SYMPTOM-BASED DISEASE PREDICTOR AND DIET SUGGESTER

A Project Report

Submitted by

Atharva Dixit	**111703067**
Gaurav Mandke	**111703071**
Abhijeet Chavan	**141803001**

in partial fulfilment for the award of the degree of

B.Tech (Computer Engineering)

Under the guidance of

Prof. V.K. Khatavkar

College of Engineering, Pune

DEPARTMENT OF COMPUTER ENGINEERING AND INFORMATION TECHNOLOGY,

COLLEGE OF ENGINEERING, PUNE-5

May, 2021

DEPARTMENT OF COMPUTER ENGINEERING AND INFORMATION TECHNOLOGY, COLLEGE OF ENGINEERING, PUNE

CERTIFICATE

Certified that the project titled, "SYMPTOM-BASED DISEASE PREDICTOR AND DIET SUGGESTER" has been successfully completed by

Atharva Dixit	**111703067**
Gaurav Mandke	**111703071**
Abhijeet Chavan	**141803001**

and is approved for the partial fulfilment of the requirements for the degree of "B.Tech. Computer Engineering".

SIGNATURE

Prof. V.K. Khatavkar

Project Guide

Department of Computer Engineering and Information Technology, College of Engineering Pune, Shivajinagar, Pune - 5.

SIGNATURE

Dr. Vahida Attar

Head

Department of Computer Engineering and Information Technology, College of Engineering Pune, Shivajinagar, Pune - 5.

Abstract

Machine Learning (ML) has, over the years, played a crucial role in shaping human lifestyle and simplifying innumerous tasks. Biomedical and healthcare domains consist of a huge volume of data, which can lead to an in-depth analysis upon processing. According to reports, it has been found that about 72 percent of people browse the Internet for obtaining health related information. In most cases, this information is present in a relatively unstructured form. It is a complex task for any ordinary citizen to conclude the meaning of this information, especially when they have limited knowledge in this field.

The current system in existence displays the probable diseases upon entering the symptoms, to the user. However, the disadvantage with this approach is, the user can only enter his symptoms first, forcing the system to take just those into consideration, without actually asking any follow-up questions. This keeps the information concise but incomplete. Occurrences of situations where the user enters mild symptoms (cold, cough) but the system still displays serious illness is seen, as it fails to take into account the symptom severity as well as co-occurring symptoms at the time of input.

Typically, these problems are handled by using a supervised learning algorithm that is trained to detect disease onset. We aim to build up on such a system by further enhancing its accuracy, by comparing several supervised learning algorithms and using ensemble learning methods and/or feature selection techniques, while also suggesting a suitable diet plan to be followed by the affected user. This project aims to ensure utmost effectiveness of such a crucial health-oriented system, so a large section of the society is benefited.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Searching for a possible infection based on the symptoms directly on the web is to be best avoided. The reason being, online sources vary widely in terms of credibility and information. Several health and psychological-related surveys have indicated that there are instances where an individual can have extreme trauma anxiety upon such searches. Misdiagnosis on behalf of such an individual may even lead to a delay in providing medical treatment, further worsening health.

This project aims to create a generalized application with a wide disease database, which takes in the initial user input symptom(s), asks follow-up questions which shows other possible co-occurring symptoms with the ones the user initially enters. Once the user feels all the relevant data has been entered, the system shows the top five possible diseases the user could be suffering from, in decreasing order of their probabilities. In order to arrive at the maximum possible accuracy for a particular supervised learning model, the project aims to apply ensemble learning methods as well as feature selection methods.

Ensemble learning, one of the two techniques to be used for accuracy enhancement, makes use of a diverse set of supervised learning models, instead of a single model, in order

to improve the net system performance for prediction purposes. For instance, in the case of a decision trees, rather than relying on a single decision tree and hoping the system chooses the correct path at a given split from a node to its children, it is much more efficient to build a final predictor that has the ability to calculate the features to be used by amalgamating multiple decision trees. Random Forest Classifier itself is considered to be an ensemble learning technique, in which each tree will be split on different features. All such trees are finally averaged to generate the final model.

Feature Selection is a process wherein the main objective is to decrease the count of input variables/features which are necessary so as to build a predictive model. This is essential because there could be certain features that can prove to be irrelevant in the system, causing the model accuracy to go down, and add to the computational cost. Systematic experimentation is needed to come up with the best suited supervised feature selection method. This is first applied to Decision Trees.

Chapter 2

Literature Review

As discussed, the project focuses on improving accuracy of prediction of disease based on input symptoms. The primary module of the project is disease detection followed by a secondary module of diet recommendation for the same; in application point of view. Research point of view dives deep into predictive and exploratory data analysis, which aims to find out an efficient algorithm by comparing different supervised algorithms applied to the considered datasets.

S.Udin Et. AL. in [1], has considered the use of a case study to recognize the critical patterns among various types of supervised machine learning algorithms, and their presentation and use for prediction of risk of disease infection.The dataset contained 48 articles, every one of which executed more than one variation of supervised machine learning algorithms for prediction of a disease. It was concluded that the Support Vector Machine (SVM) algorithm and the Naive Bayes algorithm were applied with the maximum frequency. Nonetheless, the Random Forest algorithm showed high accuracy too. Of the 17 examinations where it was applied, RF showed the highest accuracy in 9 of them, i.e., 53%. This was trailed by SVM which beat in 41% of the investigations it was considered. This study gives a wide outline of the overall behaviour of various types of

supervised machine learning algorithms for infection prediction. Two information bases were referred to: Scopus and PubMed. Scopus is an online bibliometric data set created by Elsevier. It has been picked on account of its undeniable degree of consistency and accuracy. PubMed is a free distribution internet searcher and joins reference data generally for biomedical and life science writing. It includes, in excess of 28 million references from MEDLINE, life science diaries and online books. MEDLINE is a bibliographic data set that incorporates bibliographic data for articles from scholastic diaries covering medication, nursing, drug store, dentistry, veterinary medication, and medical services.

In [2], Durai Raj Et. Al discusses about in-depth study reports of various types of data mining applications in the medical care area and to decrease the intricacy of the investigation of the medical care data transactions. Likewise, it also presents an analysis to compare various data mining applications, strategies and various techniques applied for separating information from databases created in the medical services industry. At last, the current data mining strategies with data mining calculations and its application instruments which are more important for medical care administrations are examined in detail.The benchmark dataset and the SEER data set are utilized for analysis.The comparative study concludes that data mining procedures in all the medical services applications give a really promising degree of accuracy like 97.77% for cancer detection (malignant) and around 70% for assessing the achievement pace of IVF therapy.

S. Sharmila Et. Al. in [3] do a comparative study of three different classification techniques namely, Fuzzy logic, Decision tree and Fuzzy Neural network which is used to classify liver dataset taken from the repository for ML at University of California, Irvine. From the experimental results it is noticed that out of the three techniques, Fuzzy Neural

Network also called as Neuro-Fuzzy system gave the highest accuracy score. The hybrid technique could be successfully used to help the diagnosis of liver disease.

Archana Singh Et. Al in [4], describes the techniques used for predicting the risk factor of heart disease for which they used the heart diseases symptoms dataset from UCI. The dataset included some null values or missing values, some categorical values which are required to be converted into values that can be fed into the model , This process is the preprocessing of data where the noisy or unused data are removed and the useful data is fed to the model. In this paper, the author calculates accuracy of several ML algorithms for predicting heart disease. For the same, the algorithms explored are Linear Regression, Decision Trees, Support Vector Machines, and K-Nearest Neighbors by utilizing the repository provided by UCI for training and testing.

