WEB APPLICATION FOR PREDICTION AND CLASSIFICATION OF MULTIPLE DISEASES USING MACHINE LEARNING AND DEEP LEARNING TECHNIQUES

A PROJECT REPORT

Submitted by

ABITHA R [19CS004]

AGALYA M [19CS005]

GUGAPRIYA M [19CS040]

HARSHITHAA A S [19CS051]

in partial fulfillment for the award of the degree

of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

KPR INSTITUTE OF ENGINEERING AND TECHNOLOGY
ARASUR, COIMBATORE – 641407

ANNA UNIVERSITY : CHENNAI 600 025
April – 2023

ANNA UNIVERSITY : CHENNAI 600 025

BONAFIDE CERTIFICATE

Certified that this project report " **WEB APPLICATION FOR PREDICTION AND CLASSIFICATION OF MULTIPLE DISEASES USING MACHINE LEARNING AND DEEP LEARNING TECHNIQUES** " is the bonafide work of **" ABITHA R (19CS004), AGALYA M (19CS005), GUGAPRIYA M (19CS040), HARSHITHAA A S (19CS051)"** who carried out the project work under my supervision.

Dr. YUVARAJ N

HEAD OF THE DEPARTMENT,
Professor
Computer Science and Engineering
KPR Institute of Engineering and Technology
Arasur, Coimbatore – 641407

Ms. VISHNUPRIYA B

SUPERVISOR,
Assistant professor
Computer Science and Engineering
KPR Institute of Engineering and Technology
Arasur, Coimbatore – 641407

Submitted for the university project Viva – Voce held on ___________

INTERNAL EXAMINER

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

The success shines when there is a team work and cooperation among the students and the faculties. We take this opportunity to express our gratitude and thanks to everyone who helped us during the project through every phase.

We would like to thank our beloved Chairman **Dr. K. P. Ramasamy** for his academic interest shown towards the students, by providing us excellent facilities and the constant support.

We would like to express our deep gratitude and sincere regards to our beloved and honorable Principal **Dr. M. Akila,** for providing us excellent facilities and encouragement during the course of study and project.

We owe a genuine gratitude to **Dr. Yuvaraj N,** Professor and Head, Department of Computer Science and Engineering for his encouragement and inspiration through every course.

We owe a sincere thanks to **Ms. Vishnupriya B,** Assistant Professor, Department of Computer Science and Engineering for her support and guidance throughout the project as an Internal Guide.

We owe a sincere thanks to **Mr. Nitin B Raut**, Assistant Professor, Department of Computer Science and Engineering for his support as the overall Project coordinator.

Lastly, we thank our parents for their constant encouragement without which this assignment would not have been possible.

ABSTRACT

The project presents a web application for the prediction and classification of multiple diseases using machine learning and deep learning techniques. The application allows medical professionals to input patient data and receive predictions for various diseases such as diabetes, heart disease, and cancer. The system is designed to help doctors make faster and more accurate diagnoses, improving patient outcomes and reducing costs.The web application consists of 10 pages, including a home page, disease pages, prediction pages for each disease, and a report page that displays the patient's history and offers an option to send a report via email. The system is built using ReactJS for the front-end, Firebase for data storage, and EmailJS for email communication. The machine learning models used for disease prediction were developed using the Random Forest algorithm, achieving high accuracy rates for malaria, pneumonia, breast cancer, kidney disease, diabetes, and heart disease.The project involved a thorough literature review of previous work in the field of disease prediction using machine learning techniques. The proposed system was developed based on the limitations of existing systems and their potential for improvement. The system's architecture, data flow, and the machine learning models used were described in detail.The results of testing the system showed that it could accurately predict multiple diseases with high accuracy rates. The user interface was user-friendly, making it easy for medical professionals to input data and obtain predictions. The report page offered a convenient way to access patient information and send reports to patients. Overall, this web application presents a promising solution for improving disease diagnosis and patient outcomes. Its user-friendly interface and high accuracy rates make it a valuable tool for medical professionals. The system's modular design allows for easy scalability and expansion in the future.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

CNN	CONVOLUTIONAL NEURAL NETWORK
SVM	SUPPORT VECTOR MACHINE
KNN	K - NEAREST NEIGHBOURS ALGORITHM
MRI	MAGNETIC RESONANCE IMAGING
BMI	BODY MASS INDEX
CT SCAN	COMPUTED TOMOGRAPHY SCAN
IBM	INTERNATIONAL BUSINESS MACHINE CORPORATION
ML	MACHINE LEARNING

CHAPTER 1

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

The development of Web Application for Disease Prediction and Classification using Machine Learning and Deep Learning Techniques is of great importance in the field of medical science. It provides a fast and accurate method for diagnosing diseases.

The development of a web application for predicting and classifying multiple diseases using machine learning and deep learning techniques is a challenging and exciting project. The objective of this project is to create an intuitive and efficient tool that healthcare professionals can use to diagnose and treat patients with a wide range of diseases. To accomplish this, the web application would utilize several machine learning and deep learning models, as well as modern web technologies and frameworks.

The web application would leverage machine learning algorithms such as SVM, KNN, Naive Bayes, Logistic Regression, and Random Forest to classify and predict diseases based on patient data. These algorithms would be trained on large datasets of patient medical histories, lab results, and other relevant data to provide accurate and reliable predictions and classifications. The use of multiple algorithms would help ensure the accuracy of the predictions and classifications, and the results would be presented in an easy-to-understand format for healthcare professionals. The use of deep learning techniques is another critical aspect of this project. Convolutional neural networks (CNNs), in particular, would be employed for the prediction of Malaria and Pneumonia. CNNs are well-suited for medical image classification tasks, as they can detect patterns and features in medical images that may not be apparent to the human eye. The CNNs would be trained on a large dataset of medical images to provide accurate and reliable predictions of these diseases. The user interface of the web application would be built using the React JS library, which is a popular and powerful JavaScript library that is widely used for developing user interfaces for web applications. React JS would provide the necessary flexibility and efficiency to create a dynamic and responsive user interface that is easy to navigate and use. The interface would be designed to be user-friendly, making it easy for healthcare professionals to input patient data and get predictions and classifications of diseases. Another important feature of the web application would be the ability to send reports and notifications to patients using Email JS. This feature would allow healthcare professionals to quickly and easily send reports to patients, providing them with important information about their health and the status of their treatment. The reports would be generated automatically by the machine learning and deep learning models and would be sent to patients via email in a secure and private manner.

In conclusion, the development of a web application for predicting and classifying multiple diseases using machine learning and deep learning techniques is a complex and challenging project. The application would utilize machine learning algorithms such as SVM, KNN, Naive Bayes, Logistic Regression, and Random Forest, as well as deep learning techniques such as CNNs for the prediction of Malaria and Pneumonia. The application would be built using the React JS library for the user interface, Email JS for sending reports and notifications to patients, and CSS for styling and layout. The final result would be a powerful and user-friendly tool for healthcare professionals to diagnose and treat patients, while also providing patients with valuable information about their health and treatment status.

1.2 ML IN DISEASE PREDICTION

Machine learning has shown promising results in disease prediction by analyzing large amounts of patient data and identifying patterns that can indicate the presence or likelihood of a disease. There are several ways in which machine learning is being used for disease prediction:

Predicting disease risk: Machine learning algorithms can be trained on large datasets to identify patterns and risk factors that are associated with the development of a particular disease. These algorithms can then be used to predict an individual's risk of developing the disease based on their personal health data and other factors.

Early disease detection: Machine learning can be used to analyze medical images, such as X-rays or MRIs, to identify early signs of disease that may be missed by human doctors. For example, machine learning algorithms can be trained to identify subtle changes in brain images that may indicate the early stages of Alzheimer's disease.

Diagnosis support: Machine learning can be used to analyze patient symptoms and medical histories to help doctors make more accurate diagnoses. For example, machine learning algorithms can be used to analyze patient data and suggest potential diagnoses based on patterns in the data.

Treatment recommendation: Machine learning algorithms can be trained to analyze patient data and recommend personalized treatment plans based on the patient's specific health conditions, medical history, and other factors.

Overall, machine learning has the potential to greatly improve disease prediction and diagnosis, leading to earlier interventions and better patient outcomes. However, it's important to note that machine learning algorithms should always be used in conjunction with human clinical expertise, and their predictions and recommendations should be validated through rigorous testing and evaluation.

1.3 WHY MACHINE LEARNING?

Machine learning has become an increasingly popular approach in healthcare because it can help healthcare professionals make better and more accurate decisions by analyzing large datasets of patient data. Machine learning algorithms can identify patterns and relationships in the data that might not be visible to the human eye, and use this information to make predictions and classifications. This can be particularly useful in diagnosing and treating diseases, as it can help identify patients who are at high risk of developing a particular disease, or help healthcare professionals make more accurate diagnoses.

Machine learning can also help reduce errors and improve efficiency in healthcare. For example, machine learning algorithms can analyze medical images to help healthcare professionals detect abnormalities and make more accurate diagnoses. This can help reduce the need for invasive procedures and improve patient outcomes. Additionally, machine learning can be used to automate routine tasks such as data entry, allowing healthcare professionals to focus on more complex tasks that require their expertise.

Machine learning can be a valuable tool for disease prediction, as it can analyze large datasets to identify patterns and relationships that might be missed by human analysts. Disease prediction is important for a number of reasons. Early detection of diseases can lead to more effective treatment, and can sometimes even prevent the disease from developing in the first place.

One example of a disease that can be predicted using machine learning is diabetes. Diabetes is a chronic disease that affects millions of people worldwide, and early detection is important for preventing complications such as kidney failure and blindness.

Machine learning algorithms can be trained on large datasets of patient data, such as medical history, lab results, and lifestyle factors, to predict the likelihood of a patient developing diabetes. For example, a machine learning algorithm might analyze data from thousands of patients and identify factors such as age, body mass index (BMI), and family history that are associated with a higher risk of diabetes.

Once the machine learning algorithm has been trained, it can be used to predict the likelihood of a patient developing diabetes based on their individual risk factors. This can help healthcare professionals to identify patients who are at high risk of developing diabetes, allowing them to take preventative measures such as lifestyle changes or medication to reduce the risk.

Machine learning algorithms can be trained on large datasets of patient data, such as medical history, cognitive test results, and brain scans, to predict the likelihood of a patient developing Alzheimer's disease. For example, a machine learning algorithm might analyze data from thousands of patients and identify factors such as age, cognitive test scores, and brain volume that are associated with a higher risk of Alzheimer's.

Once the machine learning algorithm has been trained, it can be used to predict the likelihood of a patient developing Alzheimer's based on their individual risk factors. This can help healthcare professionals to identify patients who are at high risk of developing Alzheimer's, allowing them to take preventative measures such as lifestyle changes or medication to reduce the risk.

Overall, machine learning can be a powerful tool for disease prediction, allowing healthcare professionals to identify patients who are at high risk of developing a disease and take preventative measures to improve treatment outcomes and quality of life.

1.4 MACHINE LEARNING

Machine learning (ML) is a subfield of artificial intelligence (AI) that focuses on the development of algorithms and models that can learn from data and make predictions or decisions without being explicitly programmed.

The main goal of ML is to develop systems that can automatically improve their performance over time as they are exposed to more data. This is achieved through the use of statistical models and algorithms that can identify patterns and relationships within large datasets.

ML algorithms can be broadly classified into three categories: supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning involves training an algorithm on labeled data, where the correct output is known for each input. The algorithm learns to predict the correct output for new inputs based on patterns it identifies in the training data.

Unsupervised learning involves training an algorithm on unlabeled data, where the correct output is unknown. The algorithm learns to identify patterns and relationships in the data without any guidance.

Reinforcement learning involves training an algorithm through trial and error by rewarding it for good decisions and punishing it for bad ones. The algorithm learns to maximize its rewards over time by making better decisions.

ML has numerous applications in various fields such as healthcare, finance, transportation, and agriculture. In recent years, it has gained significant attention in the development of intelligent systems for tasks such as image recognition, speech recognition, natural language processing, and decision-making.

Regression and classification are two of the most fundamental tasks in supervised machine learning.

Regression:

Regression is a type of supervised learning problem that involves predicting a continuous output variable based on one or more input variables. The goal of regression is to build a model that can accurately predict the output variable for new input data. In regression, the output variable is a continuous value, such as the price of a house or the temperature of a city.

There are several algorithms that can be used for regression problems, such as linear regression, polynomial regression, and decision tree regression. Linear regression is a simple and widely used algorithm that involves fitting a straight line to the data. Polynomial regression is an extension of linear regression that allows for fitting a more complex curve to the data. Decision tree regression involves building a tree-like model that can make predictions based on the values of input variables.

Classification:

Classification is another type of supervised learning problem that involves predicting a categorical output variable based on one or more input variables. The goal of classification is to build a model that can accurately predict the class label for new input data. In classification, the output variable is a categorical value, such as whether an email is spam or not, or whether a patient has a disease or not.

There are several algorithms that can be used for classification problems, such as logistic regression, decision trees, random forests, and support vector machines (SVM). Logistic regression is a simple and widely used algorithm that involves fitting a logistic function to the data. Decision trees involve building a tree-like model that can make predictions based on the values of input variables. Random forests are an extension of decision trees that involve building multiple trees and combining their predictions. SVM is a more complex algorithm that involves finding a hyperplane that separates the data into different classes.

In summary, regression and classification are two of the most important and widely used tasks in supervised machine learning. Regression involves predicting a continuous output variable based on input variables, while classification involves predicting a categorical output variable based on input variables. There are several algorithms that can be used for each of the tasks, depending on the complexity of the problem and the type of data being analyzed.

1.4.1 ADVANTAGES OF MACHINE LEARNING

Machine learning has shown great promise in the field of disease prediction. Here are some of the advantages:

Early Detection: Machine learning algorithms can identify patterns and risk factors that may not be immediately obvious to healthcare professionals. This can lead to earlier detection of diseases, which can improve treatment outcomes and save lives.

Accurate Diagnosis: Machine learning algorithms can analyze vast amounts of patient data to accurately diagnose diseases. This can lead to more effective treatment plans and better patient outcomes.

Personalized Medicine: Machine learning can analyze patient data to develop personalized treatment plans based on individual factors such as age, gender, genetics, and lifestyle. This can lead to more effective and efficient treatment plans.

Cost-effective: Machine learning algorithms can analyze data quickly and accurately, which can reduce the need for expensive diagnostic tests and procedures. This can lead to cost savings for both patients and healthcare providers.

Real-time monitoring: Machine learning algorithms can analyze patient data in real-time, allowing for early detection of disease progression and adjustment of treatment plans as needed. This can improve patient outcomes and reduce hospitalization rates.

Overall, machine learning has the potential to revolutionize the field of disease prediction and improve patient outcomes.

1.5 DEEP LEARNING IN DISEASE PREDICTION

Deep learning is a subset of machine learning that uses artificial neural networks to analyze complex data. Deep learning algorithms can be particularly useful for disease prediction, as they are able to identify subtle patterns and relationships in large, complex datasets.

One of the main advantages of deep learning for disease prediction is that it can work with unstructured data, such as medical images or free-form text. For example, a deep learning algorithm can be trained on large datasets of medical images to identify patterns associated with certain diseases. This can be particularly useful for diseases that manifest in physical symptoms, such as skin diseases or certain types of cancer.

Deep learning can also be used to analyze structured data, such as medical records or genomic data. For example, a deep learning algorithm can be trained on large datasets of patient records to identify risk factors associated with certain diseases. This can help healthcare professionals to identify patients who are at high risk of developing a disease and take preventative measures. One example of a disease that can be predicted using deep learning is diabetic retinopathy, a complication of diabetes that affects the eyes. Deep learning algorithms can be trained on large datasets of retinal images to identify patterns associated with diabetic retinopathy, allowing healthcare professionals to detect the condition early and take preventative measures to prevent vision loss.

Overall, deep learning can be a powerful tool for disease prediction, allowing healthcare professionals to identify patterns and risk factors that might not be visible to the human eye. This can help to improve early detection and prevention of diseases, leading to better outcomes for patients.

1.6 DEEP LEARNING

Deep learning with Convolutional Neural Networks (CNNs) is a type of deep learning that is particularly useful for image classification and object recognition. CNNs are a type of artificial neural network that is inspired by the way the visual cortex in the brain processes information.

CNNs use a series of convolutional layers to analyze an image and identify features such as edges, shapes, and textures. These features are then used to classify the image into one or more categories.

One area where deep learning with CNNs has been particularly successful is in medical image analysis. For example, CNNs can be trained on large datasets of medical images to identify patterns associated with certain diseases, allowing healthcare professionals to detect the disease earlier and take preventative measures.

One example of a disease that can be predicted using deep learning with CNNs is breast cancer. CNNs can be trained on large datasets of mammogram images to identify patterns associated with cancerous lesions. This can help healthcare professionals to identify suspicious areas and take action before the cancer becomes more advanced.

Another example of a disease that can be predicted using deep learning with CNNs is pneumonia. CNNs can be trained on large datasets of chest x-ray images to identify patterns associated with pneumonia. This can help healthcare professionals to diagnose the disease earlier and start treatment sooner, leading to better outcomes for patients.

In addition to disease prediction, deep learning with CNNs can also be used for medical image segmentation, which involves identifying and outlining specific structures or regions within an image. This can be useful for a variety of applications, such as identifying tumors or tracking disease progression over time.

Overall, deep learning with CNNs is a powerful tool for disease prediction and medical image analysis. By identifying subtle patterns and relationships in large datasets of medical images, CNNs can help healthcare professionals to improve early detection and prevention of diseases, leading to better outcomes for patients.

1.6.1 ADVANTAGE OF DEEP LEARNING

Deep learning has several advantages over traditional machine learning techniques. One of the key advantages is its ability to achieve higher accuracy levels than traditional machine learning algorithms. This is because deep learning algorithms are able to identify subtle patterns and relationships in data that may be difficult or impossible for traditional machine learning algorithms to detect. Additionally, deep learning algorithms can work with unstructured data such as images, audio, and text, making them particularly useful for tasks such as image classification and natural language processing.

Another advantage of deep learning is that it is able to automatically extract relevant features from data, eliminating the need for manual feature engineering. This can save time and resources and also allows for more complex features to be identified that may not be apparent to human experts. Additionally, deep learning algorithms can be easily scaled to handle large datasets, making them ideal for use in big data applications. Furthermore, deep learning algorithms can adapt to new data and improve their performance over time, making them ideal for use in applications where the data is constantly changing or evolving. Overall, deep learning is a powerful tool for analyzing and making predictions based on complex data, and it is likely to play an increasingly important role in a wide range of applications.

1.7 ORGANIZATION OF THE REPORT

The report is organized into six chapters, each of which covers an important aspect of the Web Application for Prediction and Classification of Multiple Diseases. These chapters include a literature review, existing system analysis, proposed system and working principle, results and discussion, conclusion and future work.

Chapter 2: Literature Review

In this chapter, the report provides a comprehensive review of the existing literature related to disease prediction and classification using machine learning and deep learning techniques. The review covers various studies and research papers that have been conducted in this field, including their methodologies, results, and limitations. The chapter concludes by identifying gaps in the existing literature and how the proposed project aims to address these gaps.

Chapter 3: Existing System

The third chapter of the report provides an overview of the existing disease prediction and classification systems that are currently available in the market. The chapter discusses the strengths and weaknesses of these systems, as well as their limitations. The report also highlights the need for a more accurate and reliable disease prediction and classification system, which the proposed project aims to provide.

Chapter 4: Proposed System and Working Principle

In this chapter, the report discusses the proposed system and its working principle. The chapter provides a detailed explanation of the various machine learning and deep learning techniques that have been used in the project, including their advantages and limitations. The chapter also provides an overview of the web application architecture and the data flow between the different components of the system. Additionally, the report discusses the implementation details of the system, including the programming languages and tools that were used.

Chapter 5: Results and Discussion

The fifth chapter of the report presents the results of the experiments that were conducted to evaluate the performance of the proposed system. The chapter provides a detailed analysis of the accuracy and reliability of the system in predicting and classifying various diseases. The results are further compared with the existing systems to highlight the superiority of the proposed system.

Chapter 6: Conclusion and Future Work

The final chapter of the report presents the conclusions drawn from the project and highlights its contributions to the field of disease prediction and classification using machine learning and deep learning techniques. The chapter also discusses the future work that can be done to improve the system further, including the incorporation of more advanced machine learning and deep learning algorithms, expanding the number of diseases predicted, and enhancing the user interface of the web application.

CHAPTER 2

CHAPTER 2

LITERATURE SURVEY

2.1 RELATED BACKGROUNDS

The following paper highlight the potential of machine learning and deep learning techniques for disease prediction and classification in healthcare. However, they also emphasize the need for more research and evaluation to fully assess the clinical utility of these techniques, as well as the importance of addressing challenges such as dataset quality, model interpretability, and clinical validation.

Paper 1 :"Application of machine learning and artificial intelligence in drug design and discovery" by Aslam et al. (2020) :

This review article discusses the use of machine learning and artificial intelligence in drug design and discovery, with a focus on applications in cancer treatment. It highlights the potential of machine learning techniques such as support vector machines, random forests, and neural networks for predicting drug-target interactions, toxicity, and efficacy. The article also discusses the challenges and limitations of using these techniques in drug discovery, such as the need for large datasets and the difficulty of interpreting results.

Paper 2 :"A systematic review of machine learning techniques for disease prediction" by Kavakiotis et al. (2017)

This review article evaluates the use of machine learning techniques for disease prediction in various medical fields, such as cardiology, oncology, and neurology.

It compares the performance of different machine learning algorithms, such as support vector machines, decision trees, and artificial neural networks, in predicting diseases based on patient data such as demographics, clinical history, and biomarkers. The article concludes that machine learning techniques have the potential to improve disease prediction accuracy and efficiency, but that more research is needed to fully evaluate their clinical utility.

Paper 3 : "Deep learning for healthcare: review, opportunities and challenges" by Shi et al. (2020)

This review article provides an overview of the use of deep learning techniques, particularly convolutional neural networks, for healthcare applications such as disease diagnosis, prediction, and prognosis. It discusses the advantages of deep learning over traditional machine learning techniques, such as the ability to learn complex features and patterns in large datasets, and the potential for personalized medicine. The article also highlights some of the challenges and limitations of using deep learning in healthcare, such as the need for large, high-quality datasets, and the difficulty of interpreting model predictions

Paper 4 : "Deep Learning for Healthcare: Review, Opportunities and Challenges" by Ahmed Elnahrawy, Ahmed M. Dawoud, and Mohamed A. Aboelhassan (2021)

In this comprehensive review, the authors provide an overview of deep learning techniques used in healthcare, including disease prediction, medical image analysis, and electronic health records analysis. They discuss the opportunities and challenges of using deep learning in healthcare, such as data privacy concerns and the need for explainable AI. The authors also highlight recent advances in deep learning for healthcare and provide recommendations for future research, including the need for more diverse and representative data sets and the importance of collaboration between healthcare professionals and data scientists.

Paper 5 :"A Survey of Deep Learning Techniques for Medical Image Analysis" by S. S. Khan et al.

This paper focuses specifically on the use of deep learning techniques for medical image analysis in disease prediction and diagnosis. The authors provide an overview of the different types of deep learning models and their applications in medical image analysis, including the detection of cancer, heart disease, and neurological disorders. They also discuss the challenges and limitations of using deep learning in medical image analysis and identify potential areas for future research.One notable application of deep learning discussed in the paper is the use of transfer learning, a technique in which a pre-trained deep learning model is fine-tuned for a specific task, in medical image analysis. The authors describe several studies that have demonstrated the effectiveness of transfer learning in detecting and classifying diseases from medical images. The paper also discusses the potential of deep learning for personalized medicine, in which medical treatments are tailored to the individual characteristics of each patient.

In conclusion, the literature survey highlights the importance of machine learning and deep learning techniques in the field of disease prediction and classification. Various studies have demonstrated the effectiveness of these techniques in accurately predicting and classifying different diseases, such as cancer, heart disease, and pneumonia, among others. The use of these techniques has also been found to improve the speed and efficiency of the diagnostic process, as well as reduce the need for invasive procedures and tests.

It is evident from the literature that deep learning techniques, particularly CNN, have shown promising results in improving the accuracy of disease prediction models. However, these techniques require a large amount of data and computing power, which may not be feasible in some settings.

CHAPTER 3

CHAPTER 3

EXISTING SYSTEM

3.1 INTRODUCTION

Title: MicroLab AI

MicroLab AI is an AI-powered web application that enables rapid and accurate diagnosis of diseases using machine learning and deep learning algorithms. The platform is designed to help medical professionals in low-resource settings, where access to advanced diagnostic equipment is limited.

3.2 BLOCK DIAGRAM OF EXISTING SYSTEM

The MicroLab AI platform consists of several components, including the data acquisition module, the preprocessing module, the feature extraction module, and the classification module. The block diagram for the MicroLab AI platform is shown in Figure 3.1.

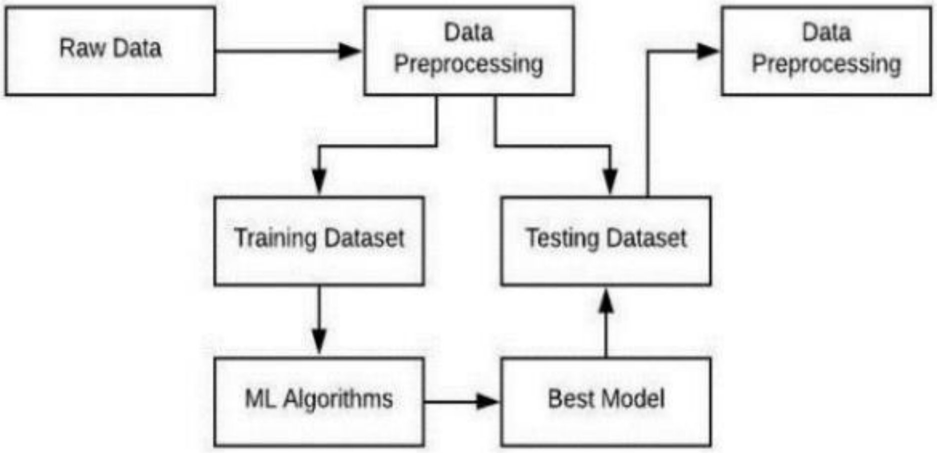


Figure 3.1. Block diagram of existing system

The data acquisition module is responsible for collecting patient data, including medical images and clinical information. The module can process data from a variety of sources, including X-ray machines, CT scans, and other diagnostic equipment. The preprocessing module then cleans and standardizes the data, removing noise and other artifacts that may affect the accuracy of the machine learning algorithms.

The feature extraction module is responsible for identifying key features in the patient data that can be used to predict disease outcomes. This module uses a combination of deep learning techniques, such as convolutional neural networks (CNNs), to extract relevant features from the patient data. The classification module then uses these features to classify the patient data into different disease categories, such as tuberculosis, pneumonia, and lung cancer.

3.3 HARDWARE DESCRIPTION

The MicroLab AI platform is designed to be used with low-cost diagnostic equipment, making it accessible to medical professionals in low-resource settings. The platform can be accessed through any internet-connected device, such as a smartphone or tablet, and is powered by a cloud-based server infrastructure. The system is designed to be scalable and can handle large amounts of patient data simultaneously.

3.4 DISADAVANTAGE OF EXISTING SYSTEM

Dependence on data quality: The accuracy of machine learning algorithms depends on the quality of the data used to train them. In low-resource settings, it may be difficult to obtain high-quality data, which could affect the accuracy of the MicroLab AI system.

Limited range of diseases: Like many machine learning-based diagnostic systems, MicroLab AI is currently limited to a specific range of diseases. As such, it may not be suitable for diagnosing more rare or complex conditions.

Dependence on internet connectivity: As a cloud-based platform, MicroLab AI requires a reliable internet connection in order to function properly. In areas with poor internet connectivity, the system may not be accessible.

Overall, the MicroLab AI system represents a promising application of machine learning and deep learning techniques in healthcare, particularly in low-resource settings. While there are some potential limitations to the system, its advantages make it a valuable tool for medical professionals working in these settings.

3.5 CONCLUSION

MicroLab AI is an example of a successful application of machine learning and deep learning techniques in healthcare. By using AI-powered algorithms to analyze patient data, the platform is able to provide rapid and accurate diagnosis of diseases, even in low-resource settings. The platform has the potential to revolutionize healthcare in developing countries, where access to advanced diagnostic equipment is limited.

