“Student Placement Guidance using Artificial Neural Networks”

Submitted in partial fulfillment of the requirement of University of Mumbai

For the Degree of

Bachelor of Engineering
(Computer Engineering)

By

Prashant Mahajan
Pratik Deshpande
Ravi Chandak
Tejas Nanaware

Under the guidance of
Prof. Mahendra Patil & Dr. Sameer Sahasrabuddhe

AET’s
Atharva College of Engineering
Atharva Educational Complex, Malad Marve Road,
Charkop Naka, Malad (W), Mumbai, 400095
(Affiliated to University of Mumbai)
April 2018

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea / data / fact / source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Mr. Prashant Mahajan
Atharva College of Engineering

Mr. Pratik Deshpande
Atharva College of Engineering

Mr. Ravi Chandak
Atharva College of Engineering

Mr. Tejas Nanaware
Atharva College of Engineering

Date: 15th April 2018

Project Report Approval for B.E.

This project report entitled **"*Student Placement Guidance using Artificial Neural Networks*"** by ***Prashant Mahajan, Pratik Deshpande, Ravi Chandak, Tejas Nanaware*** is approved for the degree of ***Bachelor of Engineering in Computer Engineering.***

Internal Examiner

External Examiner

Date: _____

Place: _____

College Seal

AET'S
ATHARVA COLLEGE OF ENGINEERING

CERTIFICATE

This is to certify that

Prashant Mahajan
Pratik Deshpande
Ravi Chandak
Tejas Nanaware

Have satisfactorily completed the requirements of the B.E Project Report

On

"Student Placement Guidance Using Artificial Neural Networks"

As prescribed by the ***University of Mumbai*** *Under the guidance of*

Guide
Prof. Mahendra Patil & Dr. Sameer Sahasrabuddhe

Prof Mahendra Patil	**Dr. S.P. Kallurkar**	**Dr. Sameer Sahasrabuddhe**	
Project Coordinator	**Head of Department**	**Principal**	**Expert Guide (IIT Bombay)**

(April 2018) **College Seal** **Date:** April 15, 2018

INDEX

TABLE OF CONTENTS

LIST OF FIGURES

List of Tables

ABSTRACT

Engineering students are skeptical about what they want to pursue after graduation. With wide options available, ranging from campus recruitments to Masters, students are perplexed, adding factors like salaries and different job opportunities makes it even worse. There aren't any reliable platforms where a student can predict the outcomes from the start of engineering and take actions to bridge this gap for a better future.

Students studying in Engineering colleges feel the exigency to know where they stand in comparison to others, and what kind of placement they would get. The training and placement offices come in the picture when a student enters final year, but they are of no use to a student planning for future studies. Prediction about the student's performance is an integral part of an education system, as the overall growth of the student is directly proportional to the success rate of the students in their examinations and extra-curricular activities. Therefore, there are many situations where the performance of the student needs to be predicted, for example, in identifying weak performing students and taking actions for their betterment.

The students have no platform to check their current position and build on their strengths. The platforms currently available, have not been trained on real and complete data sets, and do not learn from their wrong predictions which reduces the accuracy, in the long term. To achieve a better accuracy and a system that learns with every wrong prediction it has made, we intend to use Neural Networks, which will cause a continuous accuracy growth. We aim to develop one, complete, robust platform, where students can check their current status, and the range of placements they would get, on an easy to use web application. To ensure effective results, the model will be trained on a real data set and a vast number of qualitative as well as quantitative parameters will be considered.

1. Introduction

Campus placement of a student plays a very important role in a college. Campus placement is a process where companies visit colleges and identify students who are talented and qualified, before they complete their graduation. Therefore, taking a wise career decision regarding the placement after completing a particular course is crucial in a student's life. An educational institution contains a large number of student records. Therefore, finding patterns and characteristics in this large pool of data, will help find parameters that are the most important for this placement procedure.

The prediction of engineering students, about where they can be placed, from the second year and onwards, will help to improve efforts of student for proper progress. [1] It will help teachers to take proper attention towards the progress of the student during the course of time. It will help to build reputation of the institute for having such a sophisticated system in place which helps the students to train and practice for campus placements. The present study concentrates on helping the students, bridging the gap between the industry and the curriculum, and showing them the path to a better future. We apply data mining and machine learning techniques using Artificial Neural Networks, in order to interpret the potential of the student.