Larochelle Et Al. in [5] describes the technique used for enhancing the accuracy of the prediction , there are problems like overfitting and underfitting of the model where the model shows an unpredictable behavior, to optimize the model the model must be trained with the appropriate parameters, thus in machine learning hyperparameter optimization is used as a way of enhancement in the accuracy of the prediction of the solution. This paper describes the Grid search CV, rather than manually choosing the values for the hyperparameters grid search selects the appropriate values among the list of all evaluated values for the hyperparameter. MNIST basicdata set is a subset of the notable MNIST manually written digit database. The mnist background pictures database is slightly different from mnist fundamental where the white front ground digit has been composited on top of a 28x28 normal picture patch.These two datasets were utilized for analysis purposes. Grid search tests designate such a large number of trials to the investigation of measurements, that do not make any difference and experience poor coverage

of measurements that are significant. Compared to the experiments performed for grid searching, random search discovered better models in several cases and required less time for computation. Random tests are, additionally, simpler to complete than grid tests for pragmatic reasons identified with the statistical independence of each trial.

Daniel Lowd Et. Al. in [6] aims to show that for a wide scope of datasets, Naive Bayes models have a better accuracy and less learning time in contrast to other Bayesian networks.The creator utilized 47 datasets from the repository provided by the University of California, Irvine, with a varying number of factors from five to 618, and in size from 57 guides to 67,000. The extent of request of Naive Bayes derivation is quicker than Bayesian organization inference.This paper proposes Naive Bayes models as an option instead of Bayesian networks for general probability assessment tasks.Experiments on an enormous number of datasets show that the two require almost the same time to learn and are likewise exact, yet naive Bayes deduction factor is significantly larger.

Celestine Lwendi Et. Al. in [7] proposes a deep-learning based solution for a medical dataset that naturally recognizes which food ought to be given to which patient dependent on the disease and different features like age, sexual orientation, weight, calories, protein, fat, sodium, fiber, cholesterol. This study is centered around carrying out both ML and DL oriented calculations like Gated Recurrent Units, Logistic Regression, Naive Bayes, Recurrent Neural Network, Multilayer Perceptron, and Long Short-Term Memory. The clinical dataset gathered through the web and emergency clinics comprises of 30 patient's data with 13 features of various diseases and over a thousand products. The segment for products has 8 features set. The features of these IoMT data are dissected and further encoded prior to applying ML and DL based rules.

G Agapito Et. Al in [8] describes a process which is a less complex version for diet recommendation but goes through a completely different pathway than the previous one mentioned. Here the diet is recommended based on a questionnaire, which is demonstrated by using a tree, where nodes are the questions while an edge connects two nodes related to them by a particular value(answer) to the current question. Questionnaires are adaptive, that is, the following question to submit to the user is obtained by analyzing the child's node of the current node of the questionnaire tree. This solution allows conveying to the users only relevant questions related to their real health status, making it possible to characterize the health profile accurately. Thus, the system gives to the user more accurate health-related advice, pertaining to his/her health status, avoiding to give unsuitable advice.

Xiao-Yan Gao Et. Al. in [9] describes the process to split the proposed system into 6 structured stages - Data collection, Data preprocessing, feature selection, data splitting, training models, and evaluating models. Once their model has been trained, bagging and boosting (2 ensembling techniques) are applied to enable the prediction of a heart ailment, along with random forests classifiers, decision trees, KNN, and Naive Bayes. As these models need to be trained and then subsequently evaluated, the dataset has to be used, which consists of over a thousand records, 13 features, and a single target column, which outputs 1 in case of presence of the disease, and 0 in case the disease is absent.PCA and LDA (2 Feature Extraction algorithms) are also used, and it is concluded that the combination of bagging along with PCA and Decision Tree perform well due to their higher accuracy.

In [10] Loganathan Et. Al. presents a hybrid classifier model that uses logarithmic re-

gression with an accuracy of 95% in predicting diseases - which include diabetes, cerebral infection, typhoid and dengue fever. Modified Artificial Plant Optimization (MAPO) used for optimal feature selector proposed by Sharma Et.Al which is used along with numerous machine learning algorithms for predicting heart rate using fingertip video dataset.KNN and CNN algorithms are used to build up on the classifier model. Images are pre-processed using filtering techniques, thus extracting relevant features from it.The disease prediction is carried out in a centralized server with the help of hybrid Classifier Model using Logarithmic Regression. It is noted that the proposed classifier model can predict the diseases with an accuracy of 95%.This model works well with unique cases or Specific Training Data. Cases like Typhoid diagnosis are come out with a predicted value of less than 70%, whose reason (based on study) may be because of the gradual intervention of viral in the human body.

Jackins Et. Al. , in [11],have used the help of correlation coefficient and confusion matrix to predict the classification problem involving diseases like coronary heart disease, breast cancer and diabetes, and further calculate the accuracy, recall, precision and F1-score. This is used to conclude that the random forest model was best suited for training datasets, as well as real-time data. At first, illness dataset is used for the system input. Diabetes, coronary illness and cancer datasets are taken for the analysis stage, following which data preprocessing is done to remove unwanted information, and data splitting is carried out to split the data into 70% training data and 30% testing data. Data mining algorithms like random forest and Gaussian Naive Bayes are applied to estimate the performance of the system. Classification results show an improved performance over the existing results.

Latha Et. Al., in [12], make use of the Cleveland dataset within the ML repository provided by the University of California, Irvine, and test ensembling methods like boosting, bagging, stacking and majority voting on the test dataset after training them on 80% of the available initial dataset, to come to the conclusion that majority voting provides the best jump in accuracy, which can further be enhanced by employing feature selection techniques. They explore a method called ensemble classification, which allows one to improve the accuracy of certain weak classifiers by merging multiple algorithms, while also implementing the algorithm on a medical dataset from an application perspective. The conclusions of the case study show that ensemble techniques, for example, boosting and bagging, are viable in improving the accuracy of weak classifiers, and display an average achievement in distinguishing hazard of coronary illness.

Chapter 3

Research Gaps and Problem Statement

- The project encompasses an application point of view as well as a research point of view.
- With respect to the former, the project is aimed to develop a product which can predict a disease accurately based on symptoms provided by user input and recommending a diet for the same.
- A methodological study for improving the accuracy and/or efficiency for prediction of disease is what the research point of view tries to expand into. The project follows by diving deep into predictive data analysis to find out the most efficient as well as accurate model of machine learning with respect to prediction of disease.
- The goal extends further for improving the prediction accuracy more by applying Ensemble learning techniques and/or Feature Selection methods for data analysis.

3.1 Problem Statement

3.1.1 Problem

Need of a system that can simplify the job of disease identification based on symptoms, with high accuracy for best results, while also providing a suitable diet plan to tackle the disease. Once the model is trained with the training dataset, the accuracy and cross-

validation of the said supervised learning algorithms need to be calculated and compared. In all, we need to come up with the best suitable combination of accuracy-enhancing methods with the existing algorithm by extensively analysing our training and validation dataset.

3.1.2 Solution

The system is fed with datasets which are scraped from the web, so that the train-test split (data splitting) is carried out on pre-verified data. It will be capable of asking co-occurring symptoms apart from those already entered by the user.

3.1.3 Objective

To solve this problem, we are developing a prediction system which will show the top possible diseases, and hence display the diet plan.

Chapter 4

Methodology/ Solution

This project starts by initially comparing the accuracy of certain supervised machine learning algorithms used for analysing the dataset.Once this is done, the cross-validation score is calculated. Following this, Ensemble learning is on these models forming ensemble of classifiers after which feature selection is applied on all models. Overall, an efficient model is concluded.

The system will also ask the end user to enter the duration of the experienced symptoms. Along with this, a severity dataset is analysed and based on feature selection, the severity of the aforementioned symptoms can be predicted, which will further be used to offer advice as to whether the user needs a clinical checkup or not.

