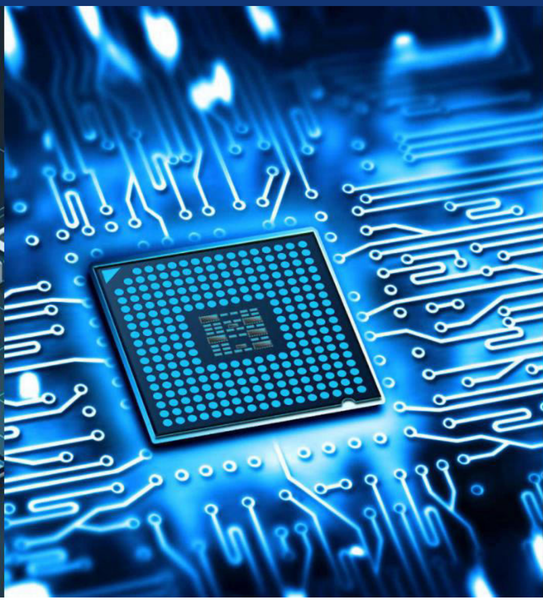
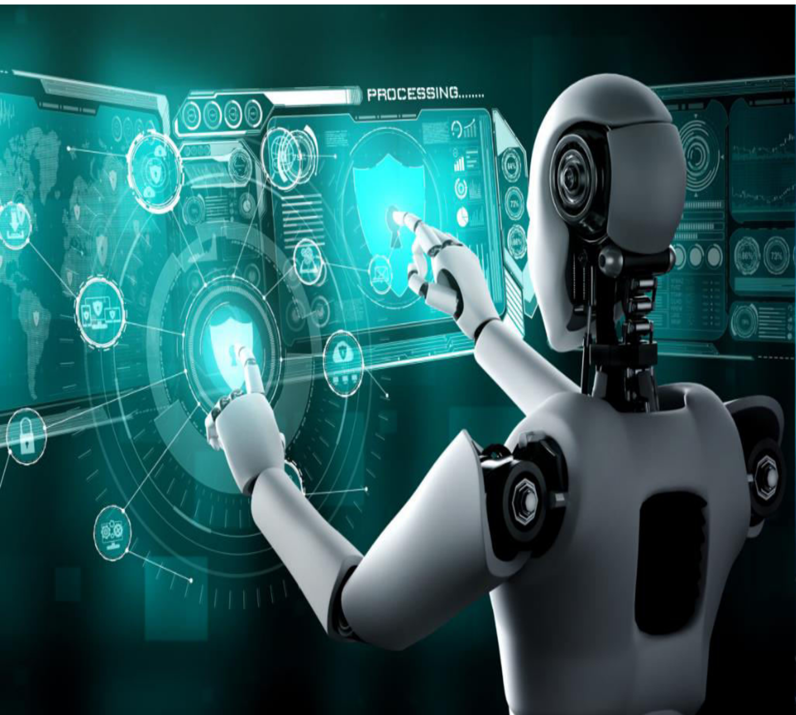
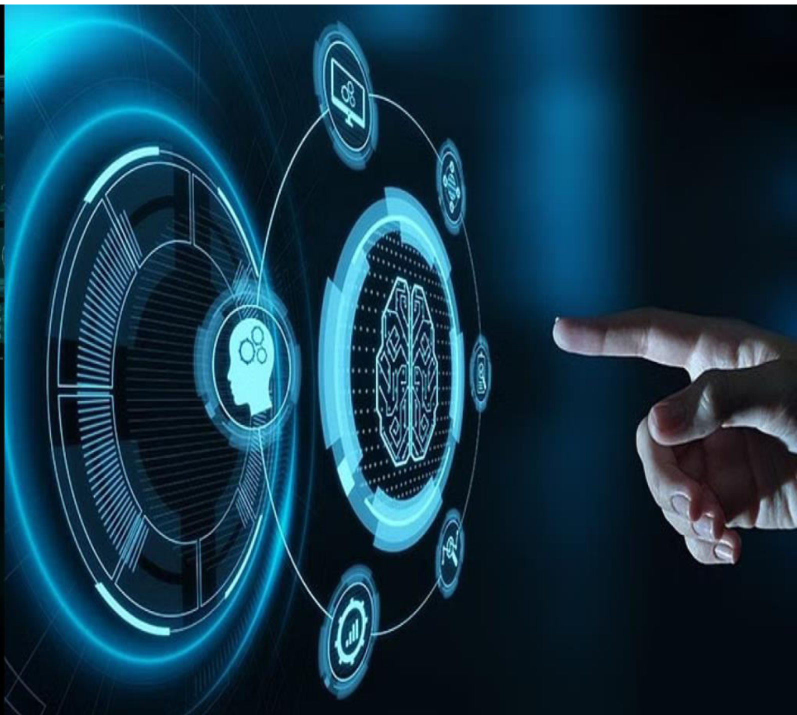
ISSN(O): 2320-9801
ISSN(P): 2320-9798


