COLLEGE PREDICTOR

Project submitted to

Shri Ramdeobaba College of Engineering & Management, Nagpur

in partial fulfillment of requirement for the award of

degree of

Bachelor of Technology

In

COMPUTER SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE
AND MACHINE LEARNING)

By

Ms. Amna Patel

Mr. Dhruvraj Solanki

Mr. Himesh Ganwani

Mr. Manav Anandani

Guide

Prof. Priya Parkhi

Computer Science and Engineering

Shri Ramdeobaba College of Engineering & Management, Nagpur
440013

(An Autonomous Institute affiliated to Rashtrasant Tukdoji Maharaj Nagpur University Nagpur)

December 2022

SHRI RAMDEOBABA COLLEGE OF ENGINEERING & MANAGEMENT, NAGPUR

(An Autonomous Institute affiliated to Rashtrasant Tukdoji Maharaj Nagpur University Nagpur)

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Thesis on **"College Predictor"** is a bonafide work of

1. Ms. Amna Patel
2. Mr. Dhruvraj Solanki
3. Mr. Himesh Ganwani
4. Mr. Manav Anandani

submitted to the Rashtrasant Tukdoji Maharaj Nagpur University, Nagpur in partial fulfillment of the award of a Degree of Bachelor of Technology, in Computer Science and Engineering (Artificial Intelligence and Machine Learning). It has been carried out at the Department Computer Science and Engineering, Shri Ramdeobaba College of Engineering and Management, Nagpur during the academic year 2022-23.

Date: 17-12-22

Place: Nagpur

Prof. Priya Parkhi
Project guide

Dr. Avinash Agrawal
H.O. D
Department of Computer Science and Engineering

DECLARATION

I, hereby declare that the thesis titled **"College Predictor"** submitted herein, has been carried out in the Department of Computer Science and Engineering of Shri Ramdeobaba College of Engineering & Management, Nagpur. The work is original and has not been submitted earlier as a whole or part for the award of any degree / diploma at this or any other institution / University

Date: 17-12-22

Place: Nagpur

Signature and Name of the Students

Ms. Amna Patel
(Roll no.: 22)

Mr. Dhruvraj Solanki
(Roll no.: 38)

Mr. Himesh Ganwani
(Roll no.: 42)

Mr. Manav Anandani
(Roll no.: 49)

ACKNOWLEDGEMENT

We would like to express our deep and sincere gratitude to our guide **Prof. Priya Parkhi,** Professor of Computer Science and Engineering Department, RCOEM, for giving us the opportunity to work on this project and providing valuable guidance throughout the project. It was a great privilege and honor to work under her guidance. We are extremely grateful for the experience we had in this project with her.

We express our sincere gratitude to **Dr. Avinash Agrawal,** Head of the Department of Computer Science Department, RCOEM for his guidance. Talent wins games, but teamwork and intelligence win championships. We would like to take this opportunity to express our deep gratitude to all those who extended their support and guided us to complete this project.

Signature and Name of the Students

Ms. Amna Patel **Mr. Dhruvraj Solanki**

Mr. Himesh Ganwani **Mr. Manav Anandani**

ABSTRACT

“Life is a matter of choices and every choice you make makes you”. Students face a lot of difficulties to secure an admission in the college of their choice. The current scenario
of an engineering admission process is little complicated and thus often students end up getting less deserved college. The college admission predictor uses historical colleges cut-off student admission data for predicting the most probable colleges. The system analyzes student academic merits, background, and college admission criteria. Based on that, it predicts the likelihood of a university college that a student may enter. For the accurate predictions we have trained a machine learning model in order to provide results. The dataset contains information on the student profile and the university details with a field detailing if the admission was positive or not. The project makes use of machine learning and database management concepts for suggesting a list of engineering colleges to the students in a sequence of the probability of getting admission to that specific college. The system would enable the student to prepare the list of colleges, which could be needed to be filled in during the admission process.

TABLE OF CONTENTS

List Of Figures

List of Formulas

Chapter 1
INTRODUCTION

1.1 Introduction

"Life is a matter of choices, and every choice you make makes you"

In recent years, competition in the industry has increased exponentially. The increased competition has also led to issues such as unemployment and high demand for newly emerging skills. In such a competitive environment, it becomes very mandatory for a student to secure admission and training from the best suitable institute. This would help the students to improve their skills as per the requirements in the industry and secure an appropriate placement.

Thus, securing admission in the best college plays a very vital role in deciding the future of a particular student. For anyone pursuing their undergraduate studies, it would be difficult for them to find out which college they deserve the most, based on their GPA, JEE scores. People may apply to many universities that look for candidates with a higher score set, instead of applying to universities at which they have a chance of getting into. This would be detrimental to their future. It is very important that a candidate should apply to colleges that he/she has a good chance of getting into, instead of applying to colleges that they may never get into. Though the admission process for engineering courses has become easier than before, it still involves some risk factors that are hard out to figure out by the student. Admissions in engineering colleges in the state of Maharashtra or any state is based upon common entrance test (CET) and since more than 1.5 lakh seats are to be allotted in more than 200 engineering colleges and over 35 different branches of engineering, for students belonging to many categories like open, home university, outside home university, reserved category (SC, ST, OBC, etc.). Thus, the problem becomes more complex and students struggle to understand which colleges they are likely to get admitted in.

