A Group Project Report
on

ADMISSION PREDICTION FOR HIGHER STUDIES IN FOREIGN UNIVERSITIES

Submitted to the Dept. of Information Technology, SNIST
in the partial fulfillment of the academic requirements for the award of

B. Tech (Information Technology)

under JNTUH

by

T. MANOHAR (19311A12P1)

under the guidance of

DR.K. KRANTHI KUMAR
Associate Professor

SNIST
Sreenidhi Institute of Science and Technology
Because, Life is all about taking the right decisions
AUTONOMOUS

Department of Information Technology
School of Computer Science and Informatics
SreeNidhi Institute of Science and Technology (An Autonomous Institution)
Yamnampet, Ghatkesar Mandal, R. R. Dist., Hyderabad – 501301

affiliated to
Jawaharlal Nehru Technological University Hyderabad
Hyderabad – 500 085

2021–2022

Department of Information Technology

School of Computer Science and Informatics
SreeNidhi Institute of Science and Technology

Certificate

This is to certify that the Group project report on “**ADMISSION PREDICTION FOR HIGHER STUDIES IN FOREIGN UNIVERSITIES**” is a bonafide work carried out by **G. Toby Merchant (19311A12N6), Samiuddin Mohammed (19311A12M3), T. Manohar (19311A12P1)** in the partial fulfillment for the award of B. Tech degree in Information Technology, SreeNidhi Institute of Science and Technology, Hyderabad, affiliated to Jawaharlal Nehru Technological University Hyderabad (JNTUH), Hyderabad under our guidance and supervision. The results embodied in the Group project work have not been submitted to any other University or Institute for the award of any degree or diploma

Internal guide	**Project Coordinator**	**Head of the Department**
(Dr. K. Kranthi Kumar)	(Dr. K. Kranthi Kumar)	(Dr. Sunil Bhutada)

External Examiner
Date:

DECLARATION

We, **G. Toby Merchant (19311A12N6), Samiuddin Mohammed (19311A12M3), T. Manohar (19311A12P1)**, students of SreeNidhi Institute of Science and Technology, Yamnampet, Ghatkesar, studying III year II semester, Information Technology solemnly declare that the Group project work, titled **"ADMISSION PREDICTION FOR HIGHER STUDIES IN FOREIGN UNIVERSITIES**" is submitted to SreeNidhi Institute of Science and Technology for partial fulfillment for the award of degree of Bachelor of technology in Information Technology.

It is declared to the best of our knowledge that the work reported does not form part of any dissertation submitted to any other University or Institute for award of any degree.

Acknowledgements

We would like to express our immense gratitude and sincere thanks to **Dr. K. Kranthi Kumar, Associate Professor** in Information Technology for his guidance, valuable suggestions and encouragement in completing the Group Project work within the stipulated time.

We would like to express our sincere thanks **Dr. C.V. Tomy**, Executive Director **Dr. Ch. Shiva Reddy,** Principal**, Dr. Sunil Bhutada**, Head of the Department of Information Technology, **Dr. K. Kranthi Kumar**, Associate Professor & Group project work coordinator of the Department of Information Technology, SreeNidhi Institute of Science and Technology (An Autonomous Institution), Hyderabad for permitting us to do our Group project work.

Finally, we would also like to thank the people who have directly or indirectly helped us and parents and friends for their cooperation in completing the project work.

G. Toby Merchant	**(19311A12N6)**
Samiuddin Mohammed	**(19311A12M3)**
T. Manohar	**(19311A12P1)**

ABSTRACT

In India every year lakhs of students are getting the graduation degree and willing to join post-graduation in other countries. Newly graduate students usually are not knowledgeable of the requirements and the procedures of the postgraduate admission and might spend a considerable amount of money to get advice from consultancy organizations to help them identify their admission chances. Human consultants and calculations might be biased and inaccurate. This project helps on predicting the eligibility of Indian students getting admission in best university based on their Test attributes like GRE, TOEFL, LOR, SOP, CGPA, University rating and Research. According to their scores the possibilities of chance of admit is calculated, but with the growth of Machine Learning methods, we have got the flexibility to search out an answer to the current issue. The present system focuses on the prediction whether a student’s score is appropriate or not by using algorithms such as Adaboost, Catboost, Support Vector Machine, Naive Bias. Random forest, decision tree and linear regression algorithms are used for predicting this model. This algorithm is trained and tested for predicting the admission for the student.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ML	MACHINE LEARNING
DL	DEEP LEARNING
SVM	SUPPORT VECTIOR MACHINE
ANN	ARTIFICIAL NEURAL NETWORK
GRE	GRADUATE RECORD EXAMINATION
TOEFL	TEST OF ENGLISH FOR FOREIGN LANGUAGE
SOP	STATEMENT OF PURPOSE
LOR	LETTER OF RECOMMENDATION
CGPA	CUMULATIVE GRADE POINT AVERAGE
DNN	DEEP NEURAL NETWORK
MSE	MEAN SQUARE ERROR
MAE	MEAN ABSOLUTE ERROR

INDEX

CHAPTER 1

INTRODUCTION

1.1 ADMISSION PREDICTON

Machine learning mainly solves classification and regression problems, which is an important research area and very useful in making decision. Both of classification and regression algorithms aim to evaluate unknown data by training labeled data sample. But they are still different. The output of the former is discrete, which means we separate data to a few classes by classification algorithm then evaluate new coming data belongs to which class. The output of the latter is continuous, which means we learn from old data then evaluate the probability of an event happening.

Prediction of USA graduate admission for Indian undergraduate appliers" based on regression algorithms. The main content is to show the process from analyzing data, making model to result evaluation. At the same time, the principles of the methods used in this project. In section we will explain my dataset and introduce the research background. We will give brief introduction about how I organize my project and the main algorithms used in my project, the completed work will be provided including selecting base model and other models fitting, and the methods of data pre-processing and best parameter search will also be mentioned in the following sections. We will also provide details about how to optimize model by stacking and analyses the results.

The dataset contains student's data and includes 7 features as predictors such as "GRE score", "TOEFL score", "University Rating", "SOP" (Statement of Purpose)," LOP" (Letter of recommendation strength)," CGPA" (out of 10)," Research" and Chance of Admission" (float datatype, ranging from 0 to 1). "Chance of admit" were evaluated by appliers. This dataset has various use. Many people have provided solutions to this problem. Most people take this problem as classification problem (The student can be admitted or not), generally, they get very high score (up to 96%). Some people take it as admission line case. They analyze the minimal line for every feature to make the admission possibility over 90%.

The most common used algorithms are Adaboost, Catboost, Linear Regression, Random Forest, Multiple Linear Regression. Linear Regression performs the best among them the projects. Importing numpy and pandas library to deal with data. Matplotlib and seaborn associate analysis by visualizing data. Sklearn is the strongest library to call popular algorithms package to train data and evaluate results. Although these tools have some limitation and not super flexible, but for basic usage they are enough.

1.2 OBJECTIVE

400 applicants have been surveyed as potential students for admission. The university weighs certain aspects of a student's education to determine their acceptance.

The objective is to explore what kind of data is provided, determine the most important factors that contribute to a student's chance of admission, and select the most accurate model to predict the probability of admission.

CHAPTER 2

RELATED WORK

2.1 INTRODUCTION

The existing system has the data about the prediction system of the students about the algorithms that are used in the project that is in the base paper. The algorithms that are present increase the accuracy and gives us the outcome as the predictive analysis of the project. The existing system is implemented by using the algorithms such as Adaboost algorithm [1][14][15], Cat boost algorithm [2], Artificial neural network [3][18][19], Support Vector Machine [4] and Naive Bias [5].

2.2 EXISTING SYSTEM

This paper establishes a machine learning model, which takes into account boundaries such as GRE Score, TOEFL Score, University Ranking, the Proposal Statement and Recommendation Letter Power, Undergraduate GPA and Study Experience. After getting all the inputs, it predicts the chance of admission. On obscure test occasions, the prepared model has substantial factual findings for the like estimate of the probability of confirmation and, accordingly, offers an unprejudiced impression of measurement.

In the model development, the dataset is consistently split into train and test set of 80% and 20%. Train set has 400 profiles and test set has 100 profiles. The dataset used for modelling looks like this. The data was pre-processed and split into two classes at random: a training set and a testing set. We selected 80 percent of the 7976 entries in our dataset as our training collection.

The variable to be predicted is Chance of Admit. The steps involved in model development are mentioned below. As cutoffs of universities changes year to year, we have put a condition in the code that GRE score should be greater than 250 and TOEFL Score should be greater than 50 and CGPA should be greater than 5 and all other conditions. The training dataset is used training the model using cat boost algorithm [2][16][17].

Adaboost algorithm [1] is an iterative algorithm, which is a strong classifier composed of different weak classifiers. The weights of each sample are determined by whether the samples are classified correctly, and the correct rate of the overall classification in the last time.

Then, the new sample data with weights is transferred to the lower level classifier to train. At this time, the classifiers we got are weak classifiers. Finally, the weak classifiers are obtained by several iterations, and a final classifier is obtained by the voting strategy. The whole process is as follows:

1) Firstly, study N samples, and obtain the first weak classifier.

2) A new group of N training samples is composed of the wrong samples and the other new data. The second weak classifiers are obtained through the learning of this sample group.

