

STUDENTS COLLEGE SEAT ALLOTMENT PREDICTION MODEL USING MACHINE LEARNING TECHNIQUES

Charanaraj N¹, Dhanush V U², Jayanth S V³, Deril Quadras⁴, Associate Prof. Vasudev Shahapur⁵

^{1,2,3,4,5} Department of Computer Science and Engineering, Alva's Institute of Engineering and Technology, Mijar.

Abstract -The ease of making better choices and making better decisions in terms of selecting colleges is the main aim of this system. Our analysis of colleges for the students makes it easier for them to make an accurate decision about their preferred colleges. For such analysis, it requires future possibilities from record data, which can make predictions and recommendations for students. Our analysis with the machine learning classification methods would help to give probable accuracy, and this requires analytical methods for predicting future recommendation. Today, most students make mistakes in their preference list due to lack of knowledge, improper and incorrect analysis of colleges, and insecure predictions. Hence repent and regret after allotment. Our project will solve the general issue of the student community by using machine learning technology. In this system, the Random Forest and Decision Tree machine learning classification algorithm is going to use.

Key Words:Machine Learning, Decision Tree and Random Forest Algorithm.

1.INTRODUCTION

At present, there are sixteen IITs in India, for which admission is governed by DTE (Directorate of Technical Education). DTE carries out the admission through CAP (Centralized Admission Process). The process is done through the cap rounds and is very confusing for students to analyze the perfect college. The student's needed to verify the documents at Facilitation Centre and are supposed to give their preference list of colleges. Then based on their Subject marks, Category, and other attributes, college is allotted to them in three or more consecutive forms. It's very difficult for the students to and out suitable colleges for them based on their Subject score, Aptitude Test, Technical Skills, English Skills, Olympiads, reading and writing skills, memory capability score, etc. Various colleges provide a degree in IITs in various branches. Though analysis of colleges and their cut-offs are required to get the most correct preference list. It is a very tedious job for a student to understand the suitable colleges which provide preferred branch and to analyses, it's last year's cut-offs 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.

The main objective of this project is to predict the College and Department allocation for the students based on their marks and skills.

2. System Requirements and Specification

A Software Requirements Specification may be a complete description of the behavior of the system to be developed. SRS may be a document that completely describes what the proposed software should do without describing how the software will roll in the hay. It is a two-way policy that both the client and the organization understand the requirements at any given point of time. SRS document itself is precise and it provides the functions and capabilities of a system that it should provide. The purpose of SRS is to bridge the communication gap between the parties involved in the development of the software. It serves as an input to design specifications. It also serves as the parent document to subsequent documents. Therefore, the SRS should be easy to know and also should contain sufficient details within the system requirements so that a design solution is often devised easily.

The document gives a detailed description of both functional and non-functional requirements. The purpose of requirements and specifications to obviously and unambiguously articulate the product's purpose, features, functionality, and behavior.

Table -1: Hardware and Software Requirements

System Processor	Core i5 8th Gen.
Hard Disk	500GB.
RAM	4GB .
Operating System	Windows 10.
Programming Language	Python.
Framework	Anaconda.
IDE	Jupyter Notebook.

3. System Analysis and Design

The analysis is a process of collecting and interpreting facts, identifying the problems, and decomposition of a system into its components. System analysis is conducted to study a system or its parts to identify its objectives. It is a problem-solving technique that improves the system and ensures that all the components of the system work efficiently to accomplish its purpose. Analysis specifies what the system should do. A system must have three basic constraints:

1. A system must have some structure and behavior which is meant to realize a predefined objective.
2. The Interconnectivity and the interdependence must exist among the system components.
3. The objectives of the organization have a better priority than the objectives of its subsystems.

The Existing system has Making a wise career decision that is extremely important for everybody. In recent years, decision support tools and mechanisms have assisted us in making the proper career decisions. This enables a student who wishes to pursue Engineering, make up good decisions, using the help of a Decision Support System. The last 3 years' information has been obtained from the web site of Directorate of Technical Education, India (DTE) which makes it freely available. Using Decision Rules, results are computed from which a student can choose which stream and college he/she can opt for based on Entrance Exam marks he/she has scored. To make the results more relevant, an inquiry within the already created decision system is performed. A student has to enter his/her Entrance Exam scores and the stream he/she wishes to opt for. Based on the entered information, the decision system will return colleges and streams categorized as Ambitious, Best Bargain, and Safe.