Other Existing systems :

IBM Watson Health: This platform offers a range of healthcare solutions, including AI-powered disease prediction and diagnosis tools.

Google Cloud Healthcare: This platform includes a range of machine learning models for predicting and diagnosing diseases, as well as tools for managing and analyzing medical data.

Ada Health: This AI-powered platform allows users to input their symptoms and receive personalized health recommendations and diagnoses.

Babylon Health: This platform offers AI-powered medical chatbots that can assist with diagnosis and provide personalized health recommendations.

CHAPTER 4

CHAPTER 4

PROPOSED SYSTEM

4.1 INTRODUCTION

The use of machine learning and deep learning techniques has become increasingly popular in the healthcare industry. In particular, the ability to predict and classify diseases using these techniques has shown promising results in improving patient outcomes. One of the applications of these techniques is the development of web applications for disease prediction and classification.

A web application for disease prediction and classification using machine learning and deep learning techniques has the potential to revolutionize the healthcare industry. By leveraging the vast amounts of patient data available, these applications can provide accurate and timely predictions of diseases, allowing healthcare providers to make more informed decisions about patient care.

The development of these web applications involves the use of both machine learning and deep learning techniques. Machine learning algorithms are used to collect and preprocess datasets of different diseases, train and test the datasets, and create models using different algorithms such as KNN, SVM, Navie Bayes, Decision Tree, Logistic Regression, and Random Forest. On the other hand, deep learning techniques involve the use of image datasets, preprocessing of images, and the creation of models using convolutional neural networks.

The architecture of a web application for disease prediction and classification typically consists of two main components: the backend and the frontend. The backend is responsible for collecting and preprocessing datasets, training and testing the models, and deploying them using Flask in Python. The frontend provides the user interface, allowing users to input patient data and receive predictions about diseases.

Overall, the development of web applications for disease prediction and classification using machine learning and deep learning techniques has the potential to greatly improve the healthcare industry. These applications can provide accurate and timely predictions of diseases, allowing healthcare providers to make more informed decisions about patient care. As the field of machine learning and deep learning continues to evolve, we can expect to see further advancements in this area, leading to even more accurate and effective disease prediction and classification.

4.2 ARCHITECTURE

4.2.1 Web application Prediction And Classification Of Multiple Diseases Using Machine Learning And Deep Learning.

The architecture Figure 4.1. of the proposed web application for disease prediction and classification using machine learning and deep learning consists of a data acquisition module, data preprocessing module, feature selection module, model training module, model testing module, and a user interface module.

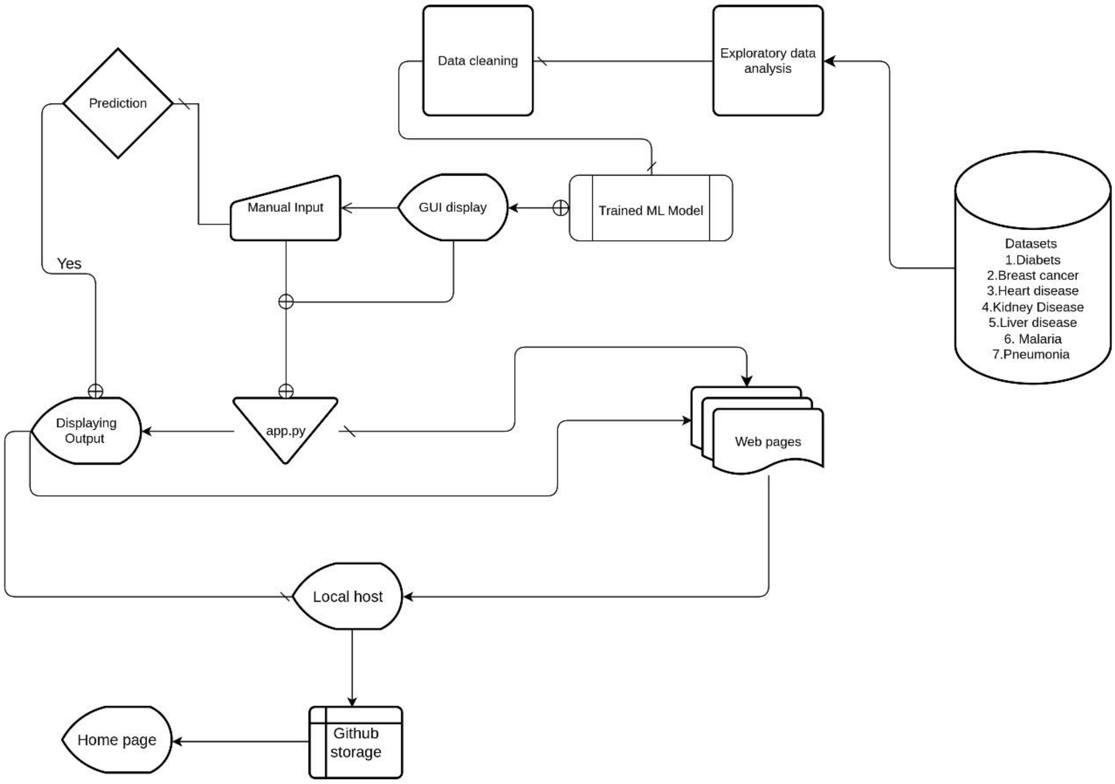


Figure 4.1. Architecture of web application Prediction And Classification Of Multiple Diseases Using Machine LearningAnd Deep Learning.

4.2.2 Web application architecture

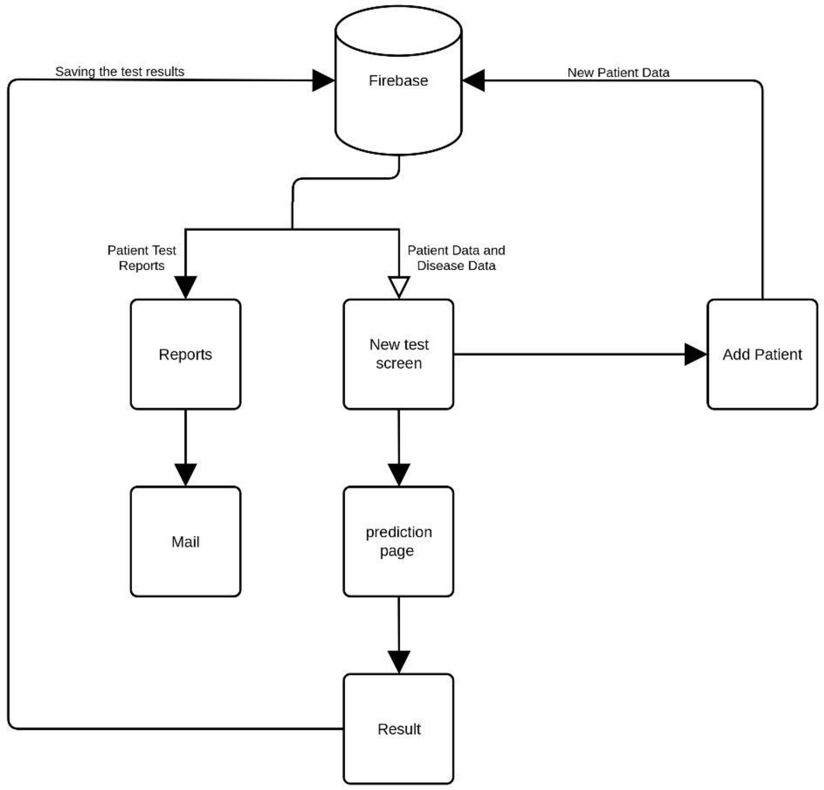


Figure 4.2. Architecture of web application Prediction Multiple Diseases .

The architecture Figure 4.2. of the proposed web application for disease prediction using machine learning and deep learning consists of a flow of pages for each prediction, a mailing system, and a database. The web application is designed to acquire user data, preprocess and select features, train the model, and test the model for disease prediction. The predicted results are then displayed to the user and sent via email. The database stores user information and predicted results for future reference.

4.3 MODULE DESCRIPTION

User-friendly Web application for prediction of disease using machine learning algorithm and deep learning .

4.3.1 Backend

a) Machine Learning

i. Collecting a datasets of different disease

ii. Data preprocessing

iii. Training and Testing of datasets

iv. Creation of model for different algorithms of KNN, SVM, Navie bayes,Decision Tree , Logistic Regression, Random forest .

v. Analysis the high accuracy algorithm

b) Deep learning

i. Using a image datasets

ii. Image preprocessing

iii. Creating a model using Convolution Neural Network

c) Model creation

i. In machine learning , model is that has high accuracy is created as model for backend using pickle

ii. In deep learning , model is saved by ‘save_weights’ in keras library.

iii. Deploying the created model using flask library in python to connect to front end.

4.3.2 Frontend

a) Home page

i. Consists of navigation to new test page and also through navigation bar and reports page .

b) Test page for different disease

i. Select and get information about patients.

i. Consists of 7 pages of prediction such that Kidney, Liver, Heart , Breast cancer, Diabetes , Malaria and pneumonia .

ii. Getting the input from patient in form.

iii. After submitting the form , it will connect to the created model and based on input it predicts the disease.

c) Reports page

i) Consists of all users report history

ii) Report can be shared to patients mail.

4.4 WORKING PRINCIPLE

1. Development Framework:
    a. React.js will be used to develop the web application.
    b. React is a popular front-end development library that enables developers to create fast and dynamic user interfaces.
2. Email JS:
    a. The Email JS library can be used to send reports via email. This library allows developers to easily send emails using JavaScript and HTML templates.
    b. It provides an easy-to-use API that can be integrated with the web application to send reports to users.
3. Disease Prediction:
    a. The web application will use machine learning algorithms to predict and classify multiple diseases.
    b. For heart, liver, breast cancer, kidney, and diabetes diseases, the machine learning algorithm Random Forest can be used to train and classify the datasets.

c. Random Forest is a powerful classification algorithm that can handle complex datasets with high accuracy.

d. For malaria and pneumonia diseases, the web application can use deep learning algorithms to train and classify the image dataset. Convolutional Neural Networks (CNNs) are commonly used for image classification tasks, and they can be used in this project to classify the images of malaria and pneumonia.

4.Datasets:

For heart, liver, breast cancer, kidney, and diabetes diseases, publicly available datasets such as the UCI Machine Learning Repository can be used.

a. These datasets contain a large number of samples and features that can be used to train the machine learning algorithm.

b. For malaria and pneumonia diseases, image datasets such as the Malaria Cell Images Dataset and Chest X-ray Images Dataset can be used.

c. These datasets contain a large number of images that can be used to train the deep learning algorithm.

5.Database:

a. Firebase can be used as a NoSQL database to store patient reports. Firebase provides a cloud-based database that can be easily integrated with React.js.

b. This will enable the web application to store and retrieve patient reports.

6. Reporting:

a. The web application can generate a report for each patient based on their medical history and the disease prediction. The report can be sent to the patient via email using the Email JS library.

b. The report can also be stored in the Firebase database for future reference.

7. User Interface:

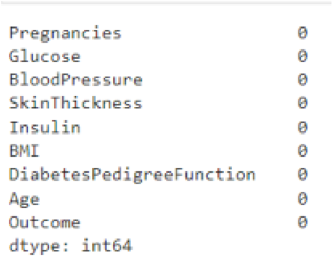
   a. React.js can be used to create a user-friendly interface for the web application.
   b. interface can allow patients to input their medical history and symptoms, view their reports, and receive email notifications when their reports are ready.

4.4.1 Data Collection

Data collection is the most efficient method of collecting the data through various sources. In this project, both crop recommendation dataset and disease detection dataset is collected from Kaggle. Datasets are collected from Kaggle, and datasets are analyzed and pre-processed for prediction.

i. Diabetes datasets

   a. Name of Dataset : Pima Indians Diabetes Database.
   b. Predict the onset of diabetes based on diagnostic measures Figure 4.3.
   c. Data - 768 Features- 9

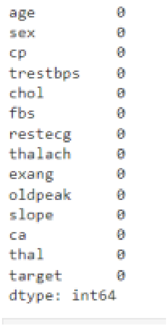


```
Pregnancies                 0
Glucose                     0
BloodPressure               0
SkinThickness               0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

Figure 4.3. Features for diabetes prediction

ii. Heart Datasets Figure 4.4.

   a. Data – 304 Features – 14

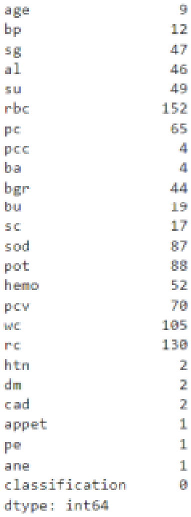


```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

Figure 4.4. Features for heart disease prediction

iii. Kidney Datasets Figure 4.5.

a. Data – 400 Features – 26

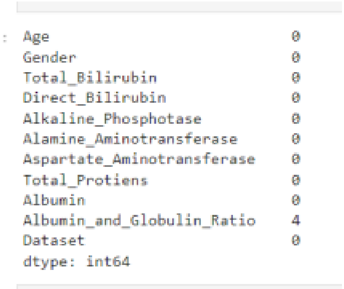


```
age                 9
bp                 12
sg                 47
al                 46
su                 49
rbc               152
pc                 65
pcc                 4
ba                  4
bgr                44
bu                 19
sc                 17
sod                87
pot                88
hemo               52
pcv                70
wc                105
rc                130
htn                 2
dm                  2
cad                 2
appet               1
pe                  1
ane                 1
classification      0
dtype: int64
```

Figure 4.5. Features for Kidney disease prediction

iv. Liver Datasets Figure 4.6.

a. Data – 583 Feature – 11

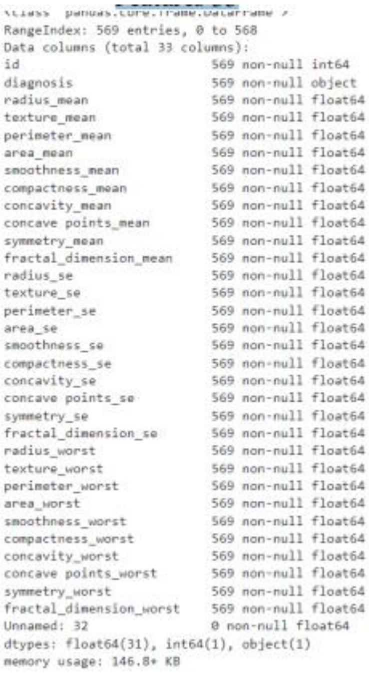


```
: Age                           0
  Gender                        0
  Total_Bilirubin               0
  Direct_Bilirubin              0
  Alkaline_Phosphotase          0
  Alamine_Aminotransferase      0
  Aspartate_Aminotransferase    0
  Total_Protiens                0
  Albumin                       0
  Albumin_and_Globulin_Ratio    4
  Dataset                       0
  dtype: int64
```

Figure 4.6. Features for Liver disease prediction

iv. Breast cancer Datasets Figure 4.7.