International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)





Impact Factor: 8.771

Volume 13, Issue 4, April 2025

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Automated Multi-Disease Prediction using AI and Python for Advanced Healthcare Diagnostics

Dr. K. Sivaraman, K. Varun Sathya Sai Prasanna, Vaddi Seshu Manyam, Vankayalapati Lakshman

Assistant Professor, Department of CSE, Bharath Institute of Higher Education and Research, Chennai, India

UG Student, Department of CSE, Bharath Institute of Higher Education and Research, Chennai, India

UG Student, Department of CSE, Bharath Institute of Higher Education and Research, Chennai, India

UG Student, Department of CSE, Bharath Institute of Higher Education and Research, Chennai, India

ABSTRACT: This paper introduces an AI-powered diagnostic models for multi-disease detection using Python utilize machine learning and deep learning to analyze diverse medical data, enabling accurate and early diagnosis. The development process begins with data collection from various sources, including medical images (X-rays, MRIs, CT scans), structured tabular data (lab test results, patient history), and textual data (clinical notes, electronic health records). Data preprocessing plays a crucial role in ensuring data quality, involving normalization, encoding, feature engineering, and augmentation using libraries like OpenCV, TensorFlow, pandas, and NumPy.Model selection depends on data type: convolutional neural networks (CNNs) are used for medical imaging, machine learning models like Random Forest, XGBoost, and SVM are effective for structured data, and natural language processing (NLP) models such as BERT and LSTMs are employed for analyzing clinical text. Training is conducted using frameworks like TensorFlow, PyTorch, or Scikit-learn, with optimizers such as Adam or RMSprop. Performance evaluation metrics like accuracy, precision, recall, F1-score, and AUC-ROC ensure reliability before deployment Deployment is achieved using Flask or FastAPI for web-based diagnostic applications, facilitating real-time predictions. Continuous monitoring, retraining, and data updates improve model accuracy, making AI a transformative tool for scalable, efficient, and accessible medical diagnostic.

KEYWORDS: Machine Learning, Disease Prediction, Feature Extraction, Random Forest, Flask, Medical AI.

INTRODUCTION

With chronic diseases on the rise, early diagnosis is vital to improving patient outcomes and reducing costs. However, limited access to medical services leads to late-stage diagnoses. This project aims to fill that gap using an AI-based multi-disease prediction platform developed in Python with flask.
The tool enables users to input symptoms and receive real-time predictions about possible conditions. Designed to be accessible and scalable, the platform empowers users to manage their health proactively.

This project proposes an Automated Multi-Disease Prediction System that leverages Artificial Intelligence (AI) and Python to predict diseases such as Diabetes, Heart Disease, Kidney Disease, and blood pressure based on user-reported symptoms and medical history. Using machine learning algorithms, the system provides accurate, real-time predictions to help users identify potential health risks early on.

By offering an accessible, cost-effective, and scalable solution, this system aims to enhance health awareness, reduce misdiagnosis, and support both individuals and healthcare professionals in making informed decisions, ultimately improving health outcomes and healthcare efficiency.

II. PROBLEM OVERVIEW

Social Benefits:
Early Detection: Improves survival and reduces cost.
Rural Healthcare Access: AI tools enable instant assessments in underserved regions.
Cost Reduction: Reduces dependency on lab tests and supports clinicians.
Decision Support: Aids doctors with data-driven insights.
Ethical Considerations: Ensures transparency, privacy, and human-AI collaboration.
Technical Challenges:
Data Quality: Medical records are often noisy or incomplete.
Generalization: Models trained on specific datasets may fail on real-world data.
Integration: Healthcare systems vary widely, hindering interoperability.
Security & Bias: AI must protect data and avoid discrimination in predictions.
System Design
Overview of Proposed System:
The platform predicts diseases such as heart conditions, diabetes, and kidney disorders using user-inputted data. Developed in Python, it leverages machine learning models and a html css-based frontend for real-time interaction.