3) The wrong samples are divided in step I and step 2 with other new samples, which constitute a new group of N training samples. Then, the third weak classifier was obtained by learning these samples.

4) Finally, a strong classifier is generated, which is combined by several weak classifiers.

To complete the study, the graduate admission dataset has been split into training dataset and testing dataset. Data normalization has been performed to accelerate the training process of the DNN [6][17] model. Using the training dataset, the DNN model has been trained with optimal hyper parameter. It has been assessed through some standard benchmarking. The outcomes of the DNN model have also been compared with the existing methods.

2.3 APPLICATION

Through the above experiment and analysis, it can be concluded that AdaBoost prediction model has a good effect on predicting the admission line of CEE [2]. The correctness of result is verified by experiment. This model considers the enrollment plan, number of applicants, the difficulty of test question, and other factors. The result of this prediction model has important reference value to the students who take part in college entrance examination. Of course, the establishment of the model is not very perfect because of the in-exhaustive data. And there are still a lot of things to be improved. And in the aspect of feature selection, we only forecast the college entrance examination of province Sichuan. If getting more data in the future, we can also do universities admission line forecast. It is also a very significance work.

2.4 CONCLUSION

Existing methodology and couple of algorithms and its implementation is discussed and also steps involved in training the model are discussed. Finally, there is an accuracy of 85 with cat boost algorithm and is the highest quantitative result of a confirmatory incentive expectation model till now.

CHAPTER 3

PROPOSED SYSTEM

3.1 INTRODUCTION

The proposed system predicts the admission with maximum accuracy. We shall talk about various machine learning, the algorithm which can help in decision making and prediction. We shall use more than one algorithm to get better accuracy of prediction. The admission dataset is given to the system which is then pre-processed so that the data is in a useable format for analysis. If the dataset is not structured or if the dataset is or if the dataset is huge or it has irrelevant features, we shall use feature extraction to extract the data. After this the data is trained and we apply a relevant machine learning algorithm of either decision tree or random forest to the dataset. And the file system is developed to store the user-given input and the predicted value.

3.2 PROPOSED SYSTEM ARCHITECTURE DESIGN

A system architecture is a conceptual model that defines the structure, behavior, and views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures and behaviors of the system. The architecture that is shown in the Fig 1 describes about the steps that are used in the project to implement.

The implementation stage of any project is a true display of the defining moments that make a project a success or a failure. The implementation stage is defined as the system or system modifications being installed and made operational in a production environment. The phase is initiated after the system has been tested and accepted by the user. This phase continues until the system is operating in production in accordance with the defined user requirements.

The Fig 1 shows us a clear picture about the implementation and the steps of the project work. This model will compare the algorithms and then select the most efficient algorithm and predicts the value for better results.

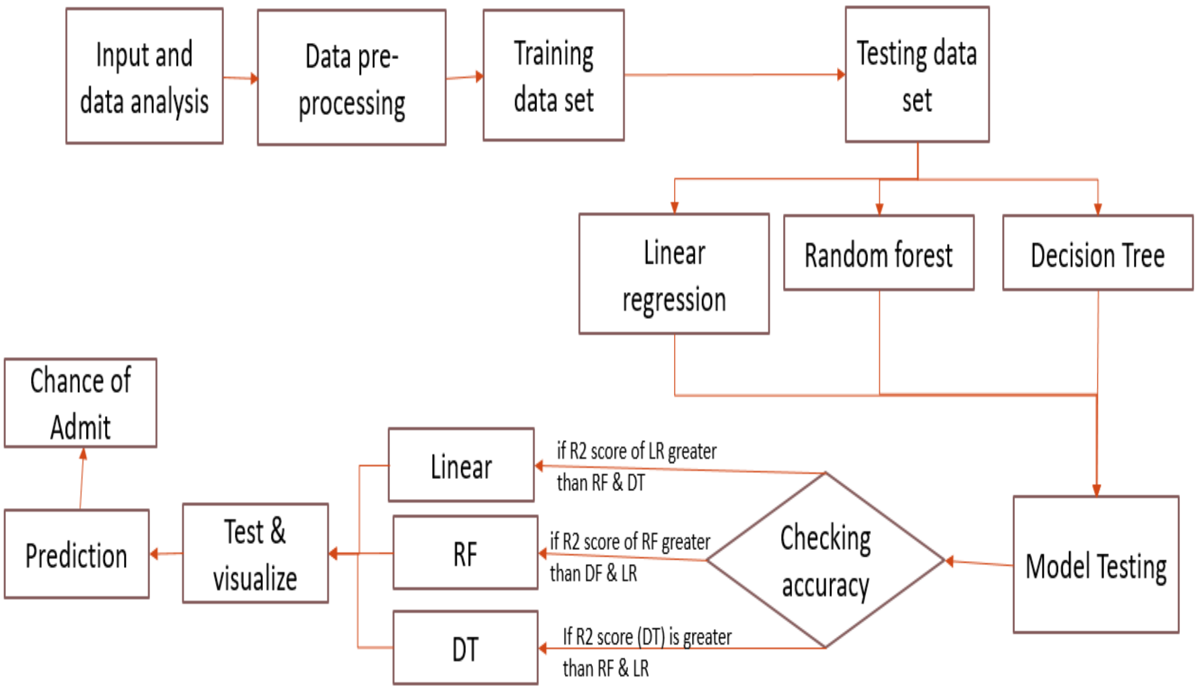


Fig 1: Architecture for Proposed System

3.2.1 INPUT AND DATA ANALYSIS

This operation helps to import the appropriate dataset to the model that is described and perform various operations.

3.2.1.1 IMPORTING DATASET

A dataset in machine learning is quite simply a collection of data pieces that can be treated by a computer as a single unit for analytic and prediction purposes. This means that the data collected should be made uniform and understandable for a machine that doesn't see data the same way as humans do. We import the dataset to analyze the details and predict the desired output.

3.2.2 DATA PREPROCESSING

This operation helps to remove the unnecessary details and processes it without any errors. Every step in the Machine learning is important as there are lots of operations while sorting the dataset. If there are any missing values in the dataset this operation helps to remove those missing values. By this there is a chance of reducing the error to an extent.

3.2.2.1 CHECKING THE CORRELATIONS

The correlation coefficient is determined by dividing the covariance by the product of the two variables' standard deviations. Standard deviation is a measure of the dispersion of data from its average. If the correlation is checked the accuracy of the model is increased.

3.2.2.2 DATA CLEANING AND ANALYSIS

In this operation you clean the noisy data which is unnecessary for the required output. Inspecting feature values that help identify what needs to be done to clean or pre-process until you see the range or distribution of values typical of each attribute.

You may find missing or noisy data or anomalies such as the incorrect data form used for a column, incorrect measuring units for a particular column or that there are not enough examples of a specific class.

There are no missing values and outliers because we analyzed the data, so for this data there is no need to fill the missing values and deal with outliers. If there are any missing values and outliers we can fill or drop using the fill method and drop method and we can also standardize the data using the minimax scaler, if necessary.

3.2.2.3 DATA VISUALIZATION

After analyzing the data from the dataset which is processed, we will be able to know what the features and labels are, so that from the above data, the label we have to consider is Chance of Admission and then we have to consider the parameters that influence or play a major role in Chance of Admission

We can get to know certain features that are more affected by the visualization or analysis or the use of feature importance method in decision tree. The process of finding trends and correlations in our data by representing it pictorially is called Data Visualization. To perform data visualization in python, we can use various python data visualization modules [7] such as Matplotlib, Seaborn, Plotly, etc. Data visualization plays a significant role in the representation of both small and large data sets, but it is especially useful when we have large data sets, in which it is impossible to see all of our data, let alone process and understand it manually.

3.2.2.4 REMOVING OUTLIERS

In this step we check that, from these charts it looks like we have no missing values. It seems as though Serial No. is just an index for students, which we can take out. Two columns also have an added space in the label which we'll take out We are also removing the blank spaces

3.2.3 SPLIT TRAINING AND TESTING DATA

In machine learning, data splitting is typically done to avoid overfitting [8]. That is an instance where a machine learning model fits its training data too well and fails to reliably fit additional data. The original data in a machine learning model is typically taken and split into different sets. The sets commonly used are the training set and the testing set.

Data should be split so that data sets can have a high amount of training data. For example, data might be split at an 80-20 or a 70-30 ratio of training vs. testing data. The exact ratio depends on the data. The training set is the portion of data used to train the model. The model should observe and learn from the training set, optimizing any of its parameters. In this project we have used 80% of the dataset as the training set. The testing set is the portion of data that is tested in the final model and is compared against the previous sets of data. The testing set acts as an evaluation of the final mode and algorithm. In this project we have used 20% of the dataset as the training set.

3.2.3.1 TRAINING LINEAR REGRESSION MODEL

Here, we import the linear regressor [9] to the model. Simple linear regression is a regression technique in which the independent variable has a linear relationship with the dependent variable. The straight line in the diagram is the best fit line. The main goal of the simple linear regression is to consider the given data points and plot the best fit line to fit the model in the best way possible. In the same way the linear regression sets a best line to predict data

3.2.3.2 TRAINING DECISION TREE MODEL

In this step the decision tree regressor [10] is imported to process the data. Decision Tree is a supervised learning method used in data mining for classification and regression methods. It is a tree that helps in decision making purposes. The decision tree creates classification or regression models as a tree structure. It separates a data set into smaller subsets, and at the same time, the decision tree is steadily developed. The final tree is a tree with the decision nodes and leaf nodes. A decision node has at least two branches. The leaf nodes show a classification or decision. We can't accomplish more split on leaf nodes. The uppermost decision node in a tree that relates to the best predictor called the root node. Decision trees can deal with both categorical and numerical data. Our project deals with the numerical data, as per the prior knowledge the algorithm choses the best possible way that can execute the desired output.

