Intelligent Personalised Health Assistance Web Application

A Project Report

Submitted to the APJ Abdul Kalam Technological University

in partial fulfillment of requirements for the award of degree

Bachelor of Technology

in

Computer Science and Engineering

by

Ashish M Nair(TCR19CS021)

Basil Pathrose(TCR19CS025)

Jonathan Nebu George(TCR19CS033)

Roshan T Joy(TCR19CS055)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GOVERNMENT ENGINEERING COLLEGE, THRISSUR

KERALA

December 2022

DEPT. OF COMPUTER SCIENCE & ENGINEERING GOVERNMENT ENGINEERING COLLEGE THRISSUR

2019 - 23

CERTIFICATE

This is to certify that the report entitled **Intelligent Personalised Health Assistance Web Application** submitted by **Ashish M Nair**(TCR19CS021),**Basil Pathrose** (TCR19CS025), **Jonathan Nebu George**(TCR19CS033) & **Roshan T Joy** (TCR19CS055) to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Computer Science and Engineering is a bonafide record of the project work carried out by him under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Ms. Princy Ann Thomas
(Project Guide)
Assistant Professor
Dept.of CSE
Government Engineering College
Thrissur

Ms. Helen K.J.
(Project Coordinator)
Associate Professor
Dept.of CSE
Government Engineering College
Thrissur

Dr. Shibily Joseph
Professor and Head
Dept.of CSE
Government Engineering College
Thrissur

DECLARATION

We hereby declare that the project report **Intelligent Personalised Health Assistance Web Application**, submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us under supervision of Ms. Princy Ann Thomas .

This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

We also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Ashish M Nair
Basil Pathrose
Jonathan Nebu George
Roshan T Joy

Thrissur
23-12-2022

Abstract

Nowadays, people are getting more and more health conscious and they indulge in activities like exercises, yoga and so on to improve fitness. To ensure that these activities are done correctly, the people are compelled to take the help of professional services. The professional services are often expensive and thus not accessible to most of the people. Thus arises the need of a system which can help such people at zero to low cost.The project intelligent personalized health assistant web application is a web application which acts as a health instructor to its users, It has mainly three features :- Yoga pose recognition and correction which helps the people in recognising the yoga pose that they are performing and helps the user in correcting it in case there is any error in posture, Early diagnosis of chronic disease which helps the early diagnosis of chronic diseases the user may have from the input of symptoms given by the user, Yoga pose recommendation based on various features like BMI, benefits and so on. Thus, this product is a socially relevant product as it helps its users in living a healthier lifestyle.

Acknowledgement

We take this opportunity to express my deepest sense of gratitude and sincere thanks to everyone who helped us to complete this work successfully. We express our sincere thanks to Dr. Shibily Joseph, Head of Department, Computer Science and Engineering, Government Engineering College, Thrissur for providing us with all the necessary facilities and support.

We would like to express my sincere gratitude to the Ms. Helen K.J., Department of Computer Science and Engineering, Government Engineering College, Thrissur for the support and co-operation.

We would like to place on record my sincere gratitude to our project guide Ms. Princy Ann Thomas, Assistant Professor, Computer Science and Engineering, Government Engineering College, Thrissur for the guidance and mentorship throughout this work.

Finally I thank my family, and friends who contributed to the succesful fulfilment of this seminar work.

Ashish M Nair

Basil Pathrose

Jonathan Nebu George

Roshan T Joy

Contents

List of Figures

List of Tables

List of Symbols

Ω Unit of Resistance

$\varepsilon^{'}$ Real part of dielectric constant

c Speed of light

λ Wavelength

δ Delta

Chapter 1

Introduction

1.1 Purpose

This report serves the purpose of providing a detailed description of the project Intelligent Personalized Health Assistance Web Application. It looks into the system architecture, decomposition of the system into modules, the data flow with the help of DFD's, the response of the system to external stimuli and the user interfaces involved. This document is intended to be of use to the developers and other stakeholders to gauge the status of the project's progress.

1.2 Motivation

Health is of major importance to all people and yoga is a popular practice that is followed by people to keep their health in check. Since, a lot of people are quite new to yoga, it is likely they'll run into trouble when they try out various yoga poses and hence would find the web application quite useful in addressing the trouble they face and to correct their mistakes while they learn the poses. The popularity of the practice of yoga has boosted in recent times. This makes the project a good business model with promising prospects.

1.3 Scope

The project is intended to help users having trouble with performing yoga poses to be guided so that they can easily correct the mistakes that they make in real time. Furthermore, the users are able to find out what diseases they're at risk for based on their current health status which would be fed as the input. Moreover, the application will be designed to personalize recommendations to users based on the health status that they provide of themselves.

1.4 Overview

The document comprises of an introduction that describes the purpose, scope, motivation of the project at hand. We then move on to the system design which intends to define the overall architecture, decomposition,and flow of data within the system. A comparison is made between several architectures that are proposed and the best one is selected with proper reasoning. We move to the conclusion, where the status of project progress is analyzed and the timeline of the work to be completed is estimated for the following phase

Chapter 2

Sample section

There are various approaches to yoga pose correction and recognition.In one of the approach,Kinect device is used which forms a human skeleton in 3D space which directly gives us the information about the joints of the body but the problem with this approach is that the kinect devices are expensive.To avoid this problem, tf pose estimation algorithm are used.It helps to create a skeleton drawing of human body from images and deduce the coordinates of joints from it, which are used to calculate the angles between joints using appropriate formulas and then passed as a feature in the model.The features are passed into the classification model like KNN, Naive Bayes etc. and their accuracy are compared so as to get the optimum method to implement this model.The flowchart of the method to implement this model is given in Fig 3.1..

2.1 Basic idea to implement the model

2.1.1 Creating skeleton

First of all, the brightness of every image is increased, let's say 2 for uniformity and the image are resized to 500x500 resolution to best fit the pose estimation algorithm for an accurate result.The tf-pose estimation algorithm is used to create a skeleton drawing of the person doing yoga in which joints are joined by skeleton sticks.(Fig. 3.2.)

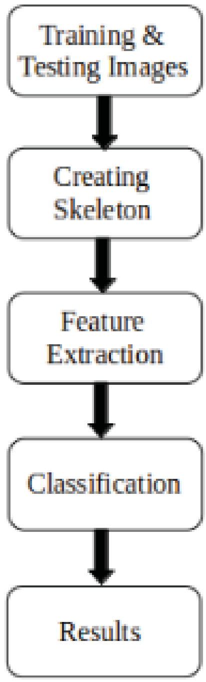


Figure 2.1: Flowchart

2.1.2 Feature Extraction

In this step,using tf-pose estimation algorithm the coordinates of the joints are extracted which is used as feature to train the model.The formula for calculating angle is give in figure 3.3 and 3.4.

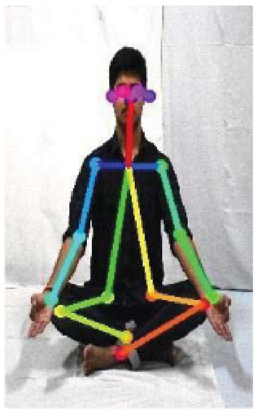
2.1.3 Classification

In the final stage the features are stored in CSV and different classification models are applied and then compared so as to find the optimum classification model.The features passed are usually angles calculated by using appropriate formulas from the coordinates of joints.Here, in this seminar,mainly six classification models are used(Logistic Regression, Random Forest, Decision Tree, SVM, Naive Bayes, KNN).Then, their accuracy score are calculated to find out the most accurate model.

2.2 Classification Algorithms

The following classification algorithms are discussed in this seminar:-

(a) (b)



(c)

Figure 2.2: Skeleton Drawing

$$a^2 = b^2 + c^2 - 2bc\cos A$$

Where,

a = Distance between point p1 and p2
b = Distance between point p2 and p3
c = Distance between point p1 and p3
A = Angle made by point p2

Figure 2.3: Formula for calculating angle

- Logistic regression

- Random Forest

$a = \sqrt{(x1 - x2)2 + (y2 - y1)2}$

Where,

(x1,y1) is the coordinate of point p1 (x2,y2) is th coordinate of point p2

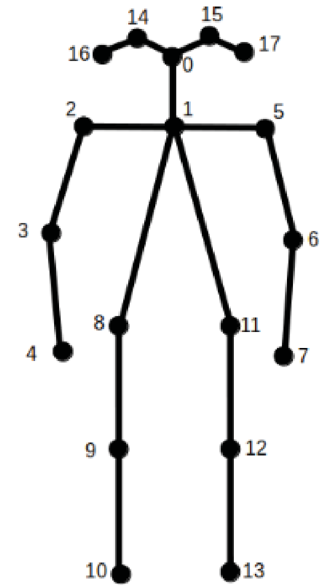


Figure 2.4: Skeleton Drawing

- SVM
- Decision Tree
- Naive Bayes
- KNN

2.2.1 Logistic Regression

Logistic regression is a supervised ML algorithm used for predicting categorical dependent variable. It is similar to linear regression, with the difference that linear Regression is used for solving regression problems, whereas Logistic regression is used for solving the classification problems. Its outcome must be a categorical or discrete value. It can be either Yes or No, 0 or 1, true or False, etc. but instead of giving the exact value as 0 and 1, it gives the probabilistic values which lie between 0 and 1.

$$Logistic function = \frac{1}{1 + e^{-x}}$$

In Logistic regression, instead of fitting a regression line like in case of linear regression, we fit an sigmoid logistic function in logistic regression as shown in the figure 3.5 .