Data Mining refers to extracting or mining useful patterns from a large database. It is knowledge discovery in large amount of data. Neural networks and Fuzzy Inference System is a part of Soft Computing, that works well with low level computing, gaining experience and knowledge from its mistakes, and the later works well with the irregularities and the incompleteness of the data. Neural Networks has various applications in number of sectors. Whilst data mining tools can be used to find patterns in the large set of data which can help understand, business requirements, market analysis and management. Learning using neural networks can be classified into three types, supervised learning and unsupervised learning. Supervised learning is so named because the data scientist acts as a guide to teach the algorithm what conclusions it should come up with. It's similar to the way a child might learn arithmetic from a teacher. Unsupervised machine learning is more closely aligned with what some call true artificial intelligence — the idea that a computer can learn to identify complex processes and patterns without a human to provide guidance along the way. We're going to use unsupervised learning techniques to provide guidance for the students.

1.1 Need

The questions this work can provide the solutions to, can be given as follows:

1. What are the types of students the college has according to their academic scoring?
2. Predict in advance, the placement status of the pre - final year students.
3. What should the students learn next, to have a better chance of placement?
4. What are the clusters of domains, students in a college fall into?
5. Provide info to the hiring companies and prompt questions and proof of the student details and help companies graphs and visuals to filter students.

So, we aim to fix the above vulnerabilities and answer those questions in our system, using Artificial neural networks.

1.2 Basic Concept

In this technical world many different techniques are used by humans for different purposes. There are many softwares and applications that are developed for reducing human effort. Data mining techniques and neural networks can be used to find patterns in large databases, and to guide decisions about future activities. It's expected that by using neural networks, the model will learn on its own, and work efficiently even with minimal input from the user to recognize [14]. The model can be useful to understand the unexpected and provide an analysis of data followed by decision-making which is examined and it ultimately leads to strategic decisions and business intelligence. The simplest word for knowledge extraction and exploration of volume data is very high and the more appropriate term is, "Exploring the hidden knowledge of the database." This process includes preparation and interpretation of results.

1.3 Applications

There are various applications of this system, few of them being

1. Student's will have an idea of where they stand and what to do next to bridge the gap and become better
2. Student's will have a clear option which will help reduce the ambiguity in their mind.
3. The college will have the statistics of all the students and what are the different domains they fall into.
4. The college, will be able to take decisions to improve students and have better, insights of the students.
5. The student's will get their resume based on the data they feed.
6. The corporate end, will be able to filter the students and download the resumes of the students, according to their needs.
7. The corporate end and the college end will be able to post there, requirements and send messages directly to the student, or maybe even globally.
8. The corporate end there will be interview questions that will be prompted, different for different students based on the student resume.

2. Review of Literature

2.1.1 ID3 classification algorithms

Hitarthi Bhatt identified relevant attributes based on quantitative and qualitative aspects of a student's profile such as CGPA, academic performance, technical and communication skills and designed a model which can predict the placement of a student using ID3[1].

2.1.2 Gradient Descent algorithm

One of the most prominent work on prediction of placement for students has been cited by Ajay Shiv Sharma, Swaraj Prince, Shubham Kapoor and Keshav Kumar where they presented the development of placement predictor system (PPS) using logistic regression model. They used Machine learning technique to design and implement a logistic classifier that predicts the probability of the student to get placed along with Gradient Descent algorithm. The results are generated from an open source GNU Octave programming tool. The developed model has been applied to predict the placement of students at training and placement office (TPO) [5].

2.1.3 Classification algorithms

S. Taruna and Mrinal Pandey implemented an empirical analysis on predicting academic performance by using classification techniques or mapping of data items into predefined groups and classes using supervised learning. They compared five classification algorithms namely Decision Tree, Naïve Bayes, Naïve Bayes Tree, K-Nearest Neighbour and Bayesian Network algorithms for predicting students' grade particularly for engineering students using a four-class prediction problem [6].

State of the art regression algorithms Kotsiantis and Pintelas, 2005 predicted the student marks (pass and fail classes) using the regression methods and available previous data. The scope of this work compares some of the state of the art regression algorithms in the application domain of predicting students' marks. A number of experiments have been conducted with six algorithms, which were trained using datasets provided by the Hellenic Open University [7].

2.1.4 Logistic Regression

Saha and Goutam applied logistic regression method on the examination result data and analyzed the data under the University Grant Commission sponsored project entitled - Prospects and Problems of Educational development (Higher Secondary Stage) in Tripura - An In-depth Study [8].