3.2.3.3 TRAINING RANDOM FOREST MODEL

In this step we perform the random forest algorithm [11] by just importing it into the model. Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML [12]. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

As the name suggests, Random Forest is a classifier that contains a number of decision trees on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting. This algorithm helps in the prediction of data through the bagging method, where the predictions are taken from various decision trees.

3.2.4 CREATING PICKLE FILE

The pickle module implements binary protocols for serializing and de-serializing a Python object structure [13]. “Pickling” is the process where a Python object is converted into a byte stream, and “unpickling” is the inverse operation, where a byte stream (from a binary file or bytes-like object) is converted back into an object hierarchy. Pickling (and unpicking) is alternatively known as “serialization”, “marshalling,” or “flattening”; however, to avoid confusion, the terms used here are pickling and unpickling. The required code for the interface which is used in the project is through this pickle file. It links the model to an application.

3.2.5 IMPLEMENT PYTHON FILE FOR BUILDING WEB FRAMEWORK

To implement python file for building webpage we use this operation and let us see the operations which are explained below. Flask is a small and lightweight Python web framework that provides useful tools and features that make creating web applications in Python easier. It gives developers flexibility and is a more accessible framework for new developers since you can build a web application quickly using only a single Python file.

It uses the Bootstrap toolkit to style the application so that it is more visually appealing. Bootstrap helps us to incorporate responsive web pages in the web application so that it also works well on mobile browsers without writing own HTML, CSS, and JavaScript code to achieve these goals. The toolkit will allow to focus on learning how Flask works. Flask uses the Jinja template engine to dynamically build HTML pages using familiar Python concepts such as variables, loops, lists, and so on.

3.2.6 IMPLEMENTING HTML FOR WEBPAGE

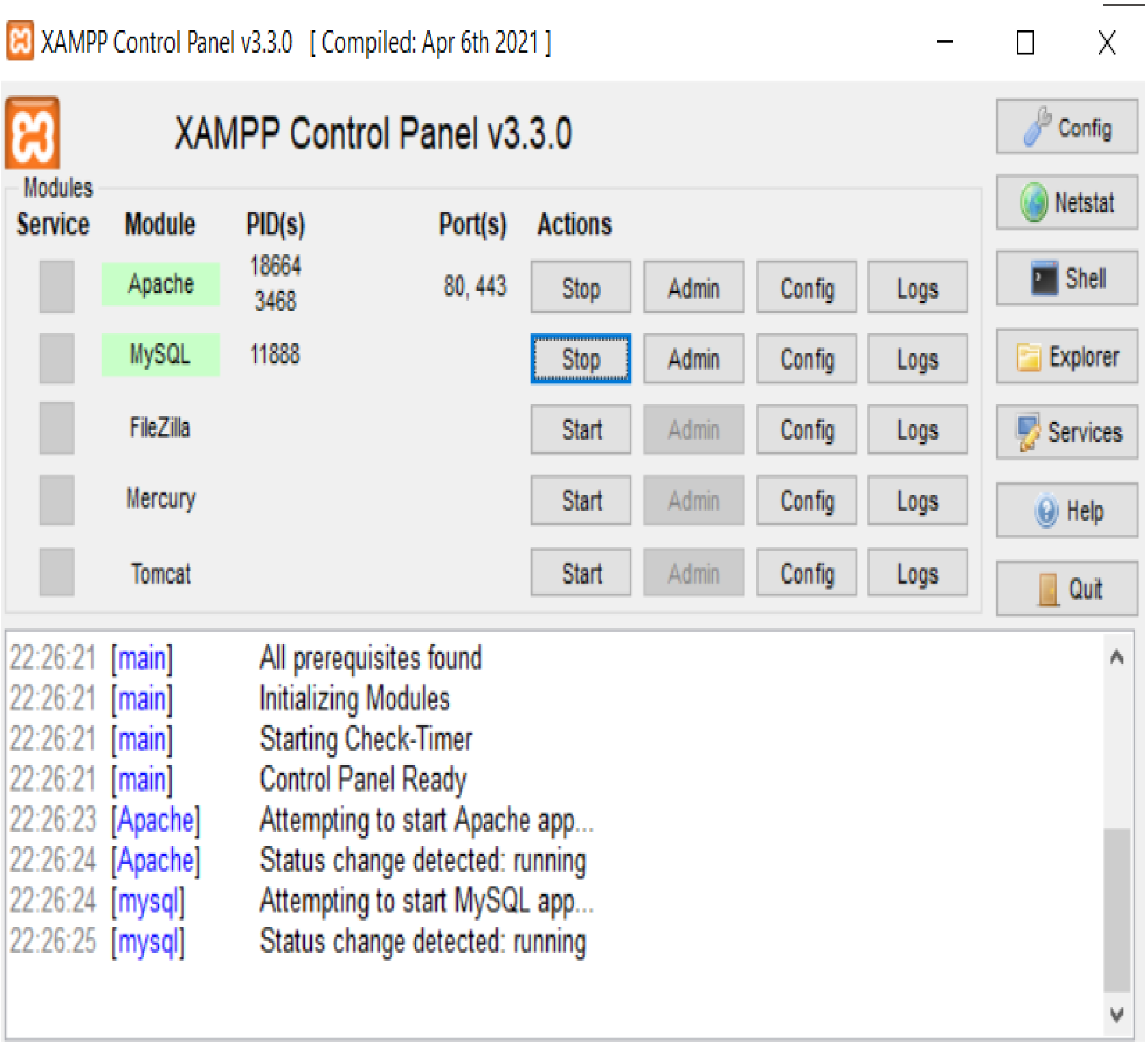
The HTML page is required for the representation of the work and to provide an interface to the user. The Hyper Text Mark-up Language or HTML is the standard markup language for documents designed to be displayed in a web browser. It can be assisted by technologies such as Cascading Style Sheets (CSS) and scripting languages such as JavaScript.

Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.

3.2.7 FILE SYSTEM SERVICE

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages[14]. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible. The Fig 2 shows the modules that needed to be launched to create the SQL database. This helps us to store the data entered to check the results of the scores.

In XAMPP a database is created to store the values. In relational database a table is created to store the user given input and the predicted values.


XAMPP Control Panel v3.3.0 [Compiled: Apr 6th 2021]
XAMPP Control Panel v3.3.0
Modules
Config
Netstat
Shell
Explorer
Services
Help
Quit

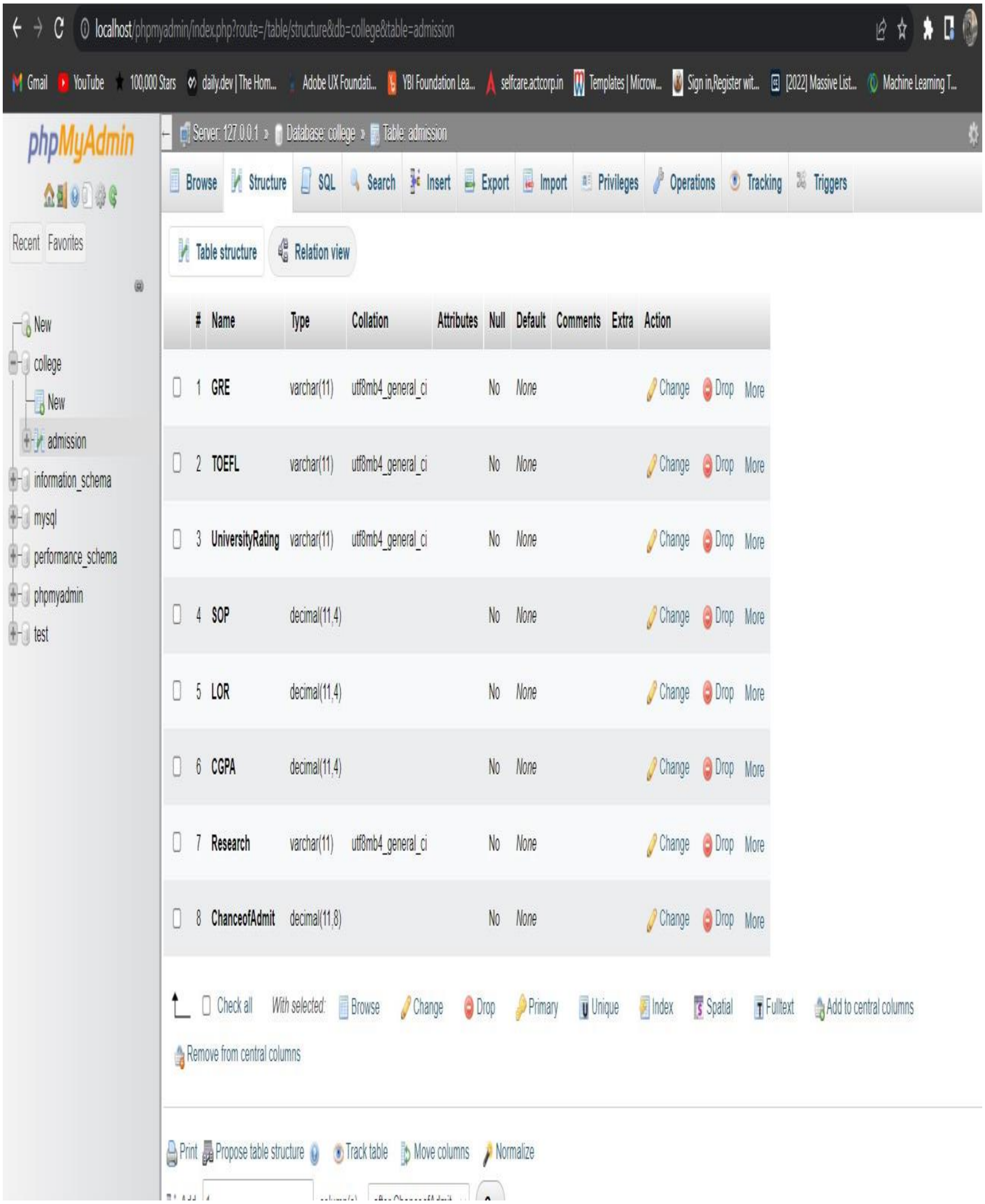
Service	Module	PID(s)	Port(s)	Actions			
□	Apache	18664 3468	80, 443	Stop	Admin	Config	Logs
□	MySQL	11888		Stop	Admin	Config	Logs
□	FileZilla			Start	Admin	Config	Logs
□	Mercury			Start	Admin	Config	Logs
□	Tomcat			Start	Admin	Config	Logs

```
22:26:21  [main]    All prerequisites found
22:26:21  [main]    Initializing Modules
22:26:21  [main]    Starting Check-Timer
22:26:21  [main]    Control Panel Ready
22:26:23  [Apache]  Attempting to start Apache app...
22:26:24  [Apache]  Status change detected: running
22:26:24  [mysql]   Attempting to start MySQL app...
22:26:25  [mysql]   Status change detected: running
```