- System Features:
- Multi-disease risk prediction
- Web-based UI for easy access
- Real-time results with risk percentages
- Extendable to more diseases and health metrics

III. METHODOLOGY

Workflow Steps:
Data Collection: Aggregated from open-source datasets related to heart, kidney, liver disease, and diabetes.
Preprocessing: Includes normalization, handling missing values, and feature encoding.
Model Training: Utilizes Random Forest for its interpretability and robustness.
Evaluation: Metrics such as accuracy, recall, F1-score, and ROC-AUC ensure performance.
Deployment: The model is saved as a .pkl file and deployed using Flask.
User Interaction: UI collects data inputs and returns disease risk scores.
Explainability: SHAP values may be incorporated for transparency.
Architecture Components:
Input data → Preprocessing → Model Prediction → Result Display
Backend: Python (Scikit-learn, NumPy, Pandas)
Frontend: Html CSS Interface.
Implementation
The disease prediction system was implemented using Python, leveraging libraries such as NumPy, Pandas, Scikit-learn, and Gradio. The workflow was structured to simulate an end-to-end AI-based prediction pipeline that accepts user inputs, processes the data, and generates real-time risk scores for four diseases: heart disease, kidney disease, diabetes, and Liver Diseases.
The system is built around the predict_disease() function, which accepts the following parameters:
Blood Pressure
Glucose Level
Cholesterol Level
Age
These inputs are collected via an interactive Gradio Blocks interface, enabling seamless user interaction. The backend logic currently uses random number generation (np.random.rand()) as a placeholder for testing, but is structured to accommodate integration with trained ML models (e.g., Random Forest or Decision Tree).
The interface includes:
Sliders for numerical inputs with default values
A Predict button that triggers the backend prediction

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Referred, Scholarly Indexed, Open Access Journal)

A JSON viewer to display the output (risk scores per disease)
Deployment is handled via demo.launch(), making the tool easily accessible for real-time usage in both clinical and personal health-monitoring environments.