There aren't many efficient ways to find out the colleges that one can get into, relatively quickly. The College Predictor helps a person decide what colleges they can apply to with their scores. In the system proposed, a person can enter their scores in the respective fields provided. The system then processes the data entered and produces an output of the list of colleges that a person could get into, with their scores. This is relatively quick and helps conserve time and money. In order to achieve this, we have proposed a novel method utilizing Machine Learning algorithms.

1.2 Motivation

Education plays a vital role in today's era. While we talk about career - a person's degree, course, university and the knowledge that he possesses - is the key factor on which the firm hires a fresher. As soon as a student completes his/her Higher Secondary Schooling, the first goal of any student is to get into an appropriate College so that he can get a better education and guidance for his future. For that, students seek help from many sources like online sites or career experts to get the best options for their future. A good career counselor charges a huge amount for providing such solutions. Online sources are also not as reliable as the data from particular sources is not always accurate. Students also perform their analysis before applying to any institutions, but this method is slow and certainly not consistent for getting actual results and possibly includes human error. Since the number of applications in different universities for each year is way too high, there is a need to build up a system that is more accurate or precise to provide proper suggestions to students.

Admissions in engineering colleges in India is based upon Joint Entrance Examination (JEE) and since more than 50,000 seats are to be allotted in about 23 IITs and 31 NITs and over 35 different branches of engineering, for students belonging to many categories like open, home university, outside home university, reserved category (SC, ST, OBC, etc.). Thus, the problem becomes more complex and students struggle to understand which colleges they are likely to get admitted in.

According to a case study, there are more than 400 engineering colleges in Maharashtra, for which admission is governed by DTE (Directorate of Technical Education). It's very difficult for the students to and out suitable colleges for them based on their MHCET Score, MHCET Rank, Category, Home University, etc. Various colleges provide degree in engineering in various branches (IT, Computer, Mechanical, Electrical, civil, etc.). Though analysis of colleges and their cut offs is required in order to get the most correct preference list. It is very tedious job for a student to understand about the suitable colleges which provides preferred branch and to analyses it's last three years cut offs in order to predict whether that he can get one of those colleges in CAP.

Most of the students make mistakes in their preference list due to lack of knowledge, improper and incorrect analysis of colleges and insecure predictions. Hence those students regret after what they get the college after allotment. Our project will solve the

issue of the student community by using a technology. To minimize the stress of students we came up with the idea of a computer-aided method which aims to automate this process and remove the risk holding factor of searching the number of eligible and best colleges within their vicinity.

1.3 Objectives

- To help students pursuing engineering identify the best colleges they can get, based on their rank and category. Thus, students will not have to make extra efforts on research about different colleges they can take admission into.
- To help students to fill their preferences at the time of option-entry process accurately.
- Ease the decision-making process for students as they would have a ready list of best colleges into which they are eligible to take admission. This would help them make better choices of college and branch before allotment.

1.4 Problem Definition

Educational organizations have always played an important and vital role in society for development and growth of any individual. There are different college prediction apps and websites being maintained contemporarily, but using them is tedious to some extent, due to the lack of articulate information regarding colleges, and the time consumed in searching the best deserving college. The problem statement, hence being tackled, is to design a college prediction/prediction system and to provide a probabilistic insight into college administration for overall rating, cut-offs of the colleges, admission intake and preferences of students. Also, it helps students avoid spending time and money on counsellor and stressful research related to finding a suitable college.

1.5 Brief Description of the System

In the system proposed, a student is required to enter can enter his JEE rank along with some basic details such as his/her quota, pool, category and round number. The system then processes the data entered and produces an output of the list of most deserving colleges that a person could get into.

1.5.1 Front End

For the frontend, a Web Application using HTML5 and CSS has been used.
A model using Python to detect the language was developed which was later integrated with the web- pages using Flask.
HTML templates: Template systems allowed us to specify the structure of the input as well as output documents, using placeholders for data that was filled in when a page was generated.

- The index.html in the templates folder is our home page- the page where the user first lands into when he/she visits our application.
- Form.html is used for getting input from our users.
- results.html file that shows the output as the detected language name.

1.5.2 Back End

Pandas library of python has been used for data loading, preprocessing and cleaning while the sklearn library has been used for Model Building. At the end flask framework is used for the integration of the backend with frontend.

To detect a language, we have to first train our model. To train the model a part of a dataset from Kaggle were used. DecisionTreeClassifier and K means algorithm have been used for creating our model which is part of the sklearn library. Firstly, the most deserved college would be selected by the Decision Tree based on the user's inputs. Then the K- Means will find the 10 most deserving college to produce them as output.

Once the model is created i.e., trained a file is created hence it is not needed to train our model every time while running the code. Once the user enters the input as text or URL of the website, that input is tested through this model and the Language with the highest value is printed and this is the emotion being recognized. We have also deployed our project on cloud so that it can be used anywhere in the world.

- 'form.py'- this file contains all of the code to get input from the user and then process it to get the desired output.

Chapter 2
LITERATURE SURVEY

Various papers had been studied related to College Predictor how they work. Some of

the papers are mentioned below with their brief description of the studies.

PAPER 1

Prediction of the Admission Lines of College Entrance Examination based on machine learning [1]

In this paper, Zhenru Wang and Yijie Shi compared two important machine learning algorithms to predict the college admission from the previous years' records. The paper shows different statistics based on the result of both algorithms. The first algorithm implemented by them was Random forests prediction algorithm. This model shows the accuracy of around 80% i.e., around 80 results out 100 predicted was correct. The second algorithm implemented was Adaboost. The model trained with Adaboost algorithm shows an accuracy of around 90% i.e., 90 accurate predictions from 100 results. The resources used for training of model and testing data are as follows: the system is windows 7, and python is used for the program. The conclusion of the paper suggests that the AdaBoost algorithm is much more efficient and reliable than Random forests while dealing with admission prediction process using machine learning.