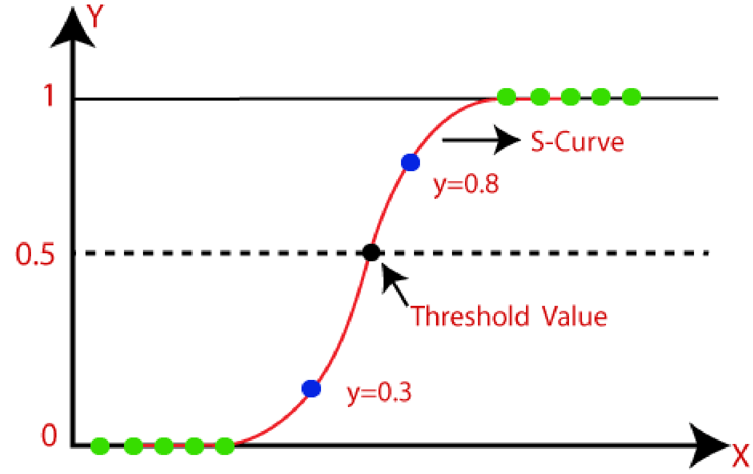


Figure 2.5: Logistic Function

Assumptions for Logistic Regression are:-

- The dependent variable must be categorical.
- The independent variable should not have multi-collinearity.

Formula for Logistic Regression is given in Fig 3.6 .

$$log\left[\frac{y}{1-y}\right] = b_0 + b_1x_1 + b_2x_2 + b_3x_3 + \cdots + b_nx_n$$

Figure 2.6: Formula for Logistic Function

where x1,x2..xn are dependent variables.

The various types of Logistic Regression are:-

- Binomial:-There are only two types of dependent variable.E.g.:-0 or 1

- Multinomial:-There are three or more unordered types of dependent variable.E.g.:-Cat,Dog,Sheep
- Ordinal:-There are three or more ordered types of dependent variable.E.g.:-Low,Medium,High.

2.2.2 Decision Trees

Decision Tree is a supervised machine learning algorithm used for both classification as well as regression.They usually mimic human thinking ability while making a decision, so it is easy to understand. In decision tree, internal nodes represent features of the dataset, branches represent rules while leaf nodes represent the outcome.In order to build a tree, we use the CART algorithm (Classification and Regression Tree algorithm. The general structure of a decision tree is given in figure 3.7. An example of decision tree is given in figure 3.8.

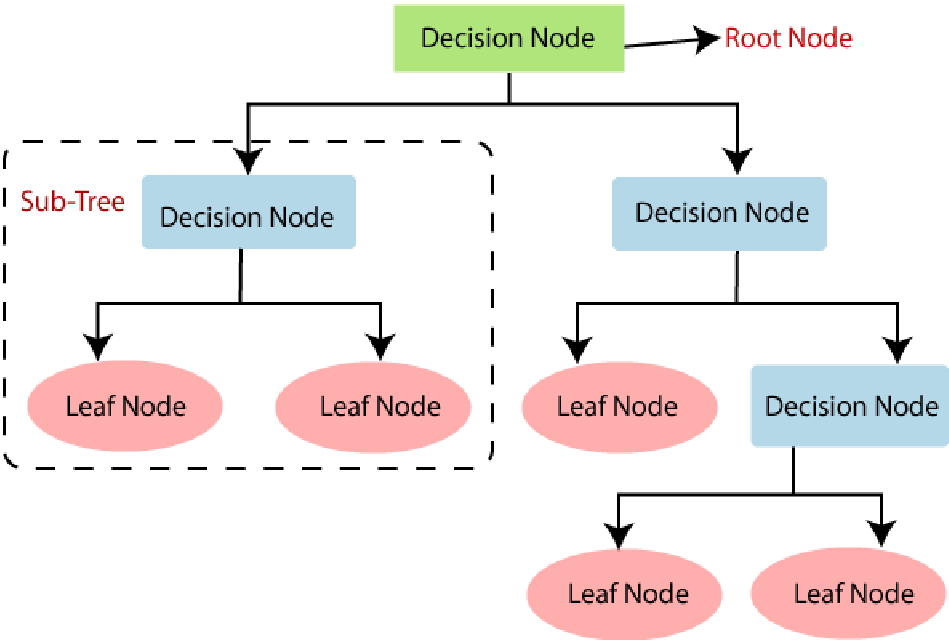


Figure 2.7: General Structure of Decision Tree

A decision tree to decide whether to accept a job offer or not.

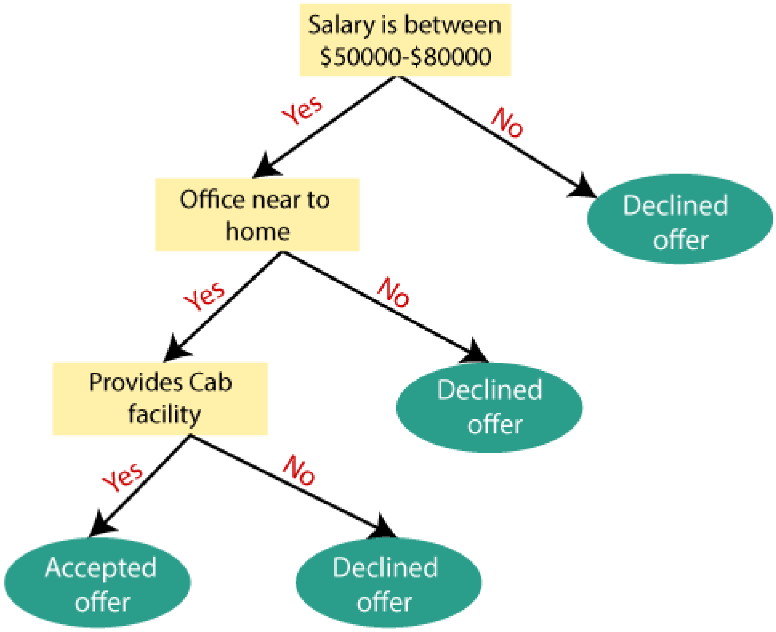


Figure 2.8: Example of Decision Tree

2.2.3 Random Forest

RF is a supervised machine learning algorithm based on the concept of ensemble learning (Combining multiple classifiers to solve a complex problem and improve performance). It contains a number of decision trees on various subsets of the given dataset and then takes the average or majority voting of prediction of each decision trees to improve the predictive accuracy of that dataset. Thus, the greater number of decision trees results in higher accuracy and prevents overfitting. It is capable of performing both classification and regression tasks.The working of RF is given in figure 3.9 and an example of RF is given in figure 3.10. In Fig. 3.10, we have a fruit basket as node and dataset is divided into various subsets in which decision tree classification is applied and then we do majority voting so as to get the result (apple in this case).

2.2.4 Naive Bayes

It is a supervised machine learning algorithm based on Bayes theorem, used for classification problems. It predicts on the basis of the probability of an object and assumes that the features are independent to each other.The following formula is used

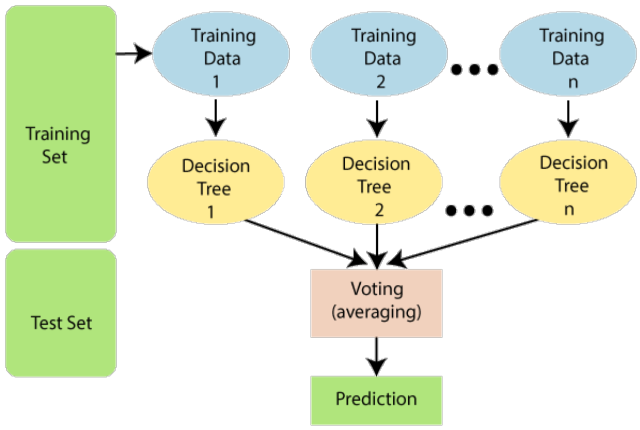


Figure 2.9: Working of Random Forest

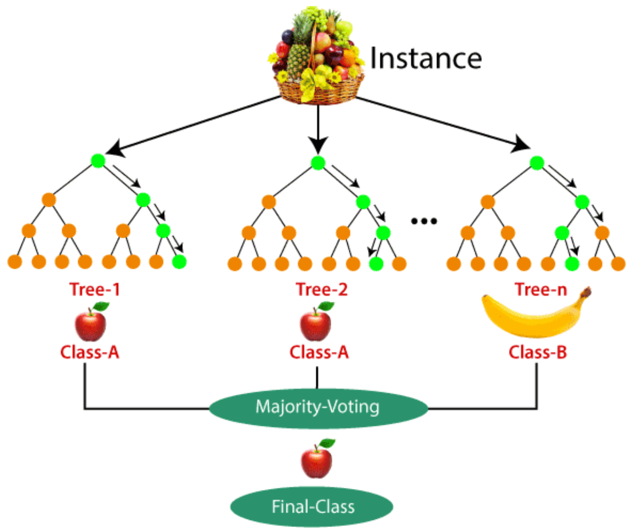


Figure 2.10: Example of Random Forest

in Naive Bayes classification to find posterior probability (Fig.3.11). The Naive Bayes classification have trhe following assumptions:-

- The features are independent.
- The features have equal contribution to the outcome.

