DecodingHealthMysteries

Dr.HarishKumarBT[1],RajashekarD[2],SaaeeshSatish P[3],PrakashRajS[4]and Priyanshu[5]
[1]AssociateProfessor,Dept.ofCS&E,BangaloreInstituteofTechnology,Bangalore.
[2-5]Under-graduateinCS&E,BangaloreInstituteofTechnology,Bangalore.

Abstract – ***The healthcare industry faces challenges in achieving accurate and timely diagnoses, often relying on subjective assessments and conventional methodsthat may be prone to errors. This paper presents an innovative approach to addressing these challenges by developing an AI/ML and NLP-based system to improve diagnostic accuracy and early disease detection. Leveraging deep learning models such as Convolutional Neural Networks (CNN) for analyzing lung X-rays and machine learning models like Random Forest for predicting heart disease, the system processes both structured and unstructured Patient data like laboratory results,clinicalnotes,andelectronichealthrecords.The project incorporates a user-friendly interface, developed using React for the front-end and FastAPI for the back-end, hosted in a Docker container on platforms like Render and Vercel. The key objective is to provide healthcare professionals with reliable and interpretable predictions for multiple diseases, enabling timely intervention and personalized treatment plans. This paper outlines the system's design, model development, evaluation metrics, and its possible uses in clinical decision-making support and early disease detection, ultimately aiming to enhance patient outcomes and reduce the strain on healthcare systems.***

Keywords – ***Artificial General Intelligence (AI), Deep Learning (DL), Disease Prediction, Early Disease Detection, Healthcare Technology, Convolutional Neural Networks (CNN), Random Forest, Clinical Decision Support.***

I. INTRODUCTION

The healthcare industry is progressively adopting advanced technologies to tackle the rising challenges of precise diagnostics and effective patient care. Conventional diagnostic approaches often depend on subjective assessments and tend to be time-consuming. whichmayresultindelayeddiagnoses,errors,andoverall inefficiencies in healthcare delivery. With the rapid evolving in the field of Generative Artificial Intelligence (GenAI), Machine Learning (ML), and Natural Language Processing (NLP), there is an unprecedented opportunity to revolutionize the way diseases are detected and managed. By utilizing these technologies, healthcare professionals can achieve quicker and more accurate diagnoses, resulting in better patient outcomes and easing thepressureonhealthcaresystemsglobally.Thispaper concentrates on the creation of an AI/ML-based system aimed at
to predict multiple diseases, including lung disease and heart disease, by analyzing both structured and unstructured patient data. The system employs CNN to process and analyse medical diagnostics such as lungX-rays, Random Forest models are employed to predict heart disease using textual data, like clinical notes and medicalhistories.ByincorporatingNLPtechniques,the system can retrieve important details from unstructured text, boosting the models' predictive accuracy.
For its good access for healthcare professionals, a good interface is developed using React for the frontend and FastAPI for the backend. The system is hosted in Docker containers, and deployed on cloud platformslike Render (for the backend) and Vercel (for the frontend),enablingseamlessintegrationandscalability.
The primary goal of our project is to improve early disease detection by providing accurate predictions based on diverse datasets. In doing so, it aims to assist healthcare providers in making more informed decisions, developing personalized treatment plans, and ultimately enhancing patient care. This paper assures to explore the methodologies employed in developing the system, This involves data preprocessing, model training, and evaluation. It will also cover the implications of using AI-driven models in clinical settings, the challenges faced during implementation, and future updates for this technology in healthcare.

II. LITERATURESURVEY

In June 2024, Kallepalli Reshma, Pasumarthi Niharika, Javvadi Haneesha, Kodithala Rajavardhan, and Sana Swaroop published a research paper on Multi Disease Prediction System using Machine Learning and Streamlit user-interface. This predicts heart, diabetes and Parkinson's diseases. The data is sourced from various Electronic Health Records and public health databases. Several data preprocessing steps were performed, and models such as SVM and Logistic Regression were used for training. The final accuracy achieved was approximately 89%. A frontend interface was developed using the Streamlit API, completing the project.
Another research paper was only for our understanding of the problem statement. This was proposed by Nevon Projects. This is like a basic foundation for the project. Itdescribeshowtowellunderstand thebackgroundand necessityoftheproblemstatement,inwhatdifferent

ways can we create a frontend interface for the models trained, which all diseases datasets can be taken andwhich types of models can be trained, how AI can get adapted to this problem statement, necessity of databases while understanding the solution, undergoing the regular life cycle of the project, how to overcome limitations of this project while testing in multiple phases, and so on.