2.1.5 Classification Algorithms

Syed Tanveer Jishan, Raisul Islam Rashu, Naheena Haque and Rashedur M Rahman predicted student's performance using attributes such as Cumulative Grade Point Average, Quiz, Laboratory, Midterm and Attendance marks. Also, in order to improve the prediction model, they introduced some preprocessing techniques so that the prediction model provides with more precise results by applying three classification algorithms, e.g., Naïve Bayes, Decision Tree and Neural Network [9].

2.1.6 Decision tree algorithm C4.5

Zhiwu Liu and Xiuzhi Zhang used decision tree algorithm C4.5 to establish a classification rule and an analysis-forecasting model for students' marks. They described how the analysis-forecasting result can be used to find out the factors which can affect students' marks, so some negative learning habits or behaviors of students can be revealed and corrected in time and the teaching effect of the teacher can be checked, the teaching management can also be assisted [10].

2.1.7 Genetic Programming Algorithm

Carlos Márquez-Vera, Alberto Cano, Cristóbal Romero and Sebastián Ventura predicted the student's failure at school using genetic programming algorithm and different data mining approaches using cost sensitive classification in order to resolve the problem of classifying imbalanced data. A genetic programming algorithm and different data mining approaches were proposed for solving the problems using real data about 670 high school students from Zacatecas, Mexico [11].

2.1.8 Data mining techniques

The initial results from Kabakchieva and Dorina's implemented research project aimed to show the high potential of data mining applications for university management. This paper is focused on the implementation of data mining techniques and methods for acquiring new knowledge from data collected by universities. The main goal of the research is to reveal the high potential of data mining applications for university management [12].

2.1.9 ID3 and C4.5 classification algorithms

Kalpesh Adhatrao, Aditya Gaykar, Amiraj Dhawan, Rohit Jha and Vipul Honrao analyzed the data set containing information about students, such as gender, marks scored in the board examinations of classes X and XII, marks and rank in entrance examinations and results in first year of the previous batch of students. By applying the ID3 and C4.5 classification algorithms on this data, they have predicted the general and individual performance of freshly admitted students in future examinations[13].

3. Report on the Present Investigation (Existing Systems)

A lot of research has been done on the topic of placement prediction in the past decade. Different researcher used different methods to produce intended results. Naik et. al. (2012) used classification algorithm to predict final result and placement of the students[2]. They used data mining techniques for producing knowledge about students of Master of Computer Application (MCA) course before admitting them to the course. The overall error occurred to classify validation data using MCA result prediction classification tree was 38.46% while for validating placement prediction classification tree it was 45.38%. Sharma et.al. (2014) used logistic regression model to create a Placement Predictor System (PPS)[5]. They generated results from an open source GNU Octave programming tool which brings about 83.33% accuracy. Another approach for placement prediction is taken by Bhatt et. al. (2015) where they used ID3 Decision Tree Algorithm[1]. While predicting the placement they incorporated both qualitative and quantitative parameters of a student to achieve better results. Giri et. al. (2016) used machine learning model of K-Nearest Classifier to predict probability of a undergrad student getting placed in an IT company[3]. They compared the results of the same against the results obtained from other models like Logistic Regression and SVM and proved that KNN produces better results.

Based on above research, we proposed usage of Artificial Neural Network for placement guidance which will provide higher accuracy compared to other algorithms. Though attempts were made to create such system taking into consideration both qualitative and quantitative parameters; amount of qualitative factors considered for the same was very less which we intend to change by using more than fifty qualitative parameters which constitutes an important role in placement of a student consequently improving the accuracy of the system.

4. Aim and Objectives

4.1 Aim

Our project aims to create placement guidance system which will use the concept of Artificial Neural Networks. We intend to combine both qualitative and quantitative parameters for the decision making process. To do so we consider the academic history of the student as well as their skill set like, programming skills, communication skills, analytical skills and teamwork, which are tested by the hiring companies during the recruitment process. Though many research has been done previously on placement prediction using different methods, none of them gave consideration to qualitative parameters to a large extent, which plays a vital role in placement of any student. Thus, by taking this into account our aim is to achieve a system with greater than 85% of accuracy.

4.2 Objectives

Predicting the placement of a student gives an idea to the placement office as well as the student on where they stand. Not all companies look for similar talents. If the strengths and weaknesses of the students are identified it would benefit the student in getting placed. The placement Office can work on identifying the weaknesses of the students and take measures of improvement so that the students can overcome the weakness and perform to the best of their abilities. Thus the key lies in assessing the capabilities of the student in the right areas and subjecting them to the right training which is essentially our objective behind creating such system.