PAPER 2

Hybrid Recommender System for Predicting College Admission [2]

In this paper, Abdul Hamid M Ragab, Abdul Fatah S. Mashat and Ahmed M Khedra proposed the novel design for college admission hybrid recommender based on data mining techniques and knowledge discovery rules, for tackling college admissions prediction problems. This system consists of two cascade hybrid recommenders working together with the help of college predictor, for achieving high performance. The first recommender assigns students tracks for preparatory year students. While the second recommender assigns the specialized college for students who passed the preparatory year exams successfully. This predictor algorithm uses previous students' admission data of colleges GPA for predicting most probable colleges. It looks over student academic merits, background, student records, and the college admission

criteria. Then, predicts the possibility of university colleges that a student may enter. In addition to the high prediction accuracy rate, flexibility in an advantage, as the system can predict suitable colleges that match the student's profile and the suitable track channels through which the students are advised to enter. The trust-ability is achieved since students' responses positively increasing as long as they allocated to the most suitable college which satisfies their desire. The design is proposed only of Saudi Arabian Universities.

PAPER 3

Shiksha.com [3]

This is a web-based application which provides the guidance and solutions for the educational queries. They developed a college predictor which takes the student data as an input and shows the possible college according to the colleges criteria. The application probably uses data mining algorithm with the previous year's cutoffs as a training data for the model. The website provides a list of colleges based on rank rather than scores. The filters are restricted to selection of branches and colleges, also the list generated by the website cannot be exported.

Chapter 3
TECHNICAL SPECIFICATION

3.1 Algorithms Used

1. Decision Trees Classification

Decision Trees Classification is a two-step process, learning and prediction. At first, model is developed based upon the given training data in its learning step. Then the model is used to predict response for the given data in prediction step. One of the most popular classification algorithms and easiest to learn and understand is decision tree. Decision tree algorithm is also used for solving classification and regression problems. Decision trees use a class label for predicting, for a record it starts from the root of tree. Then compare the values of the root with its record attribute. After the comparison, it follows the branch which is corresponding to the value and jump upon to the next node. There are two types of decision trees, Categorical variable and continuous variable. Categorical variable has a categorical target variable and continuous variable has a continuous target variable. Decision tree has three types of nodes, decision nodes, chance nodes and end nodes. Decision tree assigns a class label for each leaf node. Even the non-terminal nodes, the root and internal nodes, also contain attribute test conditions to separate records that have different characteristics.

$$\overbrace{IG(T,a)}^{\text{information gain}} = \overbrace{\mathrm{H}(T)}^{\text{entropy (parent)}} - \overbrace{\mathrm{H}(T \mid a)}^{\text{sum of entropies (children)}}$$

$$= -\sum_{i=1}^{J} p_i \log_2 p_i - \sum_{i=1}^{J} -\Pr(i \mid a) \log_2 \Pr(i \mid a)$$

(3.1.1) Calculation of Information Gain

Information gain is used to decide which feature to split on at each step in building the tree.

2. K Means Clustering

K-Means Clustering is an Unsupervised Learning algorithm, which groups the unlabeled dataset into different clusters. Here K defines the number of pre-defined clusters that need to be created in the process. It allows us to cluster the data into

different groups and a convenient way to discover the categories of groups in the unlabeled dataset on its own without the need for any training.

It is a centroid-based algorithm, where each cluster is associated with a centroid. The main aim of this algorithm is to minimize the sum of distances between the data point and their corresponding clusters. The algorithm takes the unlabeled dataset as input, divides the dataset into k-number of clusters, and repeats the process until it does not find the best clusters. The value of k should be predetermined in this algorithm.

The k-means clustering algorithm mainly performs two tasks:

- Determines the best value for K center points or centroids by an iterative process.
- Assigns each data point to its closest k-center. Those data points which are near to the particular k-center, create a cluster.

Hence each cluster has datapoints with some commonalities, and it is away from other clusters.

$$S_i^{(t)} = \left\{ x_p : \left\| x_p - m_i^{(t)} \right\|^2 \leq \left\| x_p - m_j^{(t)} \right\|^2 \ \forall j, 1 \leq j \leq k \right\},$$

where each x_p is assigned to exactly one $S^{(t)}$, even if it could be assigned to two or more of them.

$$m_i^{(t+1)} = \frac{1}{\left| S_i^{(t)} \right|} \sum_{x_j \in S_i^{(t)}} x_j$$

(3.1.2) Distributing the data points into k different clusters by calculating their centroids

3.2 Python 3.10.4

Python is an interpreted, high-level programming language that is available freely for distribution and commercial use. It is an open-source language and is OSI-approved. Various NLP libraries are available under Python for building NLP solutions. These are as follows:

3.2.1 Scikit-learn

Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

DecisionTreeClassification() and Kmeans() are two functions of scikit-learn library which helps in developing the Decision Tree and K-Means machine learning model respectively.

3.2.2 Flask

Flask is a web application framework written in Python. it's a Python module that lets you develop web applications easily. It has a small and easy-to-extend core: it's a microframework that doesn't include an ORM (Object Relational Manager) or such features.

It is basically used to integrate the front-end web pages to the back-end data processing. Flask module objects take the input from the user form, then perform some processing on it and finally displays back the output on a web page.

