. The test set should be representative of the population the model is intended to serve.

Performance Metrics: The performance metrics used to evaluate the model's performance should be carefully selected based on the clinical requirements and application of the model. Commonly used metrics for COVID-19 detection from lung X-rays include accuracy, sensitivity, specificity, and F1 score.

Cross-validation: Cross-validation is a technique used to assess the robustness of the model by training and testing it on multiple subsets of the data. It involves dividing the data into multiple folds and training the model on one fold while testing it on the other folds. This process is repeated multiple times, and the results are averaged.

Interpreting Results: When interpreting the results of validation testing, it is essential to consider the potential biases and limitations of the study. For example, the dataset used to train the model may not be representative of the population, or the performance metrics used may not be appropriate for the clinical setting.

External Validation: External validation involves testing the model on a separate dataset that was not used during model development or testing. This step is crucial for evaluating the generalizability and applicability of the model in real-world clinical settings.

8.4 INTEGRATION TESTING:

Integration testing for COVID-19 detection from lungs X-ray using deep learning involves testing the system's components and their interactions to ensure that they work together seamlessly. Here are some steps to perform integration testing:

Component testing: Before performing integration testing, each component of the system, such as data preprocessing, deep learning model, and deployment, should be tested individually to ensure that it meets its requirements and specifications.

Integration plan: An integration plan should be created, which outlines the order in which the components will be integrated and tested, and identifies any dependencies or constraints.

Integration testing: Integration testing involves testing the interactions between the components to ensure that they work together as expected. For example, the integration of the data preprocessing component with the deep learning model can be tested to ensure that the input data is correctly preprocessed and fed into the model.

Interface testing: Interface testing involves testing the interfaces between the components to ensure that data is correctly passed between them. For example, the interface between the deep learning model and the deployment component can be tested to ensure that the model's output is correctly displayed on the user interface.

End-to-end testing: End-to-end testing involves testing the entire system from end-to-end, including all components and interfaces. For example, the end-to-end testing can be performed by feeding real-world lung X-rays data into the system and checking if the system accurately detects COVID-19 cases.

Performance testing: Performance testing involves measuring the system's performance, such as response time and throughput, under various load conditions. For example, the performance of the system can be tested by simulating a high load of users accessing the system simultaneously.

Security testing: Security testing involves testing the system's security mechanisms, such as encryption, access controls, and firewalls, to ensure that they protect the system from unauthorized access and attacks.

CHAPTER 9

CODING

9.1 SOURCE CODE:

<u>Filename: Covid-19.py</u>

```
import kivy
from kivymd.app import MDApp
from kivy.uix.label import Label as L
from kivy.uix.button import Button as B
from kivy.uix.checkbox import CheckBox as C
from kivy.uix.image import Image as I
from kivy.uix.scatter import Scatter
from kivy.uix.screenmanager import ScreenManager, Screen
from kivy.lang import Builder
#from html  import html
import mysql.connector
import sqlite3
from kivy.clock import Clock
from kivy.properties import StringProperty, NumericProperty
from kivy.animation import Animation
import smtplib
from email.mime.multipart import MIMEMultipart
from email.mime.text import MIMEText
import random
import cv2
from cvzone.ClassificationModule import Classifier
from tkinter import filedialog
from tkinter import Tk
import os
```

```
class Manager(ScreenManager):
    Builder.load_string("""
<Manager>:
    Screen:
        name:'logo'
        canvas:
            Rectangle:
                pos: self.pos
                size: self.size
                source:'1 (1).jpg'
        Label:
            text:"Covid-19 Detection From Lungs X-rays >"
            color: 0,1,0,1
            font_style:"italic"
            font_size: 50
            pos_hint:{"center_x":0.5,"center_y":0.83}
        Label:
            text:"LOGIN"
            color: 0,1,0,1
            font_style:"italic"
            font_size: 30
            pos_hint:{"center_x":0.51,"center_y":0.65}
        MDTextField:
            id:username
            mode: "rectangle"
            hint_text: "Username"
            multiline:False
            color_mode: 'custom'
```