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

Figure 2.11: Bayes Theorem

2.2.5 SVM

SVM is a supervised machine learning algorithm primarily used for classification. It is used to find a decision boundary hyperplane for classification of data so that new data point can be put in correct category. The hyperplane is found with the help of extreme data point close to the hyperplane called support vectors. For an n dimensional space,the hyperplane that we get is n-1 dimensional. SVM are of two types:-Linear and Non-Linear SVM.

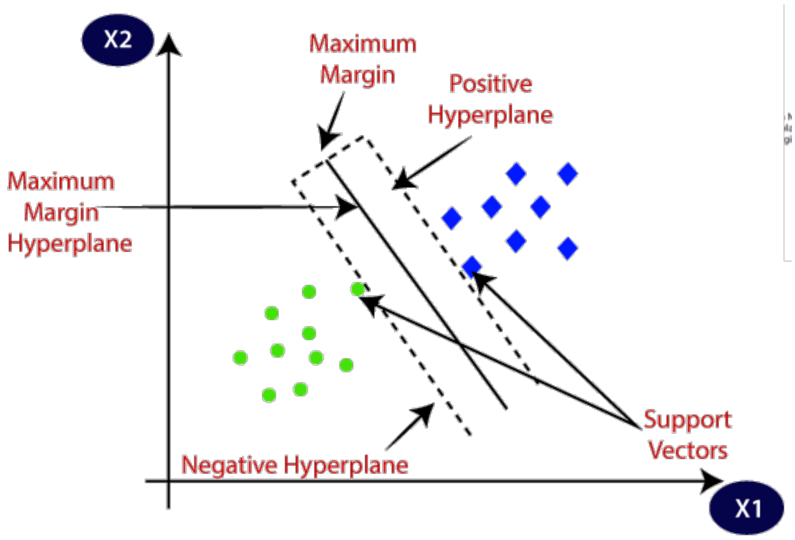


Figure 2.12: Support Vector Machine

2.2.6 KNN

KNN is one of the simplest supervised machine learning algorithm which classifies a new data point based on similarity. It can be used to solve both classification and regression problems. If majority of k samples of a datapoint belong to a particular class, then the datapoint also belong to the same class. The KNN algorithm assumes that similar things exist in close proximity. In other words, similar things are near to each other. It is also called lazy learner algorithm as it learns from the data at the time of classification. It is more effective when the training data is large.The figure 3.13 shows the how KNN classification takes place.

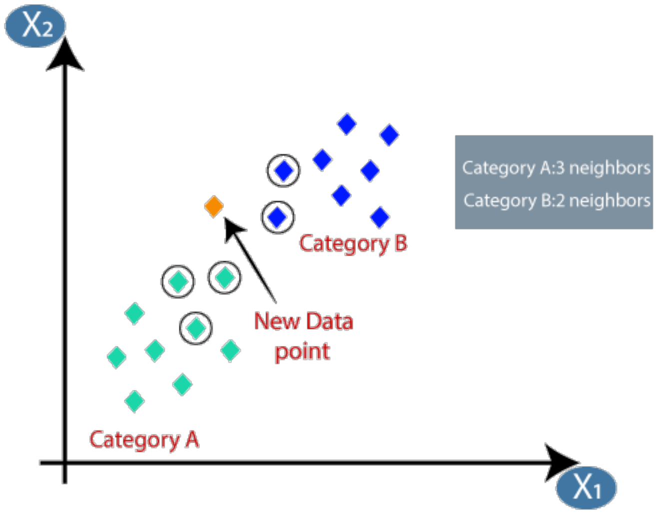


Figure 2.13: K-Nearest Neighbors

2.3 Performance Metrics in Classification

There are various ways to check the performance of the machine learning models like Accuracy,F1 score,Precision,Recall etc.In this seminar we will compare the various models based on accuracy.

2.3.1 Accuracy

Accuracy is the used to measure how much correct predictions is made by the classifier. We can define accuracy as the ratio of the number of correct predictions and the total number of predictions.

$$\textit{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

2.3.2 Precision

Precision is used to find the correct number of positive prediction from total positive predictions. Precision is useful in the cases where False Positive is a higher concern than False Negatives. The importance of Precision is in music or video recommendation systems, e-commerce websites, etc. where wrong results could lead to customer churn and this could be harmful to the business.Precision for a label is defined as the number of true positives divided by the number of predicted positives.

$$\textbf{\textit{Precision}} = \frac{TP}{TP + FP}$$

2.3.3 Recall

Recall explains how many of the actual positive cases we were able to predict correctly with our model. The formula for the TPR(True Positive Rate) in the ROC curve is same as recall. It is useful when FN is of higher concern than FP. Recall for a label is defined as the number of true positives divided by the total number of actual positives.

$$\textbf{\textit{Recall}} = \frac{TP}{TP + FN}$$

2.3.4 F1-score

The F1-score is the harmonic mean of precision and recall. It is used to compare the performance of two classifiers. Suppose that classifier A has a higher recall, and classifier B has higher precision. In this case, the F1-scores for both the classifiers can be used to determine which one produces better results.

$$\textbf{\textit{F - measure}} = \frac{2 * Recall * Precision}{Recall + Precision}$$

Chapter 3

Literature Review

3.1 e-Health Monitoring System with Diet and Fitness Recommendation using Machine Learning

Nowadays, more individuals are being diagnosed with diseases that are becoming chronic due to not following the proper diet, not doing proper exercise regularly, or not giving proper attention to the diseases because of busy schedules. Hence, we propose a system that aims at improving the health of the patients suffering from various diseases by recommending them healthier diet and exercise plans by analyzing and monitoring health parameters and the values from their latest reports related to the disease. We considered patients suffering from either Diabetes or Blood pressure or Thyroid. Our System can be essentially useful for the doctors to recommend diet and exercise based on their latest reports and personal health details. For this, we have broadly classified our system into 2 modules: 1. Health Monitoring, 2. Diet Exercise Recommendation. In the Health Monitoring module, the system would suggest follow-up sessions until the reports come normal. For the Diet and Exercise Recommendation module, the algorithm that is used is a Decision tree for classification. To be precise, C4.5 is used to given recommendations of diet and exercise. A C4.5 Decision tree will help recommend and determine if a particular food item and exercise should be given to a particular individual or not with respect to our customized datasets.

3.2 Constructing a Diet Recommendation System Based on Fuzzy Rules and Knapsack Method

Many people suffer from three chronic diseases(diabetes, hypertension, cholesterol), and they often use search engine to collect related information. However, most of dietary information on the networks is not convenient for users to collect about the diet recommendations. In this paper, a diet recommendation system is suggested which can recommend a rational diet for users. We design a diet recommendation system which has the expert knowledge of three high chronic diseases. We use Protégé to establish ontology and OWL DL to construct the structure of knowledge. The system uses fuzzy logic as a guide prior to inference. According to the patient's health information, the system infers daily calories requirement, and then use JENA inference device and JENA rule format to build our knowledge of the rules. The Knapsack-like algorithm is used to recommend suitable foods for users. The system was evaluated by nutritionists to prove it is effective.

3.3 Realizing an efficient iomt-assisted patient diet recommendation system through machine learning model

Recent studies have shown that robust diets recommended to patients by Dietician or an Artificial Intelligent automated medical diet based cloud system can increase longevity, protect against further disease, and improve the overall quality of life. However, medical personnel are yet to fully understand patient-dietician's rationale of recommender system. This paper proposes a deep learning solution for health base medical dataset that automatically detects which food should be given to which patient base on the disease and other features like age, gender, weight, calories, protein, fat, sodium, fiber, cholesterol. This research framework is focused on implementing both machine and deep learning algorithms like, logistic regression, naive bayes, Recurrent Neural Network (RNN), Multilayer Perceptron (MLP), Gated Recurrent Units (GRU), and Long Short-Term Memory (LSTM).The medical dataset collected through the internet and hospitals consists of 30 patient's data with 13 features of different diseases and 1000 products. Product section has 8 features set. The features of these IoMT data were analyzed and further encoded before applying deep and machine and learning-based protocols.

3.4 Implementation of machine learning technique for identification of yoga poses

In this paper the comparative study to implement the recognition of yoga posture is done. A dataset of 5500 images of ten different yoga pose is taken and tf-pose estimation algorithm is used to draw a skeleton drawing of human body through which we get coordinates of joints and the angles of joints of skeleton are calculated using appropriate formulas which are used as features in classification models to implement the model to recognise the yoga poses.This work is related to our seminar as it conducts a comparative study of various methods to implement yoga pose recognition and find optimum method to implement it.In this model RF was found to have highest accuracy among the six other classification algorithm(Logistic Regression, KNN, SVM, Decision Tree Naive Bayes, RF).

3.5 Yoga pose detection and classification using machine learning techniques

In this work, a dataset consisting of 1000 images of six different yoga poses is taken. The MediaPipe pose estimation model is used to find the coordinates of the joints, from which the angle between the joints are calculated. These angles are then passed as feature to the classification algorithms. In this paper, we use five different classification algorithms(LR, KNN, RF, DT, Naive Bayes) and compare them based on their performance. This paper is related to seminar as it helps in familiarizing the use of pose estimation algorithm and gives us the comparison of performance of various classification algorithms.

3.6 Classification of yoga pose using machine learning techniques

In this work, the dataset of images of sun salutation is taken. The tf-pose estimation algorithm is used to create the skeleton drawing of the image such that we get the coordinates of joints . They are then used for calculating angles between the joints using appropriate formulas which are then passed as features to the classification algorithm. Here, in this paper, four classification algorithm (LR, RF, SVM, KNN) are used and compared based on their performance measure and the model with the best performance is selected.