a. Data – 569 Features 33



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 33 columns):
id                         569 non-null int64
diagnosis                  569 non-null object
radius_mean                569 non-null float64
texture_mean               569 non-null float64
perimeter_mean             569 non-null float64
area_mean                  569 non-null float64
smoothness_mean            569 non-null float64
compactness_mean           569 non-null float64
concavity_mean             569 non-null float64
concave points_mean        569 non-null float64
symmetry_mean              569 non-null float64
fractal_dimension_mean     569 non-null float64
radius_se                  569 non-null float64
texture_se                 569 non-null float64
perimeter_se               569 non-null float64
area_se                    569 non-null float64
smoothness_se              569 non-null float64
compactness_se             569 non-null float64
concavity_se               569 non-null float64
concave points_se          569 non-null float64
symmetry_se                569 non-null float64
fractal_dimension_se       569 non-null float64
radius_worst               569 non-null float64
texture_worst              569 non-null float64
perimeter_worst            569 non-null float64
area_worst                 569 non-null float64
smoothness_worst           569 non-null float64
compactness_worst          569 non-null float64
concavity_worst            569 non-null float64
concave points_worst       569 non-null float64
symmetry_worst             569 non-null float64
fractal_dimension_worst    569 non-null float64
Unnamed: 32                0 non-null float64
dtypes: float64(31), int64(1), object(1)
memory usage: 146.8+ KB
```

Figure 4.7. Features for Breast Cancer prediction

Image dataset

v . Malaria

The dataset contains 2 folders

- Infected Figure 4.8.
- Uninfected Figure 4.9.
- And a total of 27,558 images.

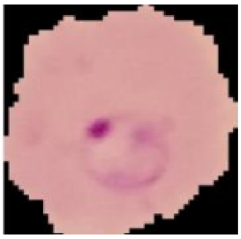


Figure 4.8. Infected Cell

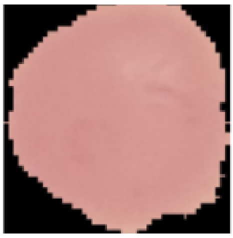


Figure 4.9. Uninfected

vi. Pneumonia

The dataset is divided into three folders: train, test, and val, with subfolders for each picture category (Pneumonia/Normal). There are 5,863 JPG X-Ray pictures and two categories (Pneumonia/Normal) Figure 4.10. and Figure 4.11 . The typical chest X-ray (left panel) shows clean lungs with no spots of aberrant opacification. Bacterial pneumonia (centre) is characterised by a focused lobar consolidation, in this example in the right upper lobe (white arrows), whereas viral pneumonia (right) is characterised by a more widespread "interstitial" pattern in both lungs.

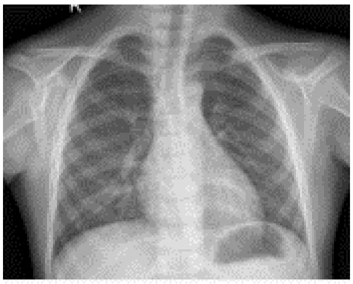


Figure 4.10. Normal Lung

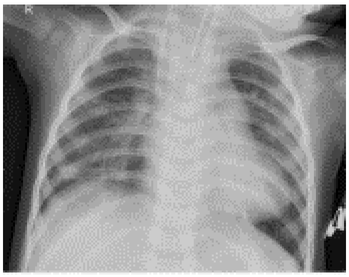


Figure 4.11. Infected Lung

4.4.2 DATA PREPROCESSING

Data preprocessing is an essential step in machine learning (ML) that involves transforming raw data into a format that is suitable for analysis and modeling. It is the process of cleaning, transforming, and organizing data to make it easier to work with, and to produce accurate and reliable results. Data preprocessing is particularly important because it can directly impact the accuracy of the model's output.

The first step in data preprocessing is data cleaning, which involves identifying and handling missing data, outliers, and errors. This is done to ensure that the dataset is complete and accurate. The next step is data transformation, which involves converting the data into a suitable format for analysis. This includes scaling and normalizing the data, feature selection, and feature engineering.

Scaling and normalization involve rescaling the features so that they have a similar scale and distribution. This is important because some machine learning algorithms are sensitive to the scale and distribution of the features. Feature selection involves selecting the most important features that have the greatest impact on the output variable. This is done to reduce the dimensionality of the dataset and to improve the accuracy of the model. Feature engineering involves creating new features that are derived from the existing features, which can improve the performance of the model.

After data preprocessing, the dataset is split into training and testing datasets. The training dataset is used to train the model, while the testing dataset is used to evaluate the performance of the model. This is done to ensure that the model can generalize well to new, unseen data.

In summary, data preprocessing is an important step in machine learning that involves cleaning, transforming, and organizing data to make it suitable for analysis and modeling. It helps to ensure that the dataset is complete, accurate, and in a suitable format for machine learning algorithms. Proper data preprocessing can significantly improve the accuracy and reliability of machine learning models.

4.4.3 MACHINE LEARNING MODEL

4.4.3.1 Decision Tree

A supervised technique for solving classification-related issues is a decision tree [8]. There is a widely used pre-defined target variable included. The decision tree may be used to input and output variables that are continuous or specified. This method may be used to tackle classification and regression issues because Based on the splitter that matters the most for the input variable, it divides the population or sample into two or more homogeneous groups called subpopulations.

Starting at the root, a decision tree is built using a top-down method that includes data segmentation. Below are the formulas for calculating Figure 4.12. entropy and the Gini index Figure 4.13.

$$Entropy = -\sum_{j} p_j \log_2 p_j$$

Figure 4.12. Entropy formula

$$Gini = 1 - \sum_{j} p_j^2$$

Figure 4.13. Gini Formula

One of the advantages of decision trees is that they are easy to understand and interpret. They also work well with datasets that have a mix of categorical and numerical variables. Decision trees are often used for tasks such as credit scoring, fraud detection, and medical diagnosis.

4.4.3.2 Naive Bayes

The Naive Bayes [10] method uses the Bayes theorem to train a classifier. In other words, it is a probabilistic classifier that was taught using the Naive Bayes approach. It computes a probability distribution for a given observation over many classes. Figure 4.14 illustrates the essential relationship of the Bayes theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Figure 4.14. Bayes theorem

P(B) is constant across all classes, hence the only product that will maximize is P(B/A)*P(A).

One of the advantages of Naive Bayes is that it requires relatively little training data compared to other algorithms. It also works well with high-dimensional datasets, where the number of features is much larger than the number of training examples. Naive Bayes is also robust to irrelevant features and can handle missing data.

4.4.3.3 SVM

SVM [6], a supervised machine learning algorithm can help with classification and regression problems. In an N-dimensional space, the SVM technique finds a hyperplane that classifies the input points. The size of the hyperplane is determined by the number of features. The hyperplane is just a line if there are only two input features. The hyperplane collapses to a two-dimensional plane when there are three input characteristics. The following are some benefits of support vector machines: effective in circumstances of enormous scope. When there are more dimensions than samples, the method is still useful. One of the advantages of SVM is that it is effective for high-dimensional datasets and can work well even when the number of features is much larger than the number of training examples. SVM is also less susceptible to overfitting compared to other algorithms, such as decision trees.

4.4.3.4 KNN

KNN [7] is a basic supervised machine learning (ML) approach that may be used to solve missing value imputation, classification, or regression problems. It is based on the idea that the observations in a data collection that is closest to a certain data point are the observations that are the most "similar," and that may thus classify unexpected points based on the values of the closest existing points. By choosing K, the user may specify the number of nearby observations to utilize in the method.

One of the advantages of KNN is that it is simple and easy to implement, requiring only the distance metric and the value of k to be specified. It can also work well for datasets that have complex decision boundaries or where the class distribution is uneven.

4.4.3.5 Logistic Regression

The Logistic Regression model is a popular statistical model in statistics that is used mostly for categorization. That is, given a collection of data, the Logistic Regression method assists us in categorizing these observations into two or more distinct groups. As a result, the target variable is discrete .

One of the advantages of logistic regression is that it is relatively simple and interpretable, as it outputs a probability score that can be easily understood. It also works well with small and medium-sized datasets and can handle both categorical and continuous input features.

4.4.3.6 Random Forest

A popular classification and regression method that uses supervised learning is called Random Forest [11]. One of the most crucial algorithms in machine learning is this one. It is based on the idea of ensemble learning, which is the practise of integrating many classifiers to solve a challenging issue and enhance the model's performance. Random Forest [11] is a classifier that uses a number of decision trees on different subsets of a given dataset and averages them to enhance the predicted accuracy of that dataset.

One of the advantages of random forest is that it can handle missing values and noisy data well, making it a useful algorithm for real-world applications. It is also less prone to overfitting compared to individual decision trees, as the ensemble approach helps to reduce the variance of the model.

4.4.4 DEEP LEARNING MODEL

Deep learning[12] is a subset of machine learning that is mostly used to train image models and video models that rely on neural networks with three or more layers. These models are capable of learning from the neurons built specifically for images.

4.4.4.1 IMAGE PREPROCESSING

Image Pre-processing The term "image pre-processing" describes operations on images that are carried out at the most fundamental level of abstraction. These operations lessen rather than increase the information content of the pictures if entropy is a measure of information. Pre-processing aims to enhance image data by reducing undesirable distortions or strengthening specific visual characteristics necessary for later processing and analysis activities. Image pre-processing techniques :

Image filtering

Image segmentation.

4.4.4.2 FLOWCHART FOR DEEP LEARNING

The flow of the deep learning procedure for the prediction of malaria and pneumonia disease is shown in the flowchart below Figure 4.15.

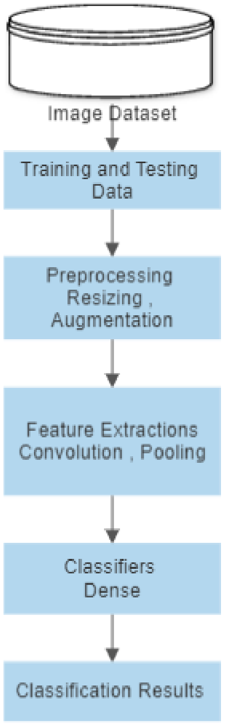


Image Dataset
Training and Testing Data
Preprocessing Resizing , Augmentation
Feature Extractions Convolution , Pooling
Classifiers Dense
Classification Results

Figure 4.15. Flowchart of Deep learning process.

4.4.4.3 Working of CNN

CNN Convolutional neural networks (CNN) [15] , are a type of deep learning technique that may apply weights and biases to different objects and characteristics in an input picture. Comparatively speaking, a CNN requires far less preprocessing than other classification techniques. CNN can learn these filters and attributes, as opposed to manually-engineered filters in basic systems. A CNN's construction Figure 4.16 is influenced by the visual cortex's organization, which is a mirror of the neural connections present in the human brain.

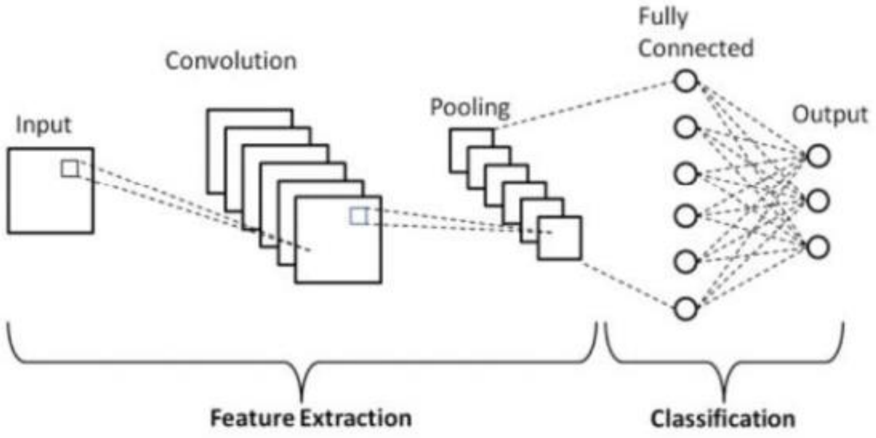


Figure 4.16 CNN Layers [16]

Individual neurons only respond to inputs in this tiny area of the visual field known as the Receptive Field. These overlapping patterns are repeated until the entire visual field is covered. Figure. 14 CNN Layers [16] A convolution neural network has multiple hidden layers Fig. 14 that help in extracting information from an image.

The four important layers in CNN are:

1. Convolution layer
2. ReLU layer
3. Pooling layer
4. Fully connected layer Convolution Layer

This is the first step in the process of extracting valuable features from an image. A convolution layer has several filters that perform the convolution operation. Every image is considered as a matrix of pixel values ReLU layer ReLU stands for the rectified linear unit. Once the feature maps are extracted, the next step is tomove them to a ReLU layer. ReLU performs an elementwise operation and sets all the negative pixels to 0. It introduces non-linearity to the network, and the generated output is a rectified feature map. Pooling Layer Pooling is a down-sampling operation that reduces the dimensionality of the feature map. The rectified feature map now goes through a pooling layer to generate a pooled feature map .

4.4.5 EVALUATION METRICS

4.4.5.1 Confusion matrix

The confusion matrix was used to analyze classifier performance using the metrics listed below.

True Positive (TP): The proportion of classes that, when projected favorably, are likewise positive.

The True Negative (TN): statistic tracks the percentage of negatively predicted classes that are truly negative.

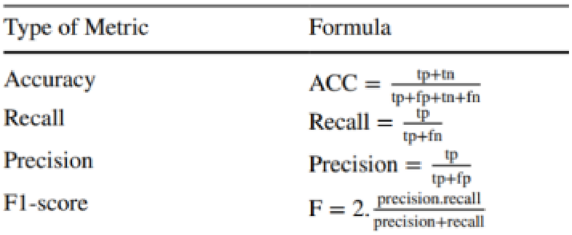
False Positive (FP): The number of classes that were incorrectly categorized while being favorably classified.

False Negative (FN): The amount of negatively classified courses that were mistakenly classified

4.4.5.2 Classification report

Equations 1-4 in Figure 4.17 state that the models are assessed based on four standard metrics: accuracy, precision, recall, F1- score, and specificity,

where TP denotes true positive, TN denotes true negative, FP denotes false positive, and FN denotes false negative.



Type of Metric	Formula
Accuracy	$ACC = \frac{tp+tn}{tp+fp+tn+fn}$
Recall	$Recall = \frac{tp}{tp+fn}$
Precision	$Precision = \frac{tp}{tp+fp}$
F1-score	$F = 2.\frac{precision.recall}{precision+recall}$

Figure 4.17. Evaluation Metrics

Figure 4.17. Evaluation Metrics Precision Precision is a classifier's ability to avoid labeling a negative occurrence as positive. For each class, it is defined as the ratio of true positives to the average of true positives and false positives.

Precision is a Positive prediction accuracy. Precision = True Positive /(True Positive + False Positive) as in Figure 4.17. Recall A classifier's recall is its capacity to discover all positive occurrences. For each class, it is defined as the ratio of true positives to the average of true positives and false negatives.

The recall is a proportion of positives Recall = True Positive /(True Positive +False Negative) as in Figure 4.17.

F1 Score The F1 score is determined as a calculated average of accuracy and recall. with 1.0 being the highest and 0.0 being the poorest. F1 scores are lower than accuracy measurements because of factors in precision and recall. Use the average value of F1 rather than global accuracy to compare classifier models. F1 Score = 2*(Recall * Precision) / (Recall + Precision) as in Figure 4.17.

4.4.5.3 Accuracy Comparison

For each disease different machine learning algorithm is applied to analyze the best algorithm .

i. Kidney Disease

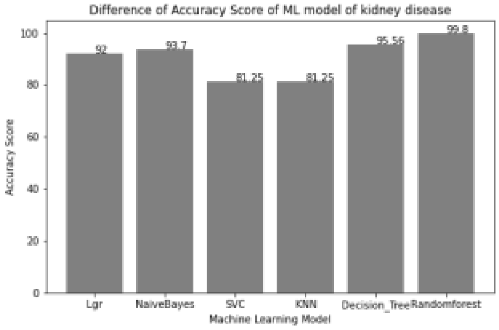


Figure 4.18. Represents the accuracy of the different algorithm on Kidney datasets

Figure 4.18 Shows that Random forest has higer accuracy of 99.8 percentage . Hence the pickel model is created for Random forest on kindey disease .

ii. Heart Disease

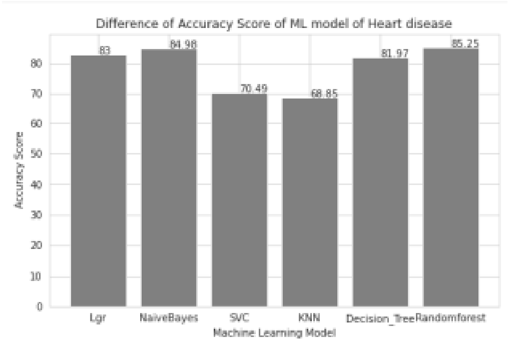


Figure 4.19 Represents the accuracy of the different algorithm on Heart datasets

Fig. 17 Shows that Random forest has higher accuracy of 85.25 percentage . Hence the pickel model is created for Random forest on Heart disease .

iii. Liver Disease

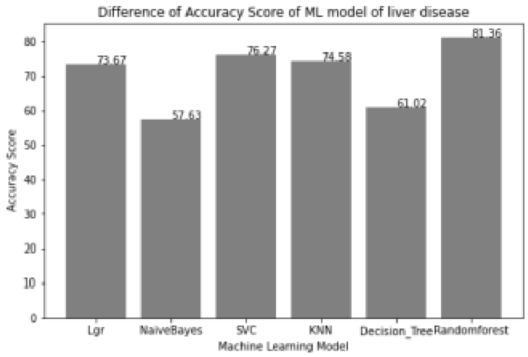


Figure 4.20. Represents the accuracy of the different algorithm on Liver datasets

Figure 4.20 Shows that Random forest has higher accuracy of 81.36 percentage . Hence the pickel model is created for Random forest on Liver disease .

iv. Breast Cancer

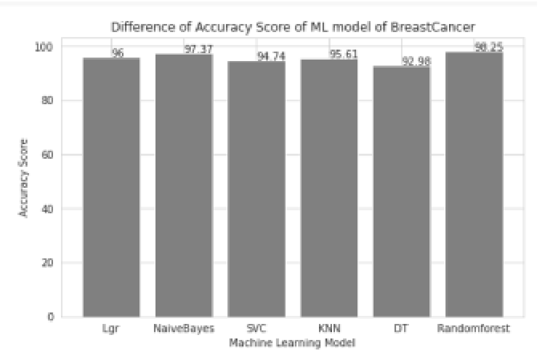


Figure 4.21. Represents the accuracy of the different algorithm on Breast datasets

Figure 4.21 Shows that Random forest has higer accuracy of 98.25 percentage . Hence the pickel model is created for Random forest on Breast Cancer .

v. Diabetes Disease

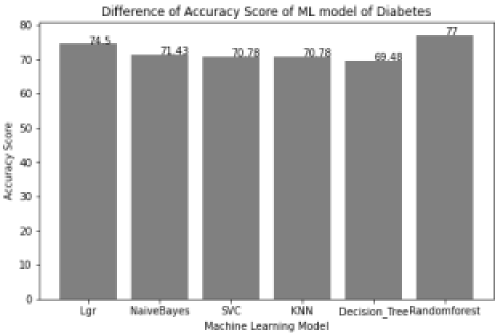


Figure 4.22. Represents the accuracy of different algorithm on Diabetes datasets

Figure 4.22. Shows that Random forest has higer accuracy of 77 percentage . Hence the pickel model is created for Random forest on Diabetes disease .

4.4.6 FRONTEND PAGES

This web application is designed to assist microlabotariest in predicting diseases based on various patient features. With the use of machine learning algorithms and deep learning techniques, the application provides accurate disease prediction for various illnesses including diabetes, heart disease, liver disease, breast cancer, kidney disease, malaria, and pneumonia. The web application consists of 10 pages, including a home page, a disease prediction page for each of the seven diseases, a report page, and pages for inputting patient data.

The pages are designed using React JS, a popular JavaScript library for building user interfaces. Email JS is used for mailing reports to patients, and Firebase is used for saving patient details and reports of predictions. Each page has a specific purpose and is designed to provide a seamless experience for doctors and patients alike.

With its user-friendly interface and accurate predictions, this web application is a valuable tool for doctors in diagnosing diseases and providing appropriate treatment. The use of machine learning and deep learning techniques ensures that the predictions are accurate and reliable. The combination of React JS, Email JS, and Firebase provides a secure and efficient platform for storing patient data and generating reports.

This web application is useful for doctors to predict disease based on the fetures .

Web application consists of 10 Pages.

1. Home Page
2. Disease Page
3. Diabetes Prediction Page
4. Heart Disease Prediction Page
5. Liver disease prediction page
6. Breast cancer Prediction page
7. Kidney Disease Prediction page
8. Malaria Prediction Page
9. Pneumonia Prediction page
10. Analysis Page

In today's world, predicting diseases has become a crucial aspect of healthcare. With the help of technology and machine learning, doctors can now predict diseases accurately. Web applications for disease prediction have made it easier for doctors to analyze the symptoms and predict the diseases. This web application is developed using React JS and Email JS for mailing, and Firebase for the database. It consists of 10 pages that include home, disease prediction pages, and report page.

4.4.6.1 PAGES DESCRIPTIONS

Home Page

The home page of this web application consists of navigation to different pages like disease prediction pages and report page. It also includes information about the web application and its features. The home page serves as the landing page for the web application.

Disease Prediction Pages

The disease prediction pages consist of seven pages that include diabetes, heart disease, liver disease, breast cancer, kidney disease, malaria, and pneumonia prediction pages. These pages are developed using React JS and are user-friendly. Users can enter their details like age, gender, and symptoms to predict the disease. The data entered by the user is sent to the backend, which is developed using machine learning algorithms to predict the disease accurately.

Diabetes Prediction Page

The diabetes prediction page is developed using machine learning algorithms to predict diabetes. Users can enter their age, gender, body mass index (BMI), and other symptoms like thirst, hunger, and fatigue. The data entered by the user is preprocessed to eliminate any errors and then fed into the machine learning model. The model then predicts whether the user has diabetes or not.

Heart Disease Prediction Page

The heart disease prediction page is developed using machine learning algorithms to predict heart diseases. Users can enter their age, gender, chest pain type, resting blood pressure, serum cholesterol levels, and other symptoms like smoking, exercise, and obesity. The data entered by the user is preprocessed to eliminate any errors and then fed into the machine learning model. The model then predicts whether the user has heart disease or not.

Liver Disease Prediction Page

The liver disease prediction page is developed using machine learning algorithms to predict liver diseases. Users can enter their age, gender, total bilirubin, direct bilirubin, alkaline phosphate, alanine aminotransferase (ALT), and other symptoms like fatigue and abdominal pain. The data entered by the user is preprocessed to eliminate any errors and then fed into the machine learning model. The model then predicts whether the user has liver disease or not.

Breast Cancer Prediction Page

The breast cancer prediction page is developed using machine learning algorithms to predict breast cancer. Users can enter their age, gender, BMI, breast density, family history, and other symptoms like breast pain and nipple discharge. The data entered by the user is preprocessed to eliminate any errors and then fed into the machine learning model. The model then predicts whether the user has breast cancer or not.

Kidney Disease Prediction Page

The kidney disease prediction page is developed using machine learning algorithms to predict kidney diseases. Users can enter their age, gender, serum creatinine, blood pressure, and other symptoms like fatigue, anemia, and swollen hands and feet. The data entered by the user is preprocessed to eliminate any errors and then fed into the machine learning model. The model then predicts whether the user has kidney disease or not.

Malaria Prediction Page

The malaria prediction page is developed using deep learning algorithms to predict malaria. Users can upload an image of their blood smear, and the deep learning model analyses the image and predicts whether the user has malaria or not.

Pneumonia Prediction Page

The pneumonia prediction page is developed using deep learning algorithms to predict pneumonia. Users can upload an image of their chest X-ray, and the deep learning model analyses the image and predicts whether the user has pneumonia or not.

In conclusion, the development of a web application for disease prediction and classification using machine learning and deep learning techniques is a significant contribution to the medical industry. The system provides doctors with a user-friendly platform to predict and classify diseases based on patients' features. The web application consists of ten pages, including the home page, disease page, and prediction pages for various diseases such as diabetes, heart disease, liver disease, breast cancer, kidney disease, malaria, and pneumonia. The report page enables doctors to access patients' histories and send reports via email to their patients.

The web application was developed using React.js, a popular JavaScript library for building user interfaces, and Email.js for email functionality. Firebase, a cloud-based platform, was utilized for data storage, ensuring that patients' details and prediction reports are safe and easily accessible.

Overall, the web application offers a convenient and reliable means of disease prediction and classification. With the use of machine learning and deep learning techniques, doctors can provide accurate diagnoses and timely treatments, leading to improved patient outcomes.

4.4.7 USE CASE DIAGRAM

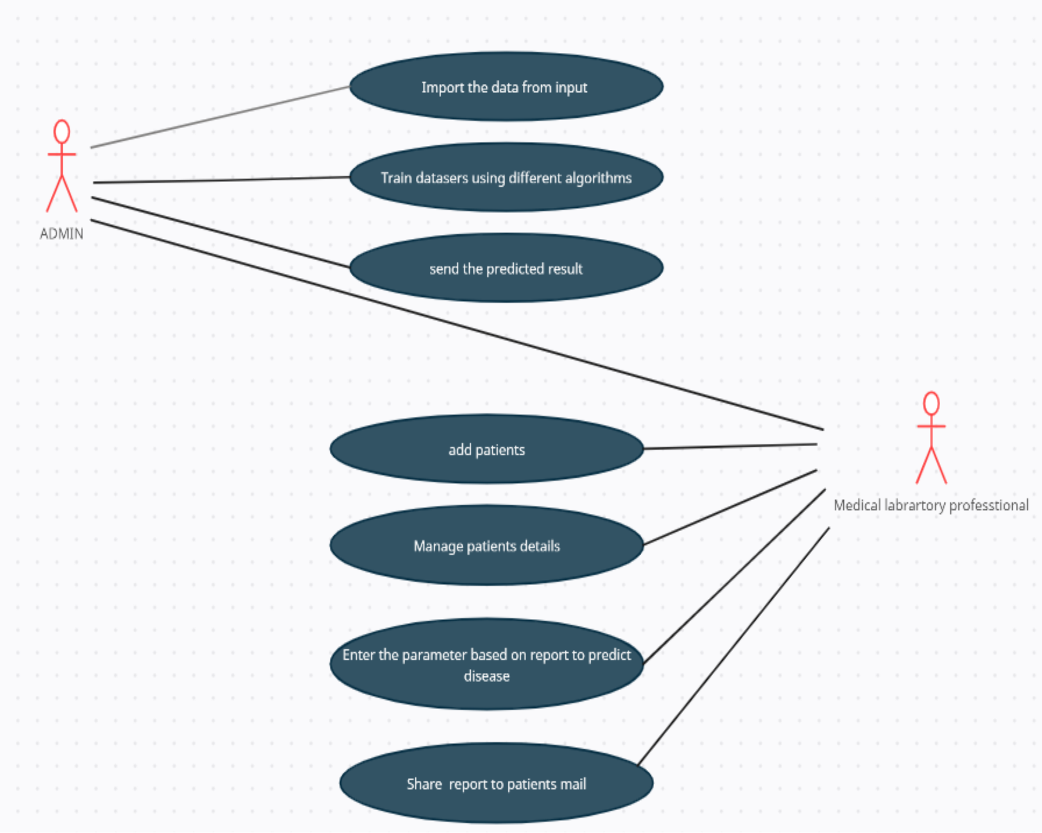

ADMIN
Import the data from input
Train datasers using different algorithms
send the predicted result
add patients
Manage patients details
Enter the parameter based on report to predict disease
Share report to patients mail
Medical labrartory professtional

Figure 4.23. Use case diagram

ADMIN ROLE

The admin role Figure 4.23. is responsible for maintaining the machine learning and deep learning models used for disease prediction. The admin has access to the entire application and can modify the algorithms used in the models. The admin can also update the database used for training the models and ensure that the system is running smoothly. Additionally, the admin can manage the accounts of other users, including medical laboratory professionals and patients.

MEDICAL LABORATORY PROFESSIONAL

The medical laboratory professional Figure 4.23. role has access to the patient report management feature of the application. These professionals are responsible for collecting patient data and inputting it into the system. They can also view the predictions made by the system and verify their accuracy. This is an important role in ensuring the accuracy of the machine learning models used in the application.

The medical laboratory professional can also manage patient information such as name, age, gender, and medical history. They can add new patient data and edit existing data, such as test results, images, and other medical information. They can also view the report history of each patient and share the reports with the patient via email.

4.4.8 SOFTWARE DESCRIPTION

The project was developed using various software tools and technologies that played a vital role in ensuring its success. In this section, we will discuss the software description and installation guide for the different software tools used in this project.

IDE: Visual Studio

Visual Studio is a fully featured integrated development environment (IDE) for building and developing web applications. It provides a rich set of tools and features for designing, coding, and testing applications. The installation process for Visual Studio is straightforward and can be completed by downloading the installation package from the official website and following the on-screen instructions.

Backend Software:

Python - Python is a popular programming language widely used in scientific computing, web development, data analysis, and artificial intelligence. It is easy to learn and offers a vast collection of libraries and modules for various tasks. To install Python, download the installer from the official website and run the installer program.

Scikit-learn - Scikit-learn is a powerful library for machine learning in Python. It offers a variety of machine learning algorithms for classification, regression, clustering, and dimensionality reduction. To install Scikit-learn, open the command prompt and type "pip install scikit-learn" and press enter.

Pickle - Pickle is a Python module used to serialize and deserialize Python objects. It allows objects to be saved to a file and then loaded later, making it useful for saving machine learning models. To install pickle, open the command prompt and type "pip install pickle" and press enter.

Pandas - Pandas is a Python library for data manipulation and analysis. It provides data structures for efficiently storing and manipulating large datasets. To install Pandas, open the command prompt and type "pip install pandas" and press enter.

NumPy - NumPy is a powerful library for numerical computing in Python. It provides tools for working with arrays, linear algebra, and other mathematical operations. To install NumPy, open the command prompt and type "pip install numpy" and press enter.

Pillow - Pillow is a fork of the Python Imaging Library (PIL) that adds support for newer Python versions. It provides tools for working with images, including opening, manipulating, and saving image files. To install Pillow, open the command prompt and type "pip install pillow" and press enter.

Matplotlib - Matplotlib is a Python library for creating static, animated, and interactive visualizations in Python. It provides a variety of tools for creating plots, charts, and graphs. To install Matplotlib, open the command prompt and type "pip install matplotlib" and press enter.

Keras - Keras is a high-level neural network library written in Python. It provides a simple and intuitive interface for building and training deep learning models. To install Keras, open the command prompt and type "pip install keras" and press enter.

Frontend Software:

React JS - React JS is a popular JavaScript library for building user interfaces. It provides a declarative syntax for describing UI components and their behavior. To install React JS, open the command prompt and type "npm install react" and press enter.

Email JS - Email JS is a JavaScript library for sending emails directly from client-side code. It provides a simple and easy-to-use API for sending emails without requiring a backend server. To install Email JS, open the command prompt and type "npm install emailjs-com" and press enter.

Node JS - Node JS is an open-source, cross-platform JavaScript runtime environment. It allows developers to run JavaScript code outside of a web browser, making it useful for building server-side applications. To install Node JS, download the installer from the official website and run the installer program.

Database Software:

Firebase - Firebase is a popular cloud-based NoSQL database platform. It provides tools and services for building web and mobile applications, including real-time databases, user authentication, hosting, and storage. To use Firebase, create an account on the official website and follow the instructions.

4.5 MERITS OF PROPOSED SYSTEM

One of the primary benefits of this web application is its high accuracy in predicting diseases. The use of machine learning and deep learning algorithms, such as KNN, SVM, Navie Bayes, Decision Tree, Logistic Regression, Random Forest, and Convolutional Neural Network, has enabled the application to predict diseases with impressive accuracy. For example, the Random Forest algorithm has achieved a 96% accuracy in predicting malaria, while pneumonia has been predicted with 95% accuracy. Breast cancer has a prediction accuracy of 98.25%, and kidney disease has a staggering 99.8% accuracy. This high accuracy can help medical professionals to identify diseases early on, providing patients with prompt treatment, ultimately leading to better health outcomes.

Another benefit of this web application is its user-friendly design. The application is easy to navigate, and patients can enter their information quickly and efficiently. The front-end of the web application has been created using ReactJS, a popular JavaScript library for building user interfaces. The use of

ReactJS has made it easy to create interactive and engaging pages, with a smooth and seamless user experience. The pages are designed to be responsive, which means they can be accessed from any device, including mobile phones, tablets, and desktop computers.

In addition to its accuracy and user-friendliness, this web application also offers a high level of security. The patient's personal and medical information is stored in Firebase, a secure cloud database. Firebase offers a range of security features, including user authentication, data encryption, and server-side security rules. This ensures that patient information is kept private and secure, preventing unauthorized access.

Moreover, the web application offers several features that enhance the patient's experience. Patients can access their reports through the Report Page, which stores a history of their previous predictions. They can also choose to receive their reports via email, which is integrated into the application using the EmailJS library. EmailJS makes it easy to send emails directly from the web application, ensuring that patients receive their reports promptly.

Finally, the web application is highly scalable, which means that it can handle a large number of users and a significant amount of data. The use of Firebase as the database and the deployment of the application on cloud-based servers means that it can easily handle a growing user base, without experiencing performance issues.

CHAPTER 5

CHAPTER 5

RESULTS AND DISCUSSION

5.1 EXPERIMENT RESULTS

5.1.1 Machine Learning Model Classifier

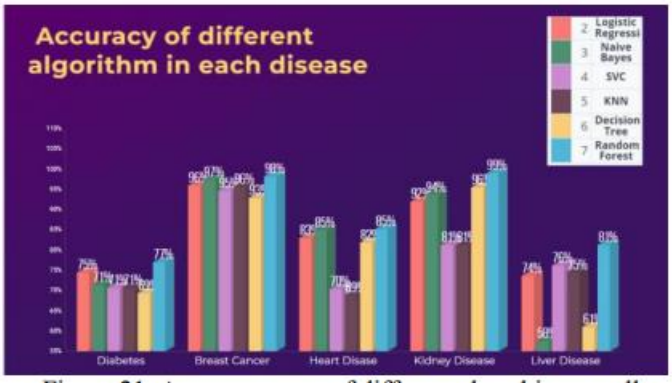


Figure 5.1. Accuracy score of different algorthims on all disease

Figure 5.1. Represents the accuracy score for different algorthms on each disase datasets on machine leaening model . Based on the analysis of Figure 5.1. , Random forest has high accuracy for all disease , Hence the Random forest is created a model for disease prediction .

Disease Accuracy	score
Diabetes	77 %
Breast Cancer	98.25%
Heart Disease	85.75%
Kidney Disease	99.8%
Liver Disease	81.36%

Table 5.1. Accuracy score of different disease of high accuracy

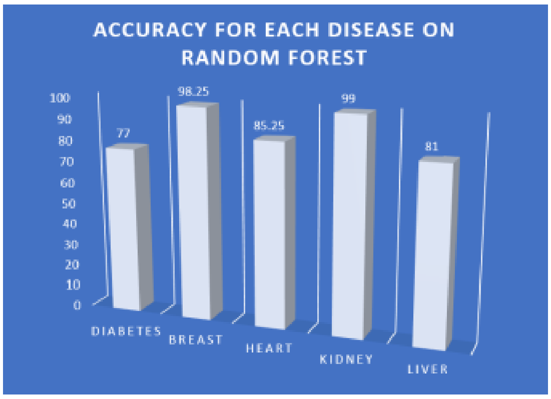


Figure 5.2. Represents the accuracy for all disease on Random forest algorithm

Figure 5.2. and Table 5.1. demonstrate the display of individual base classifiers for each disease .

5.1.2 Deep Learning

Model Deep learning model is created with CNN layers and tested with validation data which has high of Figure 5.3. and Table 5.2.

DL Model	Accuracy score
Malaria	96%
Pneumonia	95%

Table 5.2. Accuracy of CNN model

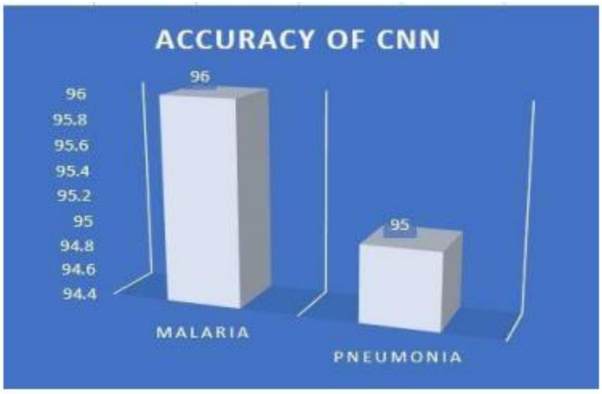


Figure 5.3. Represents the accuracy of malaria and pneumonia in CNN

Figure 5.3. and Table 5.2 demonstrate the accuracy of image datets under deep leanring model . This model is saved by save_weight from keras and used as backend for web application.

5.1.3 Front End web Design

This web application is useful for doctors to predict disease based on the fetures . Web application consists of 10 Pages.

1. Home Page
2. Test Page
3. Disease Page
4. Diabetes Prediction Page
5. Heart Disease Prediction Page
6. Liver disease prediction page
7. Breast cancer Prediction page
8. Kidney Disease Prediction page
9. Malaria Prediction Page
10. Pneumonia Prediction page
11. Report Page

Here the some pages from web applications

HOME PAGE

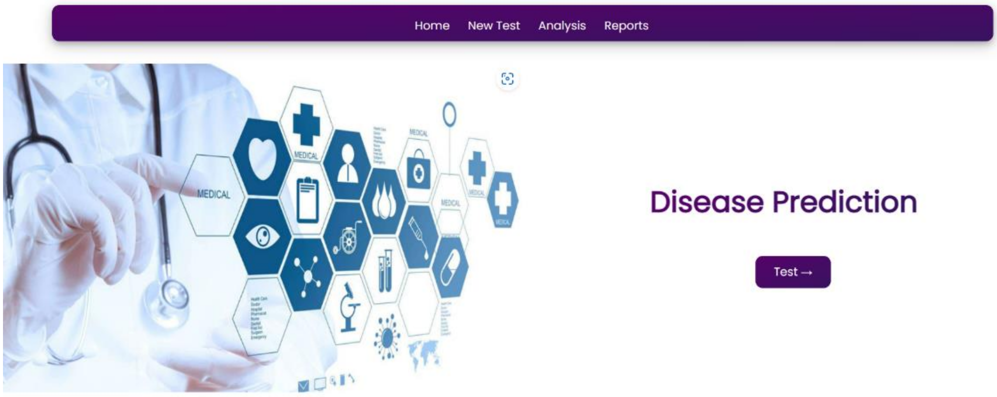


Figure 5.4. Home Page

TEST PAGE

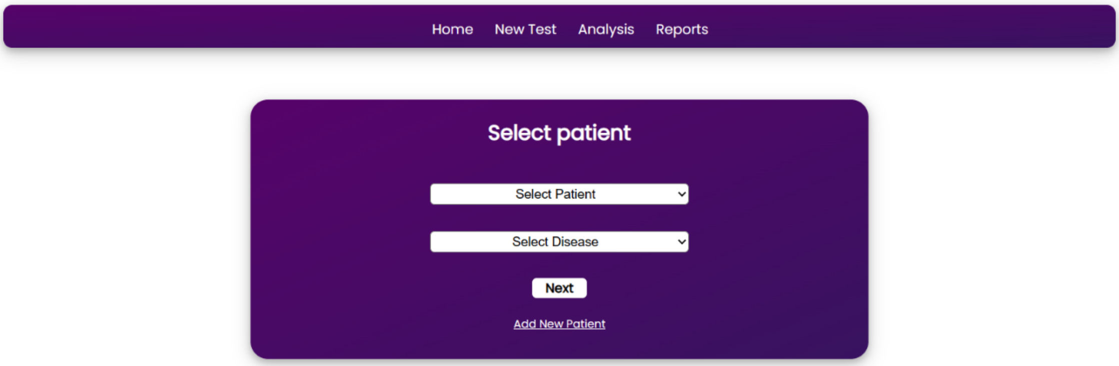


Figure 5.5. Test page

NEW PATIENTS PAGE

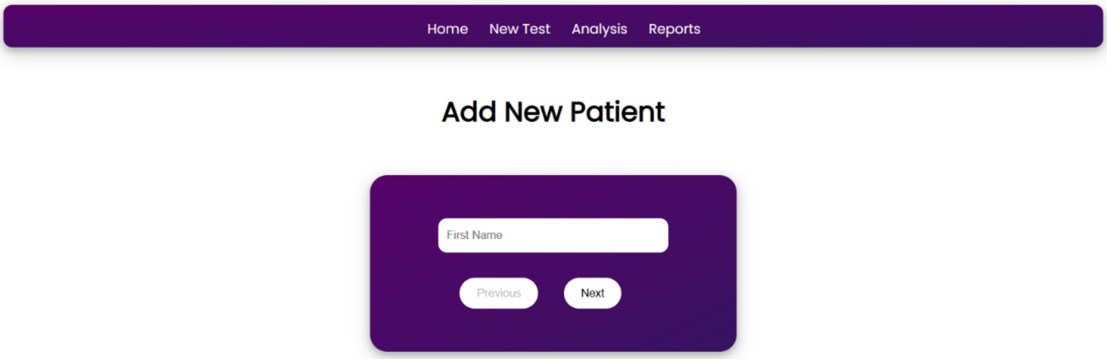


Figure 5.6. New Patient Page

TEST PAGE BASED ON DISEASE

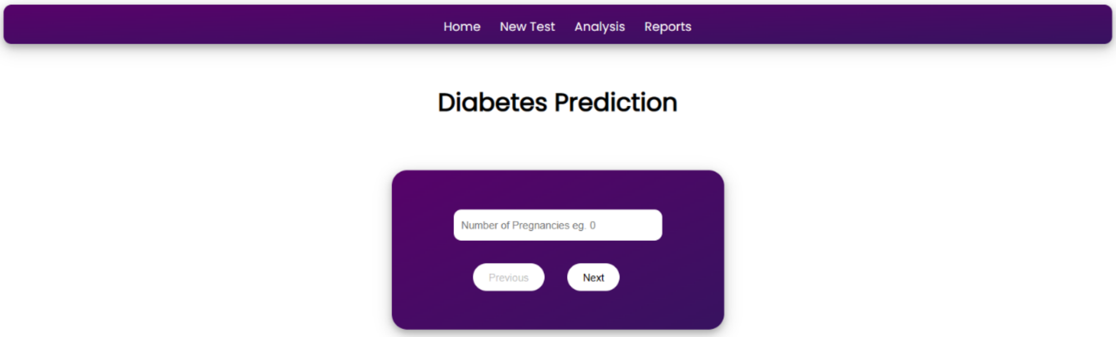


Figure 5.7. Test Page for Diabetes

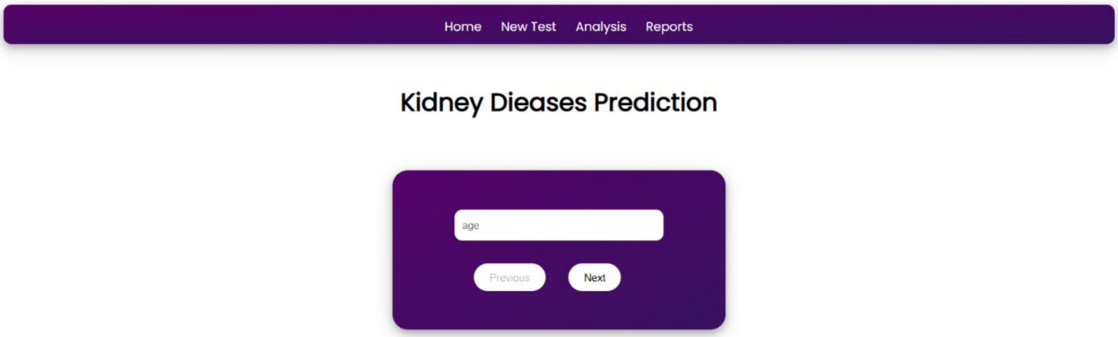
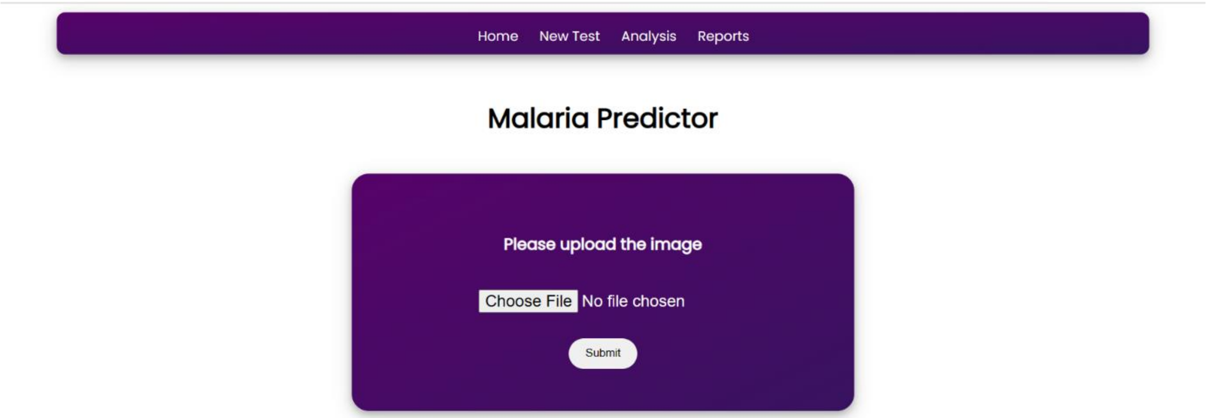


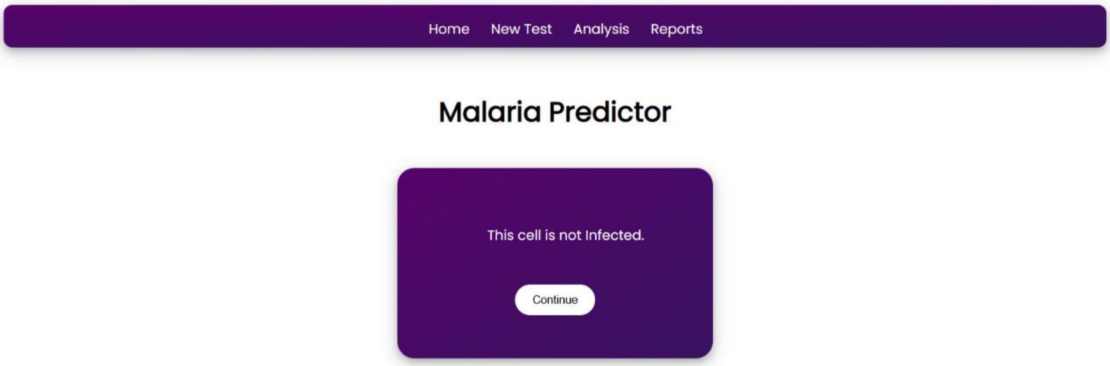
Figure 5.8. Test page for Kidney



Please upload the image

Figure 5.9. Test page for Malaria

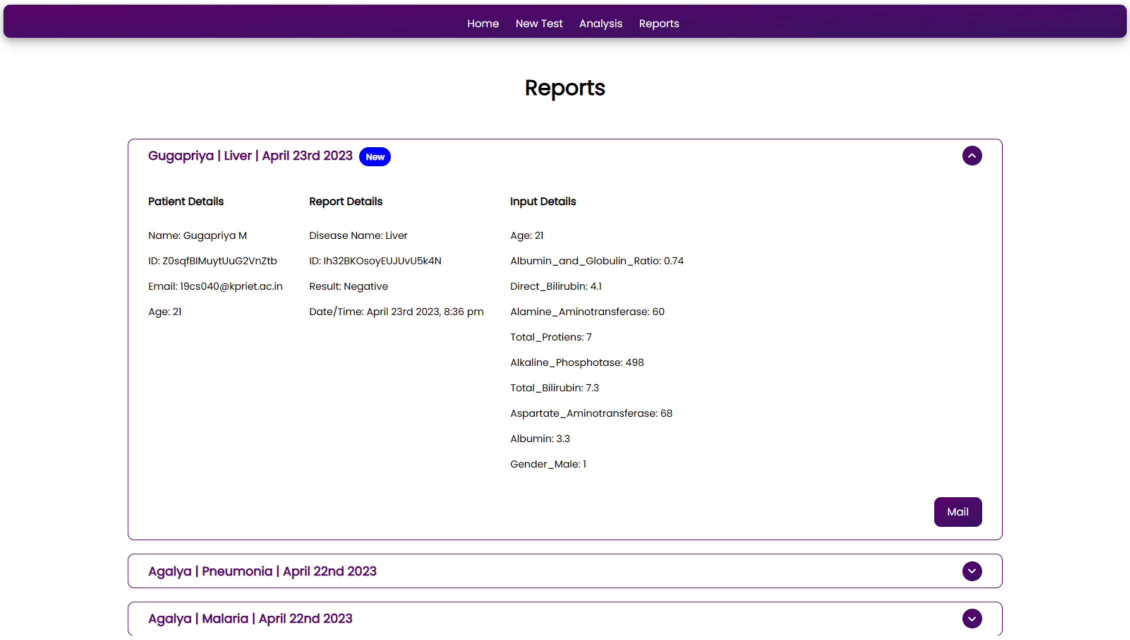
RESULT PAGE



This cell is not Infected.

Figure 5.10. Results Page

REPORTS PAGE



Reports

Gugapriya | Liver | April 23rd 2023 New

Patient Details

Name: Gugapriya M

ID: Z0sqfBIMuytUuG2VnZtb

Email: 19cs040@kpriet.ac.in

Age: 21

Report Details

Disease Name: Liver

ID: Ih32BKOsoyEUJUvU5k4N

Result: Negative

Date/Time: April 23rd 2023, 8:36 pm

Input Details

Age: 21

Albumin_and_Globulin_Ratio: 0.74

Direct_Bilirubin: 4.1

Alamine_Aminotransferase: 60

Total_Protiens: 7

Alkaline_Phosphotase: 498

Total_Bilirubin: 7.3

Aspartate_Aminotransferase: 68

Albumin: 3.3

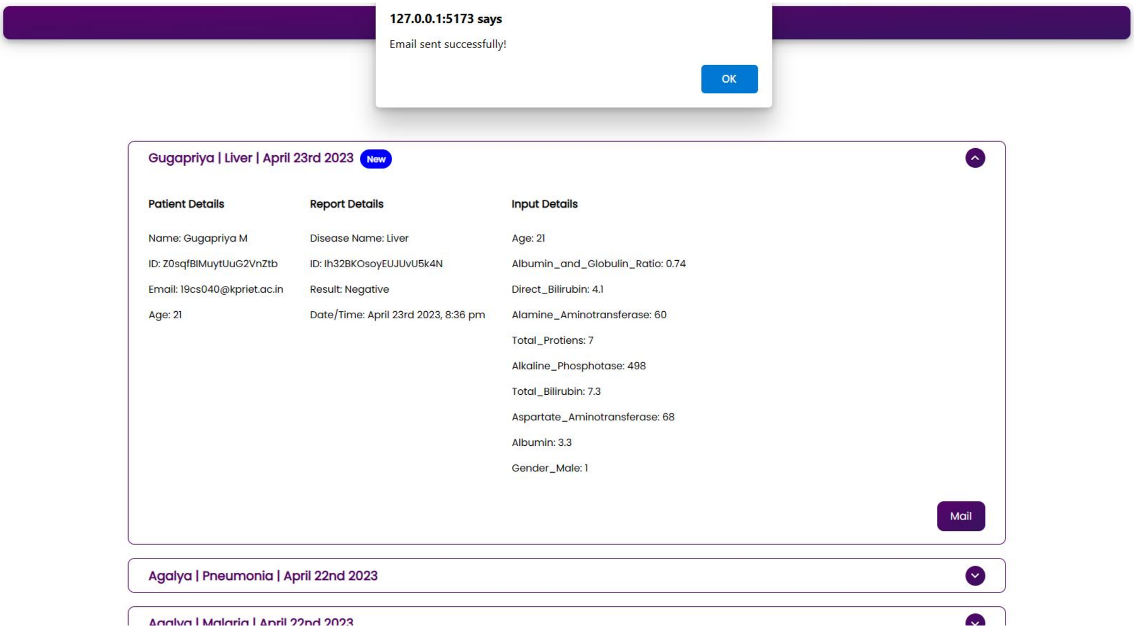
Gender_Male: 1

Agalya | Pneumonia | April 22nd 2023

Agalya | Malaria | April 22nd 2023

Figure 5.11. Reports of patients

REPORTS AFTER MAIL



127.0.0.1:5173 says

Email sent successfully!

Gugapriya | Liver | April 23rd 2023 New

Patient Details

Name: Gugapriya M

ID: Z0sqfBIMuytUuG2VnZtb

Email: 19cs040@kpriet.ac.in

Age: 21

Report Details

Disease Name: Liver

ID: Ih32BKOsoyEUJUvU5k4N

Result: Negative

Date/Time: April 23rd 2023, 8:36 pm

Input Details

Age: 21

Albumin_and_Globulin_Ratio: 0.74

Direct_Bilirubin: 4.1

Alamine_Aminotransferase: 60

Total_Protiens: 7

Alkaline_Phosphotase: 498

Total_Bilirubin: 7.3

Aspartate_Aminotransferase: 68

Albumin: 3.3

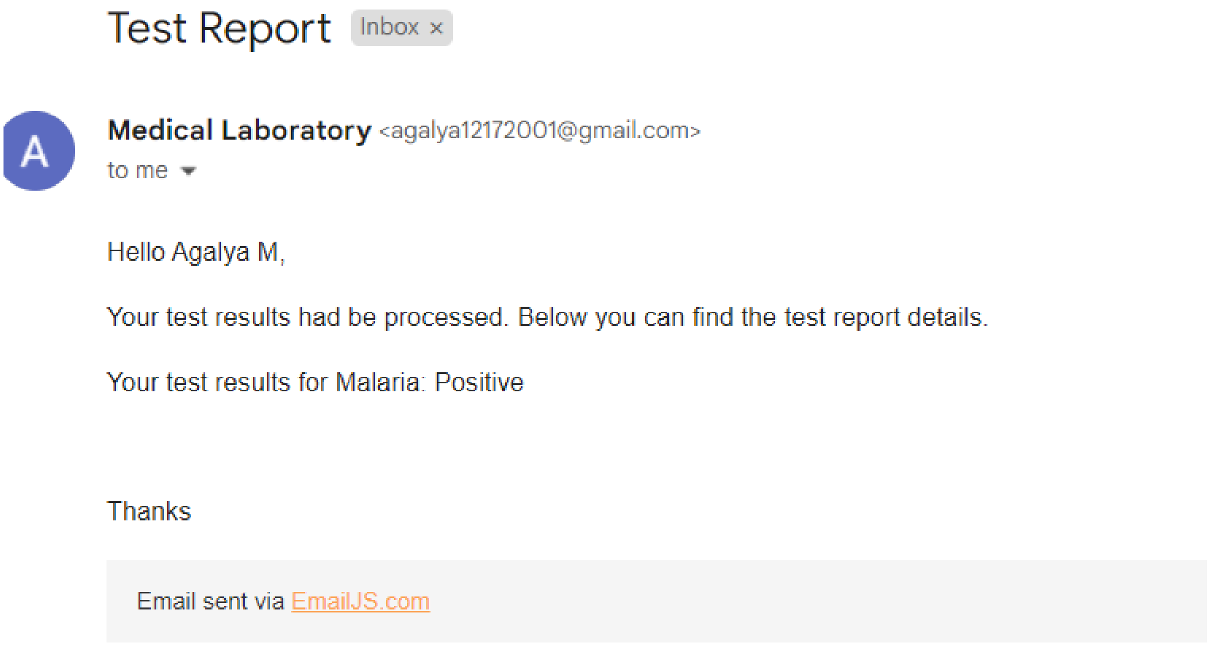
Gender_Male: 1

Agalya | Pneumonia | April 22nd 2023

Agalya | Malaria | April 22nd 2023

Figure 5.12. Page with pop up after clicking mail button

MAIL PAGE



Test Report

Medical Laboratory <agalya12172001@gmail.com>

to me

Hello Agalya M,

Your test results had be processed. Below you can find the test report details.

Your test results for Malaria: Positive

Thanks

Email sent via EmailJS.com

Figure 5.13. Mail Page

CHAPTER 6

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

In conclusion, the development of a web application for disease prediction and classification using machine learning and deep learning techniques is a significant contribution to the field of healthcare. The application provides a user-friendly interface that allows doctors to predict diseases based on various patient features. The system is equipped with an accurate and efficient prediction algorithm, which helps doctors to diagnose diseases with a high level of accuracy.

The web application is composed of ten pages, which include the home page, disease page, and various disease prediction pages such as diabetes, heart disease, liver disease, breast cancer, kidney disease, malaria, and pneumonia. The application also includes a report page that displays the patient's history and provides the option to send the report to the patient's email. The pages were developed using React.js, which is a popular JavaScript library that allows for the creation of interactive user interfaces.

The application's back-end is composed of machine learning and deep learning algorithms. The machine learning algorithm collects datasets of different diseases, preprocesses the data, trains and tests the datasets, and creates a model for different algorithms such as KNN, SVM, Navie Bayes, Decision Tree, Logistic Regression, and Random Forest. The deep learning algorithm uses image datasets, preprocesses the images, and creates a model using Convolutional Neural Networks.

The application's models are created using the pickle library in Python for machine learning algorithms and the keras library's save_weights method for deep learning algorithms. The models are then deployed using the Flask library in Python to connect to the front-end.

The application also includes Firebase for data storage, which allows medical professionals to manage and add patients' reports and information. Additionally, the application includes email.js for mailing patient reports to their respective emails.

One of the significant advantages of this web application is its accuracy in predicting diseases. The models used in the application have shown high accuracy rates for different diseases. The Random Forest algorithm has an accuracy rate of 96% for malaria, 95% for pneumonia, 77% for diabetes, 98.25% for breast cancer, 85.75% for heart disease, and 99.8% for kidney disease. These accuracy rates make the application highly reliable and useful for medical professionals.

Another significant advantage of the web application is its user-friendly interface. The application is easy to use and navigate, which makes it accessible to medical professionals with varying levels of technical expertise. The application's interface is also interactive, which enhances the user experience and provides an engaging platform for disease prediction and diagnosis.

In summary, the development of a web application for disease prediction and classification using machine learning and deep learning techniques is a significant contribution to the field of healthcare. The application's high accuracy rates and user-friendly interface make it an invaluable tool for medical professionals in disease prediction and diagnosis. Additionally, the application's integration with Firebase and email.js provides efficient data storage and management, as well as an effective means of communicating patient reports. The use of React.js for the front-end and Python for the back-end provides a robust and scalable application that can be easily maintained and updated.

Overall, the development of this web application is a promising step towards using machine learning and deep learning techniques in the field of healthcare. The use of these technologies can significantly improve disease prediction and diagnosis, leading to better patient outcomes and an overall improvement in the quality of healthcare.

6.2 FUTURE SCOPE

1) Addition of More Disease Prediction Models:
   The application can be enhanced by adding more disease prediction models. For instance, models that can predict respiratory tract infections, skin infections, and mental health diseases can be included. This will ensure that the application caters to a wider audience and offers more precise and reliable predictions.
2) Integration with Electronic Health Record Systems:
   Integrating the web application with electronic health record (EHR) systems can enhance its functionality. The integration can enable the application to pull the patient’s medical history from the EHR system, which can improve the accuracy of the predictions. Furthermore, integration with EHR systems can allow the application to suggest additional tests or treatments based on the patient's history.
3) Mobile Application:
   Developing a mobile application for the web application can increase its accessibility and convenience. A mobile application can allow users to

access the application from anywhere and at any time, making it more user-friendly. Additionally, the mobile application can provide notifications to patients reminding them of their appointments or test results.

4) Real-Time Data Visualization:

   The application can be enhanced by providing real-time data visualization of the predicted disease. This can be achieved by integrating the application with data visualization tools such as Tableau or PowerBI. Real-time data visualization can provide doctors with more comprehensive insights into the patients' condition, which can aid in better decision-making.

5) Cloud-Based Deployment:

   Deploying the web application on the cloud can improve its scalability, security, and availability. Cloud deployment can enable the application to handle a large volume of requests without experiencing downtime or latency issues. Additionally, cloud deployment can provide better security for patient data and ensure that the application is available 24/7.

6) Multilingual Support:

   Incorporating multilingual support can increase the reach of the application. This feature can enable users from different regions or countries to use the application in their native language, thereby making it more user-friendly and accessible. Providing multilingual support can also enable the application to cater to a more diverse patient population.

7) Integration with Wearable Devices:

   Integrating the web application with wearable devices can enhance its functionality. Wearable devices such as fitness trackers can provide doctors with real-time data about the patient's health, which can aid in better decision-making. Additionally, the integration can enable the application to provide personalized health recommendations to patients based on their daily activity, heart rate, and other vitals.

8) Improved User Interface:

Improving the user interface can enhance the user experience and make the application more user-friendly. The user interface can be made more visually appealing by using attractive colors, fonts, and graphics. Additionally, the interface can be made more intuitive by providing clear instructions and easy-to-use navigation.

9) Improved Machine Learning Models:

   The machine learning models can be further improved by incorporating more advanced algorithms and techniques. This can improve the accuracy of the predictions and ensure that the application provides more reliable and precise recommendations to patients. Additionally, the models can be updated regularly to include the latest research findings and medical knowledge.

10) Patient Feedback System:

   Incorporating a patient feedback system can help improve the application's functionality and usability. A feedback system can enable patients to provide feedback on their experience using the application, which can provide valuable insights for improving the application. Additionally, the feedback system can enable the application to provide personalized recommendations based on the patients' feedback and preferences.

In conclusion, the disease prediction web application has the potential to revolutionize the healthcare industry by providing accurate and reliable disease predictions to patients. However, there is always room for improvement, and the above future works can enhance the functionality and usability of the application.

APPENDIX I

SOURCE CODE

BACKEND CODE

App.py