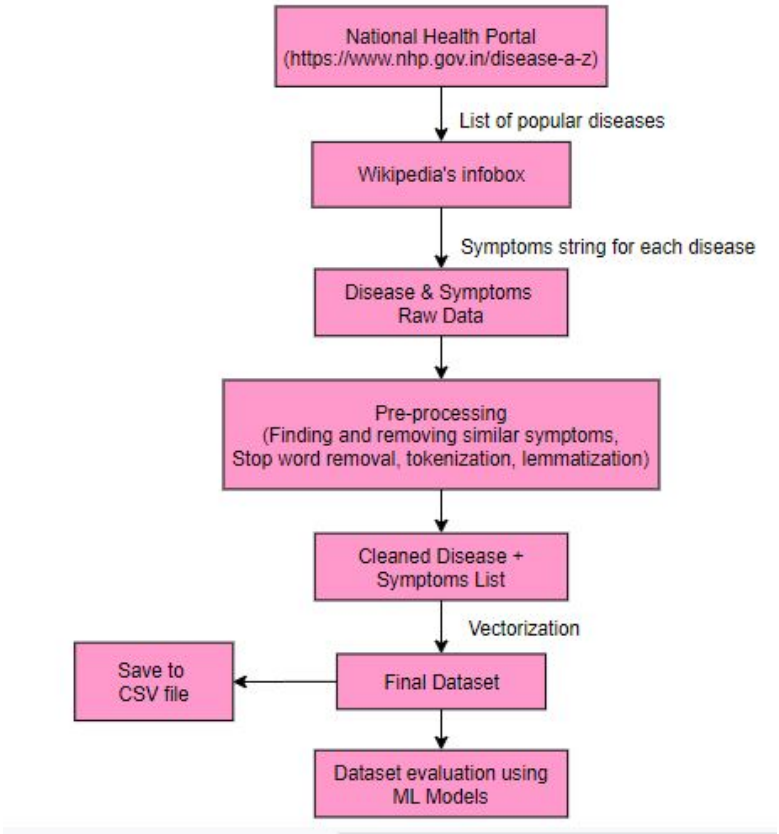

Figure 4.1: Dataset PreProcessing

Basically, the system prompts the user to enter the symptoms based on which model predicts disease with the highest probability and scores. System architecture figure describes the process of disease prediction and diet suggestion from user input. Following are the in-detailed steps involved in disease prediction:

4.1 User Input Processing

The system accepts symptom(s) in a single line, separated by comma (,). Subsequently, the following pre-processing steps are involved:

- Split user query (symptoms) into a list based on comma.
- Convert the symptoms into lowercase for easy matching.
- Removal of Stop words.

- Tokenization of symptoms to remove any punctuation marks.
- Lemmatization of tokens in the symptoms

The processed symptoms list is further used for symptoms query expansion.

4.2 Query Expansion

Each symptom in the list is expanded by appending a list of its synonyms. The synonyms are taken from thesaurus.com and Wordnet dictionary available in Python. Each symptom is broken into its different combinations for finding the synonyms set.

4.3 Symptom Selection and Suggestion

The expanded symptom query is used to find the related symptoms in the dataset. To find related symptoms, each symptom in a dataset is split into tokens and each token is checked for its presence in expanded query. Based on this, a similarity score is calculated and if the symptom's score is more than the threshold value, that symptom qualifies for being similar to the user's symptom and is suggested to the user.

The system prompts the user in the form of a question to select one or more symptoms from the list. The user selects one or more symptoms from the list. Based on the selected symptoms, which is taken as a response from the user, the system asks the question to identify and select other symptoms displayed to the user which are among the top co-occurring symptoms (with the ones selected by the user initially). The user can select any symptom, skip, or stop the series if questions i.e the symptom selection process. The final list of symptoms is then obtained for computing symptom vectors which is used for prediction.

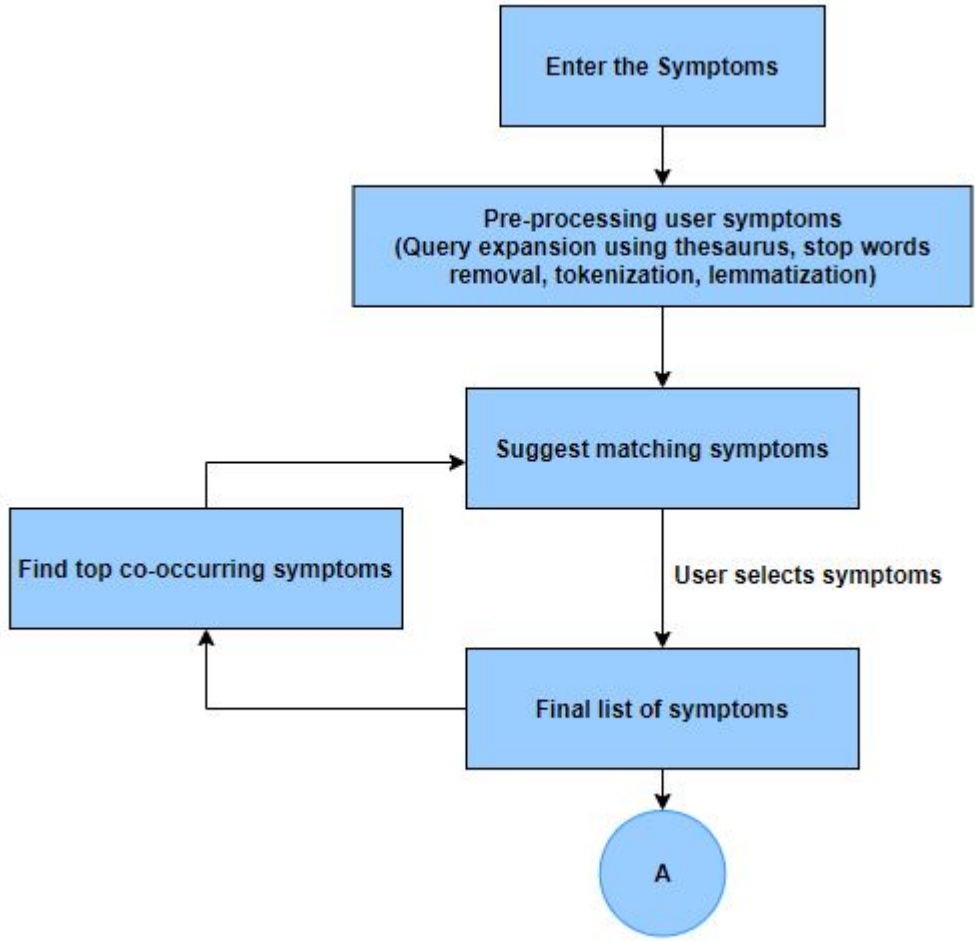

Figure 4.2: System Architecture Part A

4.4 Disease Prediction

Vectors are computed specific to the model using the final list of symptoms. A binary vector is computed that consists of 1 for the symptoms present in the user's final list of symptoms and 0 otherwise.

Different Machine learning models can be used for prediction. Dataset is split into training and testing dataset using train_test_split functionality of Python sklearn module. After dataset splitting, ML models are being trained upon the dataset. Different such models are compared, so that the most accurate and efficient model can be selected

for predicting a disease. These models accept the symptom vector and output a list of top K diseases sorted in decreasing order of probabilities.

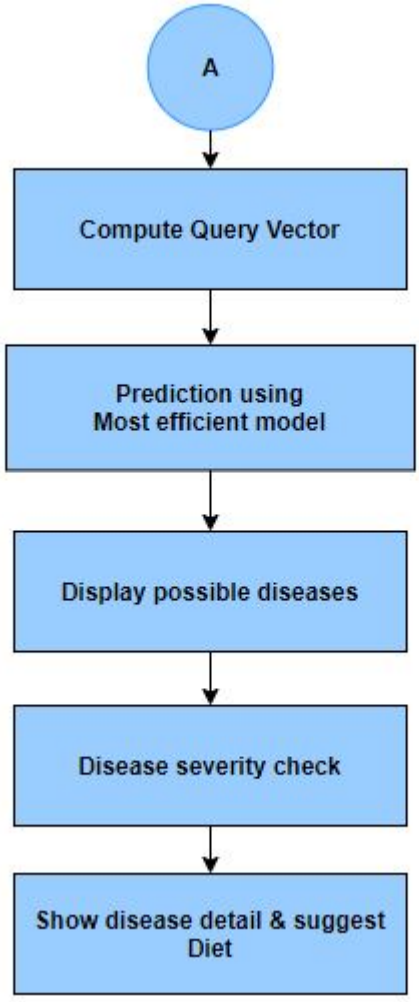

Figure 4.3: System Architecture Part B

4.5 Data Analysis

Predictive Data Analysis is done on datasets by comparing every model categorically.