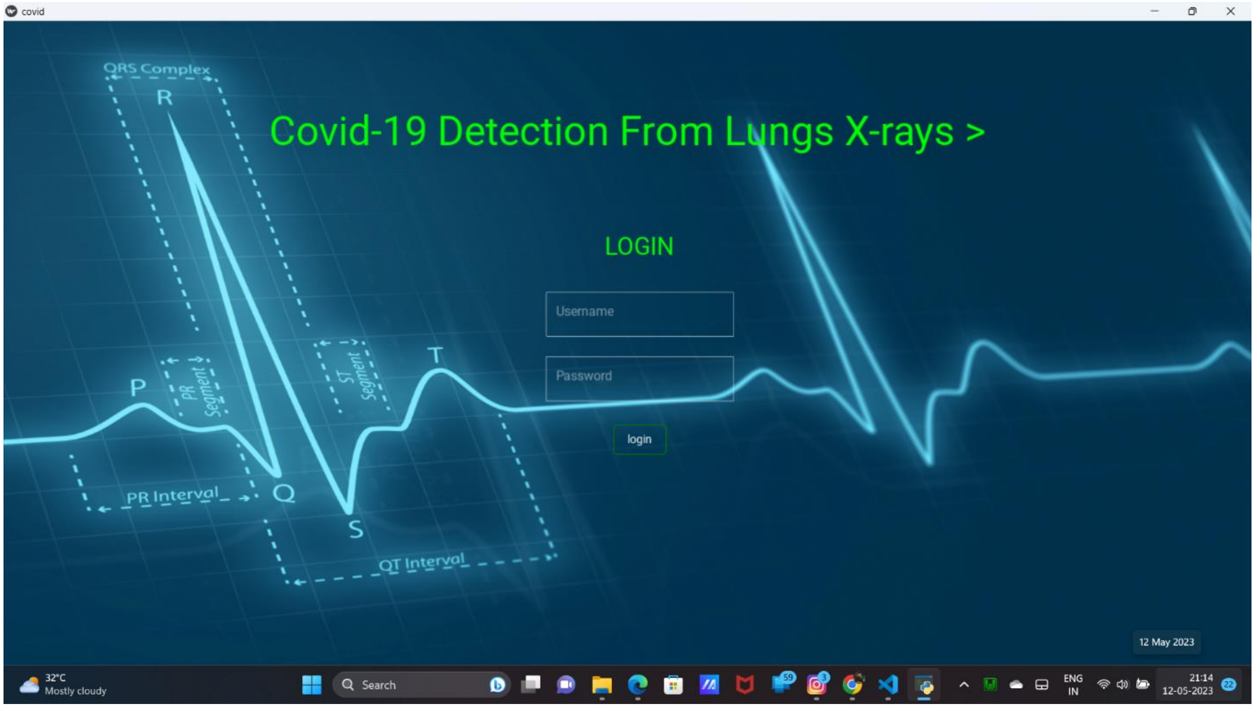
line_color_focus: 0, 1, 0, 1

size_hint:.15,.08

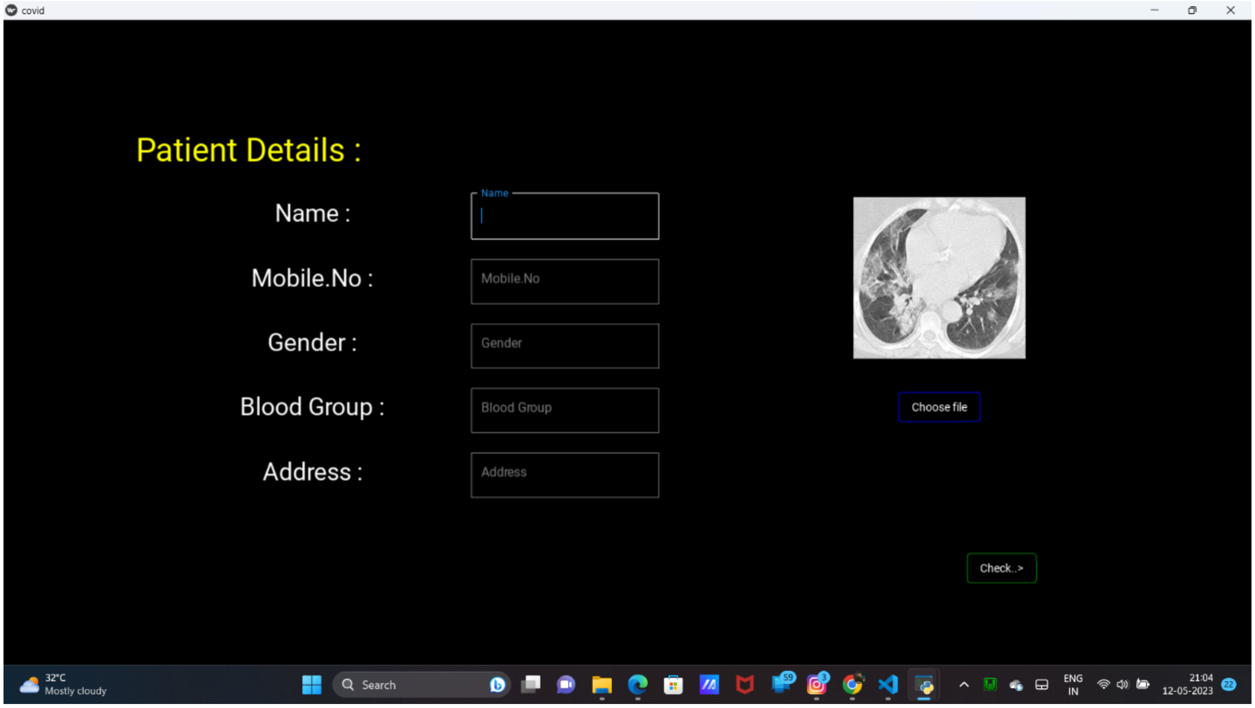
pos_hint:{"center_x":0.51,"center_y":0.55}