In June 2023, Parshant and Dr Anu Rathee conducted a literature survey on multiple disease prediction using Machine Learning. They utilised the uses of SVM model for the results of three diseases: Cardiovascular diseases, diabetes, and Parkinson's diseases. While loading the datasets from the CSV file and conducting data preprocessing using libraries like Pandas, SVM model along with KNN model was trained. It provided 98.8% accuracy and model evaluation metrics were calculated. Also, the integration of pickle model into user interface was done and accuracies were tabulated in a table.

In the same year, Banoth Ramesh, G. Srinivas, P. Ram Praneeth Reddy, MD Huraib Rasool, Divya Rawat and Madhulita Sundaray conducted a research survey on Feasible Prediction of Multiple Diseases using Machine Learning. This proposed for the results of diabetes, heart, breast cancer, kidney and liver diseases. This enabled the collection of various datasets and their preprocessing, followed by randomly splitting the data into two sets. It then trains Decision trees, Random Forest Classifier, Naïve bayes, and KNN algorithms. The accuracy wasupto 92.5%. An ROC-AUC curve is plotted and thedetails were given in a bar chart. A frontend interface named "Medibuddy" was created and along with the disease details is given when provided with inputs from the user.

In2023, Harshit Gupta, Lakshay Dahiya and Jyoti Kaushik developed a model for a multiple disease prediction system using ML to address the challenges in predicting multiple diseases such as diabetes, heart diseases, and Parkinson's diseases requiring a comprehensive approach integrating various aspects of dataanalysisapproach.TheyutilizedLogisticRegression, Random Forest Classifier and SVM models for disease classification and prediction, This was made to analysethe different aspects on how Machine Learning models works for different datasets as provided.

In May 2023, Haindavi Kothapeta. SowmyaLakkampelly, Arun Mandari, Mr.M.Sathyanarayana,and Dr.G.Vani implemented Multiple disease detection using ML and Streamlit as a research paper. Their study uses SVM for training of the model with 80% and 20% train- test splits. They conducted study on diabetes, heart and Parkinson's diseases. The model gave 88% accuracy on training data and 87% accuracy on testing data. Streamlit was used for creating frontend interface for interaction with the user in order to take input for the model.

In May-June 2021, Indukuri Mohit, K Santhosh Kumar, Avula Uday Kumar Reddy, Badhagouni Suresh Kumar proposed a Machine Learning Model to detect multiple diseases. They worked three diseases: Breast Cancer, Diabetes and Heart disease. They did used the datasets from public repositories of Kaggle. They used Logistic Regression as the model. They also performed testing on KNN and SVM algorithms. The accuracies were upto 83.84%forheartbyKNN,77.60%fordiabetesby logistic regression and 94.55% for breast cancer by logistic regression. It was only for graphical representation.

In May 2021, Palle Pramod Reddy, Dirisinala Madhu Babu, Hardeep Kumar, and Dr Shivi Kumar introduced a disease detection model using machine learning in which they datasets from different health-related websites. After performing data preprocessing and standardizing the data, this has used Random Forest Classifier for the disease prediction. There were five kinds of diseases in the prediction: Diabetes, Breast Cancer, Heart, Kidney and Liver diseases. The Accuracy was upto 98.25% in their model. They also created a frontend framework using Flask as their API.

III. TOOLSANDTECHNOLOGIES

ProgrammingLanguage

- **Python:** is a highly flexible programming language widely used in AIML projects due to itsuser-friendlynature,extensivelibraries,and strong supportfor a widerange of datascience applications.

Libraries

- **TensorFlow:** A powerful DL framework known for building and training neural networks. It is particularly effective for implementing CNN models which is time-consuming to build from scratch.
- **Keras:** A high-level neural network API running on top of TensorFlow, simplifying the creation and training of complex models.
- **Scikit-Learn:** A library used for traditional machine learning, including tasks like feature selection, data preprocessing, and model evaluation.
- **Numpy:** Provides support for large, multi-dimensional arrays and matrices, as well as mathematical functions, manipulation, and rearranging the data, converting the pixels into multi-dimensional arrays, etc.
- **OpenCV:** Used for image processing, it helps with tasks such as feature extraction and pre-processing the medical images used in the detection model.