```
from flask import Flask, render_template, request, flash, redirect, jsonify
import pickle
import numpy as np
from PIL import Image
from tensorflow.keras.models import load_model
from flask_cors import CORS
from json import JSONEncoder
import json
from flask_json import FlaskJSON, JsonError, json_response, as_json
app = Flask(__name__)
CORS(app,  resources={r"/*": {"origins": "*"}})
FlaskJSON(app)
class CustomJSONEncoder(JSONEncoder):
    def default(self, obj):
        if isinstance(obj, np.int64):
            return int(obj)
        return JSONEncoder.default(self, obj)
app.json_encoder = CustomJSONEncoder
def predict(values, dic):
    if len(values) == 8:
        model = pickle.load(open('models/diabetes.pkl', 'rb'))
        values = np.asarray(values)
        return model.predict(values.reshape(1, -1))[0]
    elif len(values) == 26:
        model = pickle.load(open('models/breast_cancer.pkl', 'rb'))
        values = np.asarray(values)
        return model.predict(values.reshape(1, -1))[0]
    elif len(values) == 13:
        model = pickle.load(open('models/heart.pkl', 'rb'))
        values = np.asarray(values)
        return model.predict(values.reshape(1, -1))[0]
    elif len(values) == 18:
        model = pickle.load(open('models/kidney.pkl', 'rb'))
        values = np.asarray(values)
        return model.predict(values.reshape(1, -1))[0]
    elif len(values) == 10:
        model = pickle.load(open('models/liver.pkl', 'rb'))
        values = np.asarray(values)
        return model.predict(values.reshape(1, -1))[0]
@app.route("/")
def home():
    return render_template('home.html')
@app.route("/diabetes", methods=['GET', 'POST'])
def diabetesPage():
    return render_template('diabetes.html')
@app.route("/cancer", methods=['GET', 'POST'])
def cancerPage():
    return render_template('breast_cancer.html')
@app.route("/heart", methods=['GET', 'POST'])
def heartPage():
    return render_template('heart.html')
@app.route("/kidney", methods=['GET', 'POST'])
def kidneyPage():
    return render_template('kidney.html')
@app.route("/liver", methods=['GET', 'POST'])
def liverPage():
    return render_template('liver.html')
@app.route("/malaria", methods=['GET', 'POST'])
def malariaPage():
    return render_template('malaria.html')
@app.route("/pneumonia", methods=['GET', 'POST'])
def pneumoniaPage():
    return render_template('pneumonia.html')
@app.route("/predict", methods=['POST', 'GET'])
def predictPage():
    try:
        if request.method == 'POST':
            print('post data', request.get_json())
            to_predict_dict = request.form.to_dict()

            print("preduct dict ", to_predict_dict)
            to_predict_list = list(map(float, list(to_predict_dict.values())))
            print("preduct list ", to_predict_list)
            pred = predict(to_predict_list, to_predict_dict)
    except:
        message = "Please enter valid Data"
        return render_template("home.html", message=message)
    return render_template('predict.html', pred=pred)
@app.route("/malariapredict", methods=['POST', 'GET'])
def malariapredictPage():
    if request.method == 'POST':
        try:
            if 'image' in request.files:
                img = Image.open(request.files['image'])
                img = img.resize((36, 36))
                img = np.asarray(img)
                img = img.reshape((1, 36, 36, 3))
                img = img.astype(np.float64)
                model = load_model("models/malaria.h5")
                pred = np.argmax(model.predict(img)[0])
        except:
            response = jsonify({'message': int(-1)})
            return response
    response = jsonify({'message': int(pred)})
    return response
@app.route("/pneumoniapredict", methods=['POST', 'GET'])
def pneumoniapredictPage():
    if request.method == 'POST':
        try:
            if 'image' in request.files:
                img = Image.open(request.files['image']).convert('L')
                img = img.resize((36, 36))
                img = np.asarray(img)
                img = img.reshape((1, 36, 36, 1))
                img = img / 255.0
                model = load_model("models/pneumonia.h5")
                pred = np.argmax(model.predict(img)[0])
        except:
            response = jsonify({'message': int(-1)})
            return response
    response = jsonify({'message': int(pred)})
    return response


if __name__ == '__main__':
    app.run(debug=True)
```

FRONTEND

kidneyperdict.jsx

```
import React, { useState } from "react";
import GroupInput from "../../Components/groupInput";
import NavBar from "../../Components/Navbar";
import "../../Styles/globelStyles.css";
import { kidneyDieasesInput } from "../../Utils/utils";

const KidneyDisease = () => {
  const [kidneyDiseaseValues, setValues] = useState({
    age: "",
    bp: "",
    al: "",
    su: "",
    rbc: "",
    pc: "",
    pcc: "",
    ba: "",
    bgr: "",
    bu: "",
    sc: "",
    pot: "",
    wc: "",
    htn: "",
    dm: "",
    cad: "",
    pe: "",
    ane: "",
  });

  const onChange = (inputName, value) => {
    console.log(inputName, value);
    setValues({ ...kidneyDiseaseValues, [inputName]: value });
  };

  const onSubmit = () => {
    console.log({ kidneyDiseaseValues });
  };

  return (
    <div className="screen-wrapper-1">
      <NavBar />
      <div className="screen-wrapper-2">
        <div className="title">
          <h1>Kidney Dieases Prediction</h1>
        </div>
        <div className="input-wrapper">
          <GroupInput
            data={kidneyDieasesInput}
            dataValue={kidneyDiseaseValues}
            onChange={onChange}
            onSubmit={onSubmit}
            />
          </div>
        </div>
      </div>
    );
  };

  export default KidneyDisease;
```

NEW TEST

Newtest.jsx

```
import { collection, getDocs } from "firebase/firestore";
import React, { useEffect, useState } from "react";
import { Circles } from "react-loader-spinner";
import { useNavigate } from "react-router-dom";
import NavBar from "../../Components/Navbar";
import { db } from "../../Firebase";
import "../../Styles/globelStyles.css";
import { getDiseaseNavigationPath } from "../../Utils/utils";
import "./styles.css";

const NewTest = () => {
  const navigate = useNavigate();

  const [selectedPatient, setSelectedPatient] = useState(null);
  const [selectedDisease, setSelectedDisease] = useState(null);
  const [patientData, setPatientData] = useState([]);
  const [diseasesData, setDiseasesData] = useState([]);
  const [loader, setLoader] = useState(false);
```

```
useEffect(() => {
  getPatientAndDiseasesData();
}, []);

const getPatientAndDiseasesData = async () => {
  setLoader(true);
  try {
    await getDocs(collection(db, "patients")).then((querySnapshot) => {
      const apiPatientData = querySnapshot.docs.map((doc) => ({
        ...doc.data(),
        patientDocid: doc.id,
      }));
      setPatientData(apiPatientData);
    });
    await getDocs(collection(db, "diseases")).then((querySnapshot) => {
      const apiDiseasesData = querySnapshot.docs.map((doc) => ({
        ...doc.data(),
        docID: doc.id,
      }));
      setDiseasesData(apiDiseasesData);
    });
  } catch (error) {
    alert("Something went wrong while fetching patients list");
  }
  setLoader(false);
};

const handleClickNext = () => {
  if (selectedDisease === null || selectedPatient === null) {
    alert(
      `Please select ${
        !selectedPatient ? "Patient" : !selectedDisease ? "Disease" : ""
      }`
    );
  }
  const selectedPatientDetails = patientData.find(
    (item) => item.patientDocid === selectedPatient
  );
  console.log(selectedDisease);
  const selectedDiseaseDetails = diseasesData.find(
    (item) => item.docID === selectedDisease
  );
  console.log({ selectedPatientDetails, selectedDiseaseDetails });
  // ...navigate to the required test page
  const navigationPath = getDiseaseNavigationPath(
    selectedDiseaseDetails.diseaseName
  );
  console.log(navigationPath);
  navigate(navigationPath, {
    state: {
      patientDetails: selectedPatientDetails,
      diseaseDetails: selectedDiseaseDetails,
    },
  });
};

const handleClickAddNewPatient = () => {
  navigate("/addNewPatient");
};

return (
  <div className="screen-wrapper-1">
    <NavBar />
    <div className="screen-wrapper-2">
      <div className="new-test-wrapper">
        <div className="new-test-container">
          <div className="title">
            <h2>Select patient</h2>
          </div>
          {!loader ? (
            <>
              <div className="input-container">
                <select
                  name="patient"
                  id="patient1"
                  onChange={(e) => setSelectedPatient(e.target.value)}
                >
                  <option value="none">Select Patient</option>
                  {patientData.length > 0 &&
                    patientData.map((item) => (
                      <option
                        key={item.patientDocid}
                        value={item.patientDocid}
                      >{`${item?.firstName} ${item?.lastName}`}</option>
                    ))}
                </select>
```

