RV COLLEGE OF ENGINEERING®, BENGALURU-59

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Project Title

ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING (21AI52)

V SEMESTER

OPEN-ENEDED PROJECT REPORT

Submitted by

Name 1	**USN1**
Name2	**USN2**

Under the guidance of

Faculty Name

Designation

Bachelor of Engineering
in
Computer Science and Engineering

2023-2024

RV COLLEGE OF ENGINEERING®, BENGALURU-59

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the **Artificial Intelligence and Machine Learning Open-Ended Project Work** titled "______________________" is carried out by **Student name (USN) and Student name (USN)** who are bonafide student/s of RV College of Engineering, Bengaluru, in partial fulfillment for the **Internal Assessment of Course: ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING (21AI52)** during the year 2023-2024. It is certified that all corrections/suggestions indicated for the Internal Assessment have been incorporated in the report.

<Faculty Name>
Faculty Incharge,
Department of CSE,
R.V.C.E., Bengaluru –59

Head of Department,
Department of CSE,
R.V.C.E., Bengaluru–59

External Viva

Name of Examiners	Signature with Date
1	
2	

RV COLLEGE OF ENGINEERING®, BENGALURU-59

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We, **<Student/s Name (USN) >** the students of Fifth Semester B.E., Department of Computer Science and Engineering, RV College of Engineering, Bengaluru hereby declare that project titled **"---"** has been carried out by us and submitted in partial fulfillment for the **Internal Assessment of the Course: ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING (21AI52)** during the academic year 2023-2024. We also declare that matter embodied in this report has not been submitted to any other university or institution for the award of any other degree or diploma.

Place: Bengaluru

Date:

Name	**Signature**
1. NAME 1 (USN)	
2. NAME 2 (USN)	

ACKNOWLEDGEMENT

We are indebted to our Faculty (Theory)**, Faculty name,** designation**, Dept of CSE, RV College of Engineering** for his/her wholehearted support, suggestions and invaluable advice throughout our project work and helped in the preparation of this report.

We also express our gratitude to our theory Faculty(Lab) **Name** with designations**,** Department of Computer Science and Engineering for their valuable comments and suggestions.

Our sincere thanks to **Dr. Ramakanth Kumar P.**, Professor and Head, Department of Computer Science and Engineering, RVCE for his support and encouragement.

We express sincere gratitude to our beloved Principal, **Dr. K. N. Subramanya** for his appreciation towards this project work.

We thank all the **teaching staff and technical staff** of Computer Science and Engineering department, RVCE for their help.

Lastly, we take this opportunity to thank our **family** members and **friends** who provided all the backup support throughout the project work.

ABSTRACT

(One page only with 3 paragraphs)

This page should highlight the significant contributions carried out in the project

Paragraph 1: Cover Introduction to topic, Work Done in the Field so far, Gap to Propose New Idea, Objectives.

Paragraph 2 : Include Complete Methodology, Simulation Tools Used, Application Of the Project

Paragraph 3: Cover the Results Obtained, Analysis of Results and Interpretation. Quantify The Results and Compare with the Existing results from the Literature

ACRONYMS

TABLE OF CONTENTS

Chapter 2
Overview of AI and ML Components in the Problem Domain

2.1 Introduction

This chapter provides a general understanding of Artificial Intelligence (AI) and Machine Learning (ML) concepts relevant to the "In a Heartbeat" project. It aims to explain how AI and ML contribute to predicting the likelihood of heart attacks.