4.5.1 Comparison of Supervised Machine Learning Algorithms

Here data analysis is performed using following algorithms

1. Decision Tree

2. K-Nearest Neighbors

3. Multinomial Naive Bayes

4. Logistic Regression

5. Support Vector Machines

4.5.2 Comparing cross-validation scores

Cross Validation is performed on all the above considered classifier and regression models. Cross Validation Scores for all the models will be compared. A 5 fold splitting is done for experimentation purposes.

Cross validation is primarily used to assess how the results of a statistical analysis will generalize to an independent dataset. In technical terms, cross-validation is used to detect overfitting. (failing to generalize a pattern)

Therefore the model having the highest cross validation score will be selected for prediction of disease for taking into account a model that will predict most efficiently and correctly for generalized data.

4.5.3 Feedforward Artificial Neural Network Model

Feedforward neural network is a type of artificial neural network, wherein associations between the nodes do not necessarily generate a cycle. The feedforward neural network was the first and easiest kind of ANN devised. In this network, the data moves just a single way (forward) from the input nodes, through the hidden nodes (assuming any), and to the output nodes.

Multilayer Perceptron Classifier: It can differentiate between data which is not linearly separable, using a learning technique known as Backpropagation, which is supervised in

nature. It features non-linear activation and has several layers. Accuracy and Cross Validation Scores are calculated for the same.

4.5.4 Ensemble Techniques

Ensemble is the art of combining a diverse set of learners (individual models) together to improvise on the stability and predictive power of the model. The project scopes to apply these techniques to form ensemble of previously mentioned classifiers to build up on the accuracy and cross-validation score of the prediction of disease

Ensemble Learning Technique Applied -

1. Random Forests (Ensemble Classifier)
2. Bagging
3. Boosting
4. Majority Voting or Hard Voting
5. Soft Voting
6. Stacking

4.5.5 Applying Feature Selection

Feature selection is the process of decreasing the count of input variables to be used to build a model for prediction. It becomes essential to do so, in order to improve model performance and simultaneously decrease the cost of computation. This technique is initially tested out on the Decision tree model which executes intrinsic feature selection. Following this, different techniques of feature selection like removing of constant features using variance threshold method, Removal of correlated features using Pearson Correlation method, Selecting K-Best Features using Chi-Squared Method, and Mutual

Information using Information Gain are tested out on dataset. The best method out of these which help in removing the redundant columns in the dataset and aid in determining the exact feature set is considered for Feature Selection. This method is then applied on all previously modelled classifiers for observing change in cross-validation score as a parameter for efficiency as well as accuracy.

- Finally the accuracy as well as cross-validation scores of all the models are compared.