3.2 HTML5 and CSS3

HTML and CSS are the core language components that are used for the construction of web pages. HTML describes the structure of the pages, primarily in regards to tables, text, headings, and images or graphics. It's the standard programming language for the overall appearance of web pages. CSS, on the other hand, is the language used for describing the presentation of each page, and primarily in regards to the layout, fonts, and colors.

3.3 AWS (Amazon Web Services)

Amazon Web Services offers cloud web hosting solutions that provide businesses, non-profits, and governmental organizations with low-cost ways to deliver their websites and web applications. AWS offers a wide-range of website hosting options whether will it be for a marketing, rich-media, or ecommerce website and much more.

Types of AWS Website Hosting Services:

1. Simple Website Hosting

Simple websites typically consist of a single web server which runs either a Content Management System (CMS), such as WordPress, an eCommerce application, such as Magento, or a development stack, like LAMP. The software makes it easy to build, update, manage, and serve the content of your website.

Simple websites are best for low to medium trafficked sites with multiple authors and more frequent content changes, such as marketing websites, content websites or blogs. They provide a simple starting point for website which might grow in the future. While typically low cost, these sites require IT administration of the web server and are not built to be highly available or scalable beyond a few servers.

2. Single Page Web App Hosting

Static web apps that require only a single load in a web browser are referred to as Single page web apps. All subsequent actions by the user are made available through HTML, JavaScript, and CSS that are pre-loaded in the browser. Backend data is accessed via GraphQL or REST APIs that fetch content from a data store and update the UI without requiring a page reload.

Single page web apps offer native or desktop app-like performance. They offer all the static website benefits (low cost, high levels of reliability, no server administration, and scalability to handle enterprise-level traffic) with dynamic functionality and blazing fast performance.

3. Simple Static Website Hosting

Static websites deliver HTML, JavaScript, images, video and other files to your website visitors and contain no server-side application code, like PHP or ASP.NET. They typically are used to deliver personal or marketing sites.

Static websites are very low cost, provide high-levels of reliability, require no

server administration, and scale to handle enterprise-level traffic with no additional work.

4. Enterprise Web Hosting

Enterprise websites include very popular marketing and media sites, as well as social, travel, and other application-heavy websites. For example, Lamborghini, Coursera, and Nordstrom use AWS to host their websites. Enterprise websites need to dynamically scale resources and be highly available to support the most demanding and highly trafficked websites.

Enterprise websites use multiple AWS services and often span multiple data centers (called Availability Zones). Enterprise websites built on AWS provide high levels of availability, scalability, and performance, but require higher amounts of management and administration than static or simple websites.

Chapter 4

SYSTEM ANALYSIS AND DESIGN

4.1 System Analysis

System analysis is a process of collecting and interpreting facts, identifying the problems and decomposing the system into components. This system analysis has been done to study the system and identify its objective. System analysis will give a brief about what the system will do.

4.1.1 Feasibility Study

The goal of our system is to detect a list of colleges for a given student. The system will detect this list in a faster and an accurate way. The actions included are extracting useful information from a student and applying decision tree algorithm to get the college of higher probability that the student can get in. After that pass this college into K-Means algorithm to get clusters and accordingly get list of colleges. This system can be used by any type of person and in any part of the world. These three tests of feasibility have been carried out.

4.1.1.1 Technical Feasibility

Technical Feasibility Assessment examines whether the proposed system can be designed to solve the desired problems and requirements using available technologies in the given problem domain. The system is said to be feasible technically if it can be deployable, operable and manageable under the current technological context of our country. Since the system aims to detect language based on user's input and to implement this using python and Flask the project can be considered technically feasible.

4.1.1.2 Operational Feasibility

The operational feasibility analysis describes how the system operates and what resources do, the system requires for performing its designated task. Being closely related to data analysis and integration, the system requires it to be easily operable for 12 different uses and operations like new datasets, concurrent use of the system should be fluent, consuming minimum cost and resources. The system to be designed is operationally feasible as the system can be operated with the resource as a personal

computer i.e., browser. The project is developed as the website allows easy access to multiple users.

4.1.1.3 Economical Feasibility

Economic Feasibility checks whether the cost required for complete system development is feasible using the available resources in hand. It should be noted that the cost of resources and overall cost of deployment of the system should be kept minimum while operational and maintenance costs for the system should be within the capacity of the organization. Since the system can be hosted on Firebase/Heroku cloud hosting service which is free of cost for limited use, the system can be considered economically feasible for the development.

4.1.2 Requirement Specification

A software requirement specification is a description of a software system to be developed. It lays out functional and non-functional requirements. It describes what the software product is expected to do and what not to do. It enlists necessary requirements that are required for the project development. It mainly aids to describe the scope of the work and provide software designers a form of reference.

4.1.2.1 Functional Requirement

The functional requirement specification of the project is mainly categorized as user requirements, and device requirements each of which are explained in detail below:

- User Requirement: Users should be able to read and understand English. The user must be familiar with the working of a web browser. The user should also have an internet connection while using the system.
- Device Requirement: System must be initiated on a web browser with an internet connection.

4.1.2.2 Non-functional Requirement

The non-functional requirement of the system can be summarized as follows:

• Performance: The system shall have quick, accurate and reliable results.

• Capacity and Scalability: The system shall be able to Detect the respected language.

•Availability: The system shall be available to users anytime whenever there is an Internet connection.

• Recovery: In case of malfunctioning or unavailability of the server, the system should be able to recover and prevent any data loss or redundancy.

•Flexibility and Portability: System shall be accessible anytime from any location.