The Proposed system consists of distinct modules like Data Acquisition and Preprocessing, Feature Selection and Data Preparation, Model Construction and Model Training, and Model Validation and Result Analysis. In addition to these, there are making better choices and making better decisions in terms of selecting colleges is the main aim of this system. Our analysis of colleges for the students makes it easier for them to make an accurate decision about their preferred colleges. For such analysis, it requires future possibilities from record data which can potentially make predictions and recommendations for students. Our analysis with the machine learning classification methods would help to give probable accuracy and this requires analytical methods for predicting future recommendation. Today, most students make mistakes in their preference list due to lack of knowledge, improper and incorrect analysis of colleges, and insecure predictions. Hence repent and regret after allotment. Our project will solve the general issue of the student community by using machine learning technology. This system using Random Forest and Decision Tree machine learning classification algorithm.

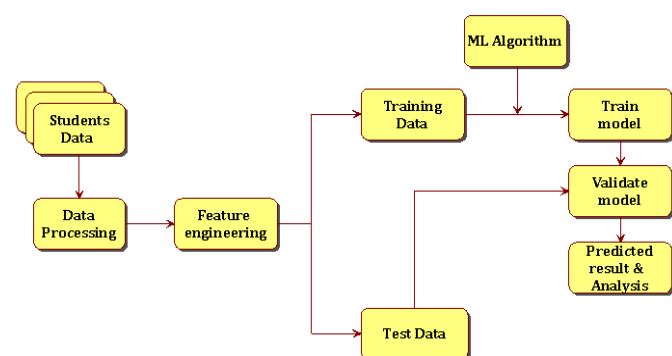


Fig -1: System Architecture

System Architecture design-identifies the overall hypermedia structure for the WebApp. Design is the goals established for a WebApp, the content to be presented, the users who will visit, and therefore the navigation philosophy that has been established. Content architecture focuses on how content objects and structured for presentation and navigation. WebApp architecture addresses how the appliance is that the structure to manage user interaction, handle internal processing tasks, affect navigation, and present content. WebApp architecture is defined within the context of the event environment during which the appliance is to be implemented.

The flow of Control is important to complete all tasks and meet deadlines. Many project management tools are available to help project managers manage their tasks and schedule and one of them is the flowchart. A flowchart is one of the seven basic quality tools used in project management and it displays the actions that are necessary to meet the goals of a particular task in the most practical sequence. Also called process maps, this type of tool displays a series of steps with branching possibilities that depict one or more inputs and transforms them into outputs. The advantage of flowcharts is that they show the activities involved during a project including the choice points, parallel paths, branching loops also because of the overall sequence of the process through mapping the operational details within the horizontal value chain. Moreover, this particular tool is very used in estimating and understanding the cost of quality for a particular process. This is done by using the branching logic of the workflow and estimating the expected monetary returns.

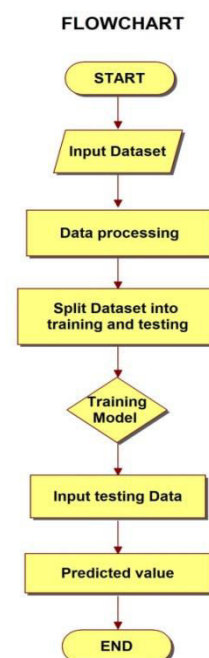
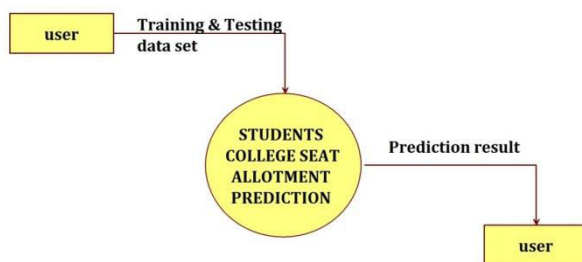