3.7 Heart Disease Prediction using Machine Learning and Deep Learning Algorithms

Heart disease rates increase gradually depending on a person's lifestyle, for example habits like smoking, high fat injection, lacking physical mobility or lack of exercise can increase the chances of a person contracting a heart disease This work predicts heart disease using various algorithms such as Support Vector Machine, Random Forest and KNN etc and find the best performing model among them.

3.8 A Comparison Based Study of Supervised ML Algorithms for Prediction of Heart Disease

A large amount of data is available to us which can not only provide us with the reliable way of predicting heart disease, but it would also reduce the pressure on the medical professionals. These algorithms can estimate the likelihood of a person developing heart disease based on factors such as age, gender, blood pressure, stress, and so on. We used a data set with 1025 samples of data including 13 attributes in our research. A comparative study of heart disease prediction using supervised algorithms such as Logistic Regression, Decision Tree, SVM, Naive Bayes, Random Forest, and KNN algorithms to predict heart disease, with the purpose of determining which method is the most reliable. In this study, Decision Tree came out at the top with 98.53% accuracy.

3.9 Disease Prediction using Machine Learning algorithms

Disease Prediction using Machine Learning is the system that is used to predict the diseases from the symptoms which are given by the patients or any user. This work deals with the detection of diseases from various symptoms using various supervised machine learning algorithms such as Decision Trees, Random Forest and Naive Bayes Classifier.

3.10 Alzheimer's Disease Detection Using Comprehensive Analysis of Timed Up andGo Test via Kinect V.2 Camera and Machine Learning

The paper focuses on the case study of prediction of AD by analysis of the TUG test. The study made use of a Kinect V@ camera for feature extraction during the TUG assessments of the participants involved. it mainly focuses on the use of Support Vector machine classifier for discriminating AD patients from Healthy Control(HC). The use of Cross validation techniques for selecting the right values for the hyperparameters involved in SVM is looked into in the paper.

3.11 Deep Learning for Accelerometric Data Assessment and Ataxic Gait Monitoring

The use of accelerometric data is signified in the paper for the purpose of discriminating between ataxic and normal gait. The methodology is based on analyzing the frequency componenets of accelerometric signals which are recorded as specific body positions at a predefined sampling frequency. The deep learning model is used for the purpose of classification. It further compares the results with other standard classification models such as SVM and the Naive Bayes Method. It analyzes the accuracy obtained for classification for different body positions where the sensors are placed which includes the spine position as well as the right foot position and compares the results.

3.12 Dual-Task Gait Assessment and Machine Learning for Early-detection of Cognitive Decline

The use of Dual task gait assessment in assessing the cognitive function of a person is a popular technique. The paper focuses on how SVM as well as gradient tree boosting can be used to differentiate subjects with Mild Cognitive Impairment(MCI) from Healthy Control(HC). The subjects involved in the study are required to do a separate independent task while walking during the assessment. Feature selection process, normalization of feature values are looked into in the paper mentioned.

AUTHOR AND YEAR	PROBLEM AREA	METHODS USED	PERFORMANCE MEASURES
Yash Agarwal ,Yash Shahet et.al. Year:-2020	Implementation of machine learning for identification for yoga poses.	The coordinates of the joints of the human body and angle are passed as features into the model and different classifiers(KNN,RF,SVM,Decision tree) are used and compared.	Accuracy of KNN is 98.26%,RF is 99.26%,SVM is 87.91%,SVM is 93.58%,Decision Tree is 97.71%).
Ajay Choudhari,Omkae Dalvi et.al. Year:-2021	Real-Time yoga pose correction using deep learning methods	The coordinates of the joints of the human body are passed as features using openpose and CNN classifier is used.	Accuracy is 95% F1 score is 72%
Varsha Bhosale,Pranjal Nandeshwar Year:-2021	Yoga pose detection and correction using posenet and KNN	The critical points in human body are identified using PoseNet and KNN classifier is used	Accuracy Rates:98.51% for a dataset of 1578 images of five different yoga poses

Table 3.1: Accuracy, Precision, Recall, F1 score and Error Values of various supervised ML algorithms on heart disease prediction.

AUTHOR AND YEAR	PROBLEM AREA	METHODS USED	PERFORMANCE MEASURES
Nidhi et al., 2022	Disease Prediction using Machine Learning	Random Forest, Decision Trees, Naive Bayes	Disease Prediction application is implemented successfully using RF,DT and NB using two methods.
Kuldeep et al.,	Heart Disease Prediction using Machine Learning and Deep Learning algorithms	DT, LR, RF, SVM, KNN	Accuracy: LR is 88.52, SVM is 87, NB is 80.33, DT is 85.25, (k = 14)KNN is 78.69, RF is 88.52).
Seifallahi et al.	Alzheimer's Detection using TUG test and ML.	SVM used for classification	5 Fold cross validation Accuracy is 97.75% Accuracy is 98.68% F-score, p-values
Prochazka et al	Deep learning for classification of ataxic gait	Deep learning	Confusion matrix for spine and foot positions Accuracy is 95.8%
Rung-Ching Chen,2013	Constructing a Diet Recommendation System Based on Fuzzy Rules and Knapsack Method	Fuzzy Leaning, Knapsack,OWL, JENA	Accuracy
CELESTINE IWEND,2019	Realizing an Efficient IoMT-Assisted Patient Diet Recommendation System Through ML Model	RNN, LSTM, GRU, MLP, naive bayes and logistic regression	Accuracy Precsion Recall Measure
Divya Mogaveera,2021	e-Health Monitoring System with Diet and Fitness Recommendation using ML.	ID3,C4.5	Accuracy Pruning Splitting criteria

Table 3.2: Accuracy, Precision, Recall, F1 score and Error Values of various supervised ML algorithms on heart disease prediction.

AUTHOR AND YEAR	PROBLEM AREA	METHODS USED	PERFORMANCE MEASURES
Chohan, D. K., & Dobhal, D. C.	A Comparison Based Study of Supervised ML Algorithms for Prediction of heart disease	Random Forest, Decision Trees, Naive Bayes	Accuracy of DT=98.53% , RF=87.80% , LR=78.53% ,SVM=80.48% ,NB=80.00% .
Kuldeep et al.,	Heart Disease Prediction using Machine Learning and Deep Learning algorithms	DT, LR, RF, SVM, KNN	Accuracy: LR is 88.52, SVM is 87, NB is 80.33, DT is 85.25, (k = 14)KNN is 78.69, RF is 88.52).
Lillian et al	Dual Task Gait assessment and Machine Learning for Early-detection of Cognitive Decline	SVM, Gradient Tree Boosting(GTB)	Accuracy(GTB - 72%, SVM - 77.17%)

Table 3.3: Accuracy, Precision, Recall, F1 score and Error Values of various supervised ML algorithms on heart disease prediction.

Chapter 4

System Development

4.1 Data Design

4.1.1 DataFlow Diagram

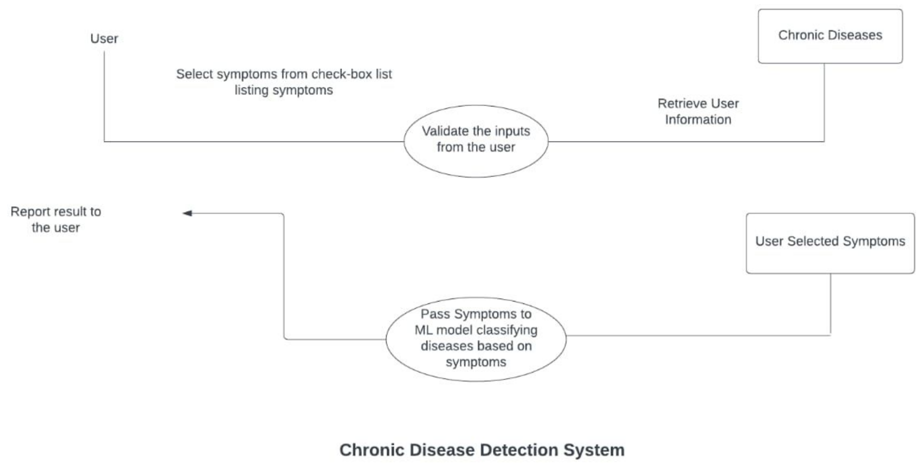


Figure 4.1: DFD for chronic disease detection from symptoms

4.1.2 Data Description

The registered user credentials are stored in the database. The user first login into their account using their account credentials which are matched against stored credentials in the database, if matched successfully the user can login to their account otherwise an error message is displayed. A screen displaying the functions that can