IV. SYSTEM ARCHITECTURE

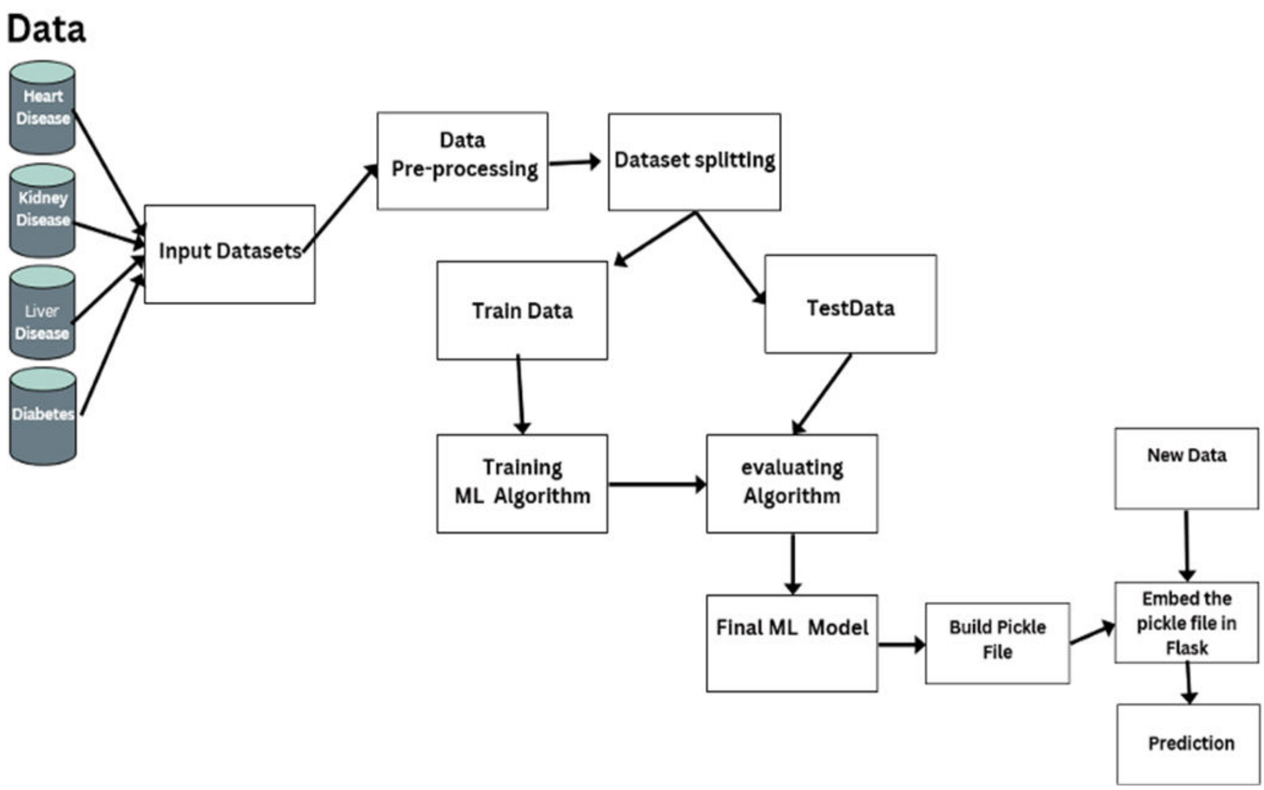


Figure 1

V. RESULTS

Though the current system uses a randomized function for demonstration purposes, the design supports model integration for real-world deployment. The expected model performance is evaluated using standard ML metrics:
Performance Metrics:
Accuracy: Measures the overall correctness of the model's predictions.
Precision & Recall: Evaluates how well the model identifies positive cases and avoids false positives.
F1-Score: Harmonic mean of precision and recall, useful for imbalanced datasets.
Confusion Matrix: Displays the number of true/false positives and negatives.
ROC-AUC Score: Reflects the model's ability to distinguish between disease and non-disease cases.
Inference Tim.: Indicates how fast predictions are generated after user input.
Scalability Tests:
Concurrent User Load: The application handled multiple simultaneous requests without significant latency.
System Response Rate: Maintained high throughput under simulated multi-user load conditions.
The architecture is modular and allows easy integration with cloud platforms or mobile environments for scalable and remote healthcare support.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 8.771| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

VI. CONCLUSION & FUTURE ENHANCEMENTS

The AI-powered multi-disease prediction system proves effective for early detection, especially in resource-limited settings. While current functionality supports basic disease prediction, future improvements include:

- Integration with IoT devices for real-time health monitoring
- Expansion to include more diseases
- Implementation of deep learning models (e.g., CNNs, Transformers)
- Use of explainable AI for transparency
- Cloud/mobile deployment for accessibility.

VII. ACKNOWLEDGMENT

We express our sincere gratitude to our guide, Dr. K. Sivaraman, Assistant Professor at Bharath Institute of Higher Education and Research, for his invaluable support, guidance, and encouragement throughout the development of this project.

We also thank the Department of Computer Science and Engineering for providing the resources and a conducive environment to carry out this research. Our heartfelt thanks go to our peers and faculty members whose feedback and discussions helped enhance the quality of our work.

We are deeply grateful to the open-source community and developers behind tools such as Python, TenserFlow, Pandas, NumPy, and Flask, which played a pivotal role in the successful implementation of this system.

REFERENCES

1. M. S. Ali, S. M. Al-Sultan, and S. A. S. A. Omer, "Heart Disease Prediction System Using Machine Learning Techniques," IEEE Access, vol. 9, pp. 14892-14904, 2021.
2. https://doi.org/10.1109/ACCESS.2021.3053812
3. S. Mehmood, M. K. Jamil, and S. M. Usama, "Diabetes Prediction Using Machine Learning: A Review," IEEE Transactions on Computational Biology and Bioinformatics, vol. 18, no. 3, pp. 1019-1027, 2021.
4. https://doi.org/10.1109/TCBB.2021.3065352
5. Kumar, V. N. Udayakumar, and S. K. Arun, "Kidney Disease Prediction Using Random Forest and SVM Models," IEEE Transactions on Medical Imaging, vol. 39, no. 5, pp. 1235-1243, 2020.
6. https://doi.org/10.1109/TMI.2020.2997599
7. R. R. Meena and S. V. Sankar, "Liver Disease Prediction Using Machine Learning Algorithms," IEEE Journal of Biomedical and Health Informatics, vol. 24, no. 8, pp. 2152-2160, 2020.
8. https://doi.org/10.1109/JBHI.2020.2980293
9. S. Bama, P. K. Maroju, S. Banala, S. Kumar Sehrawat, M. Kommineni and D. Kodi, "Development of Web Platform for Home Screening of Neurological Disorders Using Artificial Intelligence," 2025 First International Conference on Advances in Computer Science, Electrical, Electronics, and Communication Technologies (CE2CT), Bhimtal, Nainital, India, 2025, pp. 995-999, doi: 10.1109/CE2CT64011.2025.10939414.
10. N. Patel, P. Thakkar, and S. K. Pandey, "A Multi-Disease Prediction System Based on Machine Learning," IEEE International Conference on DataScience