```
</div>
<div className="input-container">
  <select
    name="disease"
    id="disease1"
    onChange={(e) => setSelectedDisease(e.target.value)}
  >
    <option value="none">Select Disease</option>
    {diseasesData.length > 0 &&
      diseasesData.map((item) => (
        <option key={item.docID} value={item.docID}>
          {item?.diseaseName}
        </option>
      ))}
  </select>
</div>
<div className="button-container">
  <div className="button" onClick={handleClickNext}>
    Next
  </div>
</div>
<div className="container-footer">
  <div
    className="add-new-patient-text"
    onClick={handleClickAddNewPatient}
  >
    Add New Patient
  </div>
</div>
```

```
              </>
            ) : (
              <Circles
                height="80"
                width="80"
                color="#fff"
                ariaLabel="circles-loading"
                wrapperStyle={{}}
                wrapperClass=""
                visible={true}
              />
            )}
          </div>
        </div>
      </div>
    </div>
  );
};

export default NewTest;
```

REPORTS PAGE

Reports.jsx

```
import { collection, getDocs } from "firebase/firestore";
import React, { useEffect, useState } from "react";
import { Circles } from "react-loader-spinner";
import NavBar from "../../Components/Navbar";
import ReportCard from "../../Components/reportCard";
import { db } from "../../Firebase";
import "../../Styles/globelStyles.css";
import "./reportsStyles.css";

const PatientReports = () => {
  const [activeCard, setActiveCard] = useState(1);
  const [reportsData, setReportsData] = useState([]);
  const [loader, setLoader] = useState(false);

  useEffect(() => {
    getReports();
  }, []);

  const getReports = async () => {
    setLoader(true);
    try {
      await getDocs(collection(db, "diseaseTests")).then((querySnapshot) => {
        const apiReportsData = querySnapshot.docs.map((doc) => ({
          ...doc.data(),
          id: doc.id,
        }));
        if (apiReportsData.length === 0) {
          setReportsData("No data");
        } else {
          setReportsData(apiReportsData);
        }
      });
    } catch (error) {
      alert("Something went wrong while fetching reports list");
    }
    setLoader(false);
  };

  const handleSelect = (id) => {
    setActiveCard(id === activeCard ? "" : id);
  };
  return (
    <div className="screen-wrapper-1">
      <NavBar />
      <div className="screen-wrapper-2">
        <div className="title">
          <h1>Reports</h1>
        </div>
        <div className="report-cards-wrapper">
          {reportsData.length > 0 ? (
            reportsData.map((item) => {
              return (
                <ReportCard
                  key={item.id}
                  data={item}
                  activeCardId={activeCard}
                  handleSelect={handleSelect}
                />
              );
            })
          ) : reportsData === "No data" ? (
            <div>No Reports Available</div>
          ) : loader ? (
            <Circles
```

```
              height="80"
              width="80"
              color="#000"
              ariaLabel="circles-loading"
              wrapperStyle={{}}
              wrapperClass=""
              visible={true}
            />
          ) : null}
        </div>
      </div>
    </div>
  );
};