Fig 2: Screenshot of XAMPP Control Panel

3.2.8 SQL CREATION

In XAMPP a database is created to store the values. In relational database a table is created to store the user given input and the predicted values[14]. The Fig 3 shows about the creation of the SQL and the attribute values that are stored. We have also created different datatypes to insert where it can predict the values with correct figures. As you can also see in the Fig 3 that describes the implementation of the SQL.



	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
□	1	GRE	varchar(11)	utf8mb4_general_ci		No	*None*			Change Drop More
□	2	TOEFL	varchar(11)	utf8mb4_general_ci		No	*None*			Change Drop More
□	3	UniversityRating	varchar(11)	utf8mb4_general_ci		No	*None*			Change Drop More
□	4	SOP	decimal(11,4)			No	*None*			Change Drop More
□	5	LOR	decimal(11,4)			No	*None*			Change Drop More
□	6	CGPA	decimal(11,4)			No	*None*			Change Drop More
□	7	Research	varchar(11)	utf8mb4_general_ci		No	*None*			Change Drop More
□	8	ChanceofAdmit	decimal(11,8)			No	*None*			Change Drop More

Fig 3: Screenshot of SQL Creation

3.3 PROPOSED SYSTEM ALGORITHM

```
Algorithm: Admission Prediction
Input: User given input
Output: Prediction of Admit
Begin
    1. Import dataset
    2. Pre-process data
    3. Split Dataset
    4. Train the model
        4.1 Import Decision tree regressor
            4.1.1 Train the model
        4.2 Import Random Forest
            4.2.1 Train the model
        4.3 Import Linear Regressor
    5. Test the model using testing data
        5.1 Test the model
    6. Finding the accuracy.
        6.1 Mean absolute error
        6.2 R2 score
    7. User input is given for the output
End
```

3.3.1 LINEAR REGRESSION

Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as GRE score, TOEFL score, CGPA, etc.

Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable. The linear regression model provides a sloped straight line representing the relationship between the variables. Consider the below image Fig 5. The linear regression is calculated with the Eq (1) as mentioned below. The graph which is mentioned in Fig 5 represents the data points a and line of regression b, it provides the straight line representing the relationship

$$y = a + bX \tag{1}$$

Where,

y= dependent variable

X= Independent variable

a = y intercept

b = Slope of the line

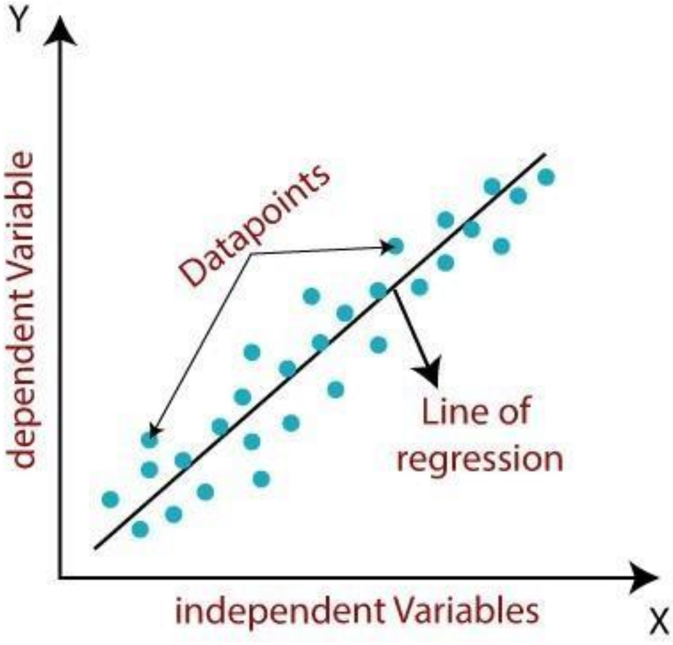


Fig 4: Linear Regression Graph

3.3.2 RANDOM FOREST

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Classification and Regression problems in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model. As the name suggests, "Random Forest is a classifier that contains a number of decision trees `on various subsets of the given dataset and takes the average to improve the predictive accuracy of that dataset." Instead of relying on one decision tree, the random forest takes the prediction from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of overfitting.

3.3.3 DECISION TREES

Decision Tree is a supervised learning method used in data mining for classification and regression methods. It is a tree that helps us in decision-making purposes. The decision tree creates classification or regression models as a tree structure. It separates a data set into smaller subsets, and at the same time, the decision tree is steadily developed. The final tree is a tree with the decision nodes and leaf nodes. A decision node has at least two branches. The leaf nodes show a classification or decision. We can't accomplish more split on leaf nodes-The uppermost decision node in a tree that relates to the best predictor called the root node. Decision trees can deal with both categorical and numerical data.

A. Entropy: Entropy refers to a common way to measure impurity. In the decision tree, it measures the randomness or impurity in data

B. Information Gain: Information Gain refers to the decline in entropy after the dataset is split. It is also called Entropy Reduction. Building a decision tree is all about discovering attributes that return the highest data gain.

C. Standard Deviation: The standard deviation of a random variable, sample, statistical population, data set, or probability distribution is the square root of its variance. A useful property of the standard deviation is that, unlike the variance, it is expressed in the same unit as the data. The below Eq (2) represents the standard deviation.

$$S(Y,X) = \sum_{c \in X} P(c)S(c) \tag{2}$$

Where,

Y id the target variable

X is the predictor

P(c) is the probability of the feature

S(c) is the standard deviation of the feature

D. Standard Deviation Reduction: Standard deviation reduction is the standard deviation of the target variable subtracted from the standard deviation of predictors, so higher standard deviation reduction means more homogeneity in data that will help to identify predictor variables for splits. The below Eq (3) represents the standard deviation reduction.

$$SDR(Y,X) = SD(Y) - S(Y,X) \tag{3}$$

Where,

Y id the target variable

X is the predictor

SDR is the Standard deviation reduction

SD is the Standard deviation

In short, a decision tree is just like a flow chart diagram with the terminal nodes showing decisions. Starting with the dataset, we can measure the entropy to find a way to segment the set until the data belongs to the same class. It enables us to analyze the possible consequences of a decision thoroughly. It provides us a framework to measure the values of outcomes and the probability of accomplishing them. It helps us to make the best decisions based on existing data and best speculations.

3.4 IMPLEMENTATION

To implement the project there should be a proper database setup, software requirements and hardware requirements.

3.4.1 DATABASE SETUP

The dataset contains student’s data and includes 7 features as predictors such as “GRE score”, “TOEFL score”, “University Rating”, “SOP” (Statement of Purpose), “LOP” (Letter of recommendation strength), “CGPA” (out of 10),” Research experience” and “Chance of Admission”. “Chance of admit” is evaluated by users as mentioned in the Table 1.