DevelopmentEnvironment

- **Jupyter Notebook:** An interactive environment where Machine Leaning or Deep Learning code can be written and can develop, visualise the outputs if correct or not.
- **Visual Studio Code:** An Editor which can be useful for creating user frontend interface. It has a vast marketplace of extensions that can enhance the coding experience.

IV. METHODOLOGY

ActivityDiagram

The activity diagram illustrates a disease prediction system that enables users to interact with the system in various ways. via chat or proceed with disease analysis based on symptoms. The process begins with the **user**, who can take one of two distinct paths: engaging in a generalchatorprovidingsymptomsfordisease

diagnosis. The chat interaction is simply a practical approach for disease prediction which doesn't contribute directly to the disease prediction, indicating it's uses such as general enquiries or guidance.

In the **disease analysis pathway**, the user provides symptoms, which is simplified into three primary groups: **lung-relateddiseases,heart-relateddiseases,andother diseases**. The system further processes lung-related symptoms using a **Lung Model** and heart-related symptoms using a **Heart Model**. However, the diagram does not specify how symptoms categorized as "Others" are handled, suggesting a potential gap in the system's design. This missing detail could indicate the need for additional models or a general classification approach for diseases beyond lung and heart conditions.

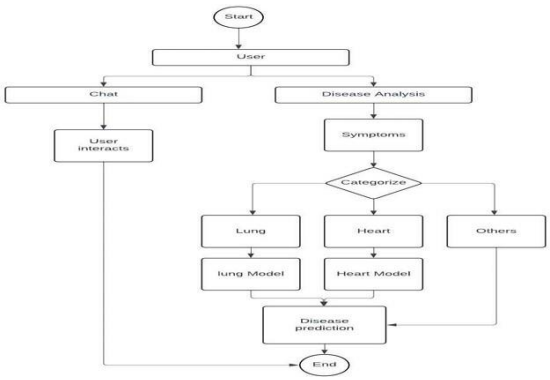

Figure:Activitydiagram

Once the symptoms are categorized and processedthrough their respective models, the system reaches the **disease prediction stage**, where the most probabledisease is identified. The process then terminates after a final prediction is made. This approach ensures that relevant ML or DL models are used to analyze specific types of diseases, improving prediction accuracy for categorized illnesses.

One key observation in the diagram is that the **chat interaction and disease analysis are entirely separate**, with no clear connection between them. If the chat functionalityismeantto assistwith diagnosis, itwould be beneficial to integrate it with the symptom analysis process. Additionally, the system currently appearslimited to **lung and heart diseases**, making it lessscalable for a broader range of illnesses. Incorporating a **generalized disease model** or additional specialized models for other diseases could enhance the system's robustness.

SystemArchitecture

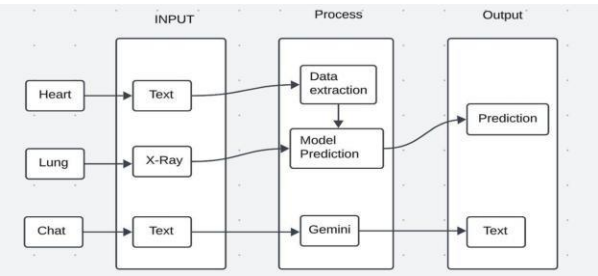

Figure:SystemArchitecture

The diagram represents a machine learning system architecture designed for disease detection using multimodal inputs. The system is divided into three main stages:

1. Input Stage: This stage accepts two types of inputs—textual data and X-ray images. For example, text inputs could be medical records, symptoms, or patient history relative to heart or other diseases. X-ray scans are primarily used to diagnose lung-related issues.
2. ProcessStage:
   - Data Extraction: Textual data undergo preprocessing to extract relevant medical information.Henceforth,featuresareextracted from X-ray images for further analysis.
   - Model Prediction: Extracted data (either from text or images) is processed by a disease-specific predictive model. This modelidentifiespatternsandmakespredictionsbased on the provided data.
   - Gemini API: The Gemini module is responsible for handling additional text-based inputs, such as chat conversations or descriptive notes.
3. OutputStage:
   - Prediction: A diagnosis or probability of the presence of a specific disease (e.g., lung disease based on X-Ray).
   - Text: A human-readable report or summary of the diagnostic results. The system efficiently integrates multimodal inputs (text and X-ray) andemploysfeatureretrievingandMLmodels to provide accurate disease predictions and detailed output reports. It is well-suited for medical diagnostics involving complex and varied input data.