export default PatientReports;
```

REFERENCES

[1] Nagasankar, Tamilselvi & Ankaryarkanni, B.. (2016). Performance Analysis of Edge Detection Algorithms on Various Image Types. Indian Journal of Science and Technology. 9. 10.17485/ijst/2016/v9i21/95207.

[2] Wenzhou Gao, Xiaoguang Zhang, Lei Yang and Huizhong Liu, "An improved Sobel edge detection," 2010 3rd International Conference on Computer Science and Information Technology, 2010, pp. 67-71, DOI: 10.1109/ICCSIT.2010.5563693.

[3] Anil K. Bharodiya, Atul M. Gonsai, An improved edge detection algorithm for X-Ray images based on the statistical range, Heliyon, Volume 5, Issue 10, 2019, e02743, ISSN 2405-8440.

[4] Y. Menga, Z. Zhang, H. Yin, et al. Automatic detection of particle size distribution by image analysis based on local adaptive canny edge detection and modified circular Hough transform Micron, 106 (2018), pp. 34-41

[5] K. Zhang, Y. Zhang, P. Wang, et al. An improved Sobel edge algorithm and FPGA implementation Procedia Comput. Sci., 131 (2018), pp. 243- 248

[6] Polat H, Danaei Mehr H, Cetin A. Diagnosis of Chronic Kidney Disease Based on Support Vector Machine by Feature Selection Methods. J Med Syst. 2017 Apr;41(4):55. doi: 10.1007/s10916-017-0703-x. Epub 2017 Feb 27. PMID: 28243816.

[7] N. Saranya, M. Sakthi Samyuktha, S. Isaac and B. Subhanki, "Diagnosing Chronic Kidney Disease using KNN Algorithm," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), 2021, pp. 2038-2041, DOI: 10.1109/ICACCS51430.2021.9442039.

[8] Ilyas H, Ali S, Ponum M, Hasan O, Mahmood MT, Iftikhar M, Malik MH. Chronic kidney disease diagnosis using decision tree algorithms. BMC Nephrol. 2021 Aug 9;22(1):273. doi: 10.1186/s12882-021-02474- z. PMID: 34372817; PMCID: PMC8351137.

[9] Subasi, A., Alickovic, E., Kevric, J. (2017). Diagnosis of Chronic Kidney Disease by Using Random Forest. In: Badnjevic, A. (eds) CMBEBIH 2017. IFMBE Proceedings, vol 62. Springer, Singapore.

[10] U. N. Dulhare and M. Ayesha, "Extraction of action rules for chronic kidney disease using Naïve Bayes classifier," 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2016, pp. 1-5, DOI: 10.1109/ICCIC.2016.7919649.

[11] Breiman, L. Random Forests. Machine Learning 45, 5–32 (2001).

[12] A. Shrestha and A. Mahmood, "Review of Deep Learning Algorithms and Architectures," in IEEE Access, vol. 7, pp. 53040-53065, 2019, doi: 10.1109/ACCESS.2019.2912200.

[13] Yang, Li & Shami, Abdallah. (2020). On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice.

[14] Jia Wu, Xiu-Yun Chen, Hao Zhang, Li-Dong Xiong, Hang Lei, Si-Hao Deng, Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimizationb, Journal of Electronic Science and Technology, Volume 17, Issue 1, 2019, Pages 26-40, ISSN 1674-862X,

[15] Alzubaidi, L., Zhang, J., Humaidi, A.J. et al. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. J Big Data 8, 53 (2021).

[17] Naveen Kishore G,V .Rajesh ,A.Vamsi Akki Reddy, K.Sumedh,T.rajesh Sai Reddy, "Prediction Of Diabetes Using Machine Learning Classification Algorithms". [18] Deepti Sisodia, Dilip Singh Sisodia," Prediction of Diabetes using classification Techniques.

[19] Archana Singh ,Rakesh Kumar, "Heart Disease Prediction Using Machine Learning Algorithms", 2020 IEEE, International Conference on Electrical and Electronics Engineering (ICE3)

[20] NisharBanu, MA; Gomathy, B.; (2013). Disease Predicting System Using Data Mining Techniques.

[21] S. Grampurohit, C. Sagarnal, Disease prediction using machine learning algorithms, 2020 Int. Conf. Emerg. Technol. (INCET) (2020) 1–7, https://doi. org/10.1109/INCET49848.2020.9154130.
[22] R.J.P. Princy, S. Parthasarathy, P.S. Hency Jose, A. Raj Lakshminarayanan, S. Jeganathan, Prediction of Cardiac Disease using Supervised Machine Learning Algorithms, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), 2020, pp. 570–575, https://doi.org/10.1109/ ICICCS48265.2020.9121169.
[23] Shrestha,Ranjit.,& Chatterjee,Jyotir. Moy. (2019).Heart Disease Prediction System Using Machine Learning . LBEF Research Journal of Science Technology and Management , 1(2), 115-132.
[24] Battineni ,Gopi. , Sagaro,Getu.Gamo. ,Chinatalapudi, Nalini. ,&Amenta,Francesco. (2020). Application Of Machine Learning Predictive Models in the Chronic Disease .International of PersonalisedMedicine , 10(21), 1- 11.
[25] Godse, Rudra A.,Gunjal,Smita S., JagtapKaran A .,Mahamuni ,Neha S., &Wankhade, Prof. Suchita. (2019). Multiple Disease Prediction Using Different Machine Learning Algorithms Comparatively.International Journal of Advance Research in Computer and Communication Engineering, 8(12), 50-52
[26] Chauhan Raj H., NaikDaksh N. ,Halpati,Rinal A., Patel,Sagarkumar J. , &PrajapatiMr. A.D. (2020). Disease Prediction using Machine Learning.International Research Journal of Engineering and Technology, 7(5) , 2000-2002.
[27] P. Prabhu, S. Selvabharathi. Deep Belief Neural Network Model for Prediction of Diabetes Mellitus. In 2019 3rd International Conference on Imaging, Signal Processing and Communication, ICISPC 2019 (pp. 138–142) Institute of Electrical and Electronics Engineers Inc. ISBN:9781728136639. 2019.

[28] Marimuthu , M. , Abinaya, M. ,Hariesh,K.S., Madhan,K.,& Pavithra, Kumar. V.(2018).A Review of Heart Disease Prediction Using Machine Learning and Data Analytics Approach .International Journal of Computer Application , 181(18), 20-25. [29] Sharma A, Nagalli S. Chronic Liver Disease. [Updated 2022 Jul 4]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2022 Jan-. Available from:

[30] R.J.P. Princy, S. Parthasarathy, P.S. Hency Jose, A. Raj Lakshminarayanan, S. Jeganathan, Prediction of Cardiac Disease using Supervised Machine Learning Algorithms, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), 2020, pp.

LIST OF PUBLICATIONS

PAPER 1 :

Conference name

International Conference on Computer Communication and Informatics(ICCCI)

Paper title

A Perlustration on the optimistic prognosis of chronic renal failure using evolutionary machine learning and deep learning techniques

Venue & Date

Sri Shakthi College of Engineering and Technology on 24 January 2023

PAPER 2 :

Conference name

Third international conference on artificial intelligence & smart computing (icaisc-2023)

Paper title

Web Application For Prediction And Classification Of Multiple Diseases Using Machine Learning And Deep Learning Techniques

Venue & Date

Bannari amman institute of technology on 10 March 2023

A Perlustration on the optimistic prognosis of chronic renal failure using Evolutionary Machine Learning and Deep learning techniques

Vishnupriya B[1],Agalya M[2], Abitha R[3],Gugapriya M[4],Harshithaa A S[5]

[1]Assistant professor, Department of CSE KPR Institute of Engineering and Technology, Coimbatore.

[2]UG-Final year, Department of CSE KPR Institute of Engineering and Technology, Coimbatore.

[3]UG-Final year, Department of CSE KPR Institute of Engineering and Technology, Coimbatore.

[4]UG-Final year, Department of CSE KPR Institute of Engineering and Technology, Coimbatore.

[5]UG-Final year, Department of CSE KPR Institute of Engineering and Technology, Coimbatore.

[1]vishnupriya.b@kpriet.ac.in ,[2]agalya12172001@gmail.com ,[3]abivaisu2221@gmail.com ,
[4]gugapriyajuly@gmail.com , [5]sujisujithaa7@gmail.com

Abstract —Millions of individuals worldwide die from chronic kidney disease each year as a result of bad lifestyle choices and inherited factors. A kind of kidney illness known as chronic kidney disease (CKD) sees a slow decline in kidney function over months or years. Initially, there is typically no symptoms. 's machine learning algorithm is a vital tool for medical practitioners to reliably identify illnesses at an early stage, and it may potentially be averted by early diagnosis and treatment of people who are at risk of getting the condition. In the framework, multiple machine learning algorithms are used to analyze renal disorders, and the best ones are chosen to determine the severity of CKD patients. For machine learning, it utilized a dataset on chronic renal illness from the UCI machine learning repository, which contained 25 functions. The DT, SVM, KNN, Logistic Regression, and NB machine learning algorithms are employed in this work. Ensemble Learning has been applied based on boosting algorithm The results were validated using four scoring techniques: Accuracy, Precision, Recall, and F1 measurement. Prior to categorization, the data were scaled to normalize the range of independent variables and preprocessed to manage missing values. The approach is therefore appropriate for the quick identification of renal illnesses that are chronic. X-ray or MRI pictures of the kidneys are used as 28x28 pixel data for deep learning. Picture filtering and image segmentation are used as part of the image pre-processing approach. A local ACED and a better Sobel on-the-edge detection technique are then used. Finally, a model is developed to forecast chronic kidney disease based on the result.

Keywords— Chronic Disorder, Machine Learning, Deep Learning, EDA, Decision Tree, SVM, Logistic Regression, Naïve Bayes, Ensemble learning, Edge detection, Improved Sobel, Local ACED.

I. INTRODUCTION

When kidneys get damaged and are unable to filter your blood as efficiently, it will develop chronic renal failure (CRF), also known as chronic kidney disease. The primary role of the kidneys is to remove extra water and waste from the blood in order to produce urine; CRF is a sign that the body has amassed wastes. This illness is chronic because the damage develops gradually over a lengthy period of time**.** There are treatments that can stop chronic kidney disease from progressing and causing complications like decreased glomerular filtration rate (GFR) and an elevated risk of cardiovascular disease, as well as improve survival and quality of life. Chronic kidney disease can be detected by routine laboratory tests. If the illness progresses to the end stage, it has two choices. Hemodialysis is one method of filtering blood; another is transplantation, which is thought to be superior to purifying methods.

The majority of techniques were there to predict kidney renal failure, which include textual analysis and using CT, and MRI images. In textual analysis, using clinical test results are organized in the CKD benchmark dataset. The data is pre-processed, and apply either machine learning or deep learning approach. CKD Dataset is not pre-processed and consists of 24 Predictive features. Even though all the features have their own importance but some features are more correlated in predicting renal kidney failure, for that feature selection is the best method. The system's overall accuracy is increased after feature selection. For classification, a variety of machine learning methods including Naive Bayes, Decision Trees, KNN, SVM, Gradient Descent, and Random Forest were used. By providing additional numerical descriptors that may be helpful in the diagnosis and prognosis of renal illness, the textural analysis added to traditional machine-learning techniques boosts the potential of medical imaging. The Deep Learning algorithm performs better with a lot of data.

II. LITERATURE SURVEY

year	ML/DL Algorithm	Feature selection methods	Dataset	Accuracy
2021	SVM, KNN, DT, and RF	Recursive feature elimination (RFE)	CKD dataset	-
2020	RF, SVM, NB, and LR	RF-FS, FS, FES, BS, and BES	CKD dataset	98.8%
2020	naïve Bayes, decision tree J48 algorithm, and random forest classifier	No	CKD dataset	97.50%
2020	An artificial neural network and multi-kernel k-means clustering	No	Image dataset	99.61
2020	ANN, C5.0, and LR CFS, Lasso, and CKD dataset LSVM, KNN, and RF	CFS, Lasso, and Wrapper method	CKD dataset	98.86%.
2020	Methods based on KNN No CKD dataset NB and DT, Bagging and Random Subspace methods	No	CKD dataset	90.1%
2019	NB and FNN No CKD dataset A hybrid model LR and RF	No	CKD dataset	99.83%

Table.1 LITERATURE SURVEY

Table.1 contains the literature survey using various algorithms. Besides textual analysis from the researcher's point of view, image analysis would be an optimistic tool for the prediction of renal kidney dysfunction. It's interesting to note that the textural analysis method of medical image quantification, which analyses picture pixels, was looked into as a useful supplementary technique. Clinical applications that strive to diagnose patients with CKD and renal function have lately looked into deep learning. This area of artificial intelligence is motivated by biological neurons, which are made up of several layers of linked nodes and provide a new, powerful tool for extracting visual features and classifying patients. Techniques for medical imaging were performed to measure kidney problems, primarily CKD. Then, Here it describes how different medical imaging modalities are used with texturing and machine learning methods, such as neural networks, to show how AI could improve the performance of predicting and diagnosing renal failure. The application of artificial intelligence (AI) in renal segmentation, which is required either before human-expert or computer-aided diagnosis, is a crucial step that can shorten the time and make renal parenchyma delineation a subject-independent challenge. Different forms of input are Radiology, MRI, UE, and CT. Radiology can yield structural and biological metrics that have been shown to aid in the prediction and follow-up of renal impairment. Using magnetic resonance imaging (MRI), it is possible to evaluate the organization, diffusion, perfusion, and oxygenation of the renal microstructure. The technique of ultrasound elastography (UE) enables the structural analysis of data pertaining to the mechanical characteristics of the tissue. Aside from exposing patients to x-rays and injecting contrast material, computed tomography (CT) is only able to offer anatomic and functional information. Scintigraphy combines functional and perfusion monitoring using radioactive tracers. The methods for feature selection that is now in use, machine learning algorithms for prediction, and deep learning to perform the same for huge datasets or for picture datasets will all be covered in this presentation.

III. METHODOLOGIES

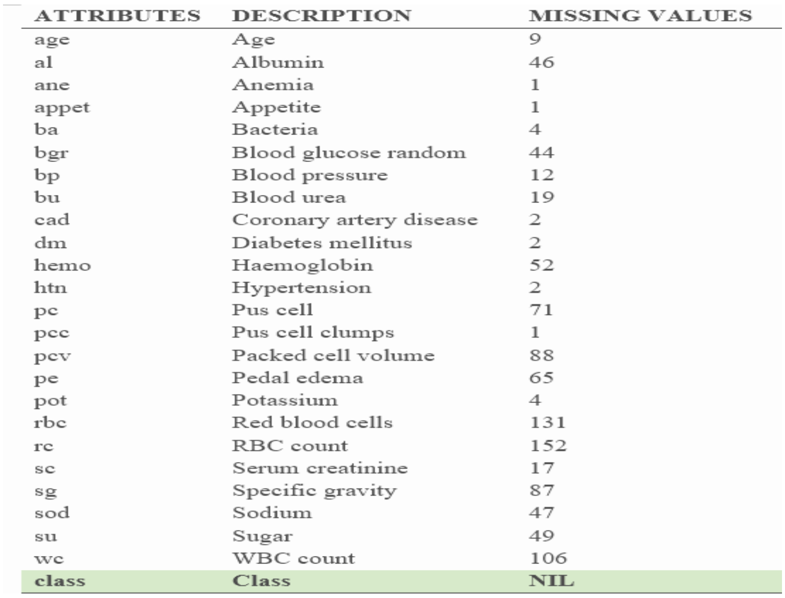
MACHINE LEARNING

A. PREDICTION USING CKD DATASET

Several machine learning techniques can predict or categorize chronic renal failure. It will not be effective to process the CKD dataset while taking into account all of its aspects. To be more optimized, the feature selection and ensemble method can be added to this process. The feature extraction or selection may be carried out using suggested evolutionary algorithms. Before prediction or classification, most correlated features are extracted and those features will be considered for the classification or prediction of renal failure using appropriate machine learning algorithms.

B. DATASET DESCRIPTION

UCI repository consists of the CKD dataset Fig.1, there are 400 instances (samples) in the CKD dataset, among which 250 were CKD patients and 150 were healthy volunteers, which were chosen during a two-month period. Each instance included 24+ class properties, 11 numerical and 14 nominals, based on measurement data from blood and urine tests or responses to a survey from all samples.



ATTRIBUTES	DESCRIPTION	MISSING VALUES
age	Age	9
al	Albumin	46
ane	Anemia	1
appet	Appetite	1
ba	Bacteria	4
bgr	Blood glucose random	44
bp	Blood pressure	12
bu	Blood urea	19
cad	Coronary artery disease	2
dm	Diabetes mellitus	2
hemo	Haemoglobin	52
htn	Hypertension	2
pc	Pus cell	71
pcc	Pus cell clumps	1
pcv	Packed cell volume	88
pe	Pedal edema	65
pot	Potassium	4
rbc	Red blood cells	131
rc	RBC count	152
sc	Serum creatinine	17
sg	Specific gravity	87
sod	Sodium	47
su	Sugar	49
wc	WBC count	106
class	Class	NIL

Fig. 1 Dataset

Fig. 2 is the HEAT MAP which describes how each attribute is correlated with the rest of the attributes.

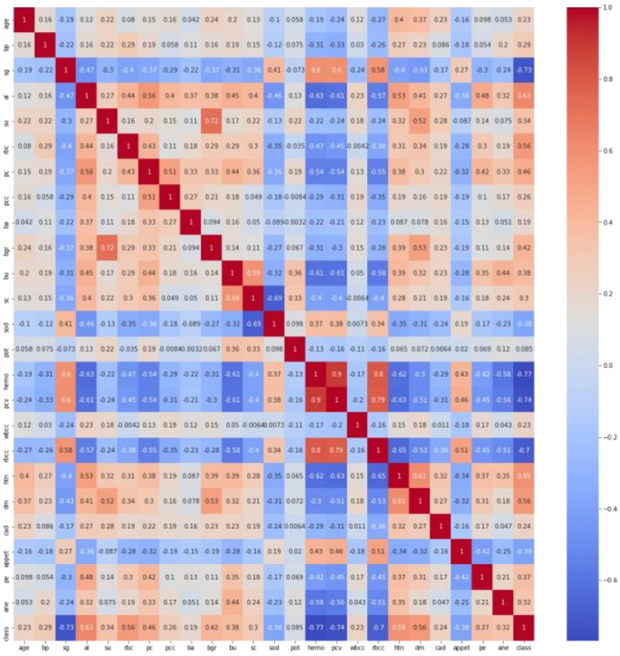


Fig. 2 HEAT MAP

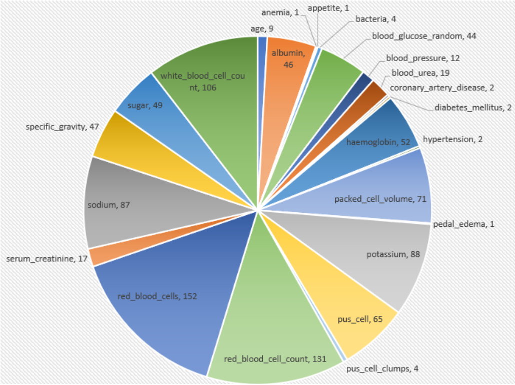


Fig. 3 Missing values in the CKD dataset

Fig. 3 represents the missing values in the CKD Dataset. To ensure good accuracy, missing values must be handled in the pre-processing stage depending on distributions. Little's MCAR test was used in this study to verify that missing data were random. The process causing the data to be missing will determine the potential bias brought on by missing data.

C. FEATURE SELECTION METHODOLOGY

Recursive Feature Elimination

It is necessary to find the significant aspects that have a strong and beneficial connection with features of relevance for illness diagnosis after computing the missing values. Building a strong diagnostic model is prevented by extracting the vector features, which removes characteristics that are unnecessary and unhelpful for prediction.

Mixed Data Feature Selection (MDFS) Model

Data pre-processing is used to handle null values of quantitative and qualitative variables in mixed data. After that, nominal and numerical data are separated from the combined data. Then, while the numerical data is treated using the Point Biserial Correlation filter, the nominal data is handled using the Chi-Square filter. Then, separately, filters compute the relationship between the dependent feature and the target. The qualities are presented in descending order from the most correlated to the least correlated based on the correlation measure.

D. MACHINE LEARNING ALGORITHM

Decision Tree

A supervised technique for solving classification-related issues is a decision tree [8]. There is a widely used pre-defined target variable included. The decision tree may be used to input and output variables that are continuous or specified. This method may be used to tackle classification and regression issues because it divides the population or sample into two or more homogenous sets called subpopulations based on the splitter that matters most for the input variable. Starting at the root, a decision tree is built using a top-down method that includes data segmentation. Below are the formulas for calculating Fig. 4 entropy and the Gini index Fig.5.

$$\text{Entropy}=-\sum_{i=1}^{n} p_i * \text{Log}_2(p_i)$$

Entropy Formula

$$\text{Gini Index} = 1-\sum_{i=1}^{n} (P_i)^2$$

Gini Index Formula

Fig. 4 Entropy formula

Fig. 5 Gini Formula

Naive Bayes

The Naive Bayes [10] approach trains a classifier using the Bayes theorem. In other words, it is a probabilistic classifier that was taught using the Naive Bayes technique. It determines a probability distribution for a given observation over various classifications.

Fig. 6 illustrates the essential relationship of the Bayes theorem:

$$P\,(C/A) = \frac{P(A/C)*P(C)}{P(A)}$$

Fig. 6 Bayes theorem

P(A) is constant across all classes, hence the only product that will maximize is P(A/C)*P(C)

Support vector machine

SVM [6], a supervised machine learning technique, may be used to solve classification and regression issues. The SVM approach seeks a hyperplane in an N-dimensional space that clearly classifies the input points. The size of the hyperplane depends on the number of features. If there are just two input characteristics, the hyperplane is merely a line. The hyperplane collapses to a two-dimensional plane when there are three input features. The following are some benefits of support vector machines: effective in circumstances of enormous scope. When there are more dimensions than samples, the method is still useful.

KNN

KNN [7] is a fundamental supervised machine learning (ML) method that may be used for missing value imputation, classification, or regression issues. It is based on the idea that the observations in a data collection that is closest to a certain data point are the observations that are the most "similar," and that may thus classify unexpected points based on the values of the closest existing points. By choosing K, the user may specify the number of nearby observations to utilize in the method.

LOGISTIC REGRESSION

The Logistic Regression model is a frequently used statistical model in statistics that is mostly used for categorization. That is, given a collection of data, the Logistic Regression method assists us in categorizing these observations into two or more distinct groups. As a result, the target variable is discrete.

ENSEMBLE LEARNING

As general diagnostic techniques for more traditional model creation, ensemble approaches can be employed. The more information the traditional statistical model is likely to be lacking, the greater the fit quality gap between one of the better ensemble approaches and a conventional statistical model. In standard statistical models, the connections between the explanatory factors and the response may be assessed using ensemble approaches. The ensemble technique allowed for a more accurate representation of the selection process and a less biased estimation of the chance of belonging to each treatment group.

BOOSTING

Boosting is a sequential procedure in which each succeeding model seeks to rectify the preceding model's mistakes. The models that follow are reliant on the prior model. Individual models may not do well on the full dataset but do well on sections of it. As a result, each model improves on the preceding model's performance, so the term is fitting.

- Ada-boost
- Gradient Boost
- xgboost

ADA-BOOST

One of the ensembles boosting classifiers is Ada-boost or Adaptive Boosting. To improve classifier accuracy, it combines many classifiers. An iterative ensemble algorithm is AdaBoost. AdaBoost classifier combines a number of ineffective classifiers to create a powerful classifier that has a high degree of accuracy. The fundamental idea underlying Adaboost is to train the data sample and adjust the classifier weights in each iteration in a way that provides accurate predictions of uncommon observations. Any machine learning method that accepts weights from the training set can be used as the basis classifier.

GRADIENT BOOST

A family of machine learning techniques known as gradient boosting classifiers combines a number of weak learning models to produce a powerful predicting model. Gradient boosting frequently makes use of decision trees. Due to success in categorizing large datasets, gradient boosting models are gaining popularity and have lately been successful in numerous Kaggle data science challenges. The Scikit-Learn machine learning toolkit for Python offers a variety of XGBoost implementations of gradient boosting classifiers.

XGBOOST

This algorithm generates decision trees in a linear manner. Weights are very significant in XGBoost. All of the independent variables are given weights, which are subsequently put into the decision tree, which predicts results. The weight of factors that the tree predicted incorrectly is raised, and these variables are subsequently put into the second decision tree. These various classifiers/predictors are then combined to form a more powerful and precise model. It can solve issues including regression, classification, ranking, and user-defined prediction.

DEEP LEARNING

Deep learning is a subset of machine learning that is mostly used to train image models and video models that rely on neural networks with three or more layers. These models are capable of learning from the neurons built specifically for images.

A. PREDICTION USING IMAGE DATASET

A renal X-ray imaging or MRI image collection is employed. In an X-ray, electrical radiation is sent into the body to record images of its internal organs for better disease diagnosis by orthopedics or radiologists. With the help of a statistical range and a Gaussian filter, In this paper an enhanced technique for identifying edges in images of humans taken with X-rays. The Gaussian filter is used to prepare and improve images. 28x28 pixels is the image size.

B. IMAGE PREPROCESSING

The term "image pre-processing" describes operations on images that are carried out at the most fundamental level of abstraction. These operations lessen rather than increase the information content of the pictures if entropy is a measure of information. Pre-processing aims to enhance image data by reducing undesirable distortions or strengthening specific visual characteristics necessary for later processing and analysis activities.

Image preprocessing techniques :

- Image filtering
- Image segmentation

C. CNN

Convolutional neural networks, or CNNs, are a type of deep learning technique that may apply weights and biases to different objects and characteristics in an input picture. Comparatively speaking, a CNN requires far less pre-processing than other classification techniques. CNN can learn these filters and attributes, as opposed to manually-engineered filters in basic systems. A CNN's construction is influenced by the visual cortex's organization, which is a mirror of the neural connections present in the human brain. Individual neurons only respond to inputs in this tiny area of the visual field known as the Receptive Field. These overlapping patterns are repeated until the entire visual field is covered.

D. FLOWCHART FOR IMAGE PROCESSING

The flow of the deep learning procedure for the prediction of chronic renal disease is shown in the flowchart below (Fig. 8).

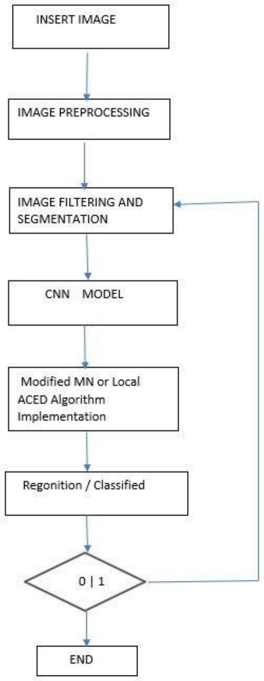

INSERT IMAGE
IMAGE PREPROCESSING
IMAGE FILTERING AND SEGMENTATION
CNN MODEL
Modified MN or Local ACED Algorithm Implementation
Regonition / Classified
0 | 1
END

Fig. 8 Flowchart of Deep learning process

E. DEEP LEARNING ALGORITHM

Edge detection

Finding breaks or sharp brightness changes in digital pictures is done using an image processing technique called edge detection. The sharpest contrasts in picture brightness occur around the image's borders (or limits).

A method for identifying the borders and edges of various objects inside a single image is edge detection.

Edge detection methods

- local ACED
- Improved Sobel

Local ACED

The local ACED [4] method only works with images of electron and transmission electron microscopy that have edges for particle detection. The Canny method suggested three rigorous edge detection requirements : (a) a high signal-to-noise ratio, (b) excellent positioning accuracy, and (c) single-edge response. The basic idea behind the Canny algorithm is to smooth the image with the Gaussian function, then obtain the gradient value and direction angle by convoluting the first-order differential operator and the smoothed image, and finally obtain the final edge by non-maximum suppression and dual-threshold detection.

Improved Sobel

A Sobel [5] technique that is more effective based on the current Sobel edge detection algorithm improved edge detection system based on the k-means clustering method. Edge detection, a fundamental method in image processing, seeks to locate the spots in an image where picture brightness varies sharply or consistently. To identify surface problems on heavy railways, a more effective and accurate Sobel algorithm has been devised. The suggested technique compensates for the Sobel algorithm's sensitivity to X and Y directions by including six templates in various orientations.

IV. EVALUATION METRICS

A.CONFUSION MATRIX

The confusion matrix was used to analyze classifier performance using the metrics listed below.

True Positive (TP): The proportion of classes that, when projected favorably, are likewise positive.

The True Negative (TN): statistic tracks the percentage of negatively predicted classes that are truly negative.

False Positive (FP): The number of classes that were incorrectly categorized while being favorably classified.

False Negative (FN): The number of incorrectly categorized negatively classed classes.

Below are Table.2 and Table. 3 shows the confusion matrix for **Logistic Regression, Naive Bayes, SVC, KNN, Decision Tree, Xgb, Ada Boost, Gradient Boost**

CLASSIFIERS	Logistic Regression		Naive Bayes		SVC		KNN	
	PREDICTED POSITIVE	PREDICTED NEGATIVE	PREDICTED POSITIVE	PREDICTED NEGATIVE	PREDICTED POSITIVE	PREDICTED NEGATIVE	PREDICTE D POSITIVE	PREDICTED NEGATIVE
ACTUAL POSITIVE	TP=48	FN=6	TP=47	FN=7	TP=41	FN=13	TP=51	FN=3
ACTUAL NEGATI VE	FP=0	TN=26	FP=0	TN=26	FP=4	TN=22	FP=4	TN=22

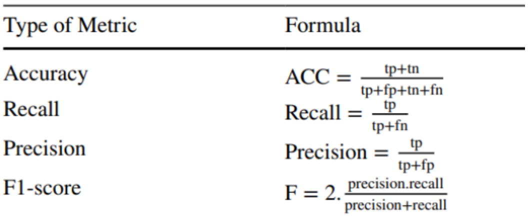
Table.2 Confusion Matrix

CLASSIFIERS	Decision Tree		Xgb		Ada Boost		Gradient Boost	
	PREDICTED POSITIVE	PREDICTED NEGATIVE	PREDICTED POSITIVE	PREDICTED NEGATIVE	PREDICTED POSITIVE	PREDICTEI NEGATIVE	PREDICTE D POSITIVE	PREDICTE D NEGATIVE
ACTUAL POSITIVE	TP=43	FN=11	TP=52	FN=2	TP=52	FN=2	TP=52	FN=2
ACTUAL NEGATI VE	FP=3	TN=23	FP=1	TN=25	FP=0	TN=2 6	FP=2	TN=24

Table. 3 Confusion Matrix

B.CLASSIFICATION REPORT

Equations 1-4 in Fig. 7 state that the models are assessed based on four standard metrics: accuracy, precision, recall, F1-score, and specificity, where TP denotes true positive, TN denotes true negative, FP denotes false positive, and FN denotes false negative.



Type of Metric	Formula
Accuracy	$\text{ACC} = \frac{tp+tn}{tp+fp+tn+fn}$
Recall	$\text{Recall} = \frac{tp}{tp+fn}$
Precision	$\text{Precision} = \frac{tp}{tp+fp}$
F1-score	$F = 2.\frac{\text{precision.recall}}{\text{precision+recall}}$

Fig. 7 Evaluation Metrics

Precision :

Precision is a classifier's ability to avoid labeling a negative occurrence as positive. It is defined for each class as the ratio of true positives to the sum of true positives and false positives.

Precision: Positive prediction accuracy.

Precision = TP/(TP + FP) as in Fig. 7

Recall :

A classifier's recall is its capacity to discover all positive occurrences. It is defined for each class as the ratio of true positives to the sum of true positives and false negatives.

Recall: Fraction of positives

Recall = TP/(TP+FN) as in Fig. 7

F1 Score :

The F1 score is a weighted mean of precision and recall, with 1.0 being the highest and 0.0 being the poorest. F1 scores are lower than accuracy measurements because of factors in precision and recall. To compare classifier models, utilize the weighted average of F1, rather than global accuracy.

F1 Score = 2*(Recall * Precision) / (Recall + Precision) as in Fig. 7

Classification Report of CKD prediction using Logistic Regression, Naive Bayes, SVC, KNN, and Decision Tree :

Fig. 9 shows the classification report of the Machine learning algorithm without ensemble learning. Logistic Regression has high accuracy based on overall performance. Logistic Regression has a weighted average of Precision and Recall of 93 as the F1 score. Support vector machine has the least accuracy, precision, recall, and F1 score.

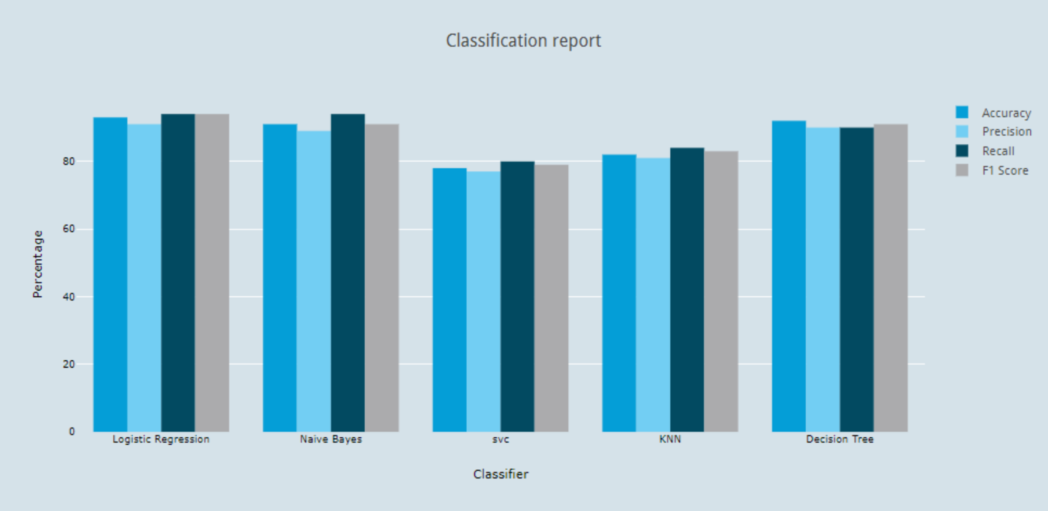


Fig. 9 classification report

Classification Report of CKD prediction using Xgb, Ada boost, and Gradient Boost :

Fig. 10 shows the classification report of the Machine learning algorithm with ensemble learning. Ada Boost has high accuracy based on overall performance. Ada Boost has a weighted average of Precision and Recall of 98 as an F1 score. Gradient Boost has the least accuracy, precision, recall, and F1 score but it is greater than the Logistic Regression. Hence Ada Boost is the best algorithm to predict chronic kidney disease based on the comparison of the overall performance of each algorithm.

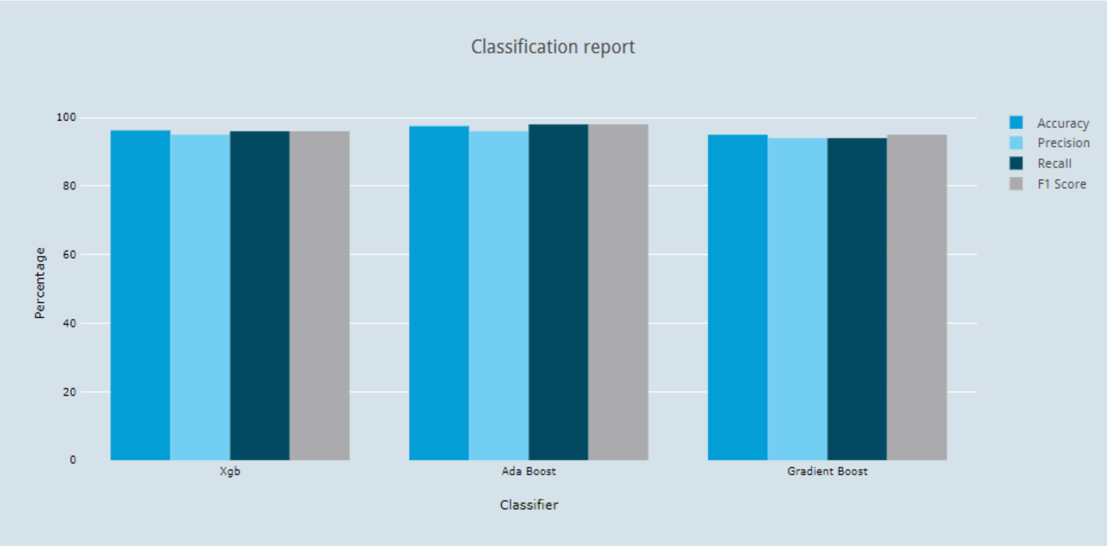


Fig. 10 Classification report for Ensemble learning

V. RESULTS

A.Machine Learning Model Classifier

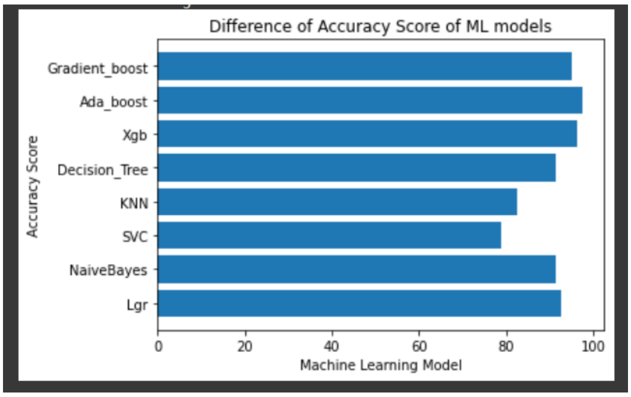


Fig. 11 Accuracy score of different model

Accuracy Score

ML Model	Accuracy score
Logistic Regression	92.5 %
Naive Bayes	91.25%
SVC	78.75%
KNN	82.5%

Decision Tree	91.25%
Xgb	96.25%
Ada Boost	97.5%
Gradient Boost	95%

Table.4 Accuracy score of different model

Fig.11 and Table.4 demonstrate the display of individual base classifiers improving ensemble learning. As shown in Table.4, Ada-boost, a quick training and robust method, outperforms other individual classifiers in the research on 25 features of the chronic renal disease dataset with a maximum accuracy of 97.5%. The ensemble approaches used in this research demonstrate the usefulness of multi-classifiers in aggregating base learners' predictions, with superior classification results than individual-based classifier performances at all levels. The established ensemble approach to chronic kidney disease diagnosis has good performance because of the ensemble method used in training the base classifiers, which combines the predictive power of the base classifiers.

As Our Ideology is to Create a system that can classify the patient's condition of CKD either by using the Dataset / Data obtained from the patients or can also able to detect the Abnormality in the X-ray images of the patients by finding the anomaly in the scan report by edge detection strategy using CNN concepts which are Successfully drafted with the Concepts and possible Algorithms can be used to obtain the better results.

VI. CONCLUSION AND FUTURE WORKS

Our ideology is practicable and, if possible, might be developed as an application. Making image detection easier by segmenting and creating images from all conceivable angles. Other than binary classification, various other features may be used to discover further classifications such as low, mild, medium, and high (chances). Paper committed such a concept for detection using both Machine Learning and Deep Learning, as well as Edge detection technology, which is almost always used in various Domains and is primarily known for its detection accuracy. Concluding here that our system can be adapted to any kind of Environment and it supports well with any Ideology to make the change.

In the future, It can add more strategies like the XGB Algorithm which is an Extreme Gradient Boosting algorithm that can make even high accuracy of predictions. It and also be developed as a Web or Android Application. The Identification and tracking system can be implemented if possible. Making React Js or Other Attractive web apps for the model can make the system more familiar among the others, both the positive and negative results should be conveyed in a good understandable manner for the user/client who uses the system.

A high level of Visualization can be implemented before developing the model to make the process understandable for the user too, since visualization is a visual technique there is no need for one to have Coding knowledge to understand it.

VII . DATASET AVAILABILITY

Source = Dr.P.Soundarapandian.M.D.,D.M
(Senior Consultant Nephrologist),
Apollo Hospitals,
Author = "L.Jerlin Rubini(Research Scholar)",
Year = "2017",
Title = "{UCI} Machine Learning Repository",
url = "https://archive.ics.uci.edu/ml/datasets/chronic_kidney_disease",
Institution = "Alagappa University,"

VIII. REFERENCE

[1]. Nagasankar, Tamilselvi & Ankaryarkanni, B.. (2016). Performance Analysis of Edge Detection Algorithms on Various Image Types. Indian Journal of Science and Technology. 9. 10.17485/ijst/2016/v9i21/95207.

[2]. Wenzhou Gao, Xiaoguang Zhang, Lei Yang and Huizhong Liu, "An improved Sobel edge detection," *2010 3rd International Conference on Computer Science and Information Technology*, 2010, pp. 67-71, DOI: 10.1109/ICCSIT.2010.5563693.

[3]. Anil K. Bharodiya, Atul M. Gonsai, An improved edge detection algorithm for X-Ray images based on the statistical range, Heliyon, Volume 5, Issue 10, 2019, e02743, ISSN 2405-8440.

[4]. Y. Menga, Z. Zhang, H. Yin, et al. Automatic detection of particle size distribution by image analysis based on local adaptive canny edge detection and modified circular Hough transform Micron, 106 (2018), pp. 34-41

[5]. K. Zhang, Y. Zhang, P. Wang, et al. An improved Sobel edge algorithm and FPGA implementation Procedia Comput. Sci., 131 (2018), pp. 243-248

[6]. Polat H, Danaei Mehr H, Cetin A. Diagnosis of Chronic Kidney Disease Based on Support Vector Machine by Feature Selection Methods. J Med Syst. 2017 Apr;41(4):55. doi: 10.1007/s10916-017-0703-x. Epub 2017 Feb 27. PMID: 28243816.

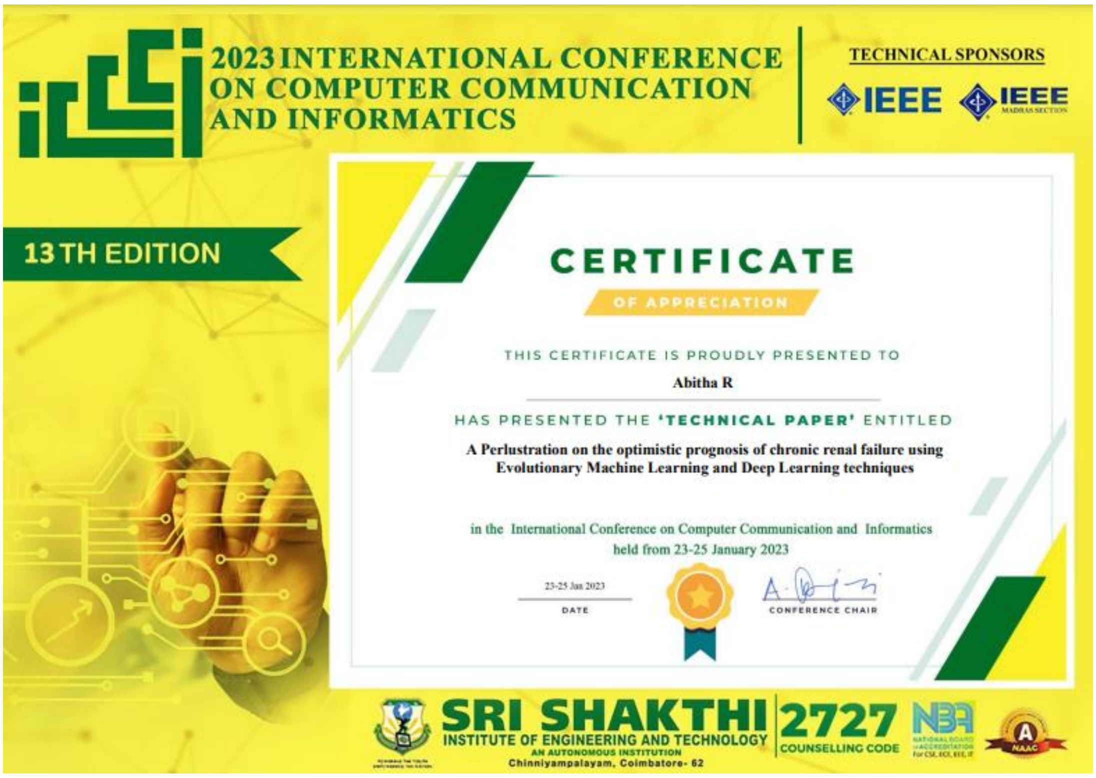
[7]. N. Saranya, M. Sakthi Samyuktha, S. Isaac and B. Subhanki, "Diagnosing Chronic Kidney Disease using KNN Algorithm," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), 2021, pp. 2038-2041, DOI: 10.1109/ICACCS51430.2021.9442039.

[8]. Ilyas H, Ali S, Ponum M, Hasan O, Mahmood MT, Iftikhar M, Malik MH. Chronic kidney disease diagnosis using decision tree algorithms. BMC Nephrol. 2021 Aug 9;22(1):273. doi: 10.1186/s12882-021-02474-z. PMID: 34372817; PMCID: PMC8351137.

[9]. Subasi, A., Alickovic, E., Kevric, J. (2017). Diagnosis of Chronic Kidney Disease by Using Random Forest. In: Badnjevic, A. (eds) CMBEBIH 2017. IFMBE Proceedings, vol 62. Springer, Singapore.

[10]. U. N. Dulhare and M. Ayesha, "Extraction of action rules for chronic kidney disease using Naïve Bayes classifier," 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2016, pp. 1-5, DOI: 10.1109/ICCIC.2016.7919649.

CERTIFICATION :



2023 INTERNATIONAL CONFERENCE ON COMPUTER COMMUNICATION AND INFORMATICS

TECHNICAL SPONSORS

IEEE IEEE MADRAS SECTION

13 TH EDITION

CERTIFICATE

OF APPRECIATION

THIS CERTIFICATE IS PROUDLY PRESENTED TO

Abitha R

HAS PRESENTED THE 'TECHNICAL PAPER' ENTITLED

A Perlustration on the optimistic prognosis of chronic renal failure using Evolutionary Machine Learning and Deep Learning techniques

in the International Conference on Computer Communication and Informatics held from 23-25 January 2023

23-25 Jan 2023

DATE

CONFERENCE CHAIR

SRI SHAKTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

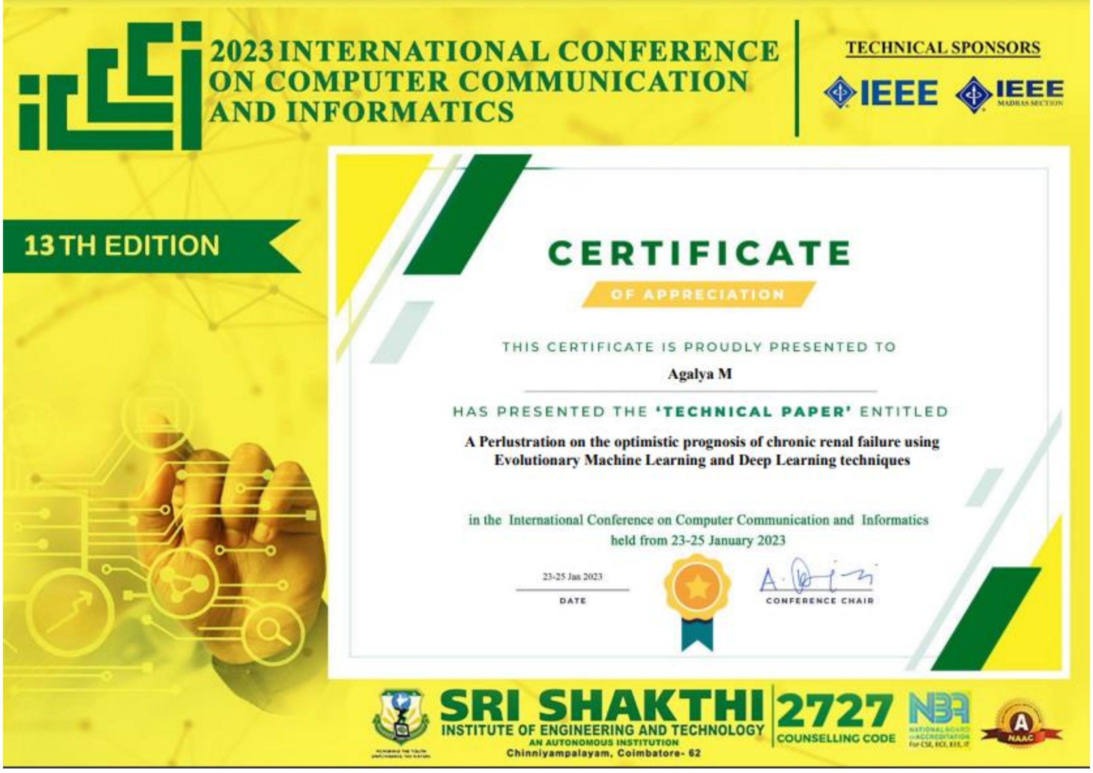
AN AUTONOMOUS INSTITUTION

Chinniyampalayam, Coimbatore- 62

2727 COUNSELLING CODE

NBA NATIONAL BOARD of ACCREDITATION For CSE, ECE, EEE, IT

A NAAC



2023 INTERNATIONAL CONFERENCE ON COMPUTER COMMUNICATION AND INFORMATICS

TECHNICAL SPONSORS

IEEE IEEE MADRAS SECTION

13 TH EDITION

CERTIFICATE

OF APPRECIATION

THIS CERTIFICATE IS PROUDLY PRESENTED TO

Agalya M

HAS PRESENTED THE 'TECHNICAL PAPER' ENTITLED

A Perlustration on the optimistic prognosis of chronic renal failure using Evolutionary Machine Learning and Deep Learning techniques

in the International Conference on Computer Communication and Informatics held from 23-25 January 2023

23-25 Jan 2023

DATE

CONFERENCE CHAIR

SRI SHAKTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

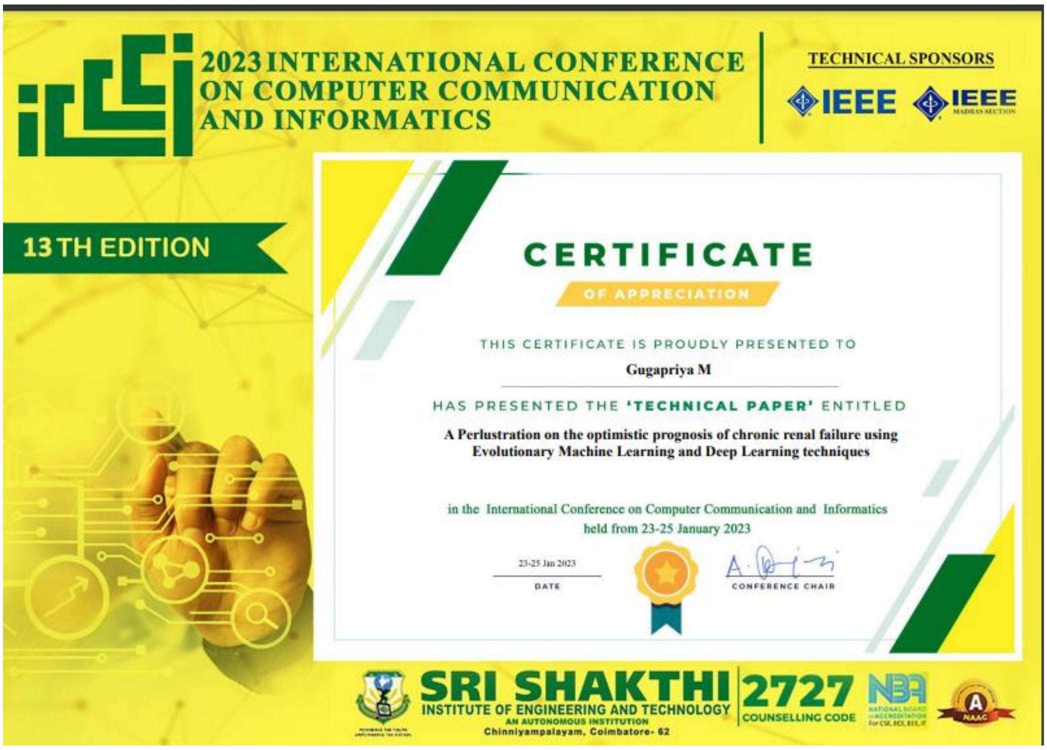
AN AUTONOMOUS INSTITUTION

Chinniyampalayam, Coimbatore- 62

2727 COUNSELLING CODE

NBA NATIONAL BOARD of ACCREDITATION For CSE, ECE, EEE, IT

A NAAC



2023 INTERNATIONAL CONFERENCE ON COMPUTER COMMUNICATION AND INFORMATICS

TECHNICAL SPONSORS

IEEE IEEE MADRAS SECTION

13 TH EDITION

CERTIFICATE

OF APPRECIATION

THIS CERTIFICATE IS PROUDLY PRESENTED TO

Gugapriya M

HAS PRESENTED THE 'TECHNICAL PAPER' ENTITLED

A Perlustration on the optimistic prognosis of chronic renal failure using Evolutionary Machine Learning and Deep Learning techniques

in the International Conference on Computer Communication and Informatics held from 23-25 January 2023

23-25 Jan 2023

DATE

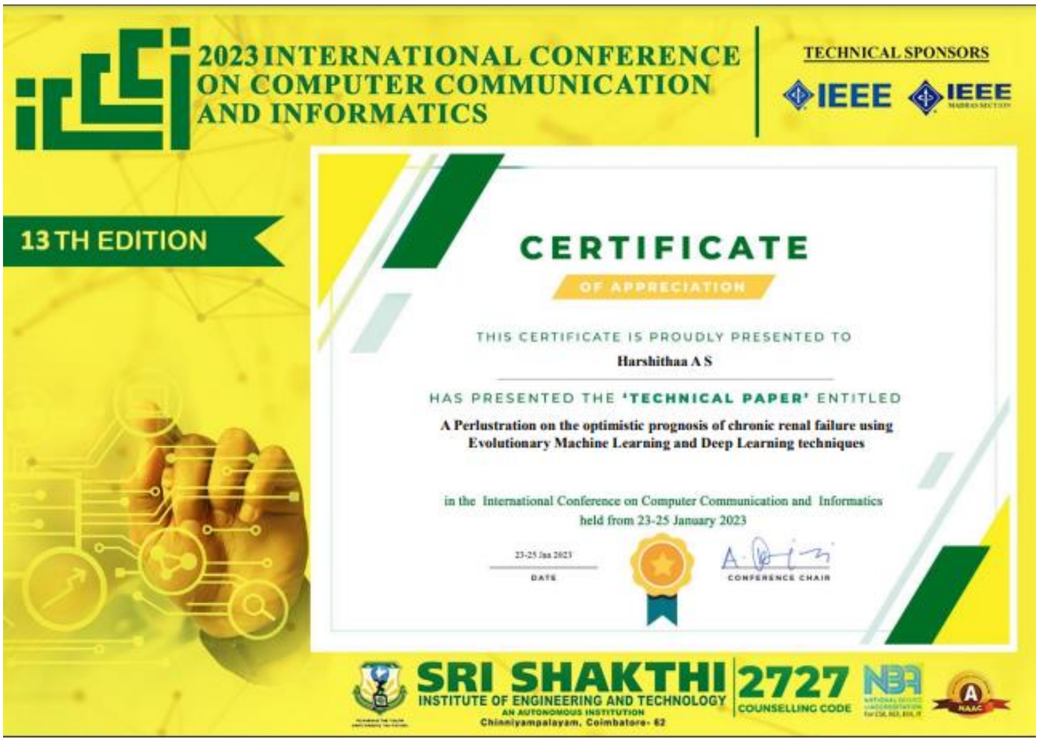
CONFERENCE CHAIR

SRI SHAKTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

AN AUTONOMOUS INSTITUTION

Chinniyampalayam, Coimbatore- 62

2727 COUNSELLING CODE



2023 INTERNATIONAL CONFERENCE ON COMPUTER COMMUNICATION AND INFORMATICS

TECHNICAL SPONSORS

IEEE IEEE MADRAS SECTION

13 TH EDITION

CERTIFICATE

OF APPRECIATION

THIS CERTIFICATE IS PROUDLY PRESENTED TO

Harshithaa A S

HAS PRESENTED THE 'TECHNICAL PAPER' ENTITLED

A Perlustration on the optimistic prognosis of chronic renal failure using Evolutionary Machine Learning and Deep Learning techniques

in the International Conference on Computer Communication and Informatics held from 23-25 January 2023

23-25 Jan 2023

DATE

CONFERENCE CHAIR

SRI SHAKTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

AN AUTONOMOUS INSTITUTION

Chinniyampalayam, Coimbatore- 62

2727 COUNSELLING CODE

Web Application For Prediction And Classification Of Multiple Diseases Using Machine Learning And Deep Learning Techniques

Vishnupriya B
Assistant professor, Department of CSE
KPR Institute of Engineering and Technology
Coimbatore, India
vishnupriya.b@kpriet.ac.in

Agalya M
UG-Final year, Department of CSE
KPR Institute of Engineering and Technology
Coimbatore, India
agalya12172001@gmail.com

Abitha R
UG-Final year, Department of CSE
KPR Institute of Engineering and Technology
Coimbatore, India
abivaisu2221@gmail.com

Gugapriya M
UG-Final year, Department of CSE
KPR Institute of Engineering and Technology
Coimbatore, India
gugapriyajuly@gmail.com
sujisujithaa7@gmail.com

Harshithaa A S
UG-Final year, Department of CSE
KPR Institute of Engineering and Technology
Coimbatore, India

Abstract — **Our proposal presents a disease prediction system using machine learning to predict diabetes , breast cancer , heart diseases , kidney diseases and deep learning to predict malaria and pneumonia . Users must go visit the hospital for a checkup for minor issues, which takes additional time. It's also quite stressful to answer phone calls for appointments. Such an issue can be fixed by employing illness prediction software and providing appropriate advice on leading a healthy lifestyle. Dueto a range of diseases and a low doctor-to-patient ratio, the usage of particular disease prediction techniques and health-related concerns has expanded over the past ten years. Therefore, with this system, we are primarily focused on providing users with rapid and accurate illness prediction based on the symptoms they enter, together with the expected severity of the condition. The most appropriate algorithm and medical advice will be provided in this project. Different machine learning algorithms like random forest, decision tree, naïve bayes, KNN, support vendor machine, logical regression and deep learning algorithms like CNN are employed to forecast illnesses to provide speedy and precise predictions. The symptoms entered will be compared to the database in one channel. Additionally, if the symptom is new, which is its primary function, it will be retained in the database, and the other channel will offer a forecast level of illness severity. A web/android application is set up for users to allow for simple portability, configuration, and remote access to areas that doctors find difficult to access. Since most users are unaware of all available treatments for a given ailment, this project also aimsto offer pharmaceutical and prescription counseling for any anticipated diseases. As a result, this approach makes health management simpler. This web application was developed using Flask Web Framework .The models used to predict the diseases were trained on large Datasets. The web application can predict diabetes, breast cancer, heart disease,kidney disease, liver disease, malaria and pneumonia.**

Keywords— Diabetes, Breast Cancer, Heart Disease, Kidney Disease, Malaria, Pneumonia, Machine Learning, Deep Learning, Random Forest, Decision Tree, Support Vector Machine (SVM), Logistic Regression, Naive Bayes, Convolutional Neural Network (CNN), K-Nearest neighbour (KNN)

I. INTRODUCTION

Heart disease is on the rise due to our busy, stressful lives nowadays. Heart disease affects people of all ages, therefore it's important to identify it early through symptoms or reports. When your kidneys become damaged and are unable to filter your blood as effectively, chronic kidney disease develops. When a disease reaches its final stage, it has two options. A technique for filtering blood is hemo dialysing; another is transplanting, which is regarded as being superior than purifying techniques. Diabetes mellitus is brought on by abnormally high blood sugar levels. It interferes with the body's capacity to generate the hormone insulin, improper glucose metabolism and elevated blood sugar levels result. The primary goal of such a system is to assist doctors in cross- verifying their diagnostic findings, which offers a viable remedy to the current death rates. In order to provide a novel platform and the most promising method for the early diagnosis of various diseases, we have put out this study.When medical data is insufficient or of low quality, it affects. Prediction of a disease based on the patient's symptoms and medical history For decades, machine learning has been a stumbling barrier. In the medical industry, Machine Learning technology provides a powerful venue for rapidly resolving healthcare challenges.

Severe liver disease [29] is characterised by the persistent degradation of liver functioning. Liver activities include the generation of clotting factors and other proteins, detoxification of toxic metabolic products, and bile excretion. This is a continual process of liver parenchymal inflammation, destruction, and regeneration that leads to fibrosis and cirrhosis. Cirrhosis is the last stage of chronic liver disease, characterised by disruption of hepatic architecture, the creation of extensive nodules, vascular rearrangement, neo-angiogenesis, and extracellular matrix deposition. The recruitment of stellate cells and fibroblasts that induce fibrosis is the basic process of fibrosis and cirrhosis at the cellular level, whereas parenchymal regeneration is dependent on hepatic stem cells.

Python pickling is used to store a model's behaviour. The significance of this system analysis is that in analysing illnesses, all of the characteristics that produce the condition are included, allowing for more efficient and accurate disease detection. The behaviour of the completed model will be preserved as a Python pickle file.

A lot of research on existing health-care systems focused on only one ailment at a time. One approach, for example, is used to analyse diabetes, another to analyse diabetic retinopathy, and still another to predict heart disease. Most systems concentrate on a single ailment. When a company wishes to analyse their patients' health records, they must use a variety of models. The current system's methodology is only suitable for analysing certain disorders. A user can analyse more than one disease on a same webpage using the multiple diseases prediction method. The user is not need to travel to multiple locations in order to anticipate whether or not he or she has a certain condition. The user need a various disease prediction system.The goal is to develop a good Machine Learning algorithm that is efficient and accurate for illness prediction. In this study, the supervised Machine Learning concept is employed for forecasting the illnesses.

The key feature will be Machine Learning, in which we will use algorithms such as Decision Tree, Random Forest, Nave Bayes, and KNN to aid in accurate illness prediction and better patient care [25].

II. LITERATURE SURVEY

This section outlines the investigation of previously proposed models for illness prediction that are linked to our proposed work. Many research have been conducted in order to diagnose various illnesses. They used several data mining approaches to accurately forecast a range of illnesses.

[17] Prediction Of Diabetes Using Machine Learning Classification Approaches was proposed by G Naveen Kishore and a few other writers. Several classification techniques, such as SVM, Logistic Regression, Decision Tree, KNN, and Random Forest, are used on the 769 instances of the Pima dataset, which comprise characteristics such as Pregnancies, Blood Pressure, Body Mass Index, and so on. The classification method Random Forest achieved the maximum accuracy of 74.4%, while the KNN achieved the lowest accuracy in this experiment.

[18] Dilip Singh Sisodia and Deepti Sisodia predicted diabetes using classification algorithms and reported an accuracy of roughly 76% on the Pima dataset.

The major goal of the study is, because the heart is crucial in living creatures. As a result, the diagnosis and prognosis of heart illness should be exact and correct since it is highly important and might cause death instances connected to heart. As a result, machine learning and artificial intelligence can help forecast any type of natural event. Hence, in this study, they analyse the accuracy of machine learning for predicting heart disease using k-nearest neighbour, decision tree, linear regression, and SVM, using the UCI repositor dataset used for training and testing. They also compared the algorithms' accuracy: SVM 83%, Decision tree 79%, Linear regression 78%, and k-nearest neighbour 87% [19].

The authors of [21] provided an automated method for addressing challenging questions about heart disease prediction. This intelligent system was built using the Naive Bayes approach to produce faster, better, and more accurate results. It might help doctors make clinical decisions regarding heart attacks. This system might be improved by adding SMS capabilities, developing Android and iOS mobile apps, and integrating a pacemaker in the order.

Diabetes and breast cancer were identified by adding the adaptivity feature into support vector machines [22]. The objective was to provide a quick, automated, and flexible diagnostic approach utilising adaptive SVM. The bias value in traditional SVM was modified to improve results. The proposed classifier generated output in the form of 'if-then' rules. The suggested technique was applied to diagnose diabetes and breast cancer, and it produced 100% correct classification rates for both illnesses. Future research should concentrate on developing more efficient methods for altering the bias value in traditional SVM.

[23] "Heart Disease Prediction System Using Machine Learning" has constructed a heart disease prediction system utilising NB algorithm that delivers 88.163% accuracy among others.

“Application of Machine Learning Predictive Models in the Chronic Disease " [24] focused on SVM and LR algorithms andevaluate the study models associated with diagnosis of chronic disease. These models are highly applicable in classification and diagnosis of CD.

III. PROPOSED SYSTEM

User-friendly Web application for prediction of disease using machine learning algorithm and deep learning .

1) Backend

a) Machine Learning

i. Collecting a datasets of different disease
ii. Data preprocessing
iii. Training and Testing of datasets
iv. Creation of model for different algorithms of KNN, SVM, Navie bayes,Decision Tree , Logistic Regression, Random forest .
v. Analysis the high accuracy algorithm

b) Deep learning

i. Using a image datasets
ii. Image preprocessing
iii. Creating a model using Convolution Neural Network

c) Model creation

i. In machine learning , model is that has high accuracy is created as model for backend using pickle

ii. In deep learning , model is saved by ‘save_weights’ in keras library.

iii. Deploying the created model using flask library in python to connect to front end.

2) Frontend

***a)* Home page**

i. Consists of navigation to all other pages and contact information .

***b)* Prediction pages for different disaese**

i. Consists of 7 pages of prediction such that Kidney, Liver, Heart , Breast cancer, Diabetes , Malaria and pneumonia .

ii. Getting the input from user in form.

iii. After submitting the form , it will connect to the created model and based on input it predicts the disease.

IV. METHODOLOGIES

1) MACHINE LEARNING

a) DATASET

Datasets are collected from Kaggle, and datasets are analasied and preprocessed for prediction

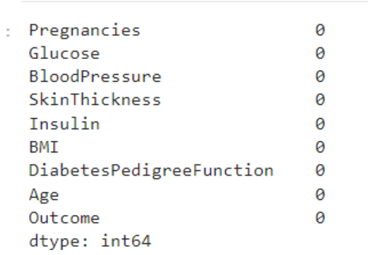
i. Diabetes datasets

Name of Dataset : Pima Indians Diabetes Database.

Predict the onset of diabetes based on diagnostic measures Fig. 1

Data - 768

Features- 9



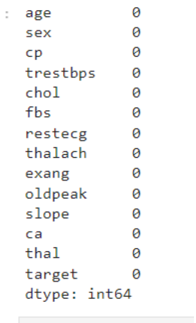
```
Pregnancies                 0
Glucose                     0
BloodPressure               0
SkinThickness               0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64
```

Figure 1. Features for diabetes prediction

ii. Heart Datasets

Data – 304 Fig. 2

Features – 14



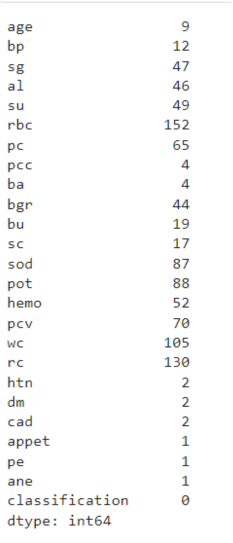
```
age         0
sex         0
cp          0
trestbps    0
chol        0
fbs         0
restecg     0
thalach     0
exang       0
oldpeak     0
slope       0
ca          0
thal        0
target      0
dtype: int64
```

Figure 2. Features for heart disease prediction

iii. Kidney Datasets

Data – 400 Fig. 3

Features - 26



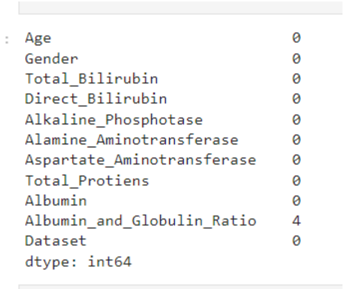
```
age                 9
bp                 12
sg                 47
al                 46
su                 49
rbc               152
pc                 65
pcc                 4
ba                  4
bgr                44
bu                 19
sc                 17
sod                87
pot                88
hemo               52
pcv                70
wc                105
rc                130
htn                 2
dm                  2
cad                 2
appet               1
pe                  1
ane                 1
classification      0
dtype: int64
```

Figure 3. Features for Kidney disease prediction

iv. Liver Datasets

Data – 583 Fig. 4

Feature – 11



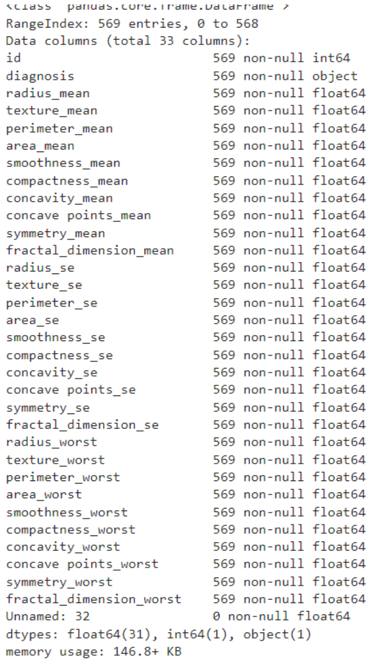
```
Age                           0
Gender                        0
Total_Bilirubin               0
Direct_Bilirubin              0
Alkaline_Phosphotase          0
Alamine_Aminotransferase      0
Aspartate_Aminotransferase    0
Total_Protiens                0
Albumin                       0
Albumin_and_Globulin_Ratio    4
Dataset                       0
dtype: int64
```

Figure 4 . Features for Liver disease prediction

v. Breast cancer Datasets

Data – 569 Fig. 5

Features 33



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 569 entries, 0 to 568
Data columns (total 33 columns):
id                         569 non-null int64
diagnosis                  569 non-null object
radius_mean                569 non-null float64
texture_mean               569 non-null float64
perimeter_mean             569 non-null float64
area_mean                  569 non-null float64
smoothness_mean            569 non-null float64
compactness_mean           569 non-null float64
concavity_mean             569 non-null float64
concave points_mean        569 non-null float64
symmetry_mean              569 non-null float64
fractal_dimension_mean     569 non-null float64
radius_se                  569 non-null float64
texture_se                 569 non-null float64
perimeter_se               569 non-null float64
area_se                    569 non-null float64
smoothness_se              569 non-null float64
compactness_se             569 non-null float64
concavity_se               569 non-null float64
concave points_se          569 non-null float64
symmetry_se                569 non-null float64
fractal_dimension_se       569 non-null float64
radius_worst               569 non-null float64
texture_worst              569 non-null float64
perimeter_worst            569 non-null float64
area_worst                 569 non-null float64
smoothness_worst           569 non-null float64
compactness_worst          569 non-null float64
concavity_worst            569 non-null float64
concave points_worst       569 non-null float64
symmetry_worst             569 non-null float64
fractal_dimension_worst    569 non-null float64
Unnamed: 32                0 non-null float64
dtypes: float64(31), int64(1), object(1)
memory usage: 146.8+ KB
```