Table 1: Sample of Dataset

Serial No.	GRE Score	TOEFL Score	University Rating	SOP	LOR	CGPA	Research	Chance of Admit
1	337	118	4	4.5	4.5	9.65	1	0.92
2	324	107	4	4	4.5	8.87	1	0.76
3	316	104	3	3	3.5	8	1	0.72
4	322	110	3	3.5	2.5	8.67	1	0.8

5	314	103	2	2	3	8.21	0	0.65
6	330	115	5	4.5	3	9.34	1	0.9
7	321	109	3	3	4	8.2	1	0.75
8	308	101	2	3	4	7.9	0	0.68
9	302	102	1	2	1.5	8	0	0.5
10	323	108	3	3.5	3	8.6	0	0.45
11	325	106	3	3.5	4	8.4	1	0.52
12	327	111	4	4	4.5	9	1	0.84
13	328	112	4	4	4.5	9.1	1	0.78
14	307	109	3	4	3	8	1	0.62

The dataset contains information about a student's:

1. GRE Score
2. TOEFL Score
3. University Ratings
4. Statement of Purpose Score
5. Letter of Recommendation Score
6. CGPA
7. Whether the Student Has Done Any Research
8. Chance of Admission

3.4.2 EXPERIMENTAL SETUP

In the experimental setup the software requirements such as Windows 10 Operating System, Python 3.7 and above version required, and several libraries like pandas, NumPy, matplotlib, seaborn, and flask are required and used. The hardware requirements tested for this project are 8GB RAM, 240GB HDD, and 2.5Gz minimum per core required here. By this experimental setup we can achieve the performance in an efficient way.

3.5 DESIGN

3.5.1 UML DIAGRAMS

A UML diagram is a diagram based on the UML (Unified Modeling Language) with the purpose of visually representing a system along with its main actors, roles, actions, artifacts or classes, in order to better understand, alter, maintain, or document information about the system [15].

3.5.1.1 CLASS DIAGRAM

Admission prediction using machine learning consist of class diagram that all the other application that consists the basic class diagram, here the class diagram is the basic entity that is required in order to carry on with the project. Class diagram consist information about all the classes that is used and all the related datasets which is mentioned in Fig 5.

A. PURPOSE OF CLASS DIAGRAMS

The class diagram is used to analyze the following they are, analysis and design of the static view of an application. It can describe responsibilities of a system. The base for component and deployment diagrams. Forward and reverse engineering

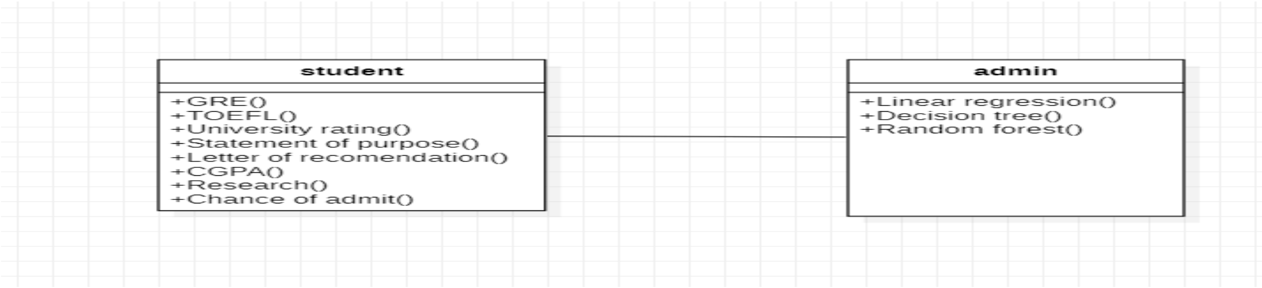


Fig 5: Class Diagram for Proposed System

3.5.1.2 USE CASE DIAGRAM

The Use Case diagram of the admission prediction using machine learning consist of all the various aspects a normal use case diagram requires. This use case diagram shows in Fig 6 that how from starting the model flows from one step to another, like he enter into the system then enters all the information’s and all other general information along with the symptoms that goes into the system, compares with the prediction model and if true is predicts the appropriate results otherwise it shows the details where the user if gone wrong while entering the information's and it also shows the appropriate precautionary measure for the user to follow.

A. PURPOSE OF USE CASE DIAGRAMS

In this we discuss the purpose of the use case diagrams they are used to gather the requirements of a system and also to get an outside view of a system. It identifies the external and internal factors influencing the system. It can also show the interaction among the requirements are actors.

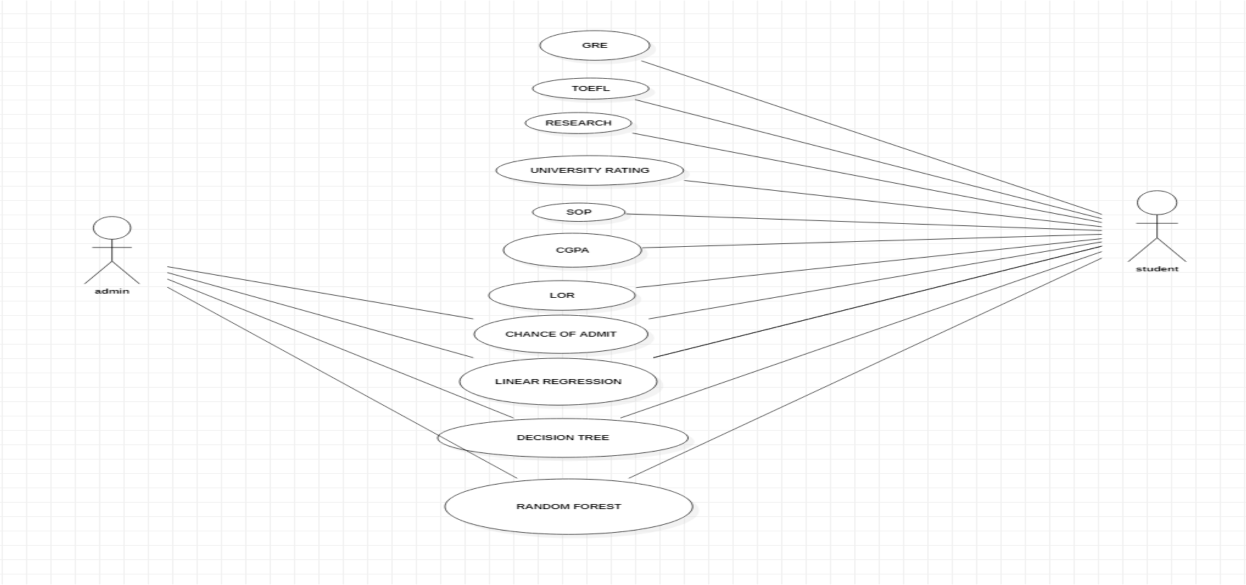


Fig 6: Use Case Diagram for Proposed System

3.5.1.2 SEQUENCE DIAGRAM

The Sequence diagram of the project Admission prediction using machine learning consist of all the various aspects a normal sequence diagram requires as mentioned in the Fig 7 This sequence diagram shows how from starting the model flows from one step to another, like he enters into the system then enters all the information and all other general information along with the symptoms that goes into the system, compares with the prediction model and if true is predicts the appropriate results.

A. PURPOSE OF SEQUENCE DIAGRAM

The admission prediction has a sequence diagram where it can discuss about the model the flow of control by time sequence. To model the flow of control by structural organizations. For forward engineering and reverse engineering.