ModuleDecomposition

1. Data Preprocessing Module: This module prepares medical images for CNN by enhancing quality and ensuring consistency.
   - Resizing: Medical images, such asX-rays, are resized to 224x224 pixels to ensure uniform input.
   - Normalization: Image pixels are scaled between 0 to 1 to stabilize model training.
   - Augmentation: Techniques like rotation, zoom, and flipping are applied to prevent overfitting and increase dataset size.
2. Model Architecture (CNN): The CNN is designed to detect lung diseases by extracting hierarchical features from X-ray images.
   - Input Layer: Defines the shape (224x224x3 for RGB images).
   - Convolutional Layers: Apply filtersto extract edge and texture features.
   - Activation Layer: ReLU is used to introduce non-linearity.
   - Pooling Layer: Max pooling reduces spatial dimensions and computational load.

- Output Layer: A sigmoid function outputs binary predictions (e.g., presence or absence of disease).

3. Training the Model: This module involves adjusting model parameters to minimize prediction error.
    - LossFunction:Sparsecategoricalcross-entropy is used for multi-class classification (if applicable).
    - Optimizer: Adam optimizer adjusts the learning rate to accelerate training.
    - Metrics: Accuracy and precision are used to assess the model's prediction performance.
4. Model Evaluation and Validation: After training, the model is tested for generalization and real-world applicability.
    - Validation/Testing: The model's performance is assessed using aseparate validation/test dataset.
    - Generalization: The evaluation checks the model's ability to handle new, unseen data.

AlgorithmDesign

AlgorithmforDataCollectionModule

1. Import necessary libraries: We imported os, numpy, and cv2 for handling image data and processing tasks.
2. Load images: We used OpenCV's imread() function to read medical images (e.g., X-rays) in grayscale mode.
3. Resize images: All images were resized to a uniform size of 224x224 pixels to maintain consistency across the dataset.
4. Store images and labels: We stored the resized images in a list and associated labels (disease status) in another list.
5. Repeatforallimages:`
6. Convert to numpy arrays: The image list was converted into a numpy array, and reshaped to match the required input format (224x224x3).
7. Label conversion: The labels were alsoconverted into a numpy array, mapping numeric values to class names (e.g., 0 for healthy, 1 for disease).
8. Function implementation: We wrapped steps 2-7 in a function and called it for the train, test, and validation datasets to return the respectiveimages and labels.

AlgorithmforDataPreprocessingModule

1. Display sample images: We used imshow() to display sample images from the training, test,and validation datasets in subplots for visual inspection.
2. Shuffle data: The image and label arrays were shuffled randomly to ensure correct pairing for training.
3. Label modification: Numeric labels were converted to human-readable categories (e.g., 0 to "Healthy", 1 to "Disease").
4. Label distribution visualization: We plottedthe distribution to visually check the balance among healthy and diseased samples.
5. Image grid display: A random grid of images was displayed using a suitable plottingfunction to visualize the dataset.
6. Image conversion: The images were converted from the original grayscale into numpy arrays, with pixel values varying between 0 and 255.
7. Normalization: The pixels were normalized from a range of [0, 255] to [0, 1] to make the data more suitable for model training.
8. Verify image shape and count: We checkedthe shape and count of the images to ensure proper conversion and preprocessing.
9. Data augmentation: We used an ImageDataGenerator to apply augmentation techniques (e.g., horizontal flip, width shift) to create robust training, testing, and validation image generators**.**

AlgorithmforModelArchitectureModule

1. CNN Model Creation: We created a CNN model using TensorFlow/Keras libraries to classify the disease from medical images.
2. Sequential Model Setup: We used a sequential model to allow for the addition of multiple layers to progressively extract features.
3. InputLayer:Theinputlayerwasdefinedusing a convolutional layer with filters, strides, padding, and input shape parameters toprocess the image data.
4. Batch normalization and pooling: We applied batch normalization and pooling layers followed by each convolutional layer to stabilize training and reduce computational load.
5. Hidden Layers: We added 5-6 hidden convolutionalandpoolinglayers,repeatingthe convolution and pooling operations to learn complex features.
6. Output Layer: The final output layer was created using a dense layer using the sigmoid activation function for classification (pneumonia, covid-19 or normal).
7. Model Compilation: We compiled the model using the Adam optimizer, binary cross-entropy loss, and accuracy as the evaluation metric.
8. Model Summary: The model summary was printed to verify the architecture.
9. Training: We trained the model with batchesof images over multiple epochs, using validation data to monitor performance.