5. Problem Statement

Students studying in Engineering colleges feel the exigency to know where they stand in comparison to others, and what kind of placement they would get. The training and placement offices come in the picture when a student enters final year, but they are of no use to a student planning for future studies. The students have no platform to check their current position and build on their strengths.

The platforms currently available, have not been trained on real and complete data sets, and do not learn from their wrong predictions which reduces the accuracy, in the long term.

Planning for future role constitutes an important role in any Engineering student's life. This necessitates a system to assist the academic planners to design a strategy to improve the performance of students that will help them in getting placed at the earliest.

In all of the previous systems placement prediction of a student was done in terms of binary values i.e. 0 and 1 which does not represent clear picture to the user. Creating a system which will guide a user in a better way like showing probability ranging from 0 to 1 is essential in such cases.

During placement prediction various attributes of a student plays vital role in whether that particular student will get selected or not. These attributes constitute both qualitative and quantitative parameters. Previous systems considered only qualitative parameters of a student overlooking personal aspects of a candidate such as his confidence, ability to work on a problem etc. Taking this into consideration a system incorporating both parameters will provide a better guidance to the students.

Some of the earlier systems used decision trees such as ID3 to provide placement prediction. But such methods are computationally too heavy and bound to break when large number of datasets are provided. A self-adaptive Artificial Neural Network will overcome this important drawback of previous systems.

6. Proposed System for Project

Students are most benefited by this application. The students can manage their profile and give tests about programming languages, logic building and other such topics. The college has the student's quantitative data like CGPA, marks, internships, projects and certifications. The test data which gives the qualitative parameters and the quantitative parameters aid the predictive model that uses maximization of entropy. Once the prediction graph is generated, we have to fit a curve to map the data and apply the entropy maximization algorithm so that prediction can be done accurately. The students get the statistical data that will help with analytics and knowing how to improve themselves to get a better package. Statistical analytics also help the TPO to verify the data and if incorrect, TPO can change the data to maintain the accuracy.