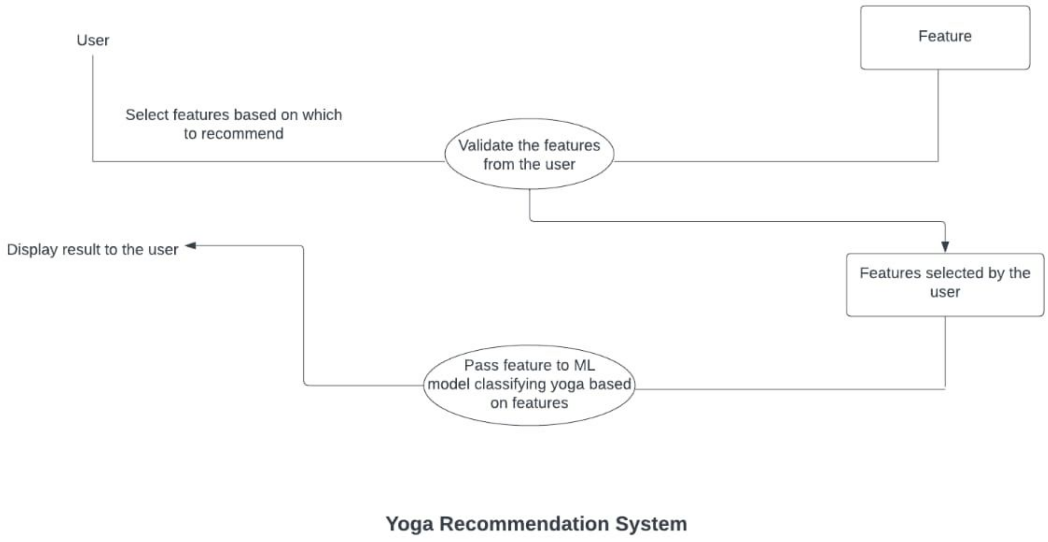


Figure 4.2: DFD for yoga recommendation for various features

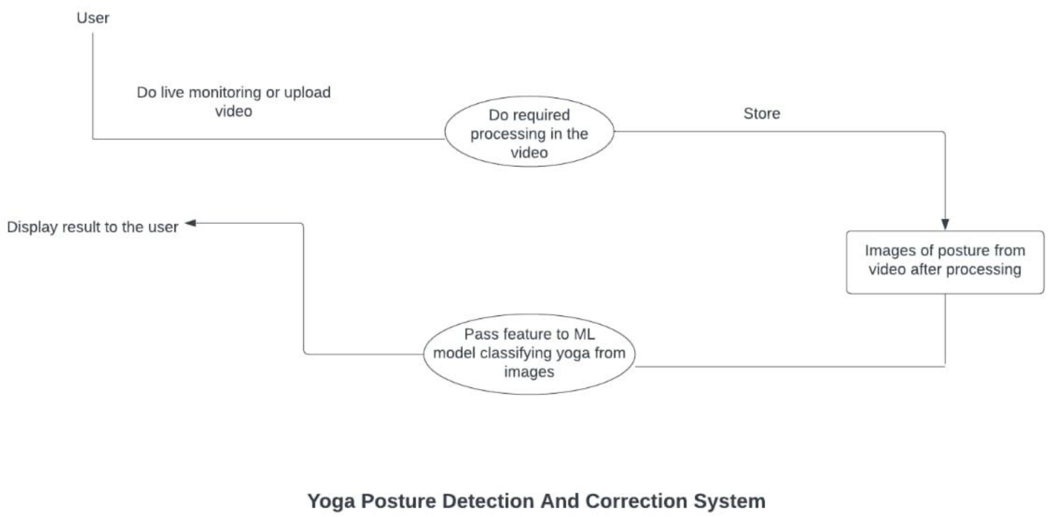


Figure 4.3: DFD for yoga posture detection and correction system

be performed(yoga pose recognition and correction, chronic disease diagnosis, yoga pose recommendation) is displayed from which the user selects the required function. In yoga pose recognition and correction, the user uploads the video of user doing yoga from local storage to frontend or video is captured realtime. This video is passed to the pre-trained model. The classification by using single frame images from video takes place in the model.The yoga pose recognised is stored in database as well as

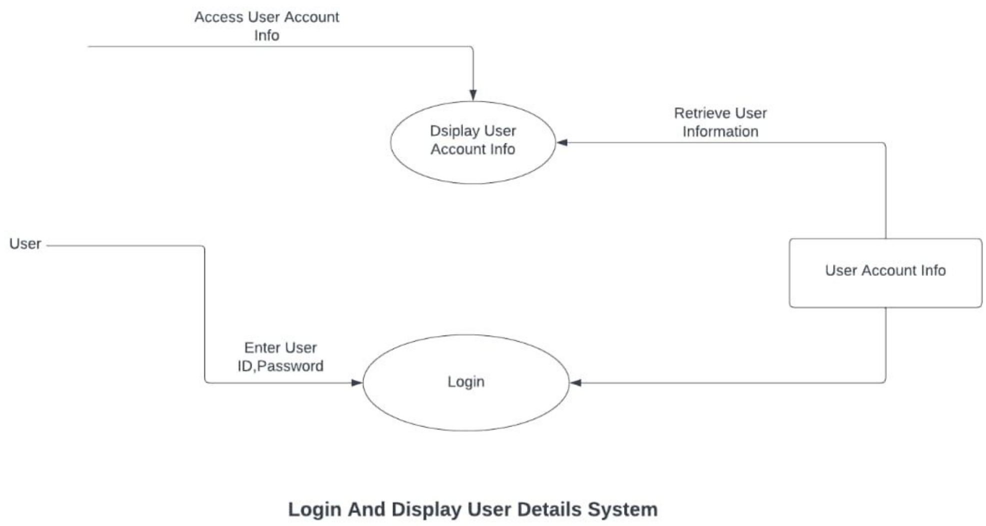


Figure 4.4: DFD for Login Subsystem

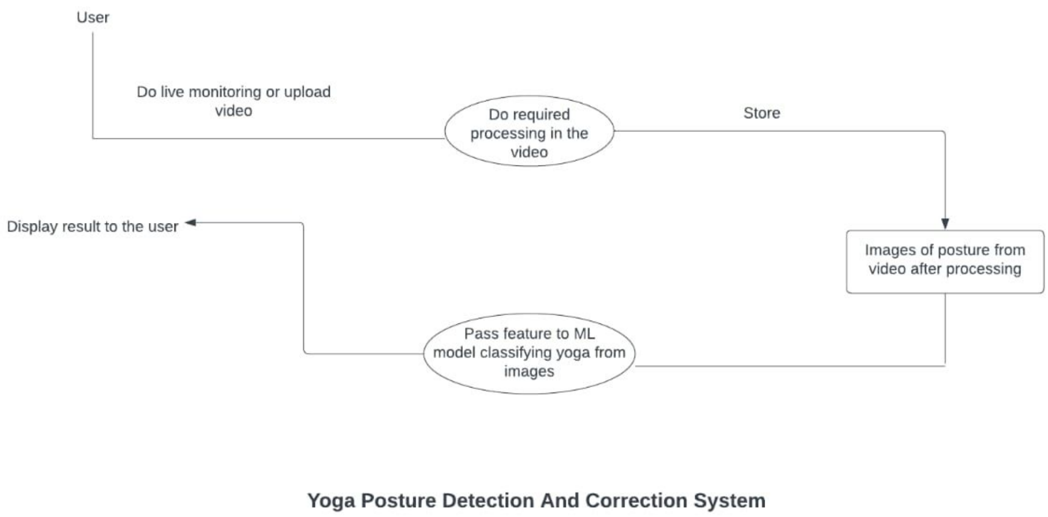


Figure 4.5: DFD for yoga posture detection and correction system

the error rate is also stored in the database.The output is then displayed from the database. Yoga pose recommendation is similar to chronic disease diagnosis with the difference that we select features instead of symptoms.In chronic disease detection, the user select symptoms from the list of common symptoms which are stored in the database.These symptoms are then passed to a pre-trained model from which we can predict disease which is cross checked with the actual disease result (if known by the patient). The predicted disease, actual disease (if known by the patient) along with symptoms is stored in the database. The predicted disease from the database is retrieved and displayed.

4.1.3 Data Dictionary

User

The account credentials are first given by the user to login to their account.Then, for yoga pose recognition and correction, the user uploads the video of the user doing yoga poses or upload video real time to the system. For chronic disease detection, the symptoms experienced by the user are given to the system. For yoga pose recommendation, the features required by the user are selected and are given to the system.

System

For yoga pose recognition and correction, the system accepts the video as input and then do suitable preprocessing to convert it into single frame images, which are then fed to ML model from which we get name of the yoga pose done as well the error which is the output , which is given to the database. For chronic disease detection, the symptoms given by the user as input is passed to ML model from which we can get the predicted chronic disease as output.If the disease experienced by the the user is known to the user it can also be given as input. The output of the system as well as actual disease(if known) are then given to the database.For yoga pose recommendation, the features given by the user as input is passed to ML model from which we can get the recommended yoga posture as output.The output of the system is then given to the database.

Database

Accepts the user credentials, yoga posture name as well as error detected by the system,.It also stores the symptoms, predicted disease as well as actual disease(if known).The yoga pose recommended by the system is also stored in the database.

4.2 System Architecture

4.2.1 Architectural Design

The architecture of the system as well as the relation between the modules is given in Figure 1. In this system, there are mainly two subsystems (functionalities which uses ML model to predict result and login subsystem). The subsystem which contains functionalities which uses ML model to predict results are yoga pose recognition and correction, early diagnosis of chronic diseases from symptoms, yoga posture recommendation based on various features like BMI, benefits and so on. As shown in the figure, first of all the ML model is trained using the dataset in the jupyter notebook which is then stored using pickle as .pkl file. Pickle is used to save ML models and helps us to minimize retraining of the model. Now the features by the user are stored into the server with the help flask, which are then passed into the corresponding pickle model and the result are predicted. These results are then passed and displayed to frontend of the web application which is built using HTML and CSS using flask. The other subsystem present in this system is login system. In this subsystem, there are mainly three functionalities:- a)Register (Used to register an account in our database.) b)Login (Used to login using correct credentials) c)Change Password(Used for changing the account password credentials in the database). The system is mainly implemented with the help of google authentication. The user information methods as well as mean of verification are stored in the database. When using the Register function, the account credentials are stored in the database and during login the matching of input with the user information in the database takes place. If matched sucessfully , then the user can login into the account.The change password functionality is used to change the password information for a particular user in the database.