Algorithm for Model Evaluation and Validation Module

1. Test Accuracy and Loss: We used the evaluate() function to assess the model's performance on test dataset, obtaining test accuracy and loss.

2. Plot Accuracy: We plotted the accuracy over training epochs to visually track improvements and overfitting.
3. Validation Metrics: We plotted validation accuracy and loss curves to monitor the model's performance during training.
4. Epoch-wise Evaluation: We tested the model at various epochs to analyze its performance on both the training and validation data.
5. Confusion Matrix: We generated a confusion matrix to assess the TP, FP, TN, and FN.
6. Classification Report: We generated a classification report for the model's predictions.
7. Model Readiness: We prepared the model for deployment,savingthetrainedweightsforfuture predictions.

AlgorithmforPredictionfromtheModel

1. Predict using the model: We used the predict() function to generate predictions on the test images.
2. Convert predictions: The predictions, whichwere probabilities, were converted to binaryclasslabels(0for"Healthy"and1for"Disease ") based on a threshold.
3. Comparison: We flattened the predicted and actual labels and compared them using the classification report to assess performance.
4. Correct/Incorrect Predictions: We counted the correct and incorrect predictions to determine if the model required further retraining.
5. Retraining: For incorrectly predicted samples,we retrained the model by adjusting hyperparameters and optimizing the training process.
6. Save Model: The final trained model is saved in pickle format for integration with the frontend interface.
7. Frontend Integration: Using frameworks like Streamlit, Flask, or React, we built an interfaceto which users can react to the model X-ray images and get disease predictions from the trained model through an AI API (e.g., Gemini API).

AlgorithmforMultipleDiseasePrediction

1. MultipleDiseaseModels:Werepeatedtheabove data analysis steps for multiple diseases (e.g., lung disease, heart disease).
2. TrainIndividualModels:Individualmodelstobe trained for individual disease, using different datasets and class labels for each condition.
3. Combine Models: We integrated the individual modelsintoasinglesystemcapableofpredicting multiple diseases from a single image input.
4. Multi-Disease Prediction System: The combined model was deployed in the frontend interface, allowing healthcare professionals to upload a single image andreceivepredictions formultiple diseases.