4.2 System Design

System design is a process of planning a new system by defining its components or modules to satisfy the requirements. System Design focuses on how to accomplish the objective of the system.

4.2.1 System Overview

The system initially asks for a student information like JEE rank, score, caste, etc. The system then analyzes the text, performs all the Machine learning algorithms on the data and produce and output list of colleges for the student.

4.2.2 Activity Diagram

Figure 4.2.2 is the activity diagram for the College Predictor. It depicts the behavior of the system as a progression of actions. The various activities of the College Predictor are as follows:

1. Cleaning and preprocessing of the data in the dataset.
2. Dividing the dataset into training and testing data.
3. Training of the model (decision tree + k-means) based on the training data.
4. Testing the trained model.
5. Taking input (round number, rank, quota, pool and category) from the user through a frontend HTML form.

6. Passing this input to the model using 'Flask' (A python framework).
7. In the model the user input is first preprocessed into numerical data.
8. The DecisionTreeClassifier then predicts the best college based on the user inputs.
9. The output of the Decision tree is passed to the K- means algorithm which then finds the best 10 colleges the user has the maximum chance of getting into.
10. The data is converted back into categorical form and the output is displayed to the user using Flask.
11. The whole project is deployed on the cloud using AWS Cloud Services.

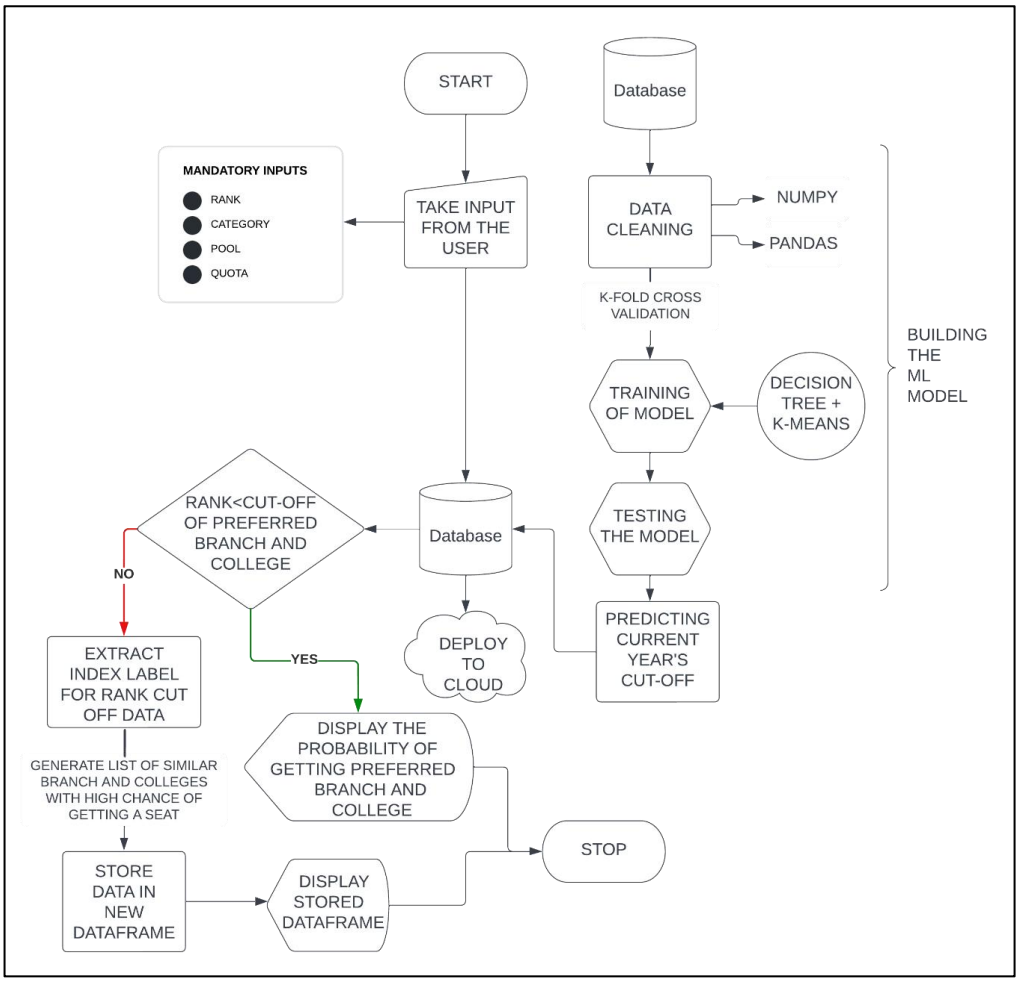

Fig 4.2.2 Working flowchart for the project

Chapter 5
WORKING

5.1 Acquiring dataset.

We found a dataset on Kaggle.com which had branch-wise opening and closing rank of IITs and NITs around the India. These ranks were dependent on some other attributes as well such as pool, category, quota etc. The dataset contained the data for past 6 years as nearly about 65,000 entries.

	year	institute_type	round_no	quota	pool	institute_short	program_name	program_duration	degree_short	category	opening_rank	closing_rank
0	2016	IIT	6	AI	Gender-Neutral	IIT-Bombay	Aerospace Engineering	4 Years	B.Tech	GEN	838	1841
1	2016	IIT	6	AI	Gender-Neutral	IIT-Bombay	Aerospace Engineering	4 Years	B.Tech	OBC-NCL	408	1098
2	2016	IIT	6	AI	Gender-Neutral	IIT-Bombay	Aerospace Engineering	4 Years	B.Tech	SC	297	468
3	2016	IIT	6	AI	Gender-Neutral	IIT-Bombay	Aerospace Engineering	4 Years	B.Tech	ST	79	145
4	2016	IIT	6	AI	Gender-Neutral	IIT-Bombay	Aerospace Engineering	4 Years	B.Tech	GEN-PWD	94	94