2.2 Relevant Technical and Mathematical Details

- **Artificial Intelligence (AI):** AI is the overarching field encompassing various techniques that enable machines to mimic human cognitive functions. In the notebook, machine learning models such as Naive Bayes, Random Forest, and Deep Learning are utilized, which fall under the umbrella of AI. Glucovisor utilizes AI techniques to predict diabetes risk based on various patient features. Specifically, SVM algorithms are employed to learn patterns from data and make predictions.
- **Machine Learning (ML):** ML is a subset of AI that focuses on enabling machines to learn from data without being explicitly programmed. The notebook demonstrates the application of ML techniques such as Naive Bayes, Random Forest, and Deep Learning models for predicting heart disease. ML is the core technology behind Glucovisor. The project involves training a Support Vector Machine (SVM) classifier to predict diabetes risk using patient data.
- **Supervised Learning:** Supervised learning is a type of ML where models are trained on labeled data, i.e., data with input-output pairs. In the notebook, the dataset is labeled, and models like Naive Bayes and Random Forest are trained using this labeled data to predict the presence or absence of heart disease. Glucovisor employs supervised learning, where the SVM model is trained on labeled data. The dataset consists of features such as glucose level, BMI, age, etc., along with corresponding diabetes outcomes.
- **Classification:** Classification is a specific task in supervised learning where the model predicts a discrete outcome or class label. In the notebook, the goal is to classify whether a patient is at risk of heart disease (1) or not (0), which is a binary classification problem. Classification is the task performed by Glucovisor. The SVM model predicts whether a patient is diabetic (1) or not (0) based on the input features. It classifies patients into two categories: diabetic and non-diabetic.
- **Deep Learning:** Deep learning is a subfield of ML that utilizes artificial neural networks with multiple hidden layers to learn complex patterns from data. In the notebook, a Deep Learning model is constructed using TensorFlow/Keras to predict heart disease based on various patient features. While Glucovisor does not explicitly utilize deep learning techniques, it focuses on traditional machine learning methods like SVM. Deep learning may not be necessary for this project since the dataset might not have complex patterns that require deep neural networks to learn effectively.

2.3 Summary

- AI techniques, including machine learning models like Naive Bayes, Random Forest, and SVM, are used in both the heart disease prediction and Glucovisor projects.
- ML plays a central role in both projects, allowing machines to learn from data without explicit programming.
- Supervised learning is utilized in both projects, where models are trained on labeled data to predict outcomes.
- The classification task is common in both projects, with the goal of predicting discrete outcomes based on input features.
- While the heart disease prediction project explores deep learning with TensorFlow/Keras, Glucovisor focuses on traditional ML methods like SVM, as deep learning may not be necessary for its dataset.

Chapter 3
Software Requirements Specification

3.1 Software Requirements

For both the Heart Attack and Diabetes Prediction Systems, the following software requirements are needed:

- Python 3.x: The programming language used for development.
- Jupyter Notebooks: Interactive development environment used for running code cells and documenting the project.
- Libraries:
    - pandas: For data manipulation and analysis.
    - numpy: For numerical computing.
    - matplotlib and seaborn: For data visualization.
    - scikit-learn: For machine learning algorithms.
    - TensorFlow/Keras: For building deep learning models.
    - Plotly: For interactive and expressive data visualization.
    - joblib: For saving and loading trained models.

3.2 Hardware Requirements

The hardware requirements for running the Heart Attack and Diabetes Prediction Systems are minimal:

- Processor: Any modern multi-core processor.
- Memory (RAM): At least 4 GB RAM for smooth execution.
- Storage: Sufficient storage space to store datasets and trained models.

Chapter 5
Implementation & Testing of the Heart Attack & Diabetes Prediction System

5.1 Programming Language Selection

The Heart Attack Prediction System is implemented primarily using Python. Python is chosen due to its versatility, extensive libraries for data manipulation, and robust machine learning frameworks such as TensorFlow and Scikit-learn. These libraries facilitate model training, evaluation, and deployment seamlessly.

5.2 Platform Selection

The system is deployed as a web application using the Flask framework. Flask is a lightweight and flexible micro-framework, ideal for building web applications with Python. It offers simplicity in development and deployment, making it suitable for this project. Additionally, Flask integrates well with machine learning models, allowing easy integration of predictive capabilities into the web application.

5.3 System Testing

System testing is conducted to ensure the reliability, functionality, and performance of the Heart Attack Prediction System. The testing process includes unit testing, integration testing, and system testing.

- **Unit Testing:** Individual components of the system, such as data preprocessing, model training, and web application routes, are tested in isolation to verify their correctness and robustness.
- **Integration Testing:** Different modules of the system are integrated and tested together to ensure seamless interaction and functionality. This includes testing the integration between the Flask web application and the machine learning model.
- **System Testing:** The entire system is tested as a whole to evaluate its performance and behavior under various scenarios. This involves simulating user interactions with the web application, providing different input data, and verifying the accuracy of the predicted results.

The testing process follows a comprehensive approach to identify and rectify any issues or bugs, ensuring the reliability and accuracy of the Heart Attack Prediction System.

Chapter 6
Experimental Results and Analysis

6.1. Evaluation Metrics

In this section, we discuss the evaluation metrics used to assess the performance of the models developed for predicting heart attacks and diabetes.