4.2.2 Decomposition Description

- **Functionalities which uses ML model to predict results**

  1. Yoga pose detection and correction system:- The DFD for this module is given in Figure 2. As shown in the diagram, the video of the user doing the yoga

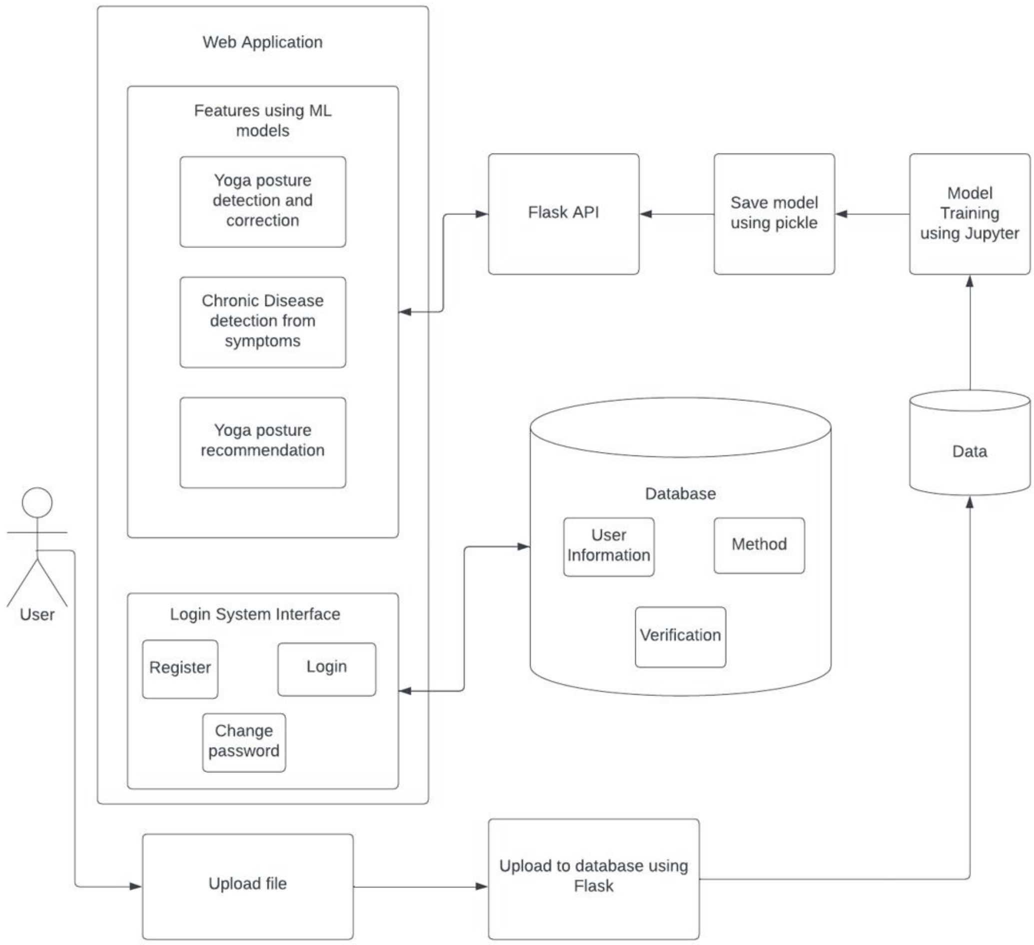


Figure 4.6: System Architecture

pose is uploaded or captured realtime in the frontend. The video is then passed to the server using flask, in which the required preprocessing takes place and frame images from video after preprocessing is obtained. These are then passed into the .pkl file which contains the ML model for classifying yoga from the images from which the result is obtained, and then the result are passed to the frontend using flask and then displayed to the user.

2.Chronic Disease Detection:- The DFD for this module is given in Figure 3. As shown in the diagram, the list of common symptoms are listed in the frontend of the website. The symptoms experienced by the user are selected and are then passed to the server using flask.The user selected symptoms are then passed into the .pkl file which contains the ML model for predicting chronic diseases based on symptoms from which the result is obtained, and then the result are passed to the frontend using flask and then displayed to the user.

3. Yoga Pose Recommendation:- The DFD for this module is given in Figure 4. As shown in the diagram, the list of features (benefits, BMI etc.) are listed in the frontend of the website. The features required by the user are selected and are then passed to the server using flask.The user selected features are then passed into the .pkl file which contains the ML model for classifying yoga posture for that particular feature from which the result is obtained, and then the result are passed to the frontend using flask and then displayed to the user.

- **Login Subsystem** The DFD for this module is given in Figure 4. As shown in the diagram, the two way communication goes between the interface and database. The user credentials are passed from the frontend to the server using flask, which are matched with the user credentials already stored in the database, and if matched successfully the user can login to the system. The register is used to add information of new user to the database while change password functionality is used to change the password information for particular user in the database.

4.2.3 Design Rationale

This simple architecture shown above consisting of the pre-trained model, flask API and a front-end is one of the most basic working architectures used in low-scale machine learning implementations. Compared to Django, Flask is open to change, and it adds a layer of flexibility to the web app development process. Furthermore, the framework is straightforward to use and understand, even for a newbie

4.3 Specific Requirements

4.3.1 Functional Requirement Specification

User Login

Use case name	:	User Login
Objective	:	Used to log in to an existing profile.
Priority	:	High
Precondition	:	User is not logged in to a profile, input profile exists in database, user password matches profile
PostConditions	:	Page data is appropriate for selected profile
Flow of Events	:	1. Basic Flow

1. Basic Flow
   (a) User enters the username and password in the text box and then clicks on the Login button.
   (b) An animated loading bar is shown while the authentication of the user profile takes place.
   (c) The Webpage is updated for the user profile.
   commands
2. Alternate Flow 1
   (a) In case of invalid username or invalid password or mismatched password, an error message is shown and it is redirected to the previous page.

Option to Upload a Video

Use case name	:	Video upload
Objective	:	The user selects and uploads a video to the site as input on which the posture is to be recognised.
Priority	:	High
Precondition	:	Upload file picker shows no file uploaded
PostConditions	:	Uploaded file name and file size along with a video icon is displayed.
Flow of Events	:	1. Basic Flow

(a) User selects the Upload Video button

(b) A file browsing window pops up for uploading video file

(c) User selects the video file to be uploaded

(d) An animated loading bar is shown while uploading

(e) Success message is shown after successful upload

Yoga Posture Detection From Uploaded Video

Use case name	:	Detect yoga posture
Objective	:	The feature allows the user to detect a yoga posture from the video uploaded.
Priority	:	High
Precondition	:	Cursor is placed at upload Detect Yoga Posture button
PostConditions	:	The window for uploading video is opened.
Flow of Events	:	1. Basic Flow

(a) User clicks on the Detect Yoga Posture button.The window for uploading video is opened and video is uploaded.

(b) User clicks on the submit button.

(c) An animated loading bar is shown till the detection of yoga posture and the percentage error in posture when done by the user is completed.

(d) The yoga posture name along with the percentage error in posture and other relevant data (such as images from video in which the posture deviates from standard posture) is given as output.

Real Time Yoga Posture Correction

Use case name	:	Correct yoga posture real-time
Objective	:	This allows user to correct yoga posture in real-time.
Priority	:	High
Precondition	:	Cursor is placed on Posture Realtime Correction button
PostConditions	:	The video is being sent to the server realtime after the clicking of Posture Realtime Correction button.
Flow of Events	:	1. Basic Flow

(a) User clicks on the Posture Realtime Correction button.

(b) The video of the user doing yoga is sent to model in the server and the error in posture is calculated.

(c) The error in posture is sent by the server and corrects the user doing yoga realtime.

(d) The number of corrections stated by the system along with other statistical data (percentage accuracy of posture,pie chart for accuracy etc.) is listed out.

Diagnosis of chronic diseases from symptoms

Use case name	:	Early detection of chronic diseases from symptoms.
Objective	:	The user can know about the chronic disease they are suffering with based on the symptoms entered.
Priority	:	Medium
Precondition	:	No box is checked in the check box list listing symptoms.
PostConditions	:	The symptoms from the list which are being experienced by the user is checked.
Flow of Events	:	1. Basic Flow

(a) User clicks on the Diagnose disease button.

(b) A window containing a list of symptoms with a check box is opened.

(c) The symptoms experienced are selected and the submit button is clicked.

(d) An animated loading bar is shown while the diagnosis of disease from symptoms is being done.

(e) The predicted disease along with the precaution that can be taken is listed out(but it is always wise to consult a doctor for confirmation).

Yoga Recommendation to reduce the burden of chronic diseases on users.

Use case name	:	Yoga recommendation for specific chronic diseases
Objective	:	This feature recommends relevent yoga to the users to control and prevent chronic diseases.
Priority	:	High
Precondition	:	Nothing is selected in the drop-down list.
PostConditions	:	The chronic disease experienced by the user is selected in the drop-down list.

Flow of Events

1. Basic Flow

   (a) User clicks on the Yoga Recommendation For Chronic Disease button.

   (b) The chronic disease experienced by the user is selected from the drop-down list.

   (c) The submit button is clicked and an animated loading bar is shown till the required yoga posture to be done by the user and other information such as precaution for that particular chronic disease is selected.