Fig 5.1.1 An Overview of the dataset

5.2 Analysis of the dataset.

The dataset acquired from Kaggle was then analyzed based on its different attributes.

	year	institute_type	round_no	quota	pool	institute_short	program_name	program_duration	degree_short	category	opening_rank
count	64958.000000	64958	64958.000000	64958	64958	64958	64958	64958	64958	64958	6.495800e+04
unique	NaN	2	NaN	7	2	54	130	2	13	10	NaN
top	NaN	IIT	NaN	AI	Gender-Neutral	IIT-Kharagpur	Computer Science and Engineering	4 Years	B.Tech	GEN	NaN
freq	NaN	32905	NaN	32905	38785	5865	8425	54434	52086	12982	NaN
mean	2020.421580	NaN	2.609348	NaN	NaN	NaN	NaN	NaN	NaN	NaN	8.259642e+03
std	1.149762	NaN	2.422558	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2.679448e+04
min	2016.000000	NaN	1.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.000000e+00
25%	2020.000000	NaN	1.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6.710000e+02
50%	2021.000000	NaN	1.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	2.309000e+03
75%	2021.000000	NaN	6.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	6.932000e+03
max	2021.000000	NaN	7.000000	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.082601e+06

Fig 5.2.1 Analysis of the dataset

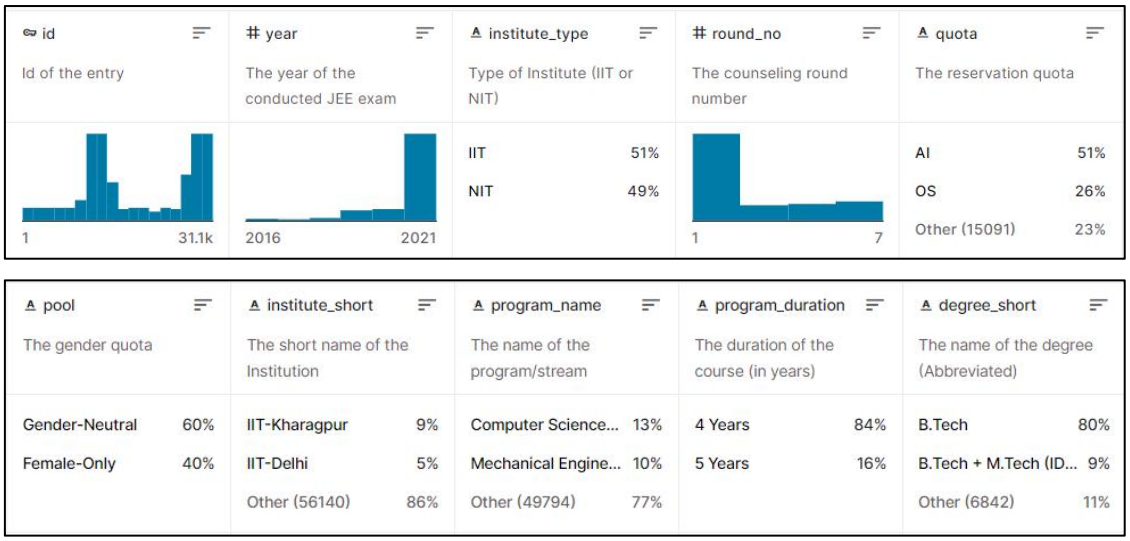

Fig 5.2.2 Attribute-wise analysis of the dataset

5.3 Data cleaning and preprocessing

Data cleaning involved merging and dropping some of the columns from the dataset. This was done to reduce the number of attributes. This eventually resulted in decreasing the complexity of the model and increasing its accuracy.

[Columns Merged- institute_type, institute_short, program_name, program_duration, degree_short]

[Columns Dropped- id, year]

Later- on the data was pre-processed. This involved converting the categorical data to numerical. This is because a model always works on numerical data and not categorical.

	round_no	quota	pool	category	closing_rank	College
0	6	1	2	1	1841	82
1	6	1	2	5	1098	82
2	6	1	2	7	468	82
3	6	1	2	9	145	82
4	6	1	2	4	94	82

Fig 5.3.1 View of the dataset after preprocessing

5.4 Building the Model

Using the preprocessed data, we first built a decision tree which can predict the best college he/she can get based on the user's input.

```
Accuracy=  72.23719676549865
```