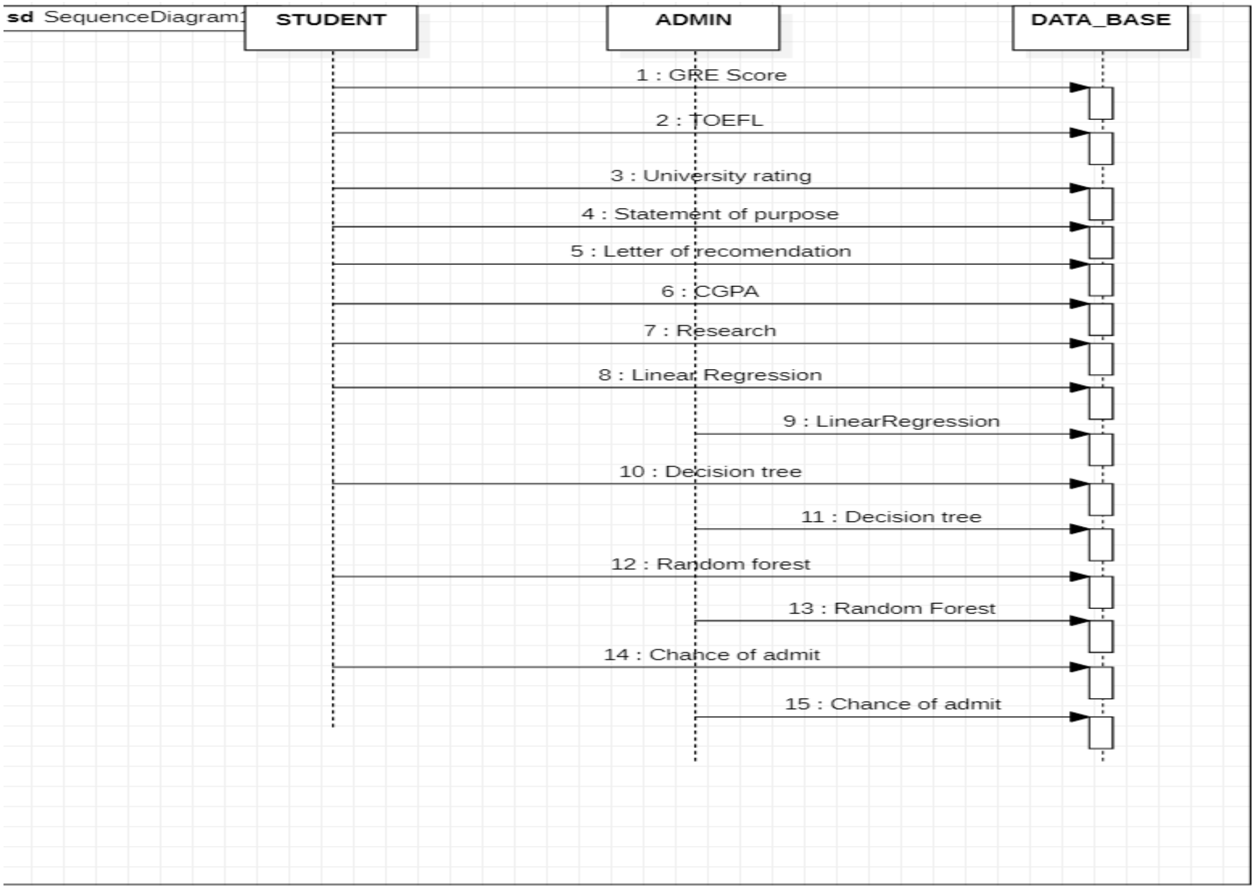


Fig 7: Sequence Diagram for Proposed System

3.6 CONCLUSION

In this chapter we can see the usage of the algorithms. The experts had proposed some models and executed it. In this proposed system we have used algorithms to predict the data. They have also published some conference and research papers. The proposed model predicts the admission according to their scores. The model’s performance has been evaluated by the R2 score. We have also developed a HTML page and a file system to store the previous data which is predicted by user input.

CHAPTER 4

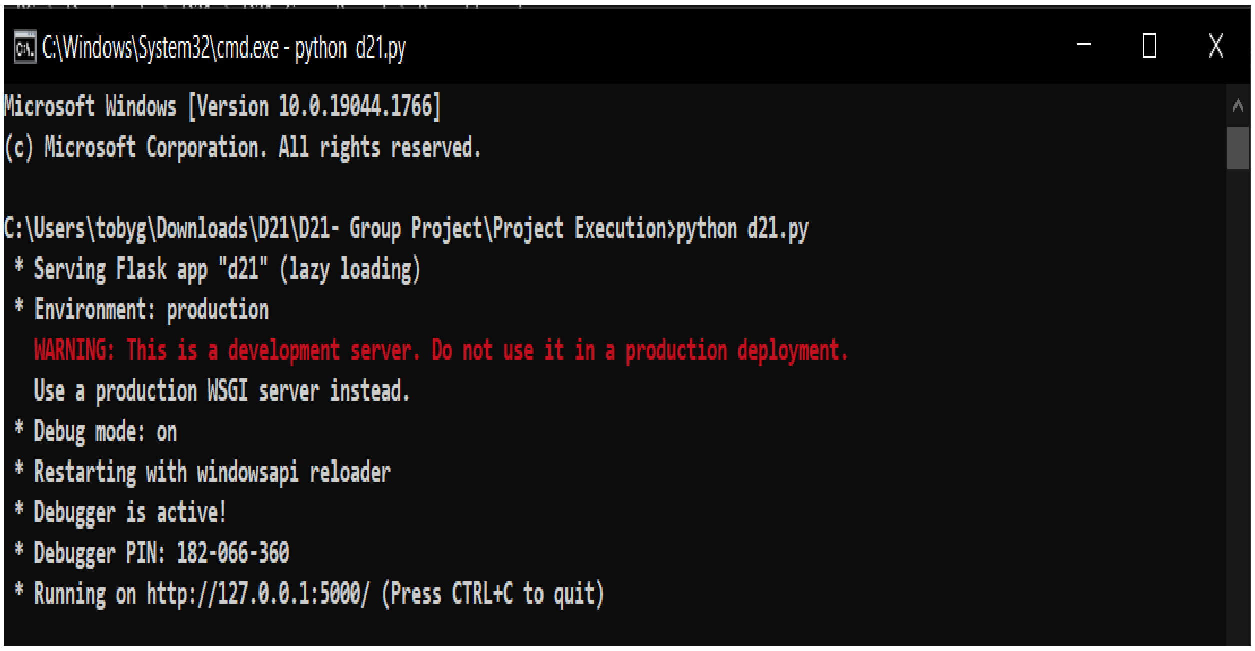
RESULTS AND OBSERVATIONS

4.1 INTRODUCTION

This chapter dealts with the results of the project work and the performance analsys where it gives a clear picture of the execution. The execution of the project is shown belows in the output.

4.2 EXECUTE APPLICATION

The first step of executing the application is shown in the below Fig 8 In this step we first run the python file which is intermediate between the html file and the pickle file. After executing the python file it displays the web page address where it loads the html file for the user on the system.