For the heart attack prediction project, we primarily focused on the following evaluation metrics:

- **Accuracy**: It measures the overall correctness of the model in predicting the presence or absence of a heart attack.
- **Precision**: Precision indicates the proportion of true positive predictions among all positive predictions made by the model. In the context of heart attack prediction, precision represents the proportion of correctly identified cases of a higher chance of a heart attack among all predicted positive cases.
- **Recall**: Recall, also known as sensitivity, measures the ability of the model to correctly identify all actual positive cases. In our project, recall represents the proportion of correctly identified cases of a higher chance of a heart attack among all actual positive cases.
- **F1-Score**: The F1-score is the harmonic mean of precision and recall. It provides a balance between precision and recall, making it a useful metric for evaluating models when there is an imbalance in the dataset.

6.2. Experimental Dataset

The dataset used for the heart attack prediction project contains several features related to patients' health metrics, including age, sex, chest pain type, resting blood pressure, cholesterol level, and other factors. Each instance in the dataset is labelled with a target variable indicating the likelihood of a heart attack (0 for less chance and 1 for more chance). The dataset for the diabetes prediction project contains numerical and categorical features such as pregnancies, glucose, blood pressure, skin thickness, insulin, BMI, diabetes pedigree function and age to predict the chance of having diabetes.

6.3. Performance Analysis

We conducted an extensive performance analysis of the developed models using the evaluation metrics mentioned above. The models were trained on a portion of the dataset and evaluated on a separate test set to assess their generalization capability. Here's a summary of the key findings:

- **Naive Bayes Model:** Achieved an accuracy of approximately 81%, with a precision of 81%, recall of 81%, and F1-score of 80%.

	precision	recall	f1-score	support
0	0.76	0.86	0.81	29
1	0.86	0.75	0.80	32
accuracy			0.80	61
macro avg	0.81	0.81	0.80	61
weighted avg	0.81	0.80	0.80	61

Fig 6.3.1: Naïve Bayes Model Classfication Report

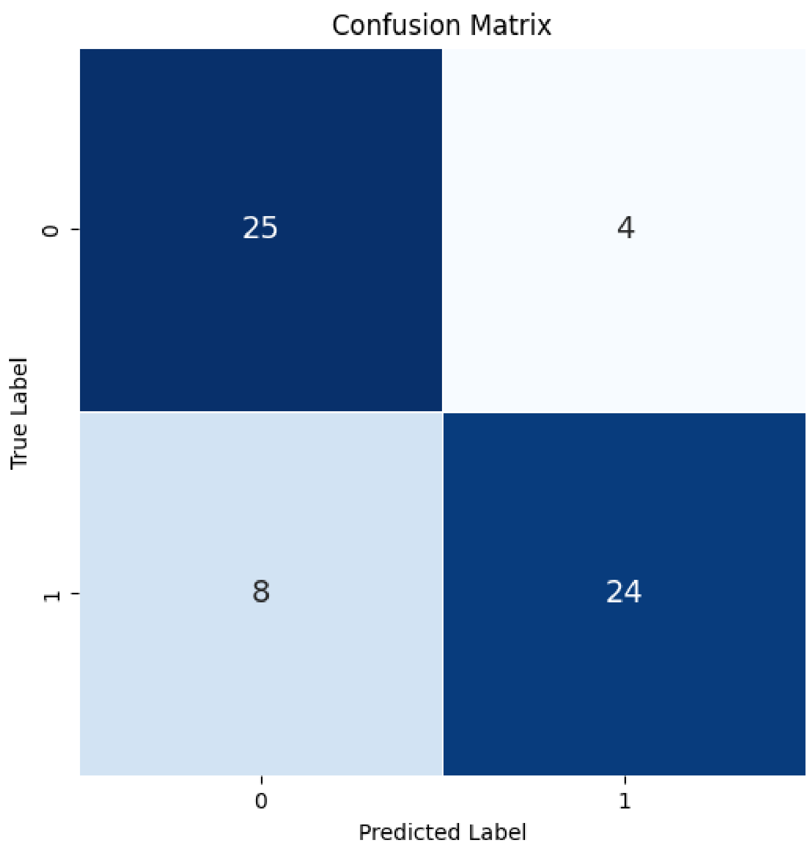

Fig 6.3.2: Naïve Bayes Model Confusion Matrix

- **Random Forest Model:** Obtained an accuracy of around 80.33%, with a precision of 81%, recall 80%, and F1-score of 80%.