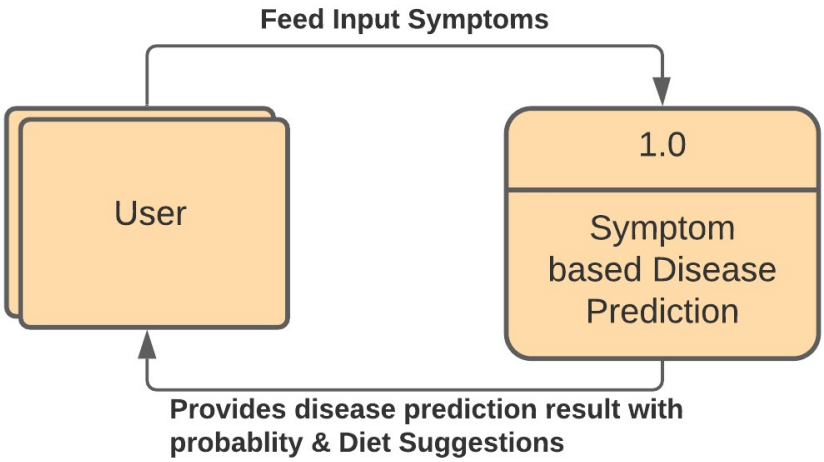

Figure 4.4: Data Flow Diagram Level 0

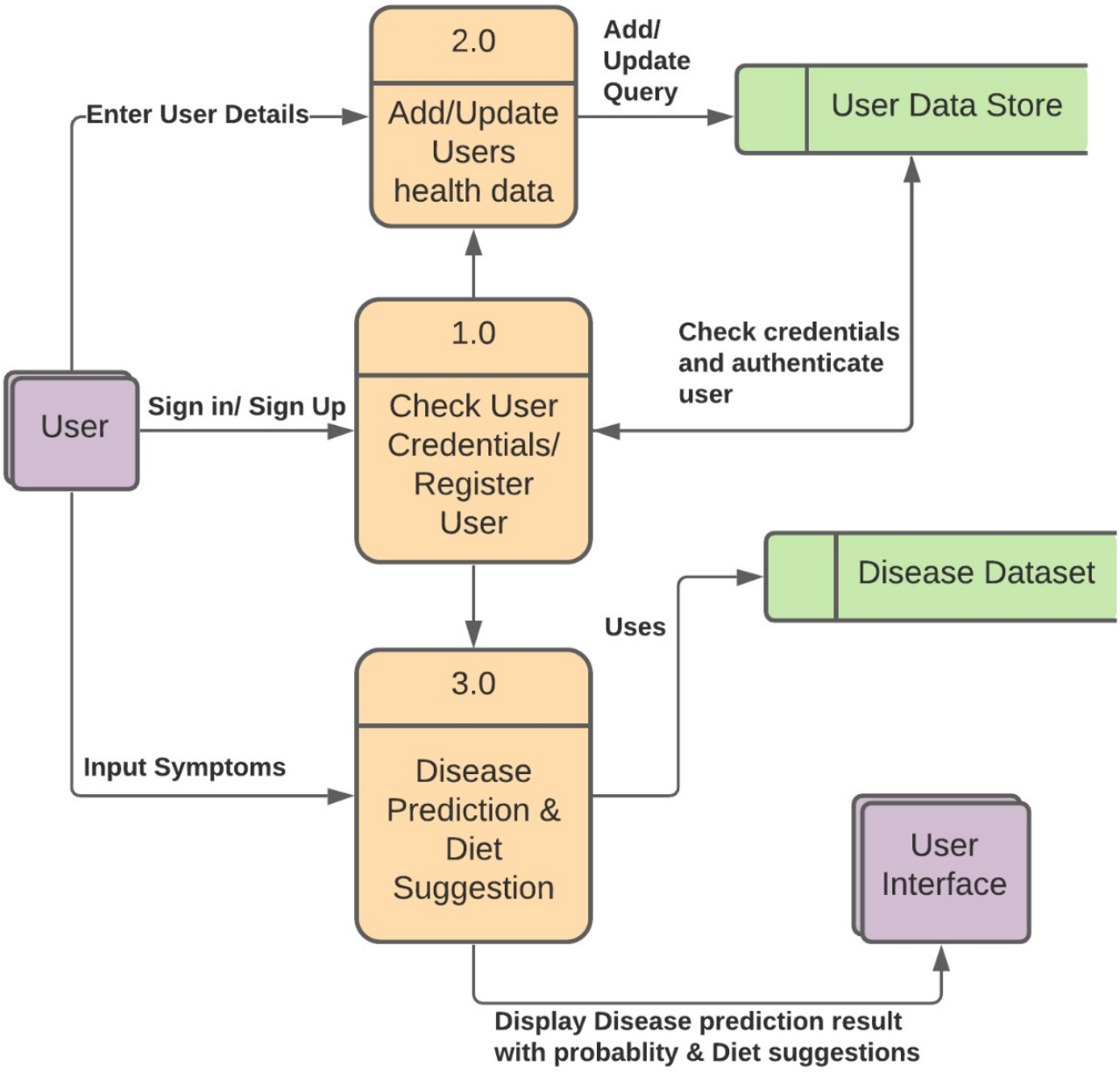

Figure 4.5: Data Flow Diagram Level 1

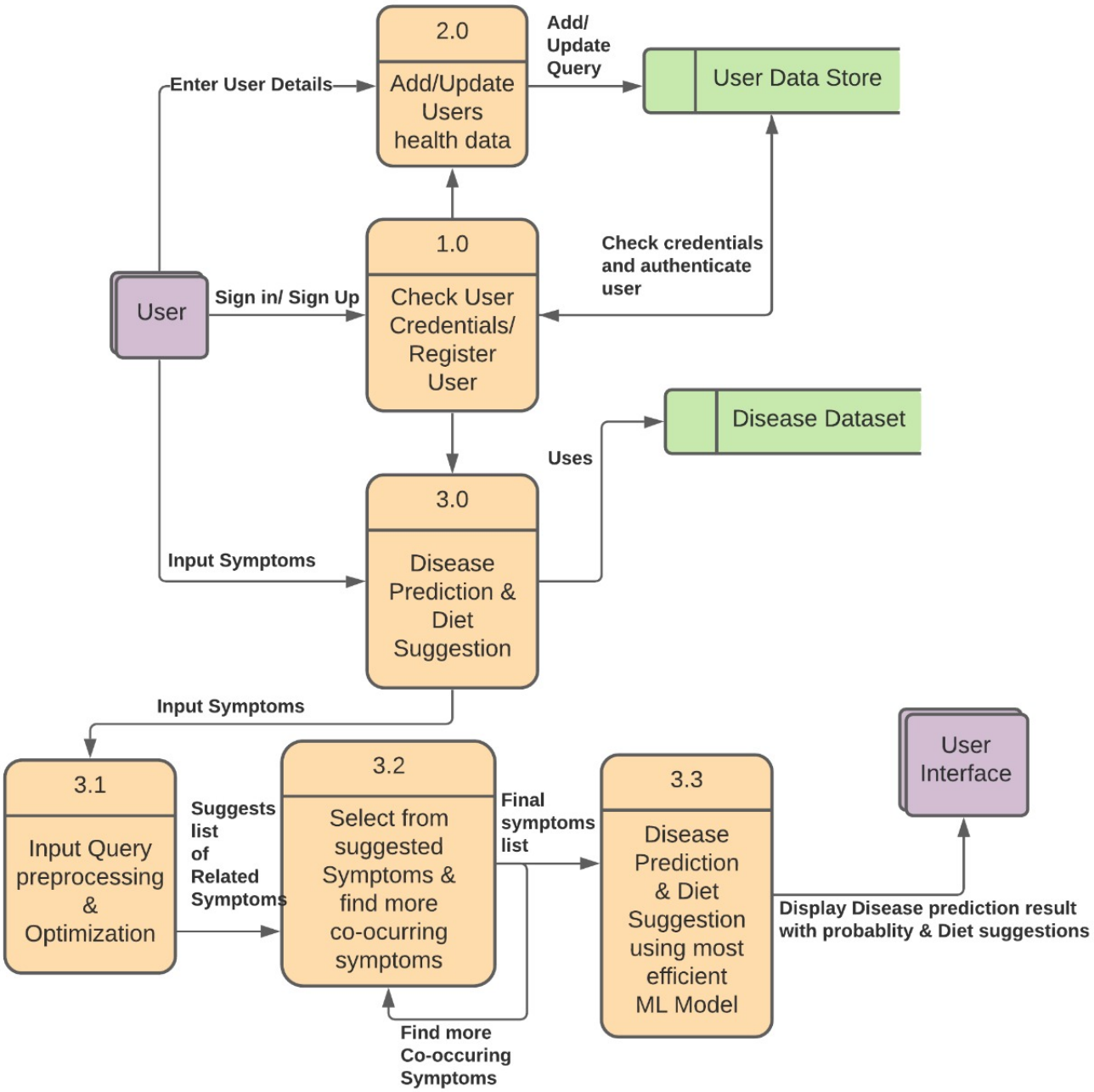

Figure 4.6: Data Flow Diagram Level 2

Chapter 5

Experimental Setup

5.1 Hardware Requirements

1. 4GB RAM (System Memory)
2. 1 GHZ Quad Core Processor
3. 25 GB Hard-Drive Space (for OS) + 2GB space for software and dependencies

5.2 Hardware Requirements for UI

1. 512MB RAM
2. 1GB ROM

5.3 Software Requirements

1. Python 3.6 or Above
2. Python Flask
3. Frontend
    - HTML

- CSS
- Javascript

4. Prominent Libraries

- Numpy, Pandas, Seaborn, Matplotlib : For Data Analysis
- Scikit Learn: For importing machine learning models
- Beautiful Soup: For web scraping
- NLTK : For parsing and lemmatizing user input as well as for formation of dataset after web scraping

5. Setup

- Linux Machine for Ease of Coding
- Jupyter Notebook for detailed data analysis
- Linux Shell for console based testing of project
- Any HTML Browser for UI based application through flask

6. General requirements for experimentation

- Form Datasets and keep all datasets ready in a defined folder for the data analysis
- Maintaining a jupyter notebook for performing data analysis and research
- Use 3-4 shell windows for different intended purposes ex. For launching jupyter notebook, For preprocessing data, For launching flask application

Chapter 6

Results and Discussion

6.1 Datasets

Initially, datasets from Kaggle pertaining to disease-symptoms were gathered for analysing. For raw start, a dataset from Kaggle having 132 symptoms as features (columns) and 4920 rows (after combination of disease and different symptoms in binary format) was considered. However, the prediction was vague for simple symptoms which made it necessary to obtain a dataset which is more diverse and which would lead to a satisfactory result after training.

A dataset was then formed by firstly scraping data from National Health Portal of India using BeatifulSoup Library fetching HTML code to filter out HTML tags which helped in obtaining diseases as labels. Secondly, the symptoms relating to the disease are scraped from Wikipedia using respective disease as input for searching by fetching the HTML code of the page. The datasets are formed by preprocessing the data and storing it in csv format.To be precise all the symptoms are extracted and a dictionary is created with key as disease and symptoms as value. Further, each disease is treated as the label and all symptoms are treated as specific attributes or columns.

A sample dataset consisting of an additional value for severity of disease (numerical value) is also used for testing feature selection on decision tree.

The severity of the disease is also predicted correctly using the severity dataset and feature selection method. However this methodology is applied currently only to decision tree as it was comparatively easier to do so.

After performing data analysis on various supervised machine learning models, comparison was done in between them based on their accuracy (by usual train-test-split).

An overall comparison was also made including all models on which data analysis was performed till now.

The comparison result was found to be as the Decision Tree model being the most accurate among all.

Followed by this result, cross validation scores were stored and compared for each model including the Ensemble Learning Model and Feedforward ANN. The comparison showed that cross validation score for Logistic Regression was highest.

Model	**Accuracy(%)**
DT	91.29
LR	90.72
KNN	91.29
SVM	90.05
MNB	83.94

Table 6.1: Model vs Accuracy

Type	**Model**	**Accuracy(%)**
Ensemble	RF	90.05
ANN	MLP	90.72

Table 6.2: (EnsembleANN) vs Accuracy

Model	**Score(%)**
DT	83.60
LR	89.19
KNN	87.03
SVM	88.62
MNB	84.50
RF	87.13
MLP	86.77

Table 6.3: Model vs Cross Validation Score

As per the proposed solution, the model having highest cross validation score is used for predicting the disease based on input symptoms.