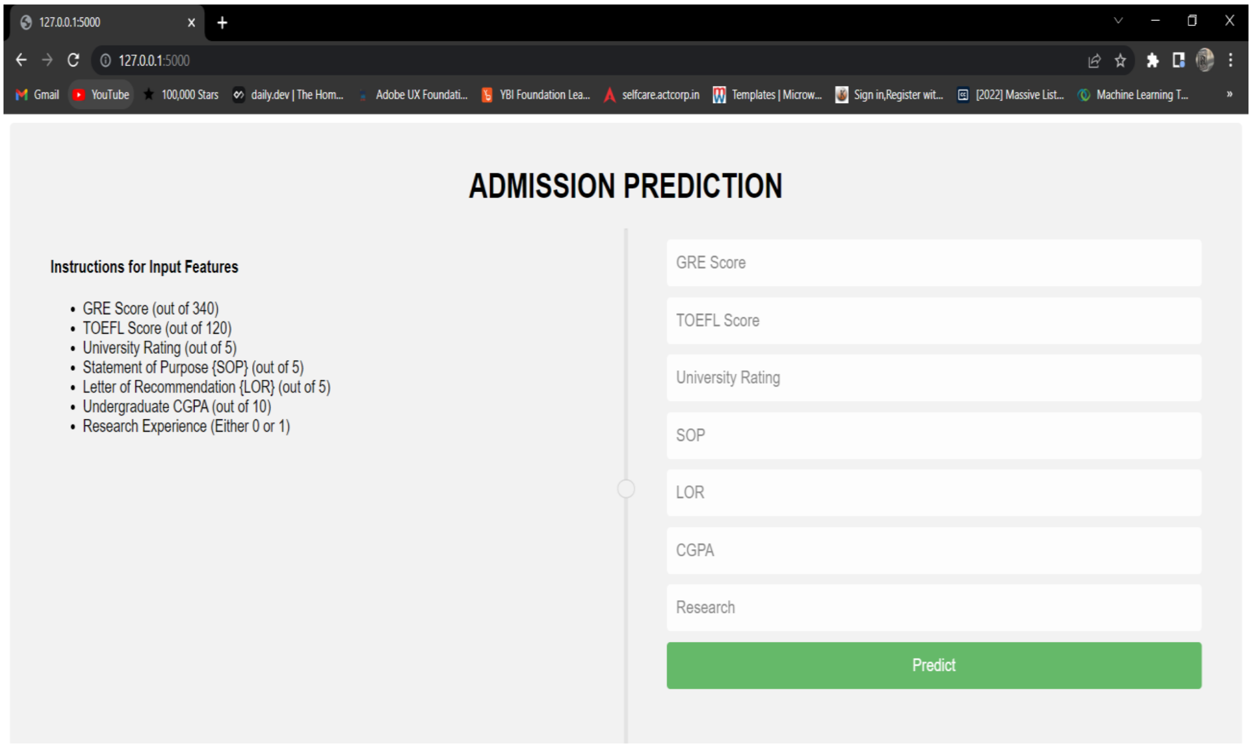
```
Microsoft Windows [Version 10.0.19044.1766]
(c) Microsoft Corporation. All rights reserved.

C:\Users\tobyg\Downloads\D21\D21- Group Project\Project Execution>python d21.py
 * Serving Flask app "d21" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not use it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Restarting with windowsapi reloader
 * Debugger is active!
 * Debugger PIN: 182-066-360
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

Fig 8: Screenshot of Running the Program

4.3 INSERT VALUES

The values are entered on the browser page as shown in Fig 9 and Fig 10. The values are entered in the html browser page are sent to the python file which acts as intermediate between the pickle file and the html page. From the python file the are fed into the pickle file for the prediction. After predicting the value, the pickle file sends the output to the python file. Again, the python file sends the predicted value to the html page for displaying the price for the user.

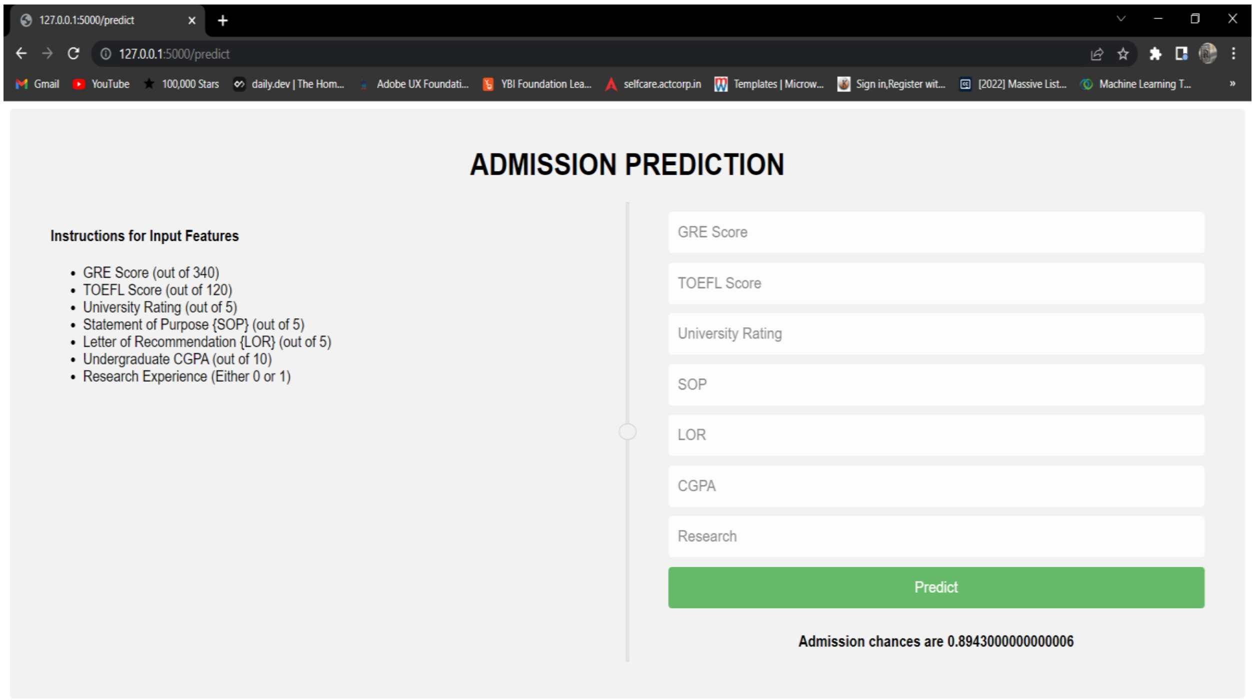


ADMISSION PREDICTION

Instructions for Input Features

- GRE Score (out of 340)
- TOEFL Score (out of 120)
- University Rating (out of 5)
- Statement of Purpose {SOP} (out of 5)
- Letter of Recommendation {LOR} (out of 5)
- Undergraduate CGPA (out of 10)
- Research Experience (Either 0 or 1)

Fig 9: Screenshot of Webpage to enter values



ADMISSION PREDICTION

Instructions for Input Features

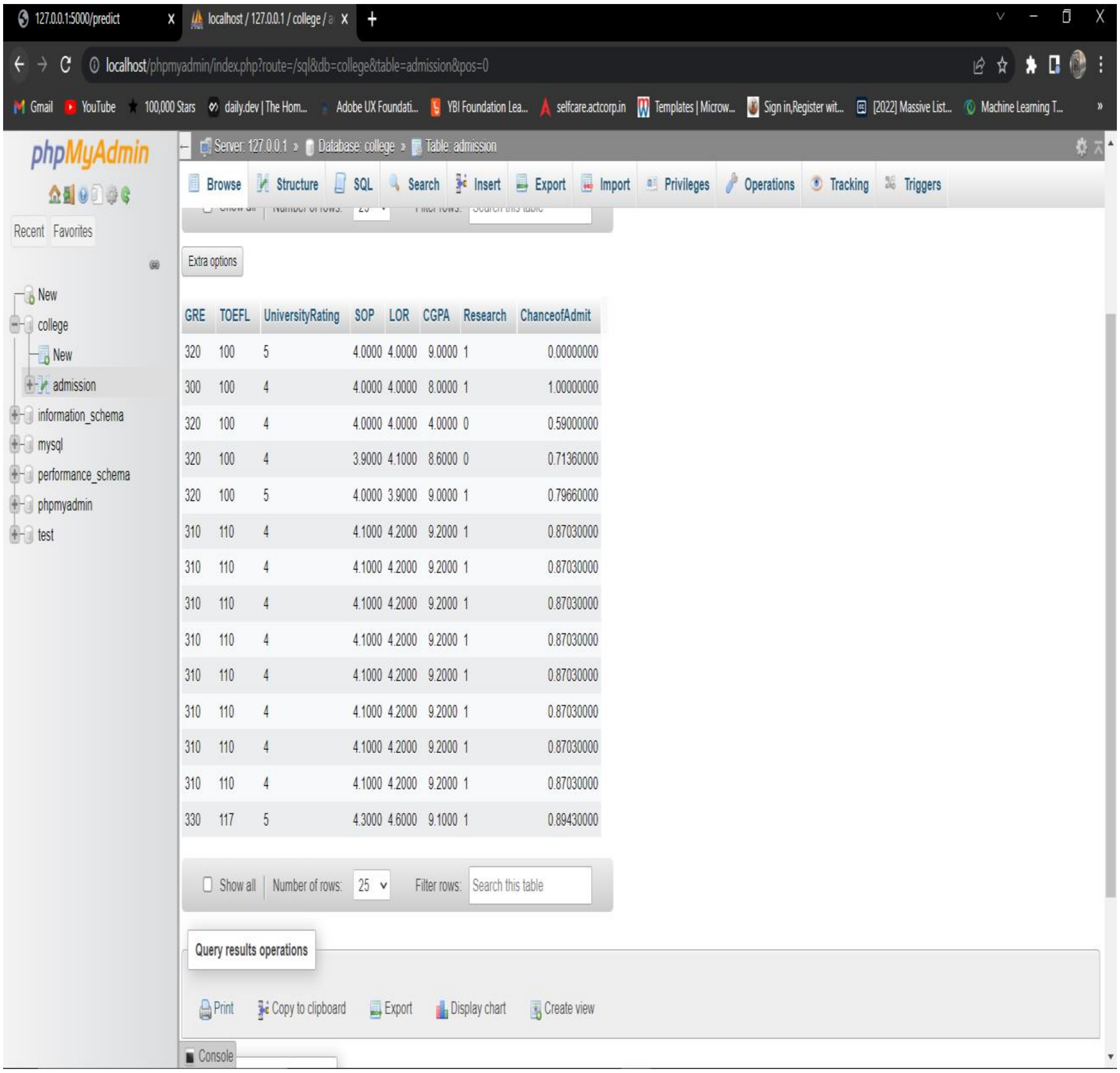
- GRE Score (out of 340)
- TOEFL Score (out of 120)
- University Rating (out of 5)
- Statement of Purpose {SOP} (out of 5)
- Letter of Recommendation {LOR} (out of 5)
- Undergraduate CGPA (out of 10)
- Research Experience (Either 0 or 1)

Admission chances are 0.8943000000000006

Fig 10: Screenshot of Output

4.4 USER GIVEN INPUT IN DATABASE

The inputs which are given and outputs obtained can be stored in a file system as shown in Fig 11. Inputs and outputs are stored to check whether the is predicting in the correct way or not.


127.0.0.1:5000/predict
localhost / 127.0.0.1 / college / a
localhost/phpmyadmin/index.php?route=/sql&db=college&table=admission&pos=0
phpMyAdmin
Recent Favorites
New
college
New
admission
information_schema
mysql
performance_schema
phpmyadmin
test
Server: 127.0.0.1 » Database: college » Table: admission
Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers
Extra options
Show all Number of rows: 25 Filter rows: Search this table
Query results operations
Print Copy to clipboard Export Display chart Create view
Console

GRE	TOEFL	UniversityRating	SOP	LOR	CGPA	Research	ChanceofAdmit
320	100	5	4.0000	4.0000	9.0000	1	0.00000000
300	100	4	4.0000	4.0000	8.0000	1	1.00000000
320	100	4	4.0000	4.0000	4.0000	0	0.59000000
320	100	4	3.9000	4.1000	8.6000	0	0.71360000
320	100	5	4.0000	3.9000	9.0000	1	0.79660000
310	110	4	4.1000	4.2000	9.2000	1	0.87030000
310	110	4	4.1000	4.2000	9.2000	1	0.87030000
310	110	4	4.1000	4.2000	9.2000	1	0.87030000
310	110	4	4.1000	4.2000	9.2000	1	0.87030000
310	110	4	4.1000	4.2000	9.2000	1	0.87030000
310	110	4	4.1000	4.2000	9.2000	1	0.87030000
310	110	4	4.1000	4.2000	9.2000	1	0.87030000
310	110	4	4.1000	4.2000	9.2000	1	0.87030000
330	117	5	4.3000	4.6000	9.1000	1	0.89430000

Fig 11: Screenshot of Data storage in File system

4.5 PERFORMANCE ANALYSIS

There is comparision of the algorithms which are used in the project work and it is selects the efficient algorithm as shown in the Fig 12.From the performance given below we can see that random forest gives better accuracy when compared to the decision tree and linear regression.

A. Mean absolute error

Eq (4) shows the MAE. This error basically is the average of the absolute difference between the actual or true values and the values that are predicted. The absolute difference means that if the result has a negative sign, it is ignored.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |yi - xi|}{n} \tag{4}$$

Where,

a. Y is the target variable
b. X is the predictor
c. P(c) is the probability of the feature
d. S(c) is the standard deviation of the feature

2. R2 Score

The coefficient of determination also called as R^2 score is used to evaluate the performance of a linear regression model. It is the amount of the variation in the output dependent attribute which is predictable from the input independent variable(s).

$$R^2 = 1 - \frac{\Sigma(yi-\hat{y}_1)^2}{\Sigma(yi-\bar{y})^2} \tag{5}$$

Where,

y_i = actual value

$\hat{y}$ = predicted value

$\bar{y}$ = mean of the predicted values

3. Mean square error

Mean squared error (MSE) measures the amount of error in statistical models. It assesses the average squared difference between the observed and predicted values. When a model has no error, the MSE equals zero. As model error increases, its value increases. The mean squared error is also known as the mean squared deviation (MSD) as in Eq (6)

$$MSE = \frac{\Sigma(y_i-\hat{y}_i)^2}{n} \tag{6}$$

Where,

y_i = Predicted value

$y_{i^\wedge}$ = Actual Value

n= number of observations

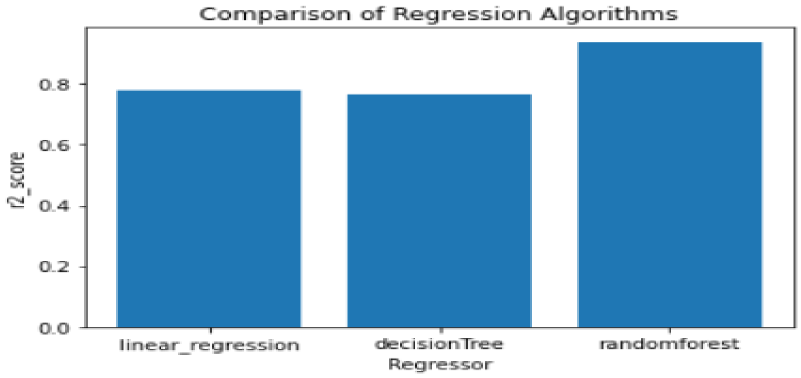


Fig 12: Bar plot of Performance Analysis

We select the best algorithm linear regression -79.89%, decision Trees - 78.48%, random Forests - 96.85% and we select the random forest algorithm as the best approach for the project. The Table 2 helps us to understand the performance of the model after using the algorithms as mentioned above.

Table 2: Performance Analysis

MODEL	MAE/RMSE	MSE	R2 score
Linear Regression & Catboost Model	0.04	0.003	0.84
Adaboost Model	-	-	0.90
DNN Model	0.03	0.0031	0.85
Proposed Model	0.04	0.0038	0.93

4.6 CONCLUSION

From the accuracies of the three models, we have got to know that the random forest produces the best accuracy as in the Table 2. We also conclude that it gives high accuracy. We have also discussed about the implementation in the webpage.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

5.1 CONCLUSION

Every year million of students apply to universities to begin their educational life. The experts had proposed some models and executed it. They have also published some conference and research papers. The reference has been taken from the discussed base papers. The proposed model predicts the admission according to their scores. The model's performance has been evaluated by the R2 score. The best chosen algorithm is uses among the algorithms. Our study found that Random forest performs very well for this project. We have also developed a HTML page and a file system to store the previous data which is predicted by user input. This is a project with good future scope, especially for the students of our age group who want to pursue their future goals in a good reputed university.

5.2 FUTURE SCOPE