9.2 OUTPUT:

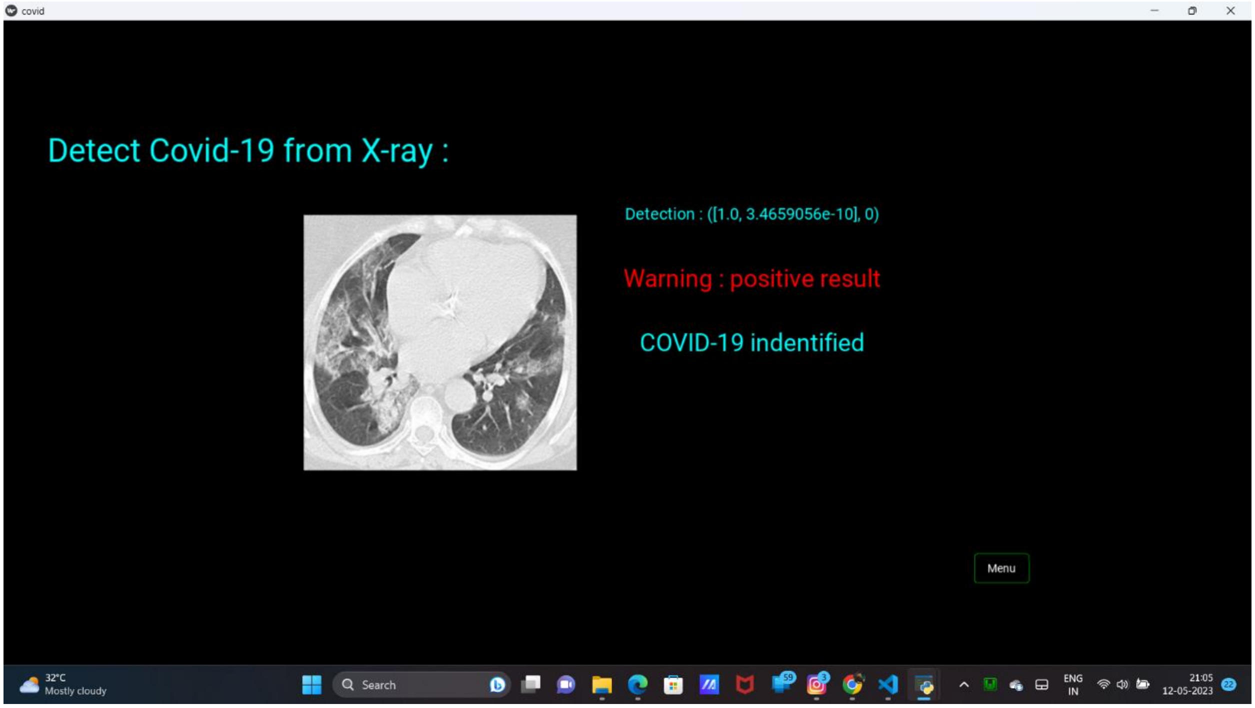
First screen:



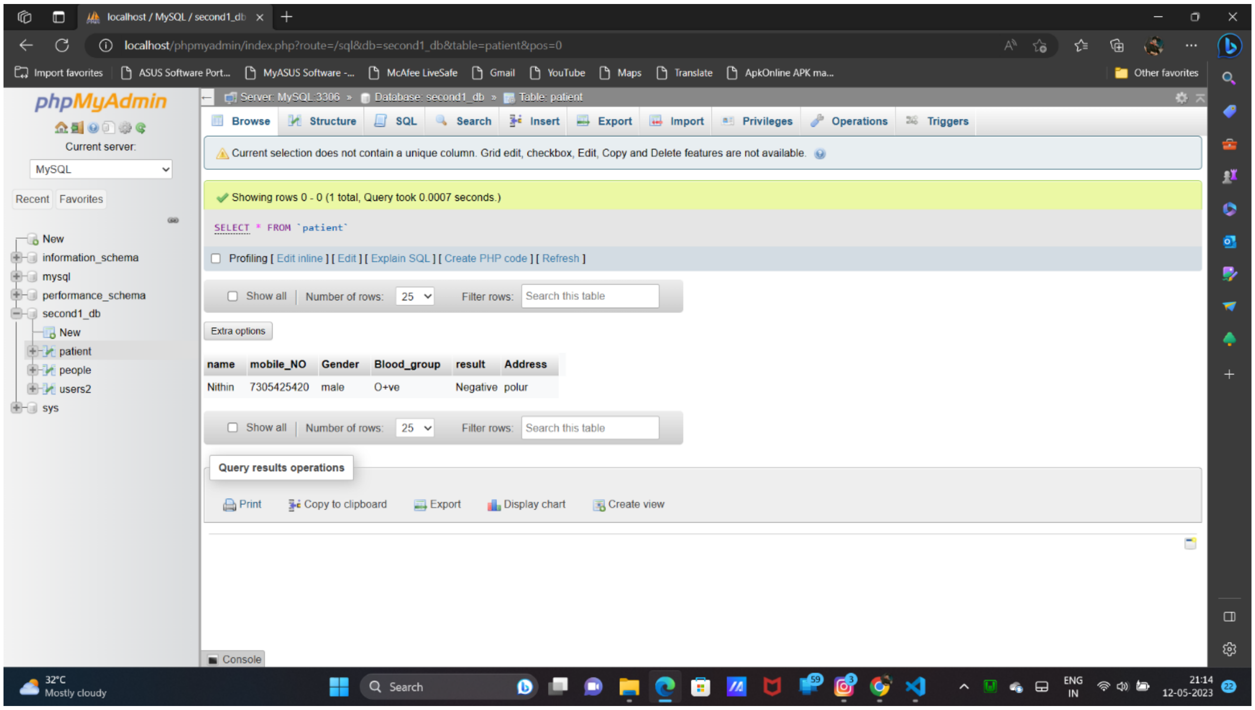
Second Screen:



Third Screen:



DATABASE:


localhost / MySQL / second1_db
localhost/phpmyadmin/index.php?route=/sql&db=second1_db&table=patient&pos=0
phpMyAdmin
Server: MySQL:3306 » Database: second1_db » Table: patient
Browse Structure SQL Search Insert Export Import Privileges Operations Triggers
Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.
Showing rows 0 - 0 (1 total, Query took 0.0007 seconds.)
SELECT * FROM `patient`
name mobile_NO Gender Blood_group result Address
Nithin 7305425420 male O+ve Negative polur
Query results operations
Print Copy to clipboard Export Display chart Create view

CHAPTER 10

CONCLUSION

In conclusion, deep learning has shown promising results in the detection of COVID-19 from lung X-rays. Developing a system architecture for COVID-19 detection using deep learning requires careful consideration of data collection, preprocessing, model selection, hyperparameter tuning, testing, deployment, and integration.The availability of large and diverse datasets, as well as the development of advanced deep learning models, has significantly improved the accuracy and reliability of COVID-19 detection from lung X-rays. The use of deep learning in healthcare systems has the potential to enhance disease diagnosis and treatment, reduce medical errors, and improve patient outcomes. However, it is important to note that deep learning models are not infallible and require careful validation and evaluation to ensure their accuracy and safety. The integration of deep learning models with other healthcare systems also raises important ethical, legal, and social issues that need to be addressed. COVID-19 detection from lung X-rays using deep learning is a promising approach that requires continued research and development to improve its accuracy, reliability, and safety. The successful deployment of such systems can significantly enhance the effectiveness of disease diagnosis and treatment and improve patient outcomes.

CHAPTER 11

REFERENCES

1. Apostolopoulos ID, Mpesiana TA. COVID-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks. Phys Eng Sci Med. 2020;43(2):635-640. doi:10.1007/s13246-020-00865-4

2. Hemdan EE-D, Shouman MA, Karar ME. COVIDX-Net: A Framework of Deep Learning Classifiers to Diagnose COVID-19 in X-Ray Images. arXiv:2003.11055 [cs.CV]. Published online March 23, 2020. https://arxiv.org/abs/2003.11055

3. Narin A, Kaya C, Pamuk Z. Automatic Detection of Coronavirus Disease (COVID-19) Using X-ray Images and Deep Convolutional Neural Networks. arXiv:2003.10849 [cs.CV]. Published online March 23, 2020. https://arxiv.org/abs/2003.10849

4. Wang L, Lin ZQ, Wong A. COVID-Net: A Tailored Deep Convolutional Neural Network Design for Detection of COVID-19 Cases from Chest X-Ray Images. arXiv:2003.09871 [eess.IV]. Published online March 20, 2020. https://arxiv.org/abs/2003.09871

5. Ozturk S, Ozkaya U, Barstugan M. Coronavirus (COVID-19) Classification Using CT Images by Machine Learning Methods. arXiv:2003.09424 [eess.IV]. Published online March 20, 2020. https://arxiv.org/abs/2003.09424