

Heart Disease Prediction Using ML

A Mini Project Report

Submitted

In Partial Fulfillment of the Requirements

For the Degree of

Bachelor of Technology(B.Tech)

in

Computer Science & Engineering

by

Vishal Kumar
(2301920100356)

Vimarsh Raina
(2301920100351)

Vivek Kumar Ankit
(2301920100366)

under the supervision of

Ms.Anju Joshi

Assistant Professor



G. L. BAJAJ INSTITUTE OF TECHNOLOGY & MANAGEMENT
GREATER NOIDA



DR. A. P. J. ABDUL KALAM TECHNICAL UNIVERSITY,
UTTAR PRADESH, LUCKNOW

2024-2025

Declaration

We hereby declare that the project work presented in this report entitled “Heart Disease Prediction Using ML”, in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science & Engineering, submitted to Dr. A.P.J. Abdul Kalam Technical University, Uttar Pradesh, Lucknow is based on our own work carried out at Department of Computer Science & Engineering, G.L. Bajaj Institute of Technology & Management, Greater Noida. The work contained in the report is true and original to the best of our knowledge and project work reported in this report has not been submitted by us for award of any other degree or diploma.

Signature

Name: Vishal Kumar

Roll No: 2301920100356

Signature

Name: Vimarsh Raina

Roll No: 23019020100351

Signature

Name: Vivek Kumar Ankit

Roll No: 2301920100366

Certificate

This is to certify that the mini-Project report entitled “Prediction of Heart Disease using ML” done by **Vishal Kumar(2301920100356), Vimarsh Raina(2301920100351 & Vivek Kumar Ankit(2301920100366)** is an original work carried out by them in Department of Computer Science & Engineering, G.L. Bajaj Institute of Technology & Management, Greater Noida under my guidance. The matter embodied in this project work has not been submitted earlier for the award of any degree or diploma to the best of my knowledge and belief.

Date:

Dr./Mr./Mrs./Coordinator

Signature of the Coordinator

Acknowledgement

The merciful guidance bestowed to us by the almighty made us stick out this project to a successful end. We humbly pray with sincere heart for his guidance to continue forever

We pay thanks to our project guide **Ms. Anju Joshi** who has given guidance and light to us during this project. her versatile knowledge has helped us in the critical times during the span of this project.

We pay special thanks to our Head of Department **Dr. Sansar Singh Chauhan** who has been always present as a support and help us in all possible way during this project.

We also take this opportunity to express our gratitude to all those people who have been directly and indirectly with us during the completion of the project.

We want to thanks our friends who have always encouraged us during this project

At the last but not least thanks to all the faculty of CSE department who provided valuable suggestions during the period of project

Abstract

Heart disease is a significant global health concern, contributing to high mortality rates worldwide. Early prediction and diagnosis of heart disease can play a crucial role in preventing adverse health outcomes. This study focuses on the application of machine learning techniques for predicting heart disease using the Framingham Heart Study dataset. The dataset comprises a wide range of clinical, lifestyle, and demographic features, including age, cholesterol levels, blood pressure, smoking habits, and diabetes history.

The methodology involves preprocessing the dataset, handling missing values, and normalizing data to ensure optimal performance of machine learning models. We will use learning algorithms, such as Logistic Regression , Confusion Matrix in this project. Additionally, feature importance analysis highlights significant contributors to heart disease, providing insights for healthcare professionals to prioritize preventive measures. This study underscores the potential of machine learning as a valuable tool in predictive healthcare and emphasizes the importance of data-driven approaches in combating heart disease.

TABLE OF CONTENTS

(i)Declaration.....	(ii)
(ii) Certificate	(iii)
(iii) Acknowledgement.....	(iv)
(iv)Abstract.....	(v)
(iv) Table of Content.....	(vi)
(v) List of Figures.....	(viii)

Chapter 1. Introduction	1
--------------------------------	----------

1.1Background

1.2Problem Statement

1.3Objectives

1.4Scope

Chapter 2 .Technology Used	3
-----------------------------------	----------

2.1 Development Environment

2.2 Libraries and tools

2.3 Data Source

Chapter 3. Plan of Work	5
--------------------------------	----------

3.1 Phases of Work

3.2 Timeline

Chapter 4. Methodology	11
-------------------------------	-----------

4.1 Data Preprocessing

4.2 Model Development

4.3 Building a Predictive Model

Chapter 5.Results & Discussion	15
5.1 Results	
5.2 Discussion	
 Chapter 6.Conclusion,Limitation & Future Scope	16
6.1 Conclusion	
6.2 Limitations	
6.3 Future Scope	
 References	17