Figure 5. Features for Breast Cancer prediction

b) MACHINE LEARNING ALGORITHM

i. Decision Tree

A supervised technique for solving classification-related issues is a decision tree [8]. There is a widely used pre-defined target variable included. The decision tree may be used to input and output variables that are continuous or specified. This method may be used to tackle classification and regression issues because Based on the splitter that matters the most for the input variable, it divides the population or sample into two or more homogeneous groups called subpopulations. Starting at the root, a decision tree is built using a top-down method that includes data segmentation. Below are the formulas for calculating Fig. 6 entropy and the Gini index Fig. 7.

$$Entropy = -\sum_j p_j \log_2 p_j$$

Figure. 6 Entropy formula

$$Gini = 1 - \sum_j p_j^2$$

Figure. 7 Gini Formula

ii. Naive Bayes

The Naive Bayes [10] method uses the Bayes theorem to train a classifier. In other words, it is a probabilistic classifier that was taught using the Naive Bayes approach. It computes a probability distribution for a given observation over many classes.

Fig. 8 illustrates the essential relationship of the Bayes theorem:

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Figure. 8 Bayes theorem

P(B) is constant across all classes, hence the only product that will maximize is P(B/A)*P(A).

iii. Support vector machine

SVM [6], a supervised machine learning algorithm can help with classification and regression problems. In an N-dimensional space, the SVM technique finds a hyperplane that classifies the input points. The size of the hyperplane is determined by the number of features. The hyperplane is just a line if there are only two input features. The hyperplane collapses to a two-dimensional plane when there are three input characteristics. The following are some benefits of support vector machines: effective in circumstances of enormous scope. When there are more dimensions than samples, the method is still useful.

iv. KNN

KNN [7] is a basic supervised machine learning (ML) approach that may be used to solve missing value imputation, classification, or regression problems. It is based on the idea that the observations in a data collection that is closest to a certain data point are the observations that are the most "similar," and that may thus classify unexpected points based on the values of the closest existing points. By choosing K, the user may specify the number of nearby observations to utilize in the method.

v. Logistic Regression

The Logistic Regression model is a popular statistical model in statistics that is used mostly for categorization. That is, given a collection of data, the Logistic Regression method assists us in categorizing these observations into two or more distinct groups. As a result, the target variable is discrete.

vi. Random forest

A popular classification and regression method that uses supervised learning is called Random Forest [11]. One of the most crucial algorithms in machine learning is this one. It is based on the idea of ensemble learning, which is the practise of integrating many classifiers to solve a challenging issue and enhance the model's performance.

Random Forest [11] is a classifier that uses a number of decision trees on different subsets of a given dataset and averages them to enhance the predicted accuracy of that dataset.

c) HYPERPARAMETER TUNING

RandomizedSearchCV [13] has "fit" and "score" methods. If the estimator supports them, it also implements "score samples", "predict", "predict proba", "decision function", "transform", and "inverse transform". Cross-validated search acrossparameter settings is used to optimise the estimator parametersrequired to apply these approaches. In contrast to GridSearchCV, a defined number of parameter settings are sampled from the given distributions rather than all parameter values being tried out. n iter specifies the number of parametersettings that are attempted. Sampling without replacement is conducted if all parameters are supplied as a list. Sampling with replacement is utilised if at least one parameter is presented as a distribution. Continuous distributions are strongly advised for continuous parameters.

2) DEEP LEARNING

Deep learning[12] is a subset of machine learning that is mostly used to train image models and video models that rely on neural networks with three or more layers. These models are capable of learning from the neurons built specifically for images.

a) Image dataset

i. Malaria

The dataset contains 2 folders

- Infected Fig. 9
- Uninfected Fig. 10
- And a total of 27,558 images.

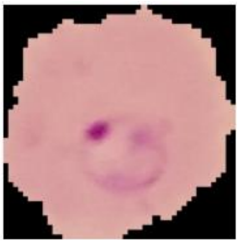


Figure 9. Infected Cell

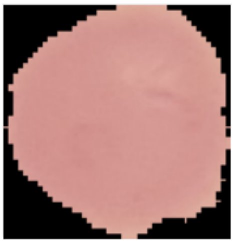


Figure 10. Uninfected

ii. Pneumonia

The dataset is divided into three folders: train, test, and val, with subfolders for each picture category (Pneumonia/Normal). There are 5,863 JPG X-Ray pictures and two categories (Pneumonia/Normal)Fig. 11 and Fig. 12. The typical chest X-ray (left panel) shows clean lungs with no spots of aberrant opacification. Bacterial pneumonia (centre) is characterised by a focused lobar consolidation, in this example in the right upper lobe (white arrows), whereas viral pneumonia (right) is characterised by a more widespread "interstitial" pattern in both lungs.

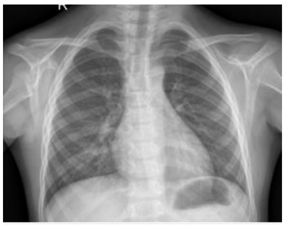


Figure 11. Normal Lung

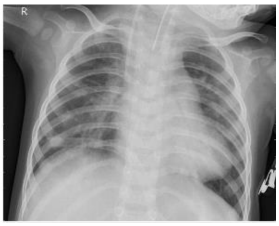


Figure 12. Infected Lung

b) Image Pre-processing

The term "image pre-processing" describes operations on images that are carried out at the most fundamental level of abstraction. These operations lessen rather than increase the information content of the pictures if entropy is a measure of information. Pre-processing aims to enhance image data by reducing undesirable distortions or strengthening specific visual characteristics necessary for later processing and analysis activities.

Image pre-processing techniques :

- Image filtering
- Image segmentation

c) Flowchart for Deep Learning

The flow of the deep learning procedure for the prediction of malaria and pneumonia disease is shown in the flowchart below Fig. 13.

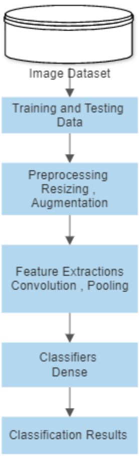


Image Dataset
Training and Testing Data
Preprocessing Resizing , Augmentation
Feature Extractions Convolution , Pooling
Classifiers Dense
Classification Results

Figure. 13 Flowchart of Deep learning process

d) CNN

Convolutional neural networks (CNN) [15] , are a type of deep learning technique that may apply weights and biases to different objects and characteristics in an input picture. Comparatively speaking, a CNN requires far less pre-processing than other classification techniques. CNN can learn these filters and attributes, as opposed to manually-engineered filters in basic systems. A CNN's construction Fig. 14 is influenced by the visual cortex's organization, which is a mirror of the neural connections present in the human brain. Individual neurons only respond to inputs in this tiny area of the visual field known as the Receptive Field. These overlapping patterns are repeated until the entire visual field is covered.

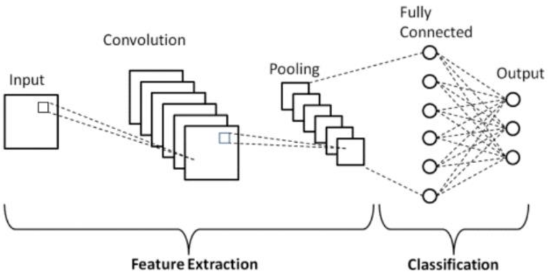


Figure. 14 CNN Layers [16]

A convolution neural network has multiple hidden layers Fig. 14 that help in extracting information from an image. The four important layers in CNN are:

1. Convolution layer
2. ReLU layer
3. Pooling layer
4. Fully connected layer Convolution Layer

This is the first step in the process of extracting valuable features from an image. A convolution layer has several filters that perform the convolution operation. Every image is considered as a matrix of pixel values

ReLU layer ReLU stands for the rectified linear unit. Once the feature maps are extracted, the next step is tomove them to a ReLU layer. ReLU performs an elementwise operation and sets all the negative pixels to 0. It introduces non-linearity to the network, and the generated output is a rectified feature map.

Pooling Layer Pooling is a down-sampling operation that reduces the dimensionality of the feature map. The rectified feature map now goes through a pooling layer to generate a pooled feature map .

3) EVALUATION METRICS

a) Confusion matrix

The confusion matrix was used to analyze classifier performance using the metrics listed below.

True Positive (TP): The proportion of classes that, when projected favorably, are likewise positive.

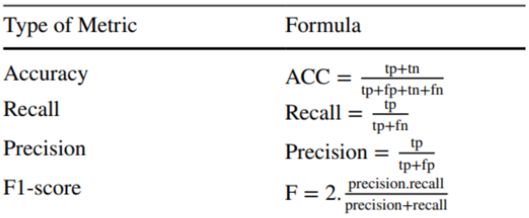
The True Negative (TN): statistic tracks the percentage of negatively predicted classes that are truly negative.

False Positive (FP): The number of classes that were incorrectly categorized while being favorably classified.

False Negative (FN): The amount of negatively classified courses that were mistakenly classified

b) Classification report

Equations 1-4 in Fig. 15 state that the models are assessed based on four standard metrics: accuracy, precision, recall, F1-score, and specificity, where TP denotes true positive, TN denotes true negative, FP denotes false positive, and FN denotes false negative.



Type of Metric	Formula
Accuracy	$\text{ACC} = \frac{\text{tp+tn}}{\text{tp+fp+tn+fn}}$
Recall	$\text{Recall} = \frac{\text{tp}}{\text{tp+fn}}$
Precision	$\text{Precision} = \frac{\text{tp}}{\text{tp+fp}}$
F1-score	$\text{F} = 2.\frac{\text{precision.recall}}{\text{precision+recall}}$

Figure 15. Evaluation Metrics

Precision

Precision is a classifier's ability to avoid labeling a negative occurrence as positive. For each class, it is defined as the ratio of true positives to the average of true positives and false positives.

Precision is a Positive prediction accuracy.

Precision = True Positive /(True Positive + False Positive) as in Fig. 15

Recall

A classifier's recall is its capacity to discover all positive occurrences. For each class, it is defined as the ratio of true positives to the average of true positives and false negatives.