V. RESULTS

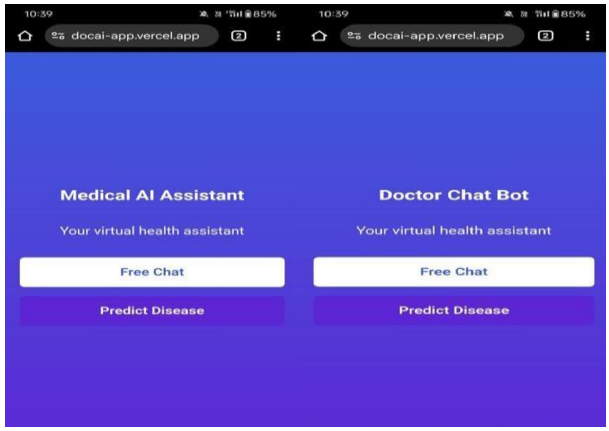

Figure:StartingInterface:Welcomepageoftheapplication

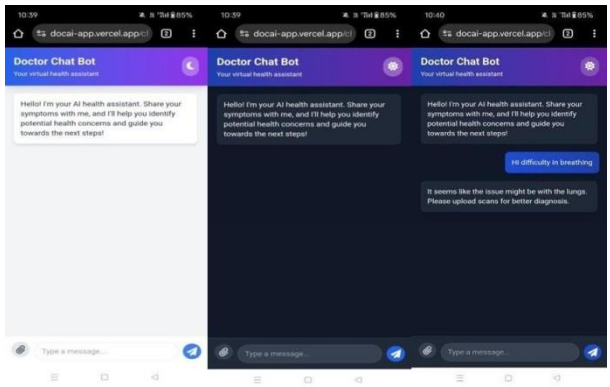

Figure:ChatsbetweentheuserandtheGenai

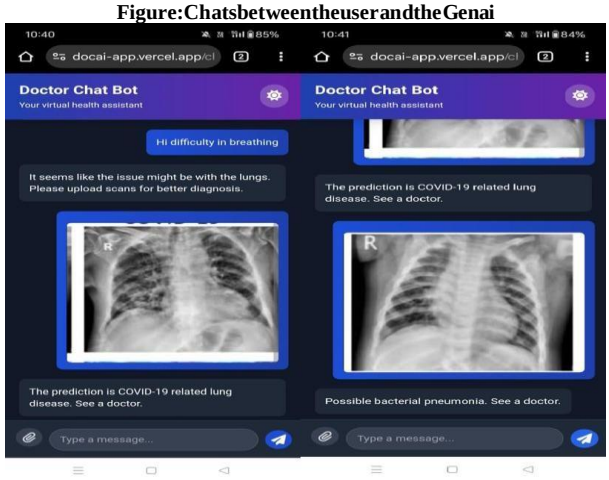

Figure:UploadingimagesoflungX-raysandpredictionofresults

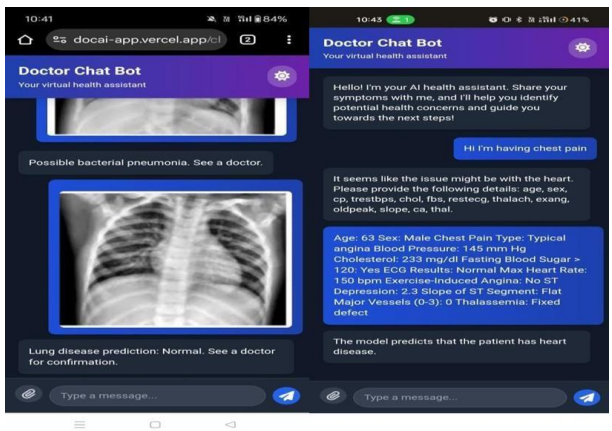

Figure:Predictiononpotentialdiseasewillbedisplayedonthescreen.

VI. CONCLUSION

This project demonstrates the power of machine learning, particular deep learning models, in enhancing early disease detection and improving diagnostic accuracy. The integration of CNN for chest X-ray analysis and Multi-Layer Perceptrons (MLPs) for clinical data prediction enables the identification of conditions like pneumonia and heart disease, respectively. By leveraging multi-disease prediction systems, we have shown, it is possible to detect multiple health risks early, improving treatment outcomes and healthcare efficiency.

Despite these advantages, challenges remain, particularly in data quality, model interpretability, and clinical validation.However,thesystemholdssignificantpromise for streamlining healthcare delivery and offering more efficient use of resources.

In the future, the system can be expanded to include additional models for diseases like diabetes, cancer, and kidney disease. By incorporating multi-modal data suchas text, medical images, and biometric information, and integrating with telemedicine platforms, the project could facilitate remote consultations, providing patients with initial assessments and supporting doctors in their diagnostic decisions.

REFERENCES

[1]. Swaroop Sana, "Multiple disease prediction system using machine learning", Researchgate.net, 2024.

[2]. Srinivas Karthik, Lambavai Kummera, Lakshmi DeepthiGopisetti,andSaiRohanPattamsetti,"Multiple DiseasePredictionSystemusingMachineLearningand Streamlit", IEEE Conference Publication, 2023.

[3]. Mallula Venkatesh, "Multiple disease prediction system using machine learning, deep learning, and streamlit", e-ISSN: 2582-5208, Volume:05/Issue:07/July-2023".

[4]. Nevon Projects, "Multiple disease predictionsystem using machine learning", 2023.

[5]. Harshit Gupta, "Multiple disease prediction system using machine learning", id=4655833, 2023.

[6].Parshant,Dr.AnuRathee,"MultipleDisease PredictionSystemusingMachineLearning,IRE Journals, Volume 6 Issue 12, ISSN: 2456-8880, 2023. [7]. Banoth Ramesh, G. Srinivas and P. Ram Praneeth Reddy, "Feasible Prediction of Multiple Diseases using MachineLearning|FeasiblePredictionofMultiple DiseasesusingMachineLearning|E3SWebof Conferences (e3s-conferences.org)", 2023.

[8]. Sai Sriram Krishna Parimi and Y.Snehith Reddy, "Prediction of Multiple Diseases using Machine Learning Techniques", 2022.

[9]. Indukuri Mohit, K. Santhosh Kumar, Badhagouni Suresh Kumar and Uday Avula Kumar Reddy, "An Approach to detect multiple diseases using machine learning algorithm", Conf. Ser. 2089 012009, 2021.

[10]. Palle Pramod Reddy, Dirisinala Madhu Babu, Hardeep Kumar and Dr. Shivi Sharma, "ijcrt.org/papers/IJCRT2105229.pdf",2021.