Fig -2: Flow Chart Diagram

A data flow diagram (DFD) is a graphic representation of the "flow" of data through an information system. A data flow chart also can be used for the visualization of knowledge processing (structured design). It is common practice for a designer to draw a context-level DFD first which shows the interaction between the system and out of doors entities. DFD's show the flow of knowledge from external entities into the system, how the info moves from one process to a

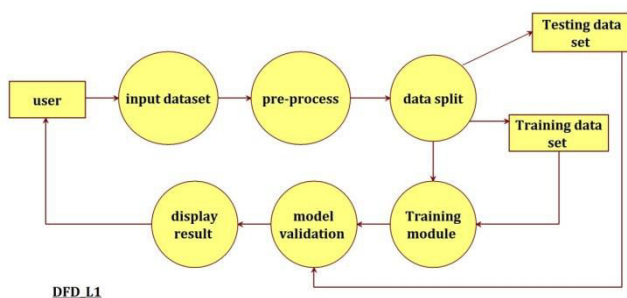
different, also as its logical storage. There are only four symbols:

1. Squares representing external entities, which are sources and destinations of information entering and leaving the system.
2. Rounded rectangles representing processes, in other methodologies, may be called 'Activities', 'Actions', 'Procedures', 'Subsystems' etc. which take data as input, do processing to it, and output it.
3. Arrows representing the data flow, which can either, be electronic data or physical items. Data can't flow from data store to the data store except via a process, and external entities are not allowed to access data stores directly.
4. The flat three-sided rectangle is representing data stores should both receive information for storing and provide it for further processing.



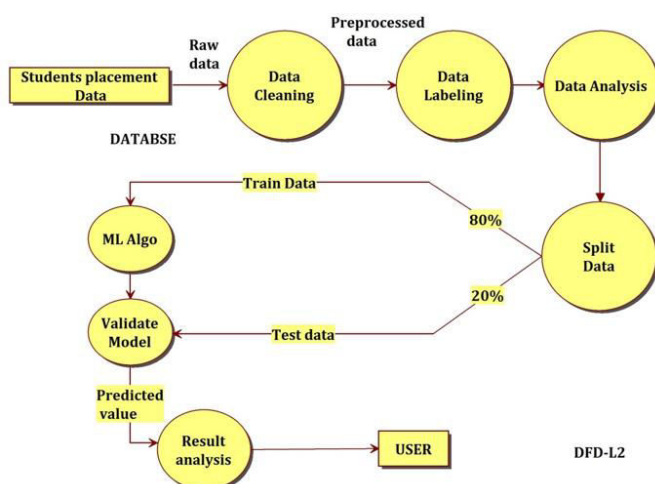
DFD L0

Fig -3: Level 0 Data Flow Diagram



DFD L1

Fig -4: Level 1 Data Flow Diagram



DFD-L2

Fig -5: Level 2 Data Flow Diagram

A use case is a set of scenarios that describing an interaction between a source and a destination. A use case diagram displays the relationship between actors and use cases. The main components of a use case diagram are use cases and actors. shows the use case diagram.

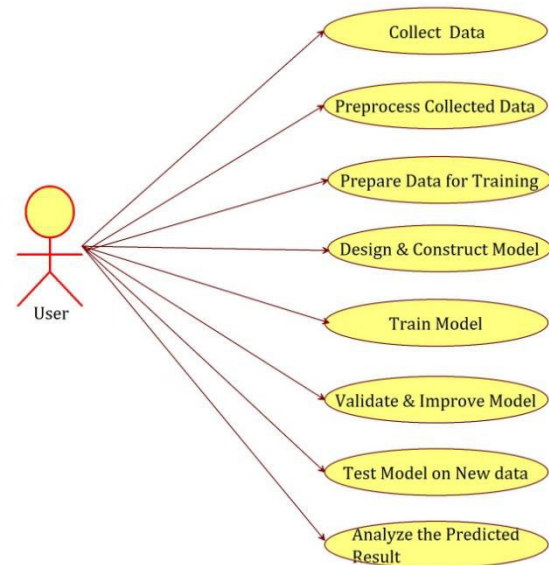


Fig -6: Use Case Diagram

4. METHODOLOGY

The system has several steps like Data Acquisition and Preprocessing, Feature Selection and Data Preparation, Model Construction and Model Training, and Model Validation and Result Analysis. We are using Machine Learning Algorithms such as Decision Tree and Random Forest to test our dataset.