Fig 5.4.2 Accuracy of Decision Tree

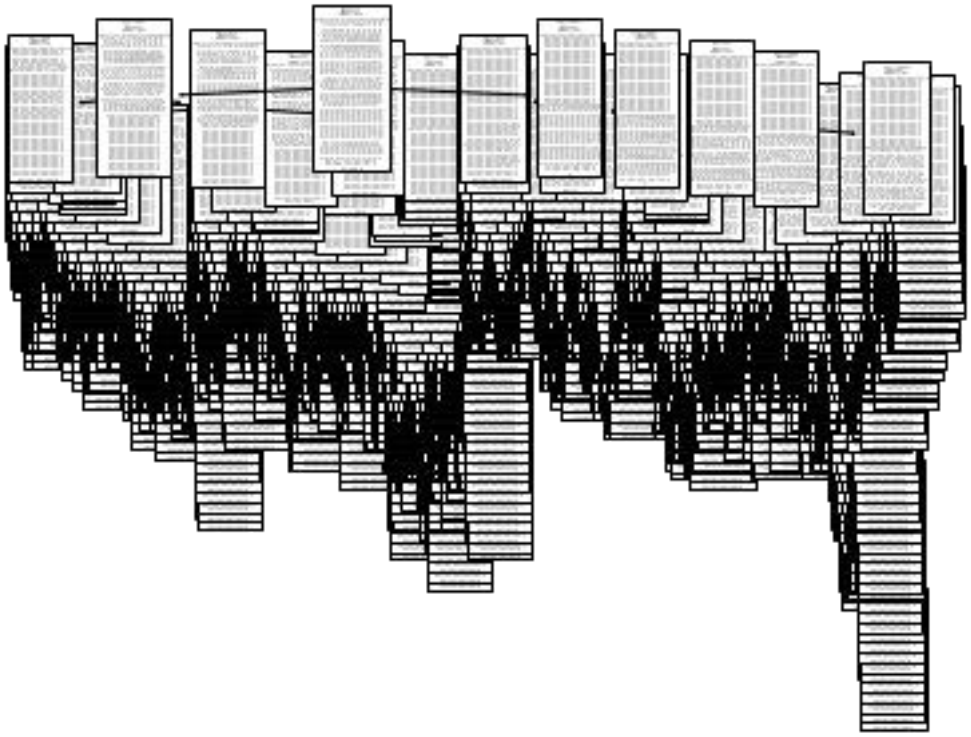

Fig 5.4.2 Decision Tree Visualization

Later-on we divided the dataset into clusters based on the closing rank. The cluster number of the output of the decision tree was found, and the first 10 entries of that particular cluster were displayed as output.

At last, the data was also converted back to its original form.

College
NIT-Calicut-B.Tech-Civil Engineering-4 Years
NIT-Jaipur-B.Tech-Mechanical Engineering-4 Years
NIT-Hamirpur-Btech + M.Tech (IDD)-Electronics ...
NIT-Meghalaya-B.Tech-Electronics and Communica...
NIT-Calicut-B.Tech-Chemical Engineering-4 Years
NIT-Rourkela-B.Tech-Metallurgical and Material...
NIT-Jalandhar-B.Tech-Electronics and Communica...
NIT-Karnataka-Surathkal-B.Tech-Civil Engineeri...
NIT-Uttarakhand-B.Tech-Electronics and Communi...
NIT-Rourkela-Btech + M.Tech (IDD)-Chemical Eng...

Fig 5.4.4 Sample Output

5.5 Building the web pages

Now for the front-end part, we built a home page and a form to take input from the user.