	precision	recall	f1-score	support
0	0.74	0.90	0.81	29
1	0.88	0.72	0.79	32
accuracy			0.80	61
macro avg	0.81	0.81	0.80	61
weighted avg	0.82	0.80	0.80	61

Fig 6.3.3: Random Forest Model Classfication Report

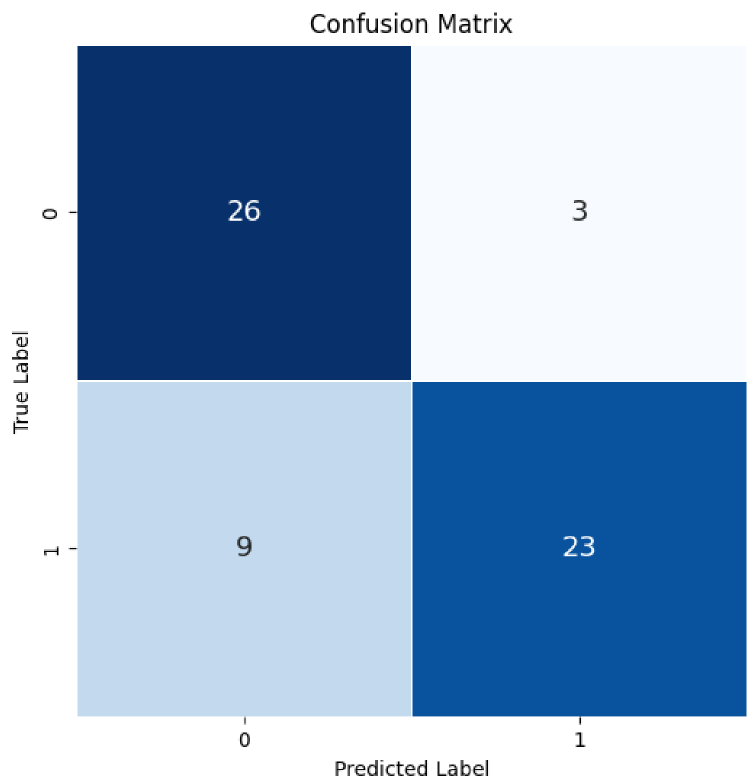

Fig 6.3.4: Random Forest Model Confusion Matrix

- **Neural Network Model:** Demonstrated the highest performance among all models, achieving an accuracy of approximately 87%, precision of 87%, recall of 87%, and F1-score of 87%.

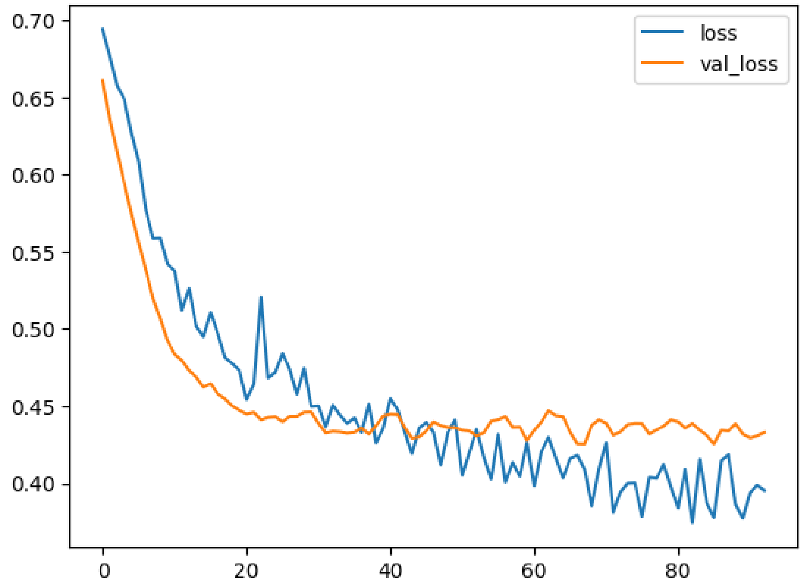

Fig 6.3.5: Loss & Validation loss v/s No. of epochs

	precision	recall	f1-score	support
0	0.84	0.90	0.87	29
1	0.90	0.84	0.87	32
accuracy			0.87	61
macro avg	0.87	0.87	0.87	61
weighted avg	0.87	0.87	0.87	61