The main aim of the project is to improve the accuracy and efficiency of disease prediction which is to be done by applying ensemble learning techniques and/or feature selection methods. This may result in a model which is more efficient/accurate than our previous results, or it may conclude the same model as efficient however the accuracy of prediction will be improved by a certain percentage, which serves the purpose of research.

6.2 Ensemble Techniques

6.2.1 Bagging

Accuracy rates of DT, LR, MLP, KNN and MNB lie in the range of 83.94%-91.29%, whereas the cross-validation accuracy ranges from 83.60%-89.19%. The DT classifier exhibits best accuracy of 91.29%, whereas the cross-validation accuracy of LR is the highest (89.19%). Bagging has been seen to increase accuracy as well as cross-validation score by upto 3-4%.

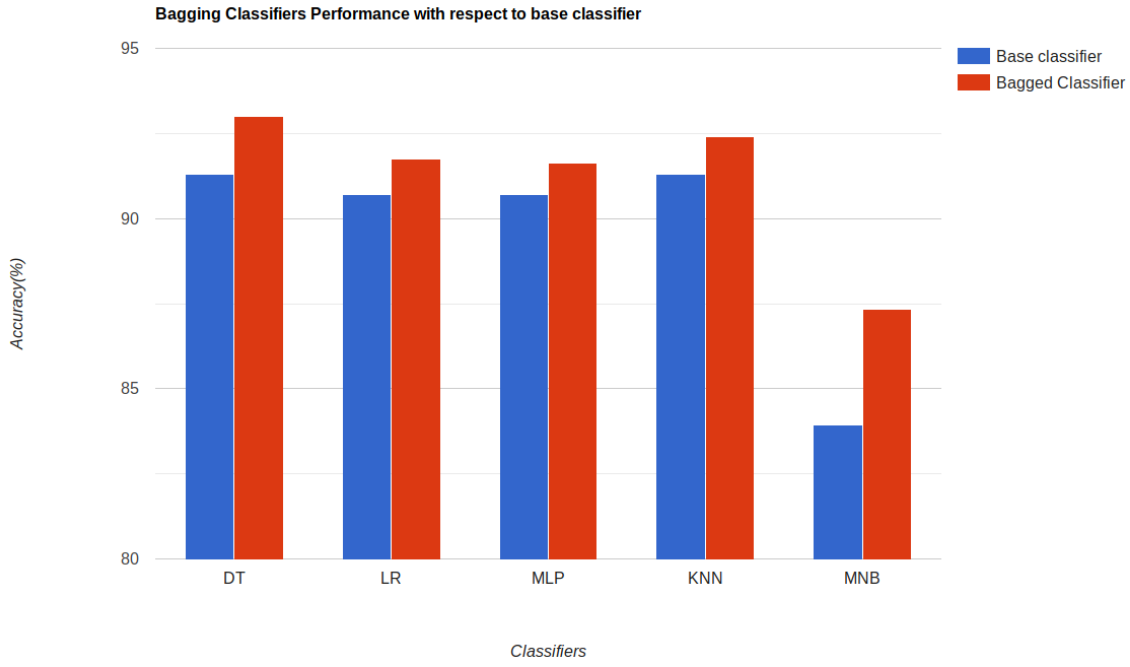

Figure 6.1: Bagging Classifier Performance w.r.t base classifier

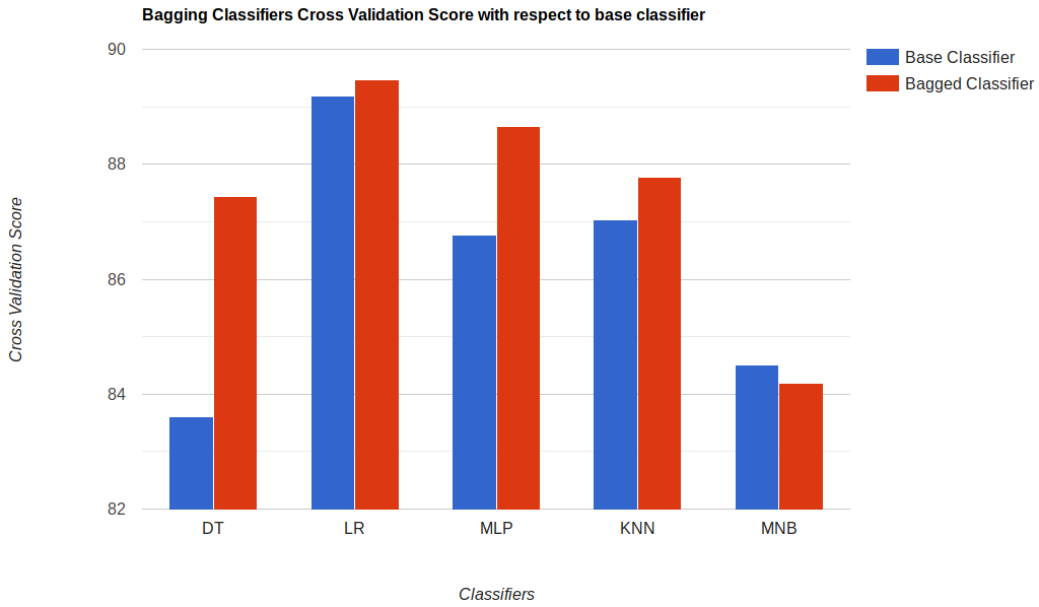

Figure 6.2: Bagging Classifier Cross Validation Score w.r.t base classifier

6.2.2 Boosting

As a part of Boosting, only DT was used for experimentation. The Adaboost model was useful in order to increase accuracy as well as cross-validation score. XGB classifier was also tested for classification - however, there was no increase in the accuracy.

6.2.3 Majority Voting

Another ensemble technique is majority voting, which incorporates several classifiers to increase their accuracy. For our dataset, the proposed method revealed that classifiers were weak with low accuracy. The following is the increasing order of models based on average accuracy (normal + cross validation): MNB, DT, MLP, RF, KNN, SVM, LR, MNB, DT, MLP, RF, KNN, SVM, LR

We can consider 4 strong models to form an ensemble with the remaining 3 weak models each. Note: Random Forests model is an ensemble of Decision Tree. However, it

can be included in the combination to increase accuracy.

Ensemble 1: LR + SVM + KNN + MNB

Ensemble 2: LR + SVM + KNN + MLP

Ensemble 3: LR + SVM + KNN + RF

Ensemble 4: LR + SVM + KNN + DT

It is inferred from Fig. 6.3 that an ensemble of weak classifiers MNB, MLP, RF, DT with strong classifiers using majority voting improves the accuracy of the weak classifier to a considerable extent. Ensembling MNB with the strong classifiers: LR, SVM, KNN improved the accuracy by 7.92%, and cross-validation score by 4.23%. Ensembling MLP with the same strong classifier set improved the accuracy by 0.23%, and cross-validation score by 1.9%. Ensembling RF with the same strong classifier set improved the accuracy by 2.14%, and cross-validation score by 1.45%. Ensembling DT with the same strong set classifier set improved the accuracy by 0.68%, and cross-validation score by 4.69%.