   (d) The yoga posture to be done along with data such as precautions to be taken are listed out.

Yoga Recommendation based on benefits.

Use case name	:	Recommendation of yoga posture based on specific benefits.
Objective	:	User can get information about a particular yoga based on their requirements like yoga for good sleep, concentration etc.
Priority	:	High
Precondition	:	No box is checked in the check box list listing benefits.
PostConditions	:	The benefits(such as improved blood circulation) required by the user are checked.

1. Basic Flow

    (a) User clicks on the Recommend Yoga By Benefits button.

    (b) A window containing a list of benefits with a check box is opened.

    (c) The benefits required are selected and the submit button is clicked.

    (d) An animated loading bar is shown while the required yoga posture is selected.

    (e) The name as well as the steps to do that particular yoga posture(in images) are listed out.

4.3.2 Non-functional Requirements

Performance Requirements

1. The web application should be able to handle multiple user requests.

2. The response time of the application should be reasonable and should not exceed a threshold.

Portability

1. The prototype of the web application should be able to run on any device provided that there is access to a web browser.

Security Requirements

1. Users are required to authenticate themselves before using the application.
2. Their personal data is protected in compliance with the IT
3. All the data is stored and processed with the user's permission.

Reliability

1. The web application should not fail to operate with varying load.
2. It should work with reasonable accuracy.

Maintainability

1. In the event of failure of the system, it should be fairly easy to restore it back to operational status.

Availability

1. The system should be up and running for the majority of the time with less down time.

Chapter 5

Results and Discussion

5.1 Proposed Model

In this section, we will propose an experimental setup which we use to build the best possible model according to the result of our seminar. The coding is done in Python 3.10 using Jupyter Notebook. As can be seen in the above literature survey RF is one of the best classifier among the six classification algorithms (RF, KNN, Logistic Regression, Decision Tree, Naive Bayes, SVM). So we will use MoveNet postural feature extraction model for feature extraction and then we will use RF classifier to implement the model. The following steps are done to implement the proposed model:-

- Gather the dataset.
- Preprocess the pose classification training data into a CSV file that specifies the keypoints detected by MoveNet along with ground truth tables.
- Build and train a pose classification model that takes the keypoints coordinates from the CSV file as input, and outputs the predicted labels.

5.1.1 Dataset Collection

Here, we get the yoga-posture dataset from TensorFlow datasets. The dataset which we will use consists of images of five yoga poses (Chair, Cobra, Dog, Tree, Warrior). The images in the dataset will be as shown in Fig.5.1. There are 425 images in total in this dataset.

Figure 5.1: Dataset image example

5.1.2 Data Preprocessing

In this step we preprocess the training images into a CSV file that contains the landmark coordinates and ground truth labels. MoveNet thunder is used to detect coordinates of 17 keypoints as shown in Fig. 5.2. Then we store detected landmark coordinates in a CSV file . We then split the images into train and test dataset. The 17 keypoints detected by MoveNet model are left ear, right ear, left shoulder, right shoulder, left elbow, right elbow, left wrist, right wrist, left hip, right hip, left knee, right knee, left ankle, right ankle as shown in Fig.5.3.

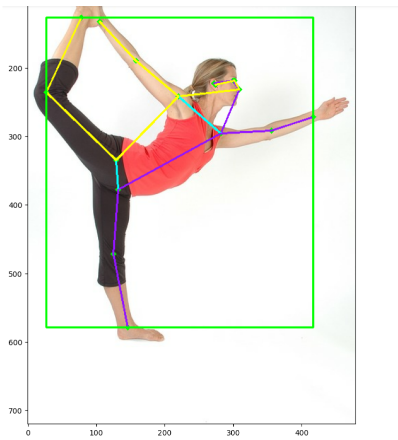


Figure 5.2: Image after preprocessing

Figure 5.3: Keypoints Detected By MoveNet

5.1.3 ML modelling using RF

After we get the CSV file, we train the dataset using RF classification algorithm. Here, in this case we use tensorflow and keras to implement the RF classification. Then, we validate the results using test dataset.

5.2 Results

The accuracy of the model is 0.99. The precision, recall and F1 score for the model is given in the following table. The confusion matrix is given in figure 5.4

Pose	Precision	Recall	F1 score
Chair	85.24	78.78	89.68
Cobra	78.68	72.22	89.65
Dog	73.77	76.92	68.96
Tree	59.01	54.54	82.88
Warrior	67.21	62.16	79.31

Table 5.1: Precision, Recall, F1 score for the proposed model

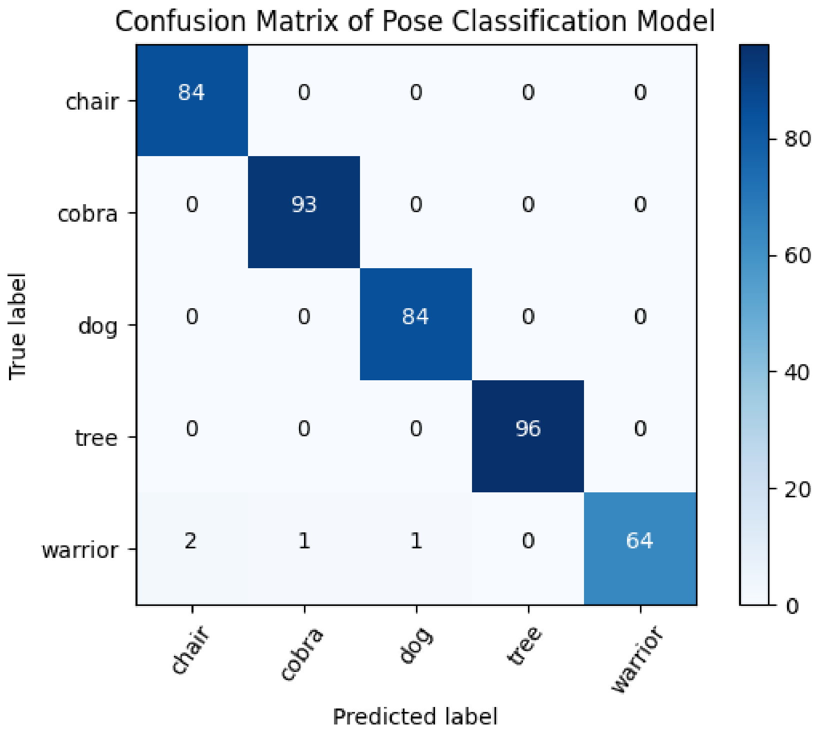


Figure 5.4: Confusion Matrix for the model

5.3 Inference

- The proposed model gives us a high accuracy of 0.99.
- The MoveNet thunder model was used for feature extraction.It detects the coordinates of 17 keypoints in human body which we store in CSV file.
- TensorFlow and Keras were used to implement the RF algorithm in the above model.
- The precision, recall and F1 score is given in the table 5.1 .

5.4 Prototype

5.4.1 Overview of User Interface

When the user visits our website initially, he or she will be redirected to the authentication page where they need to enter the required credentials to login. The they will be redirected to the home page which is the one place to login to all other pages of the website.The home page consists of the links to other pages like Yoga Posture Correction, Yoga Recommendation, Chronic Disease detection, Yoga Recommendation etc. Yoga Posture Correction page allows user to correct their yoga pose realtime. It is done by continous video monitoring of the user performing the yoga and analysing the same to provide necessary suggestions to improve the posture. Yoga Posture detection page allows the user to upload a video of the yoga they perfomed and the model predicts the name of the yoga that they uploaded. Users will be able to know the chronic disease they are affected with by uploading the required symptoms using the chronic disease detection page. Yoga recommedation is done based on user's specific needs and based on chronic diseases. BMI, diet etc are examples of user's specific needs. For people usually doing yoga, they might prefer doing yoga to prevent chronic diseases.

5.4.2 Screen Images

5.4.3 Screen Objects and Actions

Sign in & Sign Up Page

Sign in and Sign Up Page are for user authentication. When the user visits our website, this is the first page that they will be redirected to. Users have to enter their login

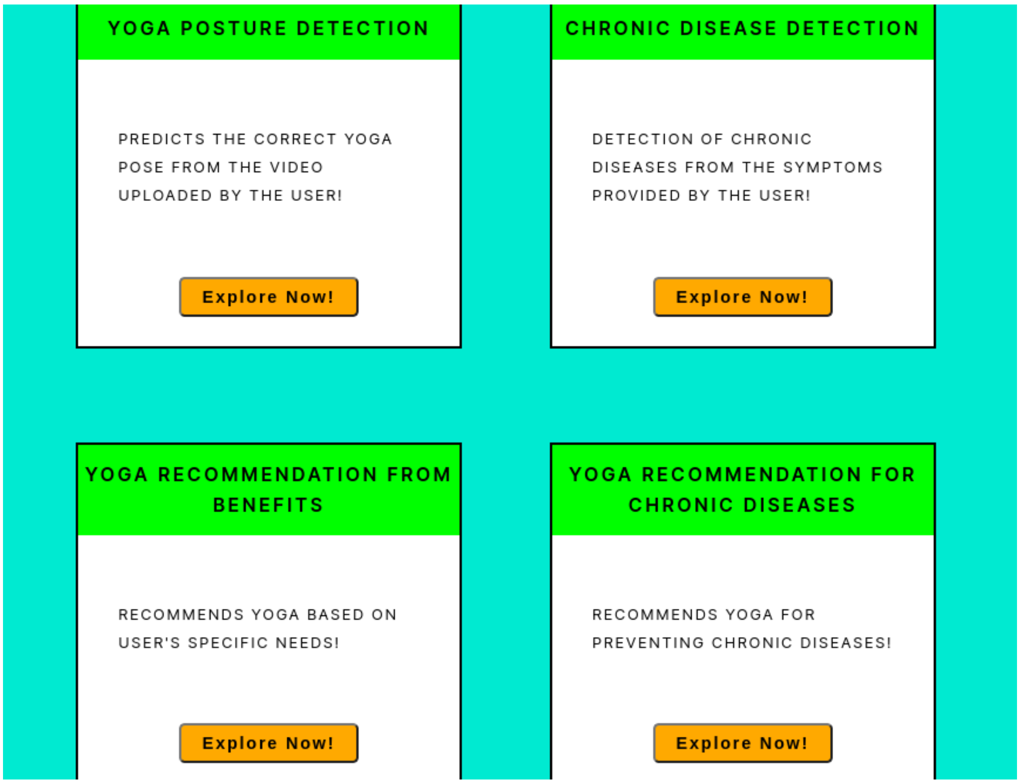


Figure 5.5: Interface for navigation to posture detection, chronic disease detection and yoga recommendation.