The recall is a proportion of positives

Recall = True Positive /(True Positive +False Negative) as in Fig. 15

F1 Score

The F1 score is determined as a calculated average of accuracy and recall. with 1.0 being the highest and 0.0 being the poorest. F1 scores are lower than accuracy measurements because of factors in precision and recall. Use the average value of F1 rather than global accuracy to compare classifier models.

F1 Score = 2*(Recall * Precision) / (Recall + Precision) as in Fig. 15

c) Accuracy Comparison

For each disease different machine leanring algorithm is applied to analysis the best algorithm

i. Kidney Disease

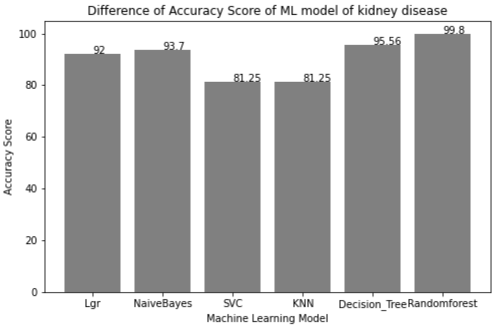


Figure 16. Represents the accurcy of different algorithm on Kidney datasets

Fig. 16 Shows that Random forest has higer accuracy of 99.8 percentage .

Hence the pickel model is created for Random forest on kindey disease .

ii. Heart Disease

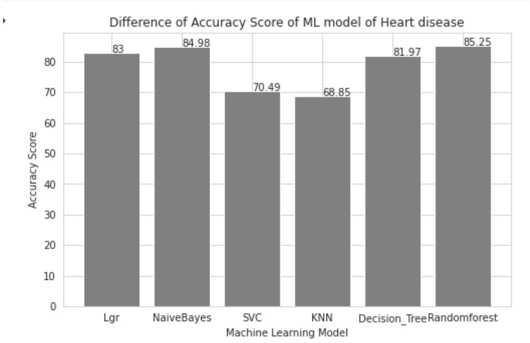


Figure 17. Represents the accury of different algorithm on Heart datasets

Fig. 17 Shows that Random forest has higher accuracy of 85.25 percentage .

Hence the pickel model is created for Random forest on Heart disease .

iii. Liver Disease

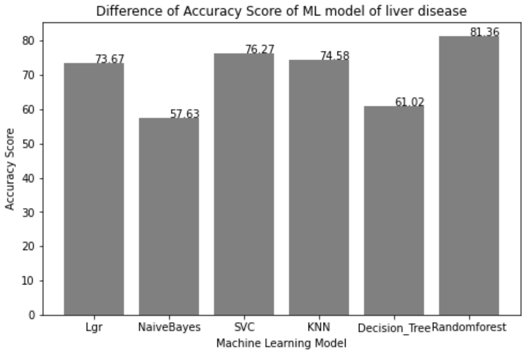


Figure 18. Represents the accury of different algorithm on Liver datasets

Fig. 18 Shows that Random forest has higher accuracy of 81.36 percentage .

Hence the pickel model is created for Random forest on Liver disease .

iv. Breast Cancer

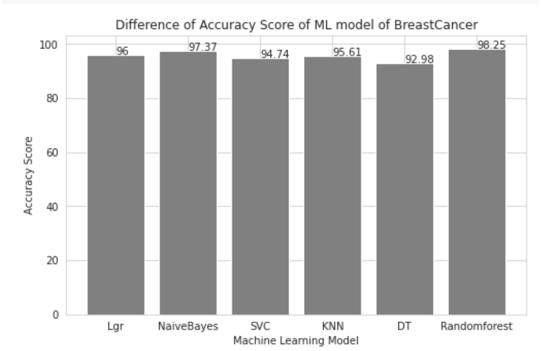


Figure 19. Represents the accury of different algorithm on Breast datasets

Fig. 19 Shows that Random forest has higer accuracy of 98.25 percentage .

Hence the pickel model is created for Random forest on Breast Cancer .

v. Diabetes Disease

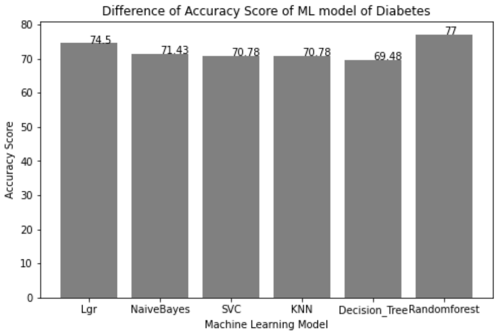


Figure 20. Represents the accury of different algorithm on Diabetes datasets

Fig. 20 Shows that Random forest has higer accuracy of 77 percentage .

Hence the pickel model is created for Random forest on Diabetes disease .

V) RESULTS

a) Machine Learning Model Classifier

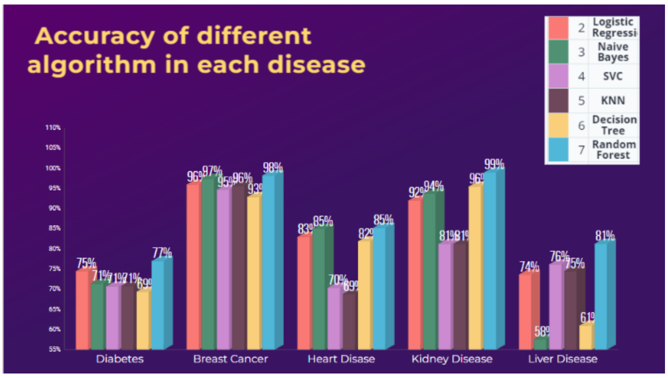


Figure 21. Accuracy score of different algorthims on all disease

Fig. 21 Represents the accuracy score for different algorthms on each disase datasets on machine leaening model .

Based on the analysis of Fig. 21 , Random forest has high accuracy for all disease , Hence the Random forest is created a model for disease prediction .

Accuracy Score

Disease	**Accuracy score**
Diabetes	77 %

Breast Cancer	98.25%
Heart Disease	85.75%
Kidney Disease	99.8%
Liver Disease	81.36%

Table 1. Accuracy score of different disease of high accuracy

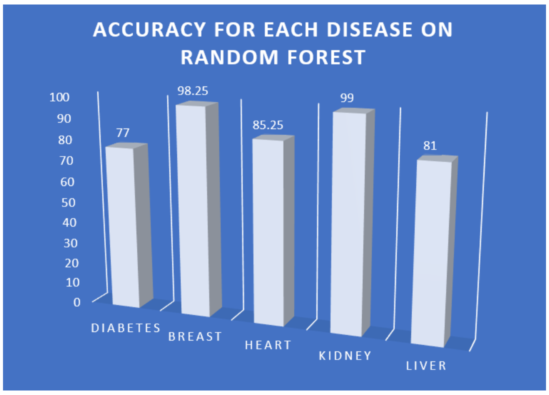


Figure 22. Represents the accuracy for all disease on Random forest algorithm

Fig. 22 and Table. 1 demonstrate the display of individual base classifiers for each disease .

b) Deep Learning Model

Deep learning model is created with CNN layers and tested with validation data which has high of Fig. 23 and Table. 2

DL Model	Accuracy score
Malaria	96%
Pneumonia	95%

Table 2. Accuracy of CNN model

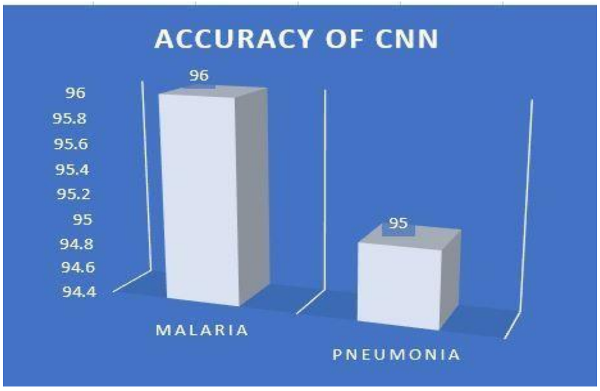


Figure 23. Represents the accuracy of malaria and pneumonia in CNN

Fig. 23 and Table. 2 demonstrate the accuracy of image datets under deep leanring model .

This model is saved by save_weight from keras and used as backend for web application.

c) Front End web Design

This web application is useful for doctors to predict disease based on the fetures .
Web application consists of 10 Pages.

1. Home Page
2. Disease Page
3. Diabetes Prediction Page
4. Heart Disase Prediction Page
5. Liver Disase prediction page
6. Breast cancer Prediction page
7. Kidney Disease Prediction page
8. Malaria Prediction Page
9. Pneumonia Prediction page
10. Analysis Page

Here the some pages from web applications

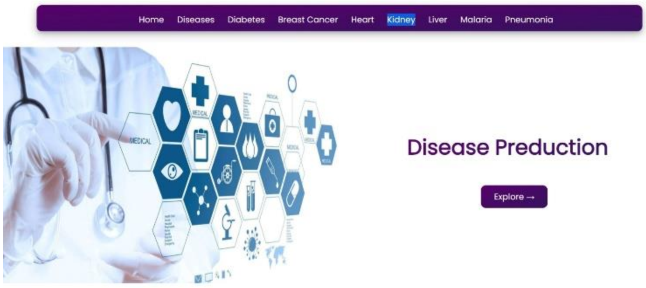


Figure 24. Home page of web application

The page Fig. 24 consists of navigation to all other pages and also contact information for email and phone numbers are included .

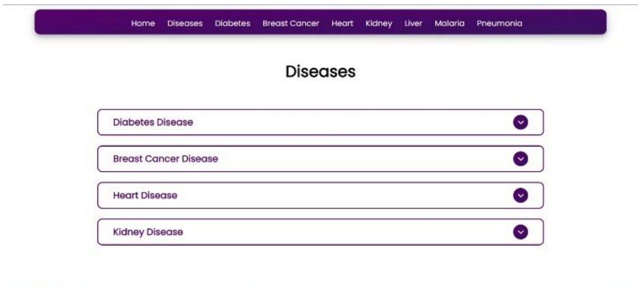
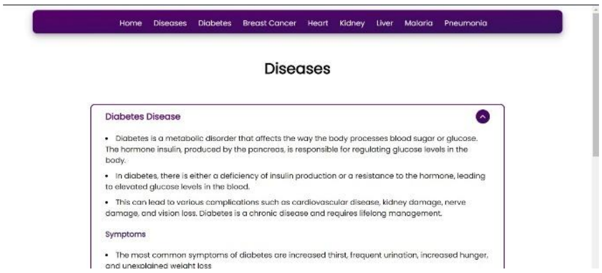


Figure 25. Shows the symptoms ,details, and awareness about each diseases



Diseases

Diabetes Disease

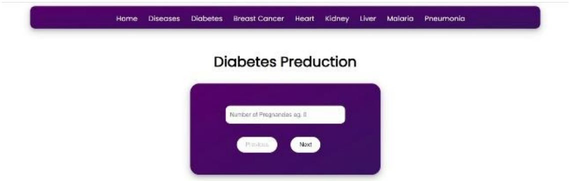
- Diabetes is a metabolic disorder that affects the way the body processes blood sugar or glucose. The hormone insulin, produced by the pancreas, is responsible for regulating glucose levels in the body.
- In diabetes, there is either a deficiency of insulin production or a resistance to the hormone, leading to elevated glucose levels in the blood.
- This can lead to various complications such as cardiovascular disease, kidney damage, nerve damage, and vision loss. Diabetes is a chronic disease and requires lifelong management.

Symptoms

- The most common symptoms of diabetes are increased thirst, frequent urination, increased hunger, and unexplained weight loss

Figure 26. i.e for Diabetes disease symtoms and information

Fig. 25 and Fig. 26 represents the Disease Page , which consists of all details and symptoms of each disease .
Same as Fig. 25 and Fig. 26, Symptoms and details are included for all other disease .

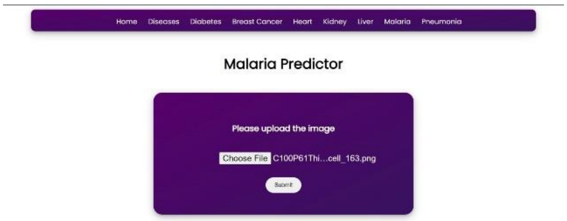


Diabetes Preduction

Number of Pregnancies eg. 0

Figure 27. i.e for Diabetes Prediction Page

Fig. 27 is the example of front end form for all disease like Liver , Breast cancer, Kidney disease ,Heart Disease , Diabetes Disesase .
Fig. 27 Form has to be filled based on the example input and submit the form .

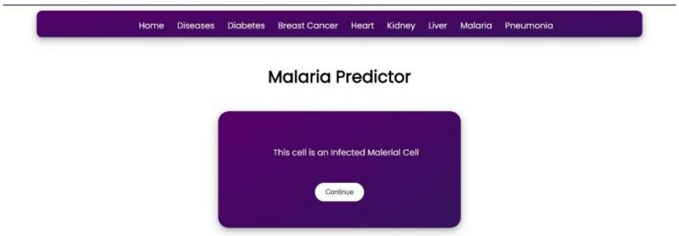


Malaria Predictor

Please upload the image

C100P61Thi...cell_163.png

Figure 28. Shows the form for uploading the image for malaria prediction



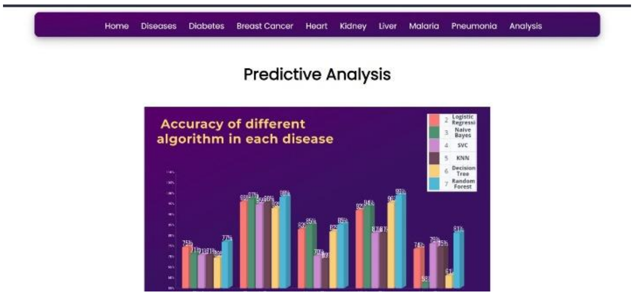
Malaria Predictor

This cell is an Infected Malerial Cell

Figure 29. Shows the Output after predictions

Fig. 28 and Fig. 29 represents the design form for the malaria prediction page where the input is uploaded as image.

Fig. 28 and Fig. 29 has design same as for Pnuemonia prediction .



Predictive Analysis

Figure 30. shows the predictive analysis

Fig. 30 Represents the predictive analysis of model created in backend process

The web application is developed with HTML , CSS , Javascript for frontend and Flask for Backend .

V. CONCLUSION AND FUTURE WORKS

The primary goal of this study was to develop a system that could accurately forecast several diseases. The doctors doesn't have to navigate via several websites thanks to this project, which also saves time. The project is set up so that the gadget uses the user's symptoms as input and produces illness prognosis as the output.

Early diagnosis of diseases can both lengthen your life and spare you from financial hardship. To attain the highest level of accuracy, we have employed a variety of machine learning techniques, including Random Forest.

Our ideology is practicable and, if possible, might be developed as an application. Making image detection easier by segmenting and creating images from all conceivable angles. Other than binary classification, various other features may be used to discover further classifications such as low, mild, medium, and high (chances). Paper committed such a concept for detection using both Machine Learning and Deep Learning, as well as Edge detection technology, which is almost always used in various Domains and is primarily known for its detection accuracy. Concluding here that our system can be adapted to any kind of Environment and it supports well with any Ideology to make the change.
In the future, it can add more strategies like the XGB Algorithm which is an Extreme Gradient Boosting algorithm that can make even high accuracy of predictions. It and also be developed as a Web or Android Application. The Identification and tracking system can be implemented if possible. Making React JavaScript or Other Attractive web apps for the model can make the system more familiar among the others, both the positive and negative results should be conveyed in a good understandable manner for the user/client who uses the system.

A high level of Visualization can be implemented before developing the model to make the process understandable for the user too, since visualization is a visual technique there is no need for one to have Coding knowledge to understand it.

The incorporation of several innovative ideas into CNN's architectural design has impacted the trajectory of research, notably in the fields of image processing and CV. The successful use of CNN to grid-like topological data illustrates its utility as an image representational model.

A promising field of study, CNN architectural design is expected to become one of the most widely utilised AI techniques in the future.

Ensemble learning is one of the possible areas of research in CNNs. The combination of multiple and diverse architectures can benefit the model by boosting generality and robustness on many categories of pictures by extracting several layers of semantic representations.

The same is true for concepts such as batch normalisation, dropout, and new activation functions, all of which should be highlighted.

VI. DATA AVAILABLITY

a) Heart Disease

Heart Disease Dataset | Kaggle

b) Kidney Disase

Chronic KIdney Disease dataset | Kaggle

c) Liver Disease

Indian Liver Patient Records | Kaggle

d) Breast Cancer

Breast Cancer Wisconsin (Diagnostic) Data Set | Kaggle

e) Diabetes

Pima Indians Diabetes Database | Kaggle

f) Malaria

Malaria Cell Images Dataset | Kaggle

g) Pneumonia

Chest X-Ray Images (Pneumonia) | Kaggle

REFERENCES

[1] Nagasankar, Tamilselvi & Ankaryarkanni, B.. (2016). Performance Analysis of Edge Detection Algorithms on Various Image Types. Indian Journal of Science and Technology. 9. 10.17485/ijst/2016/v9i21/95207.

[2] Wenzhou Gao, Xiaoguang Zhang, Lei Yang and Huizhong Liu, "An improved Sobel edge detection," *2010 3rd International Conference on Computer Science and Information Technology*, 2010, pp. 67-71, DOI: 10.1109/ICCSIT.2010.5563693.

[3] Anil K. Bharodiya, Atul M. Gonsai, An improved edge detection algorithm for X-Ray images based on the statistical range, Heliyon, Volume 5, Issue 10, 2019, e02743, ISSN 2405-8440.

[4] Y. Menga, Z. Zhang, H. Yin, et al. Automatic detection of particle size distribution by image analysis based on local adaptive canny edge detection and modified circular Hough transform Micron, 106 (2018), pp. 34-41

[5] K. Zhang, Y. Zhang, P. Wang, et al. An improved Sobel edge algorithm and FPGA implementation Procedia Comput. Sci., 131 (2018), pp. 243-248

[6] Polat H, Danaei Mehr H, Cetin A. Diagnosis of Chronic Kidney Disease Based on Support Vector Machine by Feature Selection Methods. J Med Syst. 2017 Apr;41(4):55. doi: 10.1007/s10916-017-0703-x. Epub 2017 Feb 27. PMID: 28243816.

[7] N. Saranya, M. Sakthi Samyuktha, S. Isaac and B. Subhanki, "Diagnosing Chronic Kidney Disease using KNN Algorithm," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), 2021, pp. 2038-2041, DOI: 10.1109/ICACCS51430.2021.9442039.

[8] Ilyas H, Ali S, Ponum M, Hasan O, Mahmood MT, Iftikhar M, Malik MH. Chronic kidney disease diagnosis using decision tree algorithms. BMC Nephrol. 2021 Aug 9;22(1):273. doi: 10.1186/s12882-021-02474-z. PMID: 34372817; PMCID: PMC8351137.

[9] Subasi, A., Alickovic, E., Kevric, J. (2017). Diagnosis of Chronic Kidney Disease by Using Random Forest. In: Badnjevic, A. (eds) CMBEBIH 2017. IFMBE Proceedings, vol 62. Springer, Singapore.

[10] U. N. Dulhare and M. Ayesha, "Extraction of action rules for chronic kidney disease using Naïve Bayes classifier," 2016 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2016, pp. 1-5, DOI: 10.1109/ICCIC.2016.7919649.

[11] Breiman, L. Random Forests. *Machine Learning* **45**, 5–32 (2001). https://doi.org/10.1023/A:1010933404324

[12] A. Shrestha and A. Mahmood, "Review of Deep Learning Algorithms and Architectures," in IEEE Access, vol. 7, pp. 53040-53065, 2019, doi: 10.1109/ACCESS.2019.2912200.

[13] Yang, Li & Shami, Abdallah. (2020). On Hyperparameter Optimization of Machine Learning Algorithms: Theory and Practice.

[14] Jia Wu, Xiu-Yun Chen, Hao Zhang, Li-Dong Xiong, Hang Lei, Si-Hao Deng, Hyperparameter Optimization for Machine Learning Models Based on Bayesian Optimizationb, Journal of Electronic Science and Technology, Volume 17, Issue 1, 2019, Pages 26-40, ISSN 1674-862X, https://doi.org/10.11989/JEST.1674-862X.80904120.

[15] Alzubaidi, L., Zhang, J., Humaidi, A.J. *et al.* Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. *J Big Data* **8**, 53 (2021). https://doi.org/10.1186/s40537-021-00444-8

[16] https://www.upgrad.com/blog/basic-cnn-architecture/#

[17] Naveen Kishore G,V .Rajesh ,A.Vamsi Akki Reddy, K.Sumedh,T.rajesh Sai Reddy, ”Prediction Of Diabetes Using Machine Learning Classification Algorithms”.

[18] Deepti Sisodia, Dilip Singh Sisodia,” Prediction of Diabetes using classification Techniques.

[19] Archana Singh ,Rakesh Kumar, “Heart Disease Prediction Using Machine Learning Algorithms”, 2020 IEEE, International Conference on Electrical and Electronics Engineering (ICE3)

[20] NisharBanu, MA; Gomathy, B.; (2013). Disease Predicting System Using Data Mining Techniques.

[21] S. Grampurohit, C. Sagarnal, Disease prediction using machine learning algorithms, 2020 Int. Conf. Emerg. Technol. (INCET) (2020) 1–7, https://doi. org/10.1109/INCET49848.2020.9154130.

[22] R.J.P. Princy, S. Parthasarathy, P.S. Hency Jose, A. Raj Lakshminarayanan, S. Jeganathan, Prediction of Cardiac Disease using Supervised Machine Learning Algorithms, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), 2020, pp. 570–575, https://doi.org/10.1109/ ICICCS48265.2020.9121169.

[23] Shrestha,Ranjit.,& Chatterjee,Jyotir. Moy. (2019).Heart Disease Prediction System Using Machine Learning . LBEF Research Journal of Science Technology and Management , 1(2), 115-132.

[24] Battineni ,Gopi. , Sagaro,Getu.Gamo. ,Chinatalapudi, Nalini. ,&Amenta,Francesco. (2020). Application Of Machine Learning Predictive Models in the Chronic Disease .International of PersonalisedMedicine , 10(21), 1- 11.

[25] Godse, Rudra A.,Gunjal,Smita S., JagtapKaran A .,Mahamuni ,Neha S., &Wankhade, Prof. Suchita. (2019). Multiple Disease Prediction Using

Different Machine Learning Algorithms Comparatively.International Journal of Advance Research in Computer and Communication Engineering, 8(12), 50-52

[26] Chauhan Raj H., NaikDaksh N. ,Halpati,Rinal A., Patel,Sagarkumar J. , &PrajapatiMr. A.D. (2020). Disease Prediction using Machine Learning.International Research Journal of Engineering and Technology, 7(5) , 2000-2002.

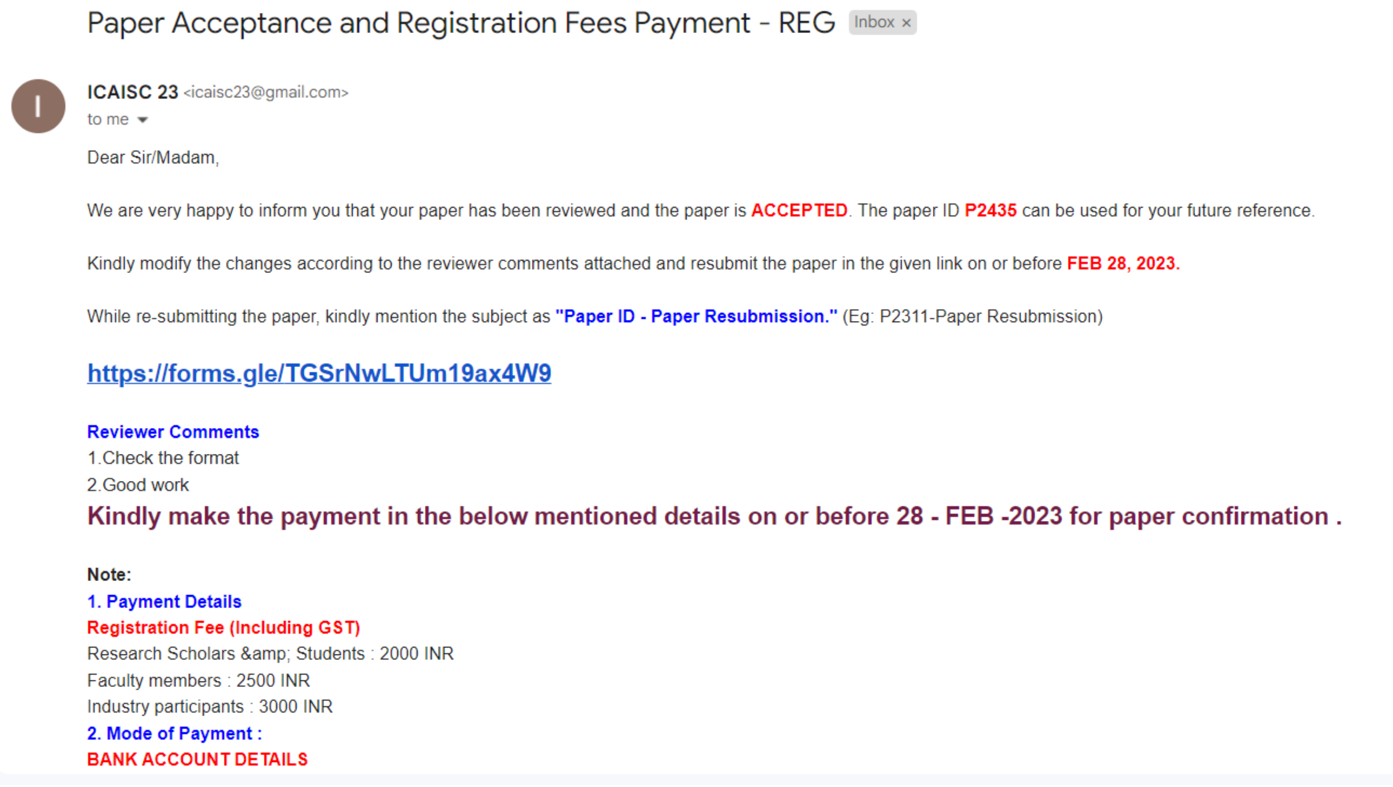
[27] P. Prabhu, S. Selvabharathi. Deep Belief Neural Network Model for Prediction of Diabetes Mellitus. In 2019 3rd International Conference on Imaging, Signal Processing and Communication, ICISPC 2019 (pp. 138–142) Institute of Electrical and Electronics Engineers Inc. ISBN:9781728136639. 2019.

[28] Marimuthu , M. , Abinaya, M. ,Hariesh,K.S., Madhan,K.,& Pavithra, Kumar. V.(2018).A Review of Heart Disease Prediction Using Machine Learning and Data Analytics Approach .International Journal of Computer Application , 181(18), 20-25.

[29] Sharma A, Nagalli S. Chronic Liver Disease. [Updated 2022 Jul 4]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2022 Jan-. Available from: https://www.ncbi.nlm.nih.gov/books/NBK554597/

[30] R.J.P. Princy, S. Parthasarathy, P.S. Hency Jose, A. Raj Lakshminarayanan, S. Jeganathan, Prediction of Cardiac Disease using Supervised Machine Learning Algorithms, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), 2020, pp. 570–575, https://doi.org/10.1109/ ICICCS48265.2020.9121169.

PAPER ACCEPTED :



Paper Acceptance and Registration Fees Payment - REG

ICAISC 23 <icaisc23@gmail.com>
to me

Dear Sir/Madam,

We are very happy to inform you that your paper has been reviewed and the paper is **ACCEPTED**. The paper ID **P2435** can be used for your future reference.

Kindly modify the changes according to the reviewer comments attached and resubmit the paper in the given link on or before **FEB 28, 2023.**

While re-submitting the paper, kindly mention the subject as **"Paper ID - Paper Resubmission."** (Eg: P2311-Paper Resubmission)

https://forms.gle/TGSrNwLTUm19ax4W9

Reviewer Comments
1.Check the format
2.Good work

Kindly make the payment in the below mentioned details on or before 28 - FEB -2023 for paper confirmation .

Note:
1. Payment Details
Registration Fee (Including GST)
Research Scholars &amp; Students : 2000 INR
Faculty members : 2500 INR
Industry participants : 3000 INR
2. Mode of Payment :
BANK ACCOUNT DETAILS