In the future, we plan to improve the model by different parameters to advise students to join the University of their choice. Future work is to test this model on various datasets and also to predict the price by including the external factors such as list of universities, etc.

REFERENCES

[1] *A.Sivasangari, V.Shivani, Y.Bindhu, D.Deepa, R.Vignesh,* " Prediction Probability of Getting an Admission into a University using Machine Learning"**,** IEEE conference on ICCMC, Tamilnadu, India, May 2021, pp: 1706 – 1709.

[2] *Zhenru Wang, Yijie Shi, "*Prediction of the Admission Lines of College Entrance Examination based on machine learning", IEEE conference on ICCC, Paris, France, July 2016, pp: 332-334.

[3] *Md. Omaer Faruq Goni, Abdul Matin, Tonmoy Hasan, Md. Abu Ismail Siddique, Oishi Jyoti, Fahim MD Sifnatul Hasnain,"* Graduate Admission Chance Prediction Using Deep Neural Network*",* IEEE conference on WIECON-ECE, Bhubhaneswar, India, December 2020, pp: 259-262.

[4] *Robert Ries, Mehmet Ozbek, "*Predicting the Performance and Success of Construction Management Graduate Students using GRE Scores*".*

[5] *Naveen S.Sapare, Sahana M.Beelagi,"* Comparison study of Regression Models for the prediction of Post-Graduation admissions using Machine Learning Techniques*"* IEEE conference on ICCDSE, January 2021, Uttar Pradesh, India, pp: 822-828

[6] *Vandit Manish Jain, Rihaan Satia, "*College Admission Prediction using Ensemble Machine Learning Models*",* IRJET, December 2021, Tamilnadu, India, pp: 403-407

[7] *Matthew N O Sadiku, Adebowale E. Shadare, Sarhan M. Musa, Cajetan Akujuobi, "*Data Visualization" ResearchGate, Texas, December 2016

[8] *Xue Ying* "An Overview of Overfitting and its Solutions", February 2019

[9] *Douglas C Montgomery, Elizabeth A Peck and G Geoffrey Vining*, Introduction to linear regression analysis, John Wiley & Sons, vol. 821, 2012.

[10] *Harsh patel, purvi prajapti*, "Study and Analysis of Decision Tree Based Classification Algorithms",research gate, 2018

[11] *leo breiman*, "random forest",springer, 2001

[12] *Pedro Strecht, Luis Cruz, Carlos Soares, Joao Mendes Moreira*, "A Comparative Study of Classification and Regression Algorithms for Modelling Students' Academic Performance", Research Gate,December 2016

[13] *Marco Slaviero* , "Sour Pickles" ,senspost

[14] *Mishra, S. and Sahoo*, S. (2016). A Quality Based AutomatedAdmission System for Educational Domain, pp. 221–223, International conference on Signal Processing, Communication, Power and Embedded System (SCOPES)-2016.

[15] *Kumar, NM Saravana*. "Implementation of artificial intelligence in imparting education and evaluatingstudent performance." Journal of Artificial Intelligence 1, no. 01 (2019): 1-9.

[16] *Thomas. G. Dietterich*. Ensemble Methods in Machine Learning[J]. Multiple Classifier Systems, 2000, 1857: I-IS

[17] *G. Valentini & F. Masulli*. Ensembles of Learning Machines[J]. Workshop on Neural Nets, 2002, (2486): 3-20

[18] *Thomas. G. Dietterich*. Ensemble Methods in Machine Learning[J]. Multiple Classifier Systems, 2000, 1857: I-IS

[19] *G. Valentini & F. Masulli*. Ensembles of Learning Machines[J]. Workshop on Neural Nets, 2002, (2486): 3-20

APPENDIX-A: PYTHON

In machine learning projects, we never know the right solution at the start. We need many experiments and iterations to finalize our approach. Having a language that allows us to iterate fast means we can improve our solution faster. As a result, a lot of people are using Python. And a lot of libraries are written for Python. This virtuous cycle makes Python a mature language with a powerful ecosystem. Python is important because it opens the door for us to:

- Use the amazing machine learning libraries such as scikit-learn, TensorFlow, and PyTorch
- Connect to other systems, such as web applications or file systems, easily
- Communicate our idea to other people through our code, even though they didn't learn Python before.

NUMPY:

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

PANDAS:

Pandas is an open source Python package that is most widely used for data science/data analysis and machine learning tasks. It is built on top of another package named Numpy, which provides support for multi-dimensional arrays. Pandas makes it simple to do many of the time consuming, repetitive tasks associated with working with data, including:

- Data cleansing
- Data fill
- Data normalization
- Merges and joins
- Data visualization
- Statistical analysis
- Data inspection
- Loading and saving

APPENDIX-B:

UNIFIED MODELING LANGUAGE

The Unified Modeling Language (UML) is a general-purpose visual modeling language that is used to specify, visualize, construct, and document the artifacts of a software system. It captures decisions and understanding about systems that must be constructed. It is used to understand, design, browse, configure, maintain, and control information about such systems. It is intended for use with all development methods, lifecycle stages, application domains, and media. The modeling language is intended to unify past experience about modeling techniques and to incorporate current software best practices into a standard approach. UML includes semantic concepts, notation, and guidelines. It has static, dynamic, environmental, and organizational parts. It is intended to be supported by interactive visual modeling tools that have code generators and report writers.

The UML specification does not define a standard process but is intended to be useful with an iterative development process. It is intended to support most existing object oriented development processes. The UML captures information about the static structure and dynamic behavior of a system. A system is modeled as a collection of discrete objects that interact to perform work that ultimately benefits an outside user. The static structure defines the kinds of objects important to a system and to its implementation, as well as the relationships among the objects. The dynamic behavior defines the history of objects over time and the communications among objects to accomplish goals. Modeling a system from several separate but related viewpoints permits it to be understood for different purposes. The UML also contains organizational constructs for arranging models into packages that permit software teams to partition large systems into workable pieces, to understand and control dependencies among the packages, and to manage the versioning of model units in a complex development environment.

The UML is not a highly formal language intended for theorem proving. There are a number of such languages, but they are not easy to understand or to use for most purposes. The UML is a general-purpose modeling language. For specialized domains, such as GUI layout, VLSI circuit design, or rule-based artificial intelligence, a more specialized tool with a special language might be appropriate. UML is a discrete modeling language.