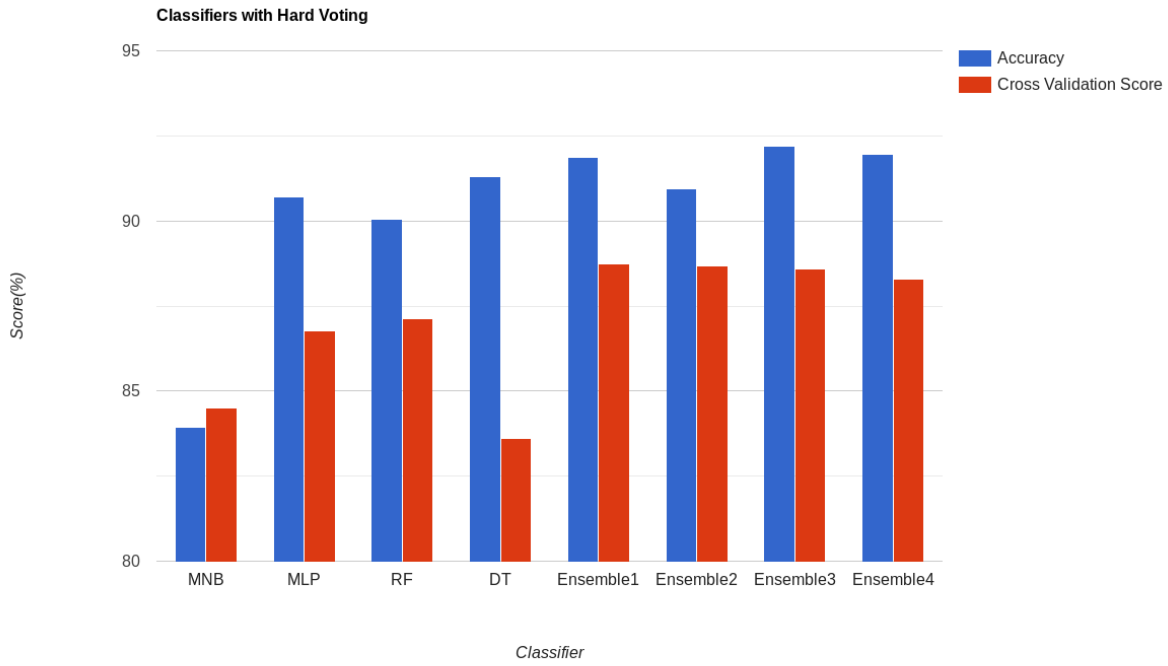

Figure 6.3: Classifier with Hard Voting

6.2.4 Soft Voting

Soft voting is a type of voting scheme in which each individual classifier generates a probability value for a certain data point to show if it is a part of a particular target class. Once the average of all these weighted probabilities is taken, the target label with the highest value wins the vote. It can be inferred from Fig 6.4 that the accuracy for weak classifiers increases more significantly, compared to hard voting (majority voting). However, the jump in the cross-validation score is smaller as compared to the one in case of majority voting.

MNB was ensembled with strong classifiers such as LR, SVM, and KNN, which increased accuracy by 8.59% and the cross-validation score by 3.32%. Using the same strong classifier collection to ensemble MLP increased accuracy by 2.27% and the cross-validation

score by 1.04%. Using the same strong classifier collection to ensemble RF increased accuracy by 2.37% and the cross-validation score by 0.97%. Ensembling DT with the same strong set classifier set improved the accuracy by 1.02%, and cross-validation score by 2.66%.

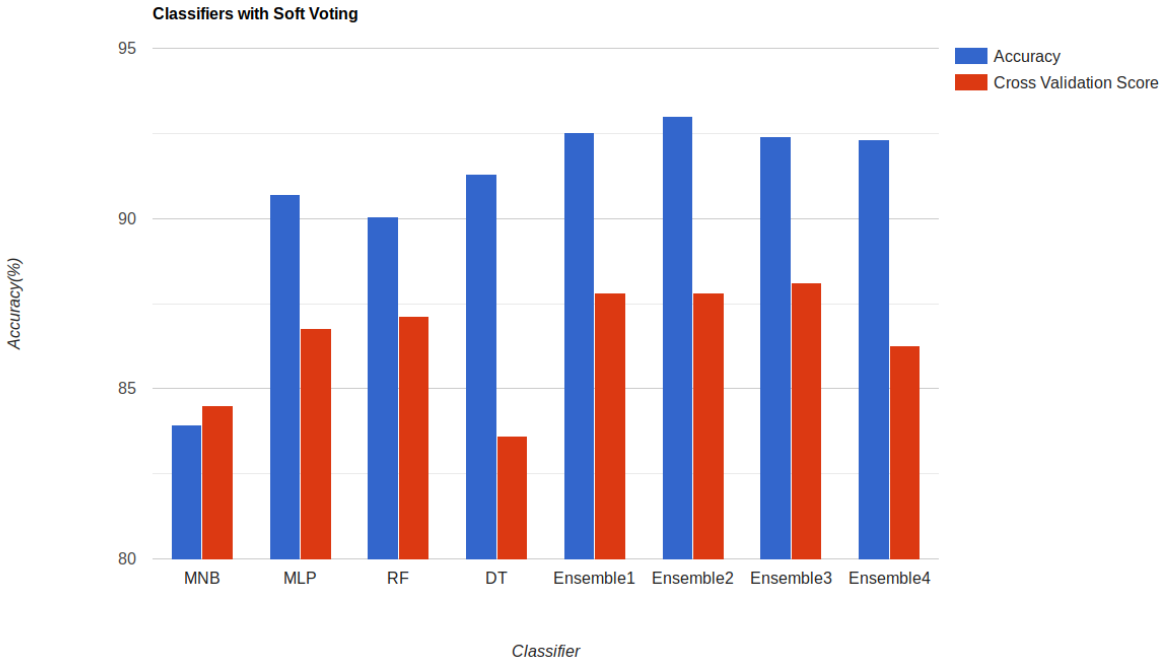

Figure 6.4: Classifier with Soft Voting

6.2.5 Stacking

Stacking is a method used to ensemble classification models. It consists of two-layer estimators, where the first baseline model layer predicts the output of the testing dataset, and the second meta-classifier layer generates further predictions based on the first layer output . In this paper, LR is used as a meta classifier. KNN, SVM, RF, MLP and DT combinations of these classifiers are used as base learners. Stacking base learners KNN, SVM and RF with LR generated an accuracy of 89.82%, and cross-validation score of 84.32%. Stacking base learners KNN, SVM, RF, MLP and DT with LR generated an accuracy of 90.27%, and cross-validation score of 87.09%. Stacking base learners KNN, SVM and MLP with LR generated an accuracy of 90.27%, and cross-validation score of

86.29%. Exception: whatever used in meta classifier can be used in base learner also, to observe the enhancement in accuracy. Following that, we have used KNN, SVM, RF, MLP, LR as base learners, maintaining LR as a meta classifier as well. This gave an accuracy of 90.38%, and cross-validation score of 86.89%.