[Inputs taken from user-

- JEE Rank
- Category
- Quota
- Round No.
- Pool

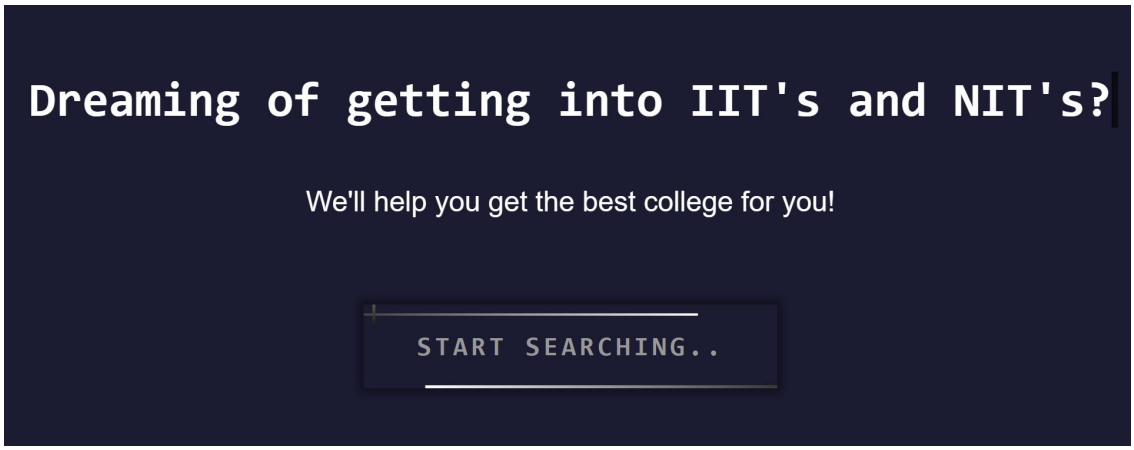

Fig 5.5.1 Home Page

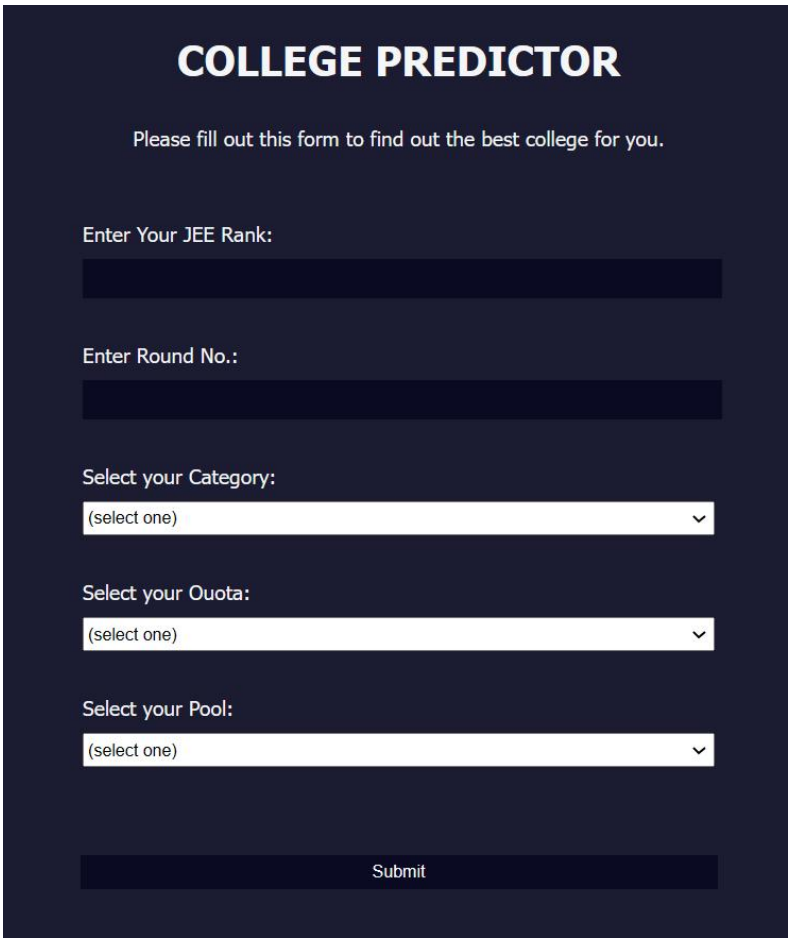

Fig 5.5.2 User Input form

5.6 Integrating Front-end and Back-end

The web-pages were then integrated with the model using Flask- a Python Framework. This framework proved very useful in taking input from the user, then passing the input to the model and then finally displaying the final results back to the user. In this project we have provided the user an option to download the result as a .csv file so that he/she can access it anytime in the future.

5.7 Deploying to Cloud

The whole of the project would be then deployed to the cloud using AWS Cloud Services so that it could be accessed by anyone from anywhere in the world.
The steps involved in deploying the website on the cloud are:

1. Create a S3 bucket and name it. The name should be the same name as your domain name for the website.
2. Configuring the S3 Bucket for Static Website Hosting using the properties section for the bucket.
3. Map the domain name to the S3 website URL. This mapping is often referred to as a CNAME record inside of the Domain Name Servers (DNS) records.
4. Upload the website and make sure it is running properly.

Chapter 6
CONCLUSION, FUTURE SCOPE

6.1 Conclusion

Every year millions of students apply to universities to begin their educational life. Most of them don't have proper resources, prior knowledge and are not cautious, which in turn creates a lot of problems as applying to the wrong university/college, which further wastes their time, money and energy. With the help of our project, we have tried to help out such students who are finding difficulty in finding the right university for them. It is very important that a candidate should apply to colleges that he/she has a good chance of getting into, instead of applying to colleges that they may never get into. This will help in reduction of cost as students will be applying to only those universities that they are highly likely to get into. Our prepared models work to a satisfactory level of accuracy and may be of great assistance to such people. The web-application includes a user-friendly interface. It requires the user to fill in some of the mandatory details and provides the student with the probability of securing admission in the college.

6.2 Future Work

The project can be scaled in multiple dimensions for future work. Currently, the prediction of appropriate college selection is limited to NIT-IIT Colleges, this can gradually be increased to cover other exams too like NEET, MHT-CET, and many more, offering service to students of multiple streams, grades and entrance exams.

Further research and advancements in data mining or machine learning can lead to replacement of Decision Tree Classifier with a better alternative. Thus, the accuracy of the system can be improved.

Chapter 7

REFERENCES

[1] Subba Reddy.Y and Prof. P. Govindarajulu," A survey on data mining and machine learning techniques for internet voting and product/service selection", IJCSNS International Journal of Computer Science andNetwork Security, VOL.17 No.9 September 2017.

[2] Zhibo Wang, Jilong Liao, Qing Cao, Hairong Qi, and Zhi Wang, "Friend book: A Semanticbased Friend Recommendation System for Social Networks IEEE Transactions on Mobile Computing.

[3] J. Bobadilla et al. "Knowledge-Based System" Elsevier B.V.

[4] Hector Nunez, Miquel sanchez-Marre, Ulises Cortes, Joaquim Comas, Montse Martinez, Ignasi RodriguezRoda, Manel Poch, "A Comparative study on the use of similarity measure in case-based reasoning to improve the classification of environmental system situations,", ELSEVIER, Environmental Modeling and Software (2003)

[5] DINO IENCO, RUGGERO G. PENSA and ROSA MEO, "From Context to Distance: Learning Dissimilarity for categorical Data Clustering," Journal Vol. X. 10 2009, pages 1- 10. [6] Duc Thang Nguyen, Lihui Chen, Chee keong Chan, "Clustering with Multi viewpoint Based Similarity Measure," IEEE Transactions on Knowledge and Data Engineering. Vol. 24. No. 6. June 2012.

[6] Abdul Hamid M. Ragab, Abdul Fatah S. Mashat, Ahmed M. Khedra," Hybrid Recommender System for Predicting College Admission", 12th International Conference on Intelligent Systems Design and Applications (ISDA), 2012, pp. 107-113.

[7] Higher Education in India — Shiksha
https://www.shiksha.com

[8] Kaggle Dataset
https://www.kaggle.com/datasets/rumbleftw/iit-nit-data