Random Forest Algorithm

- Randomly select "K" features from total "m" features where $k \ll m$
- Among the "K" features, calculate the node "d" using the best split point
- Split the node into daughter nodes using the best split.
- Repeat the a to c steps until "l" number of nodes has been reached
- Build forest by repeating steps a to d for "n" number times to create "n" number of trees

Algorithm – 1: Random Forest Algorithm

Decision tree Algorithm

- It begins with the original set S as the root node.
- On each iteration of the algorithm, it iterates through the very unused attribute of the set S and calculates Entropy(H) and Information gain(IG) of this attribute.
- It then selects the attribute which has the smallest Entropy or Largest Information gain.
- The set S is then split by the selected attribute to produce a subset of the data.
- The algorithm continues to recur on each subset, considering only attributes never selected before.

Algorithm –2: Decision Tree Algorithm

First, the data is preprocessed to remove all errors and then split into training and testing datasets. The training dataset is then fit into the model to predict the values then the trained model is again tested and accuracy is calculated.

Planning to identify all the information and requirement such as hardware and software, planning must be done properly. The planning stage has two main elements namely data collection and the requirements of hardware and software. Machine learning needs two things to figure, data (lots of it) and models. When acquiring the data, be sure to have enough features (an aspect of data that can help for a prediction, like the surface of the house to predict its price) populated to train correctly your learning model. In general, the more data you've got the higher so make to return with enough rows. The primary data collected from the online sources remain in the raw form of statements, digits, and qualitative terms. The raw data contains errors, omissions, and inconsistencies. It requires corrections after carefully scrutinizing the completed questionnaires. The following stages are involved in the processing of primary data. A huge volume of data collected through field survey must be grouped for similar details of individual responses. Data Preprocessing may be a technique that's wont to convert the data into a clean data set. In other words, whenever the data is gathered from different sources it's collected in raw format which isn't feasible for the analysis. Therefore, certain steps are executed to convert the info into a little clean data set. This technique is performed before the execution of the Iterative Analysis. The set of steps is known as Data Preprocessing.

In this final stage, we'll test our classification model on our prepared image dataset and also measure the performance on our dataset. To evaluate the performance of our created classification and make it comparable to current approaches, we use accuracy to measure the effectiveness of classifiers. After model building, knowing the power of model prediction on a new instance is a very important issue. Once a predictive model is developed using the historical data, one would be curious about how the model will perform on the info that it's not seen during the model building process. One might even try multiple model types for an equivalent prediction problem, and then, would like to understand which model is that the one to use for the real-world decision-making situation, just by comparing them on their prediction performance (e.g., accuracy). To measure the performance of a predictor, there are commonly used performance metrics, like accuracy, recall, etc. First, the most commonly used performance metrics will be described, and then some famous estimation methodologies are explained and compared to each other. Performance Metrics for Predictive Modelling classification problems, the first source of performance measurements may be a coincidence matrix (classification matrix or a contingency table). The above figure shows a coincidence matrix for a two-class classification problem. The equations of the most commonly used metrics that can be calculated from the coincidence matrix are also given in Fig 7

As being seen in the figure, the numbers along the diagonal from upper-left to lower-right represent the correct decisions made, and the numbers outside this diagonal represent the errors. The true positive rate of a classifier is

		True Class	
		Positive	Negative
Predicted Class	Positive	True Positive Count (TP)	False Positive Count (FP)
	Negative	False Negative Count (FN)	True Negative Count (TN)

$$\text{True Positive Rate} = \frac{TP}{TP + FN}$$

$$\text{True Negative Rate} = \frac{TN}{TN + FP}$$

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

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

Fig -7: Confusion Matrix and formulae.

estimated by dividing the correctly classified positives (the true positive count) by the total positive count. The false-positive rate (also called a false alarm rate) of the classifier is estimated by dividing the incorrectly classified negatives (the false negative count) by the total negatives. The overall accuracy of a classifier is estimated by dividing the total correctly classified positives and negatives by the total number of samples.

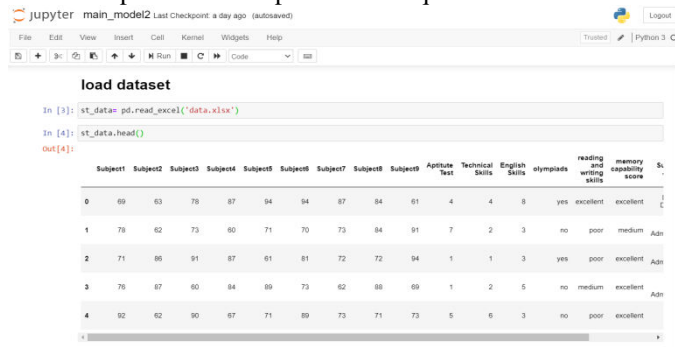
5. TESTING AND RESULTS

Testing is an important phase in the development life cycle of the product. This is the phase where all the errors remaining in all the phases will be detected. Hence testing plays a very critical role in quality assurance and ensuring the reliability of the software. During the test, the program to be tested is executed with a set of test cases and the output of the program for n test cases is evaluated to determine whether the program is performing as expected. The testing of software or hardware is conducted on a complete system to evaluate its compliance with the specified requirement. System testing is performed on the entire system in the context of functional requirements specification and/or system requirement specification. Testing is an investigatory stage, where the focus is to have almost a destructive attitude and test not only the design but also the behavior and even the believed expectation of the customer. The testing objectives are as follows:

1. Testing is the process of executing the program with the intent of finding an error.
2. an honest test suit is one that features a high probability of finding a mistake.
3. Testing cannot show the absence of defects.

A result is the final step consequence of actions or events expressed qualitatively. Performance analysis is an

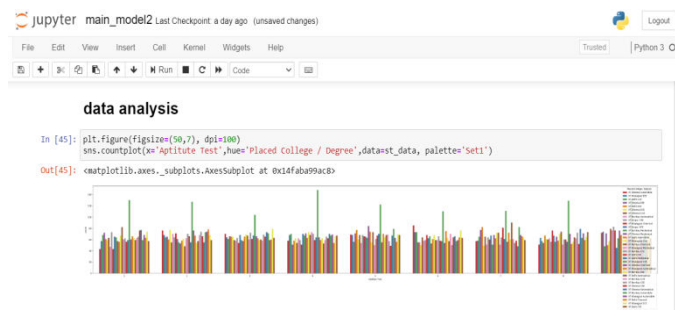
operational analysis, is a set of the basic quantitative relationship between the performance quantities.



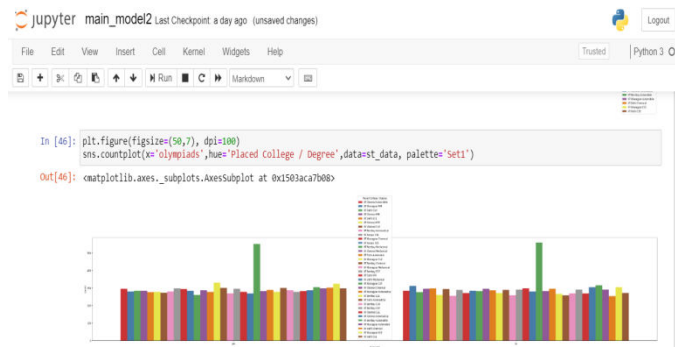
Result-1: The above figure says we are loading the dataset manually in .xlsx format and it shows results in table format



Result-2: The model Understand only numerical value that's why we are converting all the sting values into numerical value.

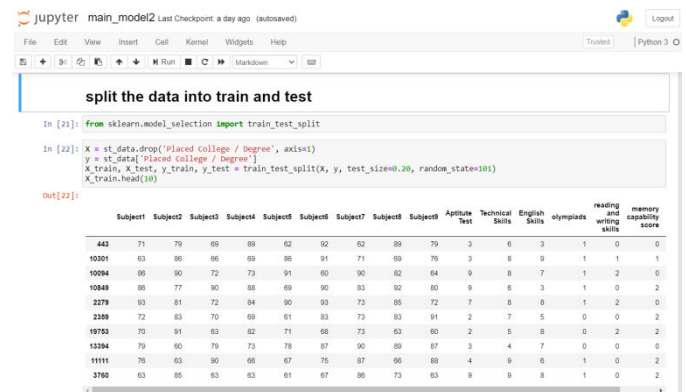


Result-3: analyzing the aptitude test results data using matplotlib function for a bar graph representation. Here dpi is a dots per inch with default value 100 and palette is a color brightness of the seaborn bar graph.

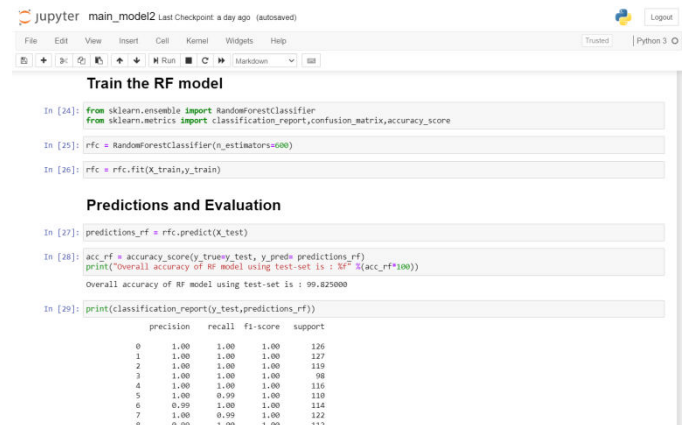


Result-4: analyzing the Olympiads data using matplotlib function for bar graph representation. Here '0' means NO and

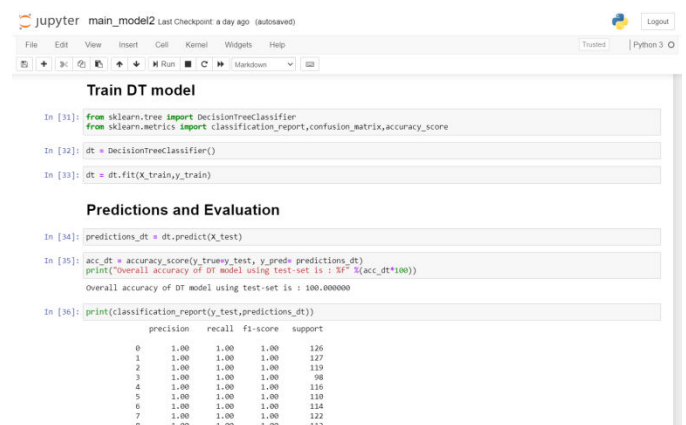
'1' means YES. Out of 34 colleges how many students are attended in Olympiads exam.



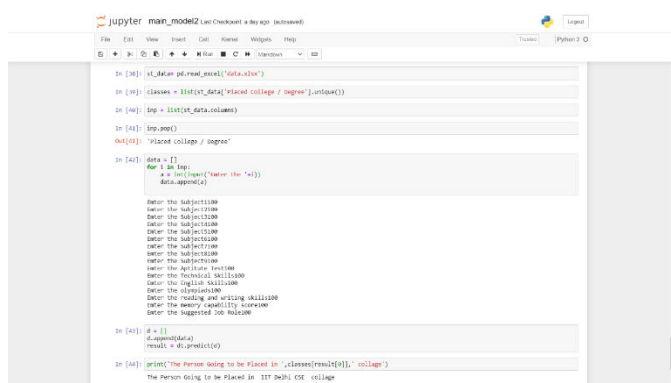
Result-5: Splitting the data into train and test. Here X is a input, Y is a Output in train test split function accepting four parameters like input, output, 20% data size, and random value. In .head(10) prints only ten rows of randomly generated value and we stored it in separate excel file.



Result-6: In Random Forest Algorithm First we importing random classifier and classification report, confusion matrix, accuracy score, and 600 estimators to fit x train and y train. Using x test we are getting accuracy of 99.82% and also we can find where it went wrong.



Result-7: In Decision Tree Algorithm First we importing Decision tree classifier and classification report, confusion matrix, accuracy score, and Her we are not calling estimators because decision tree algorithm itself a different algorithm. Using x test we are getting accuracy of 100% and also we can find where it went wrong



```

In [34]: data = pd.read_csv('data.csv')
In [35]: classes = list(data['placed college / degree'].unique())
In [36]: imp = list(data.columns)
In [37]: imp.pop()
In [38]: y_label = 'placed college / degree'
In [39]: data = {}
for i in len(data):
    data.append(data[i])
    data.append(i)
In [40]: data = {}
for i in len(data):
    data.append(data[i])
    data.append(i)
In [41]: print("The person going to be placed in :", classes[result[0]], " college")
The person going to be placed in : IT DDU CSE college

```

Result-8: This is the last stage of a model here we can enter manually to getting a results. In this step what happens means if one subject marks is less than 40 it will not going to predict and through error “ the person is not meet the criteria” and other one is if the student marks is more than 40 it will predict the college and department for that student.

6. CONCLUSION

We find that the best model for predicting the success, whether a student will pass their college course with better marks of a student differs between our datasets best prediction model is the Random Forest with an accuracy of 99.82% and better precision, recall, and support values than the other models. Using the Decision tree algorithm at an accuracy of 100% and likewise better precision, recall, and support values than the other models.

Interestingly, all of the models we utilized outperformed the current standards by approximately 200%. We have proved that machine learning models are more capable and accurate than current seat allotment standards. Using these models will increase the accuracy and precision of student seat allotment in either remedial or transfer-level coursework. Employing the models will decrease the duration and funding wasted by both the student and college. Finally, these models provide a chance for more students to be able to stand on their abilities throughout the admission process.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany a successful completion of any task would be incomplete without the mention of people who made it possible, success is the epitome of hard work and perseverance, but steadfast of all is encouraging guidance.

So, with gratitude we acknowledge all those whose guidance and encouragement served as beacon of light and crowned the effort with success.

We thank our project guide **Vasudev Shahapur**, Associate Professor in Department of Computer Science & Engineering, who has been our source of inspiration. He has been especially enthusiastic in giving his valuable guidance and critical reviews.

The selection of this project work as well as the timely completion is mainly due to the interest and persuasion of my project coordinator **Vasudev Shahapur**, Associate

Professor, Department of Computer Science & Engineering. We will remember his contribution for ever.

We sincerely thank, **Dr. Manjunath Kotari**, Professor and Head, Department of Computer Science & Engineering who has been the constant driving force behind the completion of the project.

We thank Principal **Dr. Peter Fernandes**, for his constant help and support throughout.

We are also indebted to **Management of Alva's Institute of Engineering and Technology, Mijar, Moodbidri** for providing an environment which helped us in completing the project.

Also, we thank all the teaching and non-teaching staff of Department of Computer Science & Engineering for the help rendered.

Finally we would like to thank my parents and friends whose encouragement and support was invaluable.

REFERENCES

1. Kandapriya Basu, Treena Basu, Ron Buckmire and Nishu Lal “Predictive Models of Student College Commitment Decisions Using Machine Learning”.
2. Annam Mallikharjuna Roa, Nagineni Dharani, A. Satya Raghava, J. Buvanambigai, and K. Sathish Says in the paper “College Admission Predictor”.
3. Charushila Patil, Akshay Diwate, Jayesh Baviskar, and Tejas Gholap Says in the paper “EFFICIENT CAP ROUND PREDICTION FOR STUDENTS”.
4. Anthony Dalton, Justin Beer, Sriharshasai Kommanapalli, and James S. Lanich, “Machine Learning to Predict College Course Success”.
5. Rahul Sathawane, Rohan Battulwar, Prasheel Fuley, Ananiya Mahajan, Shivam Joshi5, and Roshan Chaturpale “College Guesstimate”