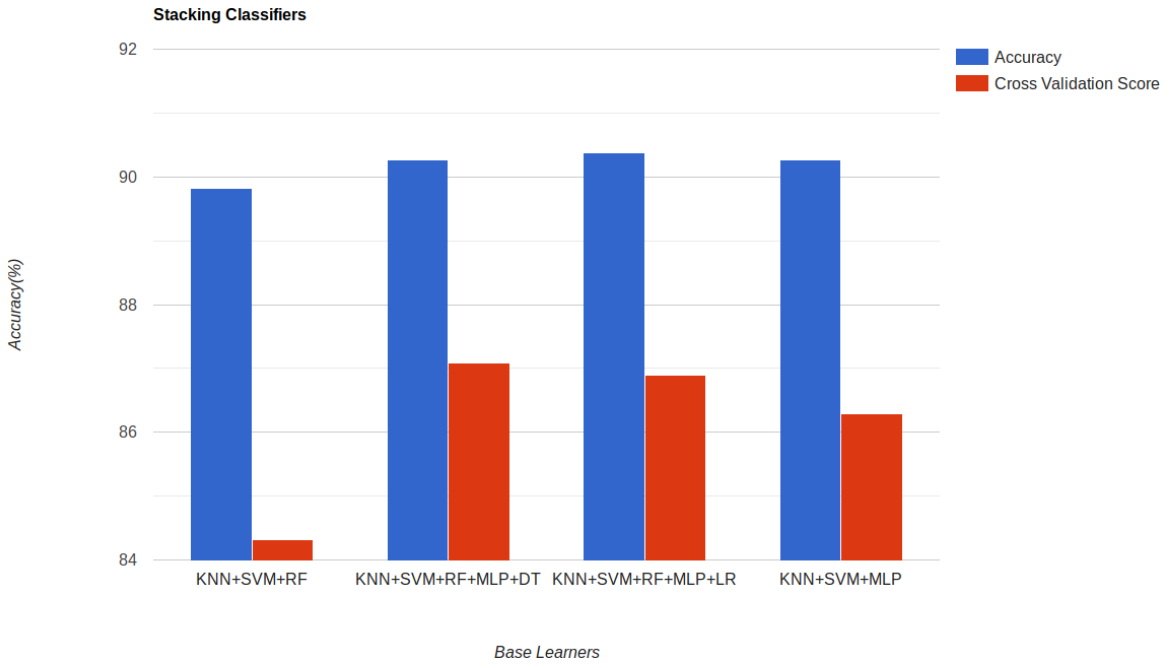

Figure 6.5: Performance of Stacking Ensemble Classifiers

6.3 Feature Selection

Our dataset consists of symptoms as features and diseases as the target variable. As discussed in the proposed methodology, the size of our dataset has been increased by performing combinations. Our dataset is categorical in nature, where each row is represented as a binary vector. Following a recent study that shows possible parameters involved to choose a feature selection method based on the dataset type, since the project contains several categorical inputs and a single categorical output, we went ahead by chi-squared method and mutual information. However, we performed preliminary feature selection

methods of dropping constant features using Variance Threshold method, which is used for removing features having low variance. Initially, train-test split was performed on the dataset for avoiding overfitting. Initially, we followed the primary approach for removing features setting the threshold value to zero. The training set was fit into the same. This resulted in three columns which had zero variance, leading to them being dropped. Following the secondary approach of variance threshold using Pearson Correlation, we generated a heatmap using correlation matrix, from which it could be inferred that there were zero correlation features.

Moving on to Chi-Squared method and information gain using Mutual Information, the former involves selecting K best features, which are determined using feature scores - that in this case, fall in a close range, showing neither a uniform increase nor decrease. The same inference can be applied to the latter. This observation was, as a result of features being the symptoms, which are considered to be discrete. Thus, the best features in such a dataset cannot be determined.

Generally, feature selection involves selecting the best features for improving the performance in terms of accuracy (prediction) and time complexity. But, as discussed previously, only the preliminary method of dropping constant features using variance threshold was applicable. This led to the combination of this feature selection method with the best performing ensemble classifier, in accordance with the aforementioned parameters.

The highest increase of 0.18% in cross-validation score was observed in the case of soft voting ensemble of LR+SVM+KNN+MLP when feature selection was applied to all the ensemble classifiers. However, the highest cross validation score was obtained for

bagged-logistic regression when the obtained feature set was used which was 89.50%.

The results of ensemble classifers combined with feature set obtained are mentioned in below tables.

Bagged Model	**Before FS(%)**	**After FS(%)**
LR	89.47	89.50
DT	87.44	87.52
MLP	88.66	88.75
KNN	87.78	87.65
MNB	84.19	84.25

Table 6.4: Comparison of Cross-Validation Score(%) for Bagged Classifiers before and after Feature Selection

Hard Voting Ensemble	**Before FS(%)**	**After FS(%)**
LR+SVM+KNN+MNB	88.73	88.73
LR+SVM+KNN+MLP	88.67	88.73
LR+SVM+KNN+RF	88.58	88.64
LR+SVM+KNN+DT	88.29	88.25

Table 6.5: Comparison of Cross-Validation Score(%) for Hard Voting Ensembles before and after Feature Selection

Soft Voting Ensemble	**Before FS(%)**	**After FS(%)**
LR+SVM+KNN+MNB	87.82	87.88
LR+SVM+KNN+MLP	87.81	87.98
LR+SVM+KNN+RF	88.10	88.15
LR+SVM+KNN+DT	86.26	86.23

Table 6.6: Comparison of Cross-Validation Score(%) for Soft Voting Ensembles before and after Feature Selection

Chapter 7

Conclusion

The dataset plays an extremely important role in training the model. If we have a number of diseases associated with a particular set of symptoms, it helps keep the system domain sufficiently wide. But at the same time, it should not be skewed or sparse in nature.

It can be observed that Majority Voting (Hard Voting) works well with the weak classifier set, as they show the highest percentage increase with respect to accuracy as well as cross-validation score. However, the highest cross-validation score is seen when Logistic Regression and Bagging are combined, making them the most viable algorithm choices. The feature selection techniques also aid in improving accuracy with regards to the ensemble algorithms.

Chapter 8

Future Scope

The future scope of this project is to integrate with various clinics and hospitals. This will enable the doctors to get a better idea of a patient.

The system can accept input from the user related to family history, and personal details like age, gender, city and other relevant information like blood test reports, which usually the doctor maintains, in order to obtain a relation between these factors with disease onset. For instance, in case of genetic diseases, and hereditary disorders. Other factors like the eating habits, diet, and routine symptoms may also be taken as valuable pieces of information. Region Wise traits also play a factor for disease determination. This can further improve the system's accuracy. Medline Dataset for the mentioned non numerical traits is the recommended dataset for data analysis for the aforesaid concept. Prediction can be made concrete using this concept.

Bibliography

[1] Khan A. Hossain M. Uddin, S. Comparing different supervised machine learning algorithms for disease prediction. *BMC Medical Informatics and Decision Making*, 19:281, 2019.

[2] V. Ranjani M. Durairaj. Data mining applications in healthcare sector: A study. *International Journal of Scientific Technology Research*, 2:29–35, 2013.

[3] Dharuman C Venkatesan Perumal Sharmila, Leoni. Disease classification using machine learning algorithms-a comparative study. *International Journal of Pure and Applied Mathematics*, 114:1–10, 01 2017.

[4] Archana Singh and Rakesh Kumar. Heart disease prediction using machine learning algorithms. In *2020 International Conference on Electrical and Electronics Engineering (ICE3)*, pages 452–457, 2020.

[5] Guilherme Caponetto. Random search vs grid search for hyperparameter optimization, 2019.

[6] Daniel Lowd and Pedro Domingos. Naive bayes models for probability estimation. pages 529–536, 01 2005.

[7] Celestine Iwendi, Suleman Khan, Joseph Anajemba, Ali Bashir, and Fazal Noor. Realizing an efficient iomt-assisted patient diet recommendation system through machine learning model. *IEEE Access*, 8:1–1, 01 2020.

[8] G. Agapito, B. Calabrese, P. H. Guzzi, M. Cannataro, M. Simeoni, I. Caré, T. Lamprinoudi, G. Fuiano, and A. Pujia. Dietos: A recommender system for adaptive diet monitoring and personalized food suggestion. In *2016 IEEE 12th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 1–8, 2016.

[9] Xiao-Yan Gao Et. Al. Improving the accuracy for analyzing heart diseases prediction based on the ensemble method. 2021.

[10] Priya Et. Al Loganathan. A novel intelligent diagnosis and disease prediction algorithm in green cloud using machine learning approach. *Journal of Green Engineering*, 10:3421–3433, 07 2020.

[11] Vimal et al. Jackins, V. Ai-based smart prediction of clinical disease using random forest classifier and naive bayes. *The Journal of Supercomputing*, 77:5198–5219, 2021.

[12] C. Beulah Christalin Latha and S. Carolin Jeeva. Improving the accuracy of prediction of heart disease risk based on ensemble classification techniques. *Informatics in Medicine Unlocked*, 16:100203, 2019.

[13] https://link.springer.com/chapter/10.1007/978-3-319-13728-5_42

[14] http://www.ijecs.in/index.php/ijecs/article/view/1855

Appendix A

List of Abbreviations

1. CSV: Comma Separated Values
2. ANN: Artificial Neural Network
3. RF: Random Forest
4. MLP: Multilayer Perceptron
5. DT: Decision Tree
6. LR: Logistic Regression
7. SVM: Support Vector Machines
8. MNB: Multinomial Naive Bayes
9. KNN: K-Nearest Neighbors
10. ML: Machine Learning
11. AI: Artificial Intelligence