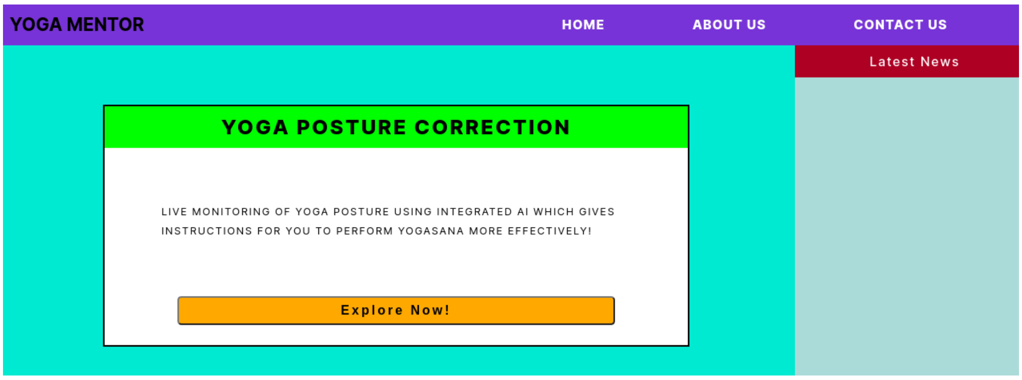


Figure 5.6: Interface for navigation to yoga posture correction.

credentials in the sign in page for logging in to website successfully. If they does not have an account, then they need to create an account using sign up page.

Home Page

This is the page users are redirected to once they have successfully logged in. This page contains the link to navigate to the core features of the application such as Yoga Posture Correction, Yoga Posture Detection, Chronic Disease Detection etc.

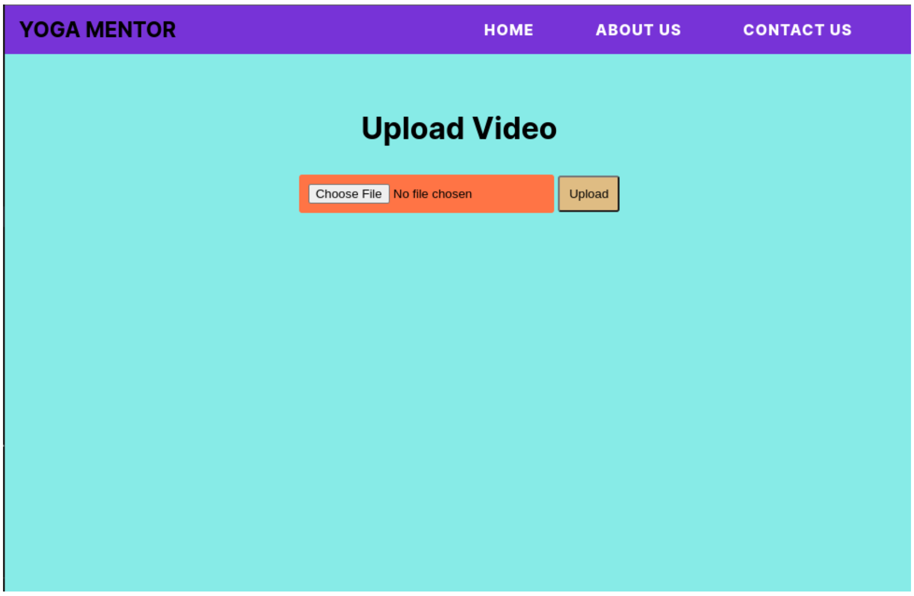


Figure 5.7: Interface to upload video for yoga posture detection.

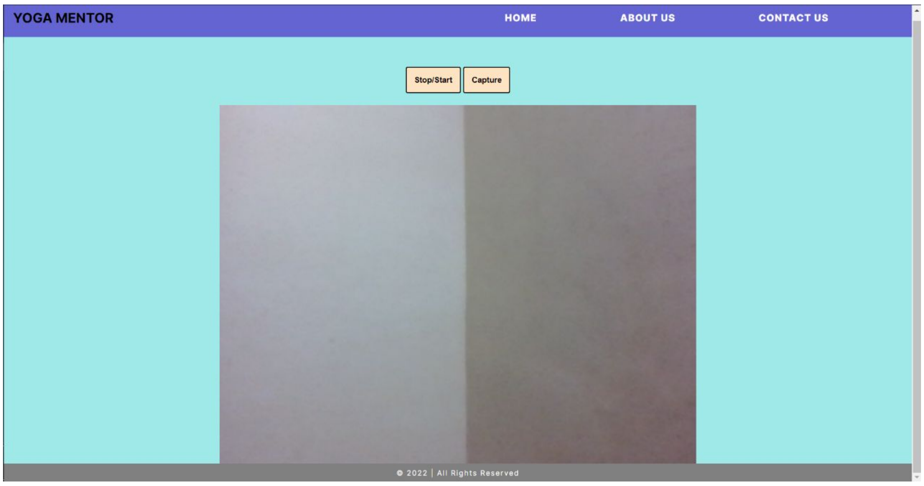


Figure 5.8: Interface of live video monitoring.

Interface for Yoga Posture Correction

This page continuously displays video monitoring screen for it to record the live yoga performed by the user. For this page to work, the laptop or PC should have a web cam or inbuilt camera in it. The video is analysed frame by frame with the trained yoga posture dataset to find the faults in the yoga posture and giving necessary corrections through text or voice.

Interface to detect Chronic Diseases

This page contains various list down box to select various symptoms to predict the chronic disease they are affected with. Users can select one or more symptoms and the system predicts the chronic disease by using already trained dataset.

Yoga Recommendation

Yoga recommedation is done based on user's specific needs and based on chronic diseases. BMI, diet etc are examples of user's specific needs. For people usually doing yoga, they might prefer doing yoga to prevent chronic diseases.

Chapter 6

Conclusion

Each chapter is to begin with a brief introduction (in 4 or 5 sentences) about its contents. The contents can then be presented below organised into sections and subsections.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

References

[1] D. Mogaveera, V. Mathur, and S. Waghela, “e-health monitoring system with diet and fitness recommendation using machine learning,” *Inventive Computation Technologies (ICICT)*, 2021.

[2] L. M, B. Sharon, K. Ashley, and Naimi, “Machine learning can improve the development of evidence-based dietary guidelines,” *Cambridge University Press*, 2022.

[3] D. Kirk, E. Kok1, M. Tufano1, B. Tekinerdogan2, E. J. Feskens1, and G. Camps1, “Machine learning in nutrition research,” *Journal of Advances in Nutrition*, 2022.

[4] D. S. G. Utkarsh Bahukhandi, “Yoga pose detection and classification using machine learning techniques,” *International Research Journal of Modernization in Engineering Technology and Science*, 2021.

[5] A. S. Yash Agarwal, Yash Shah, “Implementation of machine learning technique for identification of yoga poses,” *2020 IEEE 9th International Conference on Communication Systems and Network Technologies (CSNT)*, 2020.

[6] K. J.Palanimeera, “Classification of yoga pose using machine learning techniques,” *Materials Today,IEEE*, vol. 37, no. 2, 2021.

[7] A. Sharma, J. Pathak, and P. Rajakumar, “Disease prediction using machine learning algorithms,” in *2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)*. IEEE, 2022, pp. 995–999.

[8] K. Vayadande, R. Golawar, S. Khairnar, A. Dhiwar, S. Wakchoure, S. Bhoite, and D. Khadke, “Heart disease prediction using machine learning and deep learning

algorithms," in *2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*. IEEE, 2022, pp. 393–401.

[9] D. K. Chohan and D. C. Dobhal, "A comparison based study of supervised machine learning algorithms for prediction of heart disease," in *2022 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*. IEEE, 2022, pp. 372–375.

[10] P. C. H. I. M. Z. P. M. V. O. V. Aleš Procházka, Ondˇrej Dostál, "Deep learning for accelerometric data assessment and ataxic gait monitoring," *SIGCOMM Comput. Commun. Rev.*, vol. 29, no. 4, Mar. 2021.

[11] A. R. M. I. T. Lillian N. Boeetcher, Behnaz Ghoraani, "Dual-task gait assessment and machine learning for early-detection of cognitive decline," 2022.