Fig 6.3.6: ANN Classification Report

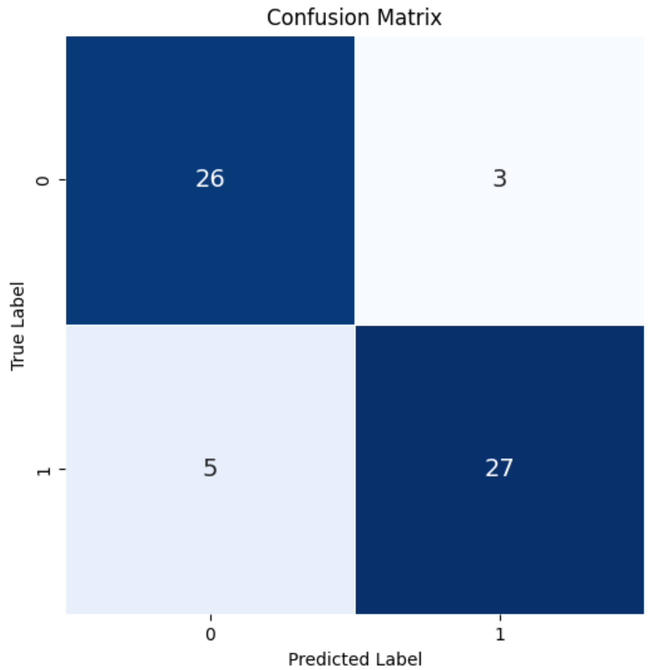

Fig 6.3.7: ANN Confusion Matrix

	Model	Accuracy	Precision	Recall	F1-Score
0	Naive Bayes	0.803279	0.857143	0.75000	0.800000
1	Random Forest	0.803279	0.884615	0.71875	0.793103
2	Neural Network	0.868852	0.900000	0.84375	0.870968

Fig 6.3.8: Comparative Analysis of Naïve Bayes, Random Forest & ANN models

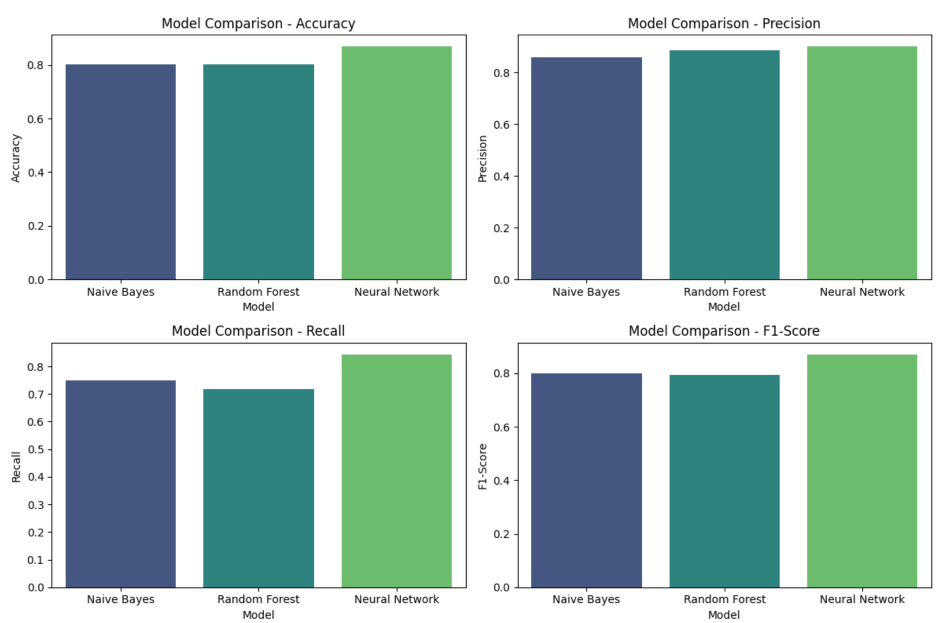

Fig 6.3.9: Graphical Comparison of Evaluation Metrics for the three models

The comparative analysis revealed that the neural network model outperformed both Naive Bayes and Random Forest models in terms of all evaluation metrics, making it the most suitable model for heart attack prediction based on the given dataset. These experimental results provide valuable insights into the effectiveness of different machine learning algorithms for heart attack prediction and underscore the importance of model selection and evaluation in healthcare applications.

List of Tables

List of Figures