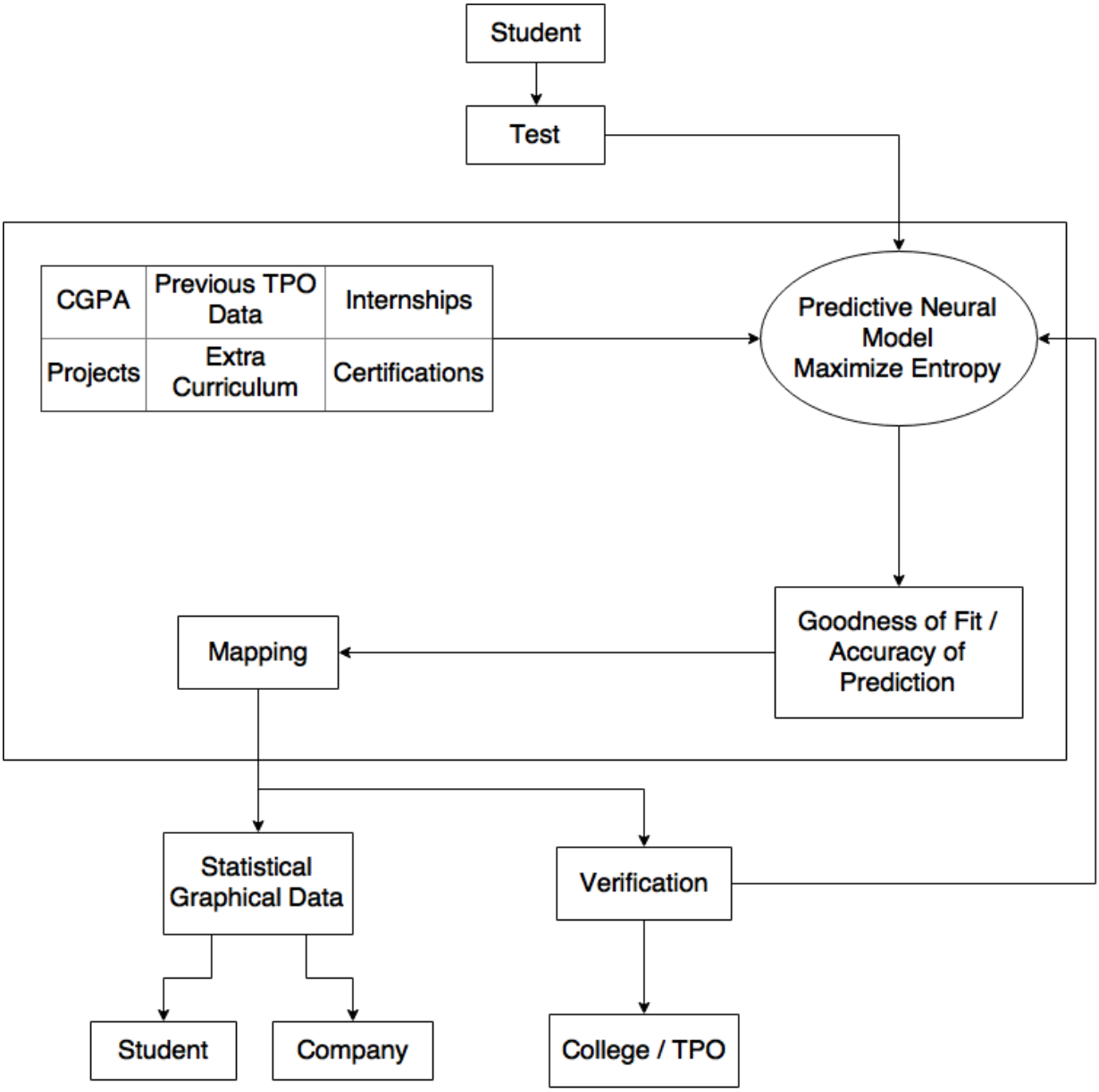


Figure 6-1: System Architecture

7. Requirement Analysis (SRS)

7.1 Functional Requirements

This section describes the functional requirements of the system for those requirements which are expressed in the natural language style. Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Our Web Application has three modules which are design for three different users. First is Students Portal where they can login through their portal page or registered themselves. Students can create their profile using personal dashboard. An Administrator Portal can login into his account and he/she will send emails regarding placement and companies and verify the details and apply filters on the data. View placement prediction analysis reports generated by our model .The Company Portal where interviewer can view student profile while taking interview and will be prompted questions based on the student resume. The main backend of our system is our Logistic Model, a mathematical model/machine which continually learns with every student's test data from database and it will process this information and will give final numeric value/probability of success or getting placed.

7.2 Non-Functional Requirements

A description and, where possible, target values of associated non-functional requirements. Non-functional requirements detail constraints, targets or control mechanisms for the new system. They describe how, how well or to what standard a function should be provided. For example, levels of required service such as response times; security in the form Login Authentication for Student and Admin and access requirements; technical constraints; required interfacing with users' and other systems. Service level requirements are measures of the quality of service required, and is crucial to capacity planning and physical design. Identify realistic, measurable target values for each service level. These include service hours, service availability, and responsiveness due to UI design, throughput and reliability regarding the display of results on searching about company or students. Access restrictions should deal with what data needs protected; what data should be restricted to a particular user role; and level of restriction required, e.g. physical, password, view only. Non-functional requirements may cover the system as a whole or relate to specific functional requirements.

8. Scope (Feasibility of Project)

8.1 Scope

1. The work currently deals with predicting the results based on last 5 year's training and placement data, examination department's data, and the extra curricular data of the students.
2. The present work will be using a regression model which is training itself, so the accuracy of the system will increase over time, making the system more reliable over time.
3. Even if the parameters change or the system will adapt to it over time.
4. In future, this system will be extended on the web end to give better outputs and competitive insights and also providing more statistics to the corporate end.
5. This system can be used for many years, as the system is adapting with the data, and the accuracy is increasing over time.

8.2 Feasibility

8.2.1 Operational Feasibility

Operational Feasibility is the ability to utilize, support and perform the necessary tasks of a system such as model training and mapping the different types of parameters. It includes everyone, from who creates, operates or uses the system. To be operationally feasible, the system must fulfill a need required by the business.

8.2.2 Technical Feasibility

Technical Feasibility, involves development of a working model of the product or a service. The concept of Predictive analysis is used for finding the chances of prediction and other guiding parameters using Artificial Neural Network algorithm. It is not necessary that the initial materials and components of the working model represent those that actually will be used in the finished product or the service.

8.2.3 Economic Feasibility

Economic feasibility is the cost and logistical outlook for a business project or endeavor. Prior to embarking on a new venture, most businesses conduct an economic feasibility study, which is a study that analyzes data to determine whether the cost of the prospective new venture will ultimately be profitable to the college since they would be able to have more information about the ability of their students. College can conduct seminars and extra training sections to improve the caliber of the students. Economic feasibility is sometimes determined within the organization, while other times companies hire an external company, who has an expertise in this domain, to do the task for them.

9. Methodology

Methodology being used here is Agile Methodology. This method promotes continuous iteration of development and