LIST OF FIGURES

No. Of Figures	Name Of Figures	Page No.
Fig1	Libraries	3
Fig2	Info Of Dataset	5
Fig3	Missing Values	6
Fig4	Fill Missing Values	6
Fig5	Bar Graph	7
Fig6	Scatter Plot	8
Fig7	Split (Train and Test)	8
Fig8	Logistic Regression	9
Fig9	Accuracy Of Model	9
Fig10	Scaling	11
Fig11	Check Duplicates	11
Fig12	Logistic Regression	12
Fig13	Confusion Matrix	13
Fig14	Plot Confusion Matrix	13
Fig15	Build Predictive Model	14

CHAPTER-1

INTRODUCTION

1.1 Background

Heart disease is a significant public health concern, and its early detection is critical for effective treatment and management. Machine learning provides an innovative approach to predictive analysis in healthcare, allowing for more accurate and efficient diagnosis.

1.2 Problem Statement

Existing methods for predicting heart disease often rely on manual analysis and predefined thresholds, which may not account for complex patterns in the data. This project addresses the need for an automated, data-driven approach to improve prediction accuracy.

1.3 Objectives

1. To analyze the Framingham Heart Study dataset and identify significant features related to heart disease.
2. To build and evaluate machine learning models for heart disease prediction.
3. To compare the performance of various machine learning algorithms.
4. To provide insights into the factors contributing to heart disease risk.

1.4 Scope

The project focuses on using machine learning techniques to predict heart disease. It leverages the Framingham dataset, which contains relevant health and demographic information.

CHAPTER-2

TECHNOLOGIES USED

2.1 Development Environment

Google Colab: A cloud-based platform for writing and running Python code, providing access to GPUs and TPUs for faster computation.

2.2 Libraries and Tools

▼ IMPORT LIBRARIES



```
import warnings
warnings.filterwarnings('ignore')
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib.colors import ListedColormap
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import GridSearchCV, StratifiedKFold
from sklearn.metrics import classification_report, accuracy_score
from sklearn.linear_model import LogisticRegression
from imblearn.over_sampling import SMOTE
from sklearn.preprocessing import StandardScaler
```

**tools that were used in fig-1
are given below**

fig-1 Libraries

Python: A powerful programming language used for data analysis and machine learning.

Scikit-learn: A robust library for machine learning algorithms and model evaluation.

Pandas: For data manipulation and analysis.

NumPy: For numerical computations.

Matplotlib and Seaborn: For data visualization and exploratory data analysis.

2.3 Data Source

The Framingham Heart Study dataset, a publicly available resource for cardiovascular research, includes variables such as age, gender, cholesterol, smoking habits, and blood pressure.

CHAPTER-3

PLAN OF WORK

3.1 Project Phases

Data Collection:

Collecting data from sources like kaggle, GitHub etc.

Analysing which dataset is accurate and contains more number of observations.

Data Acquisition:

Understanding and analyzing the dataset.

We will analyse how our dataset looks like.

Info of dataset

```
✓ [803] heart_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4240 entries, 0 to 4239
Data columns (total 16 columns):
 #   Column                Non-Null Count  Dtype  
---  -
 0   male                  4240 non-null  int64  
 1   age                   4240 non-null  int64  
 2   education             4135 non-null  float64 
 3   currentSmoker         4240 non-null  int64  
 4   cigsPerDay            4211 non-null  float64 
 5   BPMeds                4187 non-null  float64 
 6   prevalentStroke       4240 non-null  int64  
 7   prevalentHyp          4240 non-null  int64  
 8   diabetes              4240 non-null  int64  
 9   totChol               4190 non-null  float64 
10   sysBP                 4240 non-null  float64 
11   diaBP                 4240 non-null  float64 
12   BMI                   4221 non-null  float64 
13   heartRate             4239 non-null  float64 
14   glucose               3852 non-null  float64 
15   TenYearCHD            4240 non-null  int64  
dtypes: float64(9), int64(7)
memory usage: 530.1 KB
```

this fig2 indicates how our
dataset looks like

fig2 Info Of Dataset

Data Filtering:

Remove outliers

Remove duplicate values

Remove the unwanted values

Remove the missing values as done in fig-3 and fig-4

```
[807] heart_data.isnull().sum()
```



	0
male	0
age	0
education	105
currentSmoker	0
cigsPerDay	29
BPMeds	53
prevalentStroke	0
prevalentHyp	0
diabetes	0
totChol	50
sysBP	0
diaBP	0
BMI	19
heartRate	1
glucose	388
TenYearCHD	0

fig3 Missing Values

The function `heart_data.isnull().sum()` is used to find missing values. as we can see in fig-3 for more understanding.

FILLING MISSING VALUES

```
[811] # Create a copy of heart_data and assign it to heart_data1
      heart_data1 = heart_data.copy()
```

```
[812]
      heart_data1['glucose'] = heart_data1['glucose'].fillna(heart_data1['glucose'].mean())
      heart_data1['education'] = heart_data1['education'].fillna(heart_data1['education'].mean())
      heart_data1['BPMeds'] = heart_data1['BPMeds'].fillna(heart_data1['BPMeds'].mean())
      heart_data1['totChol'] = heart_data1['totChol'].fillna(heart_data1['totChol'].mean())
      heart_data1['BMI'] = heart_data1['BMI'].fillna(heart_data1['BMI'].mean())
      heart_data1['sysBP'] = heart_data1['sysBP'].fillna(heart_data1['sysBP'].mean())
      heart_data1['cigsPerDay'] = heart_data1['cigsPerDay'].fillna(heart_data1['cigsPerDay'].mean())
```

fig4 Fill Missing Values

In fig-4 we can see how to fill these missing values. we will use `heart_data1['glucose'] = heart_data1['glucose'].fillna(heart_data1['glucose'].mean())` this function to fill missing values.

Data Transformation:

We will convert the datatype of the variables(male,cigPerDay, prevalentStroke diaBP etc) into suitable datatype which is suitable for our model.

Data Exploration:

We will observe:-

Trends

Relations

Patterns between our dataset to know our data better

```
✓ [826] heart_data1.groupby('male')['TenYearCHD'].mean().plot(kind='bar')
```

<Axes: xlabel='male'>

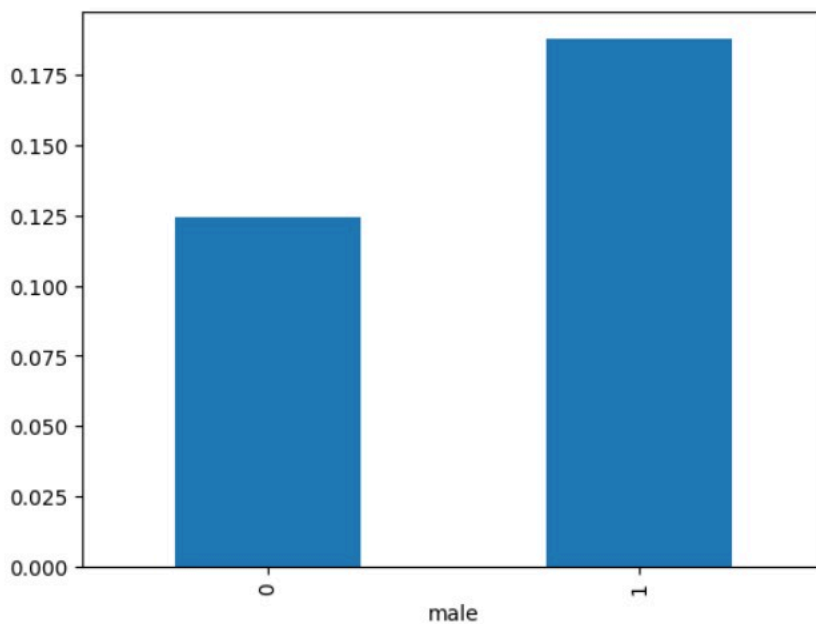


fig5 Bar Graph

We will use libraries matplotlib.pyplot,seaborn to plot.For example we will use

`heart_data1['TenYearCHD'].value_counts(normalize=True).plot(kind='bar')` to plot graph between 'male' and 'TenYearCHD' as shown in fig-5 and fig-6.

```
[833] sns.scatterplot(data=heart_data1,x='age',y='heartRate',hue='TenYearCHD')
```

<Axes: xlabel='age', ylabel='heartRate'>

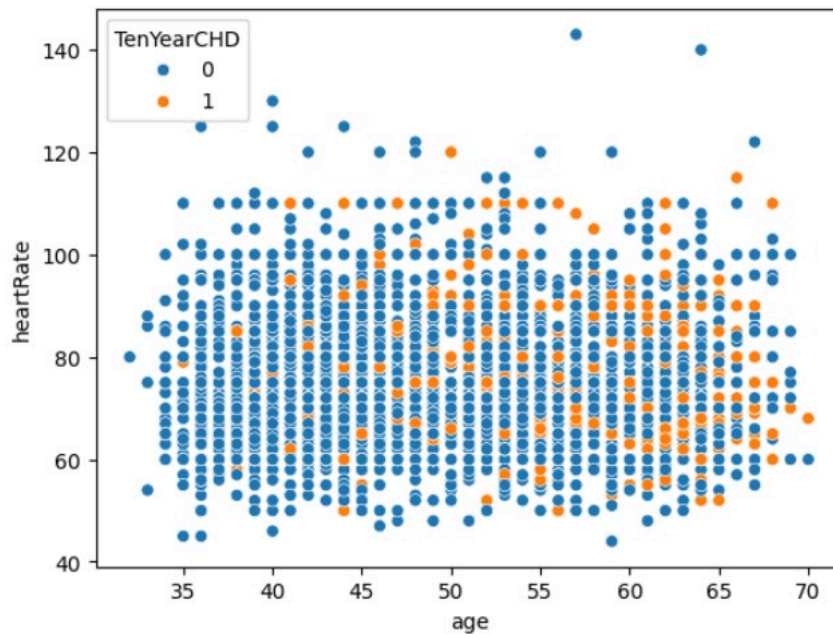


fig6 Scatterplot

Train and test:

We will split the dataset into two parts to train and test.

Generating model:

We will create our machine learning model using logistics regression.

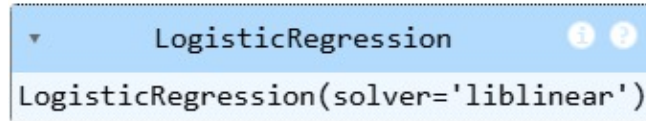
```
✓ [861] #divide in test and train  
0s X_train,X_test,Y_train,Y_test=train_test_split(X_os,Y_os,test_size=0.3,random_state=0)
```

fig7 Split(Train and Test)

As shown in fig-7 we can see we divided the dataset in train and test.


```
model_lr=LogisticRegression(solver='liblinear')
```

```
#use to train  
model_lr.fit(X_train_sc,Y_train)
```

A screenshot of a Jupyter Notebook cell. The cell has a blue header bar with the text 'LogisticRegression' and two circular icons on the right. Below the header, the text 'LogisticRegression(solver='liblinear')' is displayed in a monospaced font.

```
LogisticRegression  
LogisticRegression(solver='liblinear')
```

fig8 Logistic Regression

As shown in fig-8 we will use LogisticRegression to train our model.

Analysing accuracy:

We will analyse the accuracy of our model that how accurately it predicts.

```
#test the performance  
results=model_lr.score(X_test_sc,Y_test)  
results*100
```

```
69.0454124189064
```

```
results=model_lr.score(X_train_sc,Y_train)  
results*100
```

```
67.75923718712752
```

fig9 Accuracy Of Model

we can test the accuracy of our model as done in fig-9

3.2 Timeline

Week 1-2: Data collection and Data acquisition.

Week 3-4: Data filtration, Data transformation, and Data exploration.

Week 5-6: Training, testing, generating our model and testing accuracy of our model.

CHAPTER-4

METHODOLOGY

4.1 Data Preprocessing

Handling Missing Values: Imputation methods were used to replace missing data. we have already explained the concept of missing values in fig-3 and fig-4

Normalization: Scaling features to ensure uniformity. **as shown in fig-10**

scaling the values because there are so many variations in values

```
[863] sc_train=StandardScaler().fit(X_train)
      X_train_sc=sc_train.transform(X_train)
```

fig 10 Scaling

Encoding: Converting categorical variables into numerical formats.

Checking For Duplicate Values: We can check the duplicate rows that are present in our dataset as in fig-11

✓ how to check duplicate rows

```
[817] heart_data1.duplicated().sum()
      #0 means no duplicate rows
```

⇒ 0

fig-11 Check Duplicates

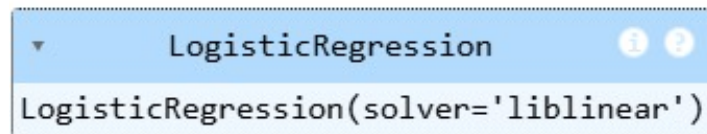
4.2 Model Development

Logistic Regression:

A baseline model for binary classification.

```
model_lr=LogisticRegression(solver='liblinear')
```

```
#use to train  
model_lr.fit(X_train_sc,Y_train)
```



The screenshot shows a Jupyter Notebook cell with a blue header bar containing a dropdown arrow, the text 'LogisticRegression', and information and help icons. The code area below the header contains the same two lines of Python code as the previous blocks: `LogisticRegression(solver='liblinear')` and `model_lr.fit(X_train_sc,Y_train)`.

```
LogisticRegression(solver='liblinear')
```

fig-12 Logistic Regression

As shown in fig-12 we will use Logistic Regression to train our model.

Confusion Matrix:

A confusion matrix is a table used to evaluate the performance of a classification model. It compares the predicted labels with the actual labels and provides a comprehensive summary of how well the model is performing. Here's how it works as in fig-13 and fig-14:

Structure of a Confusion Matrix

For a binary classification problem, the confusion matrix is a 2x2 table:

Actual/Predicted	Positive	Negative
Positive	True Positive (TP)	False Negative (FN)
Negative	False Positive (FP)	True Negative (TN)

- **True Positive (TP):** The model correctly predicted the positive class.
- **True Negative (TN):** The model correctly predicted the negative class.
- **False Positive (FP):** The model incorrectly predicted the positive class (Type I error).
- **False Negative (FN):** The model incorrectly predicted the negative class (Type II error).

fig-13 Confusion Matrix

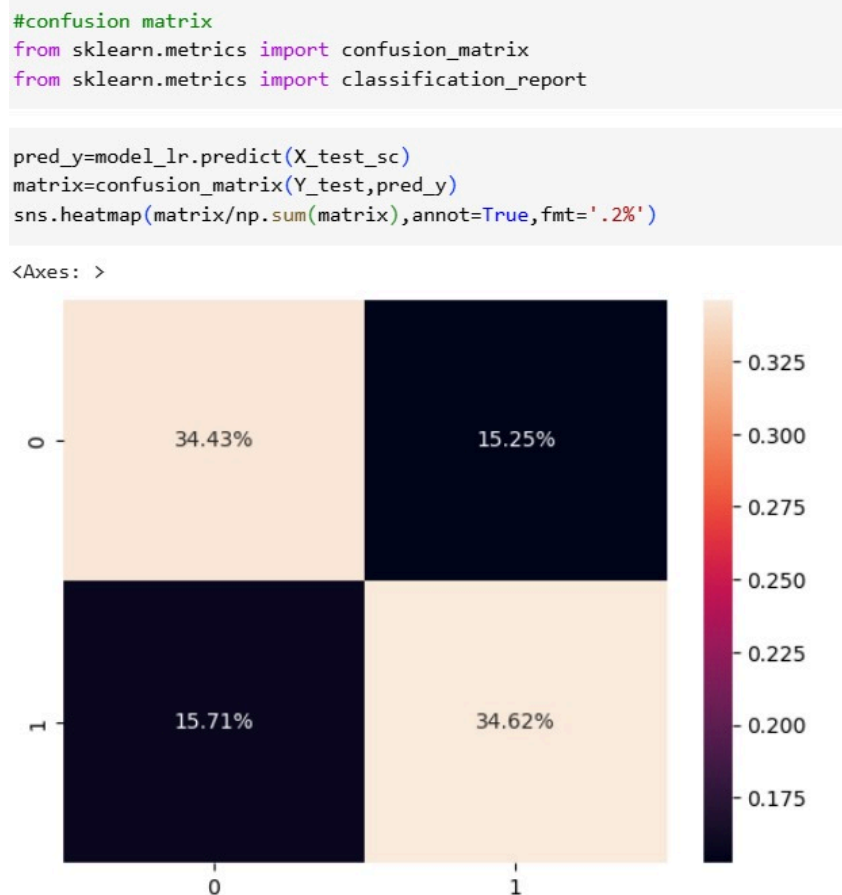


fig-14 Plot Confusion Matrix

4.3 Building a Predictive Model

After successfully training our machine learning model on the dataset, we can now proceed to use it for making predictions. The primary goal of the model is to classify whether an individual is at risk of developing heart disease based on their medical and lifestyle data.

As done in fig-15

✓ BUILDING A PREDICTIVE MODEL

```
895] #(male,age,educ,currsmoke,cigs,BPmed,prevstrole,prevhyp,diab,totchol,sysBP,diaBP,BMI,heartrate,glucose,CHD)
input_data = (17, 12, 300, 120, 100, 25, 1, 123, 1, 0, 0, 0, 2,2)

# Convert the tuple to a NumPy array
input_data_as_numpy_array = np.asarray(input_data)

# Reshape the array for prediction (1 row, multiple columns)
input_data_resaped = input_data_as_numpy_array.reshape(1, -1)
#reshape the numpy array as we are predicting for only one instance kyuki machine will think ki hum check kr rhe all values ke liye

# Assuming you have your StandardScaler object (sc) from training
# Scale the input data
#Instead of using 'sc', use the correct StandardScaler object
#that was fitted on the training data: sc_train or sc_test
input_data_scaled = sc_train.transform(input_data_resaped) # Use sc_train or sc_test

# Make the prediction using the scaled data
prediction = model_lr.predict(input_data_scaled)

print(prediction)

if prediction[0] == 0:
    print('THE PERSON DOES NOT HAVE A HEART DISEASE')
else:
    print('THE PERSON HAS HEART DISEASE')
```

[0]
THE PERSON DOES NOT HAVE A HEART DISEASE

fig15 Build Predective Model

CHAPTER-5

RESULTS & DISCUSSION

5.1 Results

In this project, we trained a Logistic Regression model to predict heart disease using the Framingham dataset. The following results were observed:

Testing Accuracy: 69.0454124189064%

Training Accuracy: 67.75923718712752%

To evaluate the model's performance, a confusion matrix was used. It provided insights into the classification results, showing the number of true positives, true negatives, false positives, and false negatives. This helped us understand the model's strengths and areas of improvement.

5.2 Discussion

Logistic Regression provides interpretability, making it suitable for real-world applications where understanding feature importance is crucial.

CHAPTER-6

CONCLUSION, LIMITATION AND FUTURE SCOPE

6.1 Conclusion

We applied various machine learning models to predict the likelihood of heart disease us. The dataset, consisting of various medical features such as age, cholesterol levels, blood pressure, and lifestyle factors, was preprocessed to handle missing values and scaled appropriately for the model training process.

Machine learning techniques demonstrate significant potential in predicting heart disease. Logistic Regression provided valuable insights into feature importance.

Machine learning models can provide reliable predictions for heart disease risk, helping in early detection and preventive healthcare measures.

Overall, machine learning has the potential to assist healthcare providers in predicting heart disease risk more accurately, offering significant value in personalized healthcare and early intervention strategies.

6.2 Limitations

Limited Dataset: Results may not generalize to other populations.

Feature Availability: Certain critical health attributes were unavailable in the dataset.

6.3 Future Scope

Dataset Expansion: Incorporate diverse datasets to improve model robustness.

Advanced Models: Explore deep learning techniques for prediction.

Real-time Application: Develop an application for real-time risk assessment.

REFERENCES

Framingham heart disease dataset from

<https://www.kaggle.com/datasets/aasheesh200/framingham-heart-study-dataset?resource=download>

Used

<https://colab.research.google.com/>

to write the code

Used

<https://openai.com/index/chatgpt/>

for suggestions

Watched

<https://youtu.be/tSBAag6lAQo?si=8vKOjXvlOKH-vCVs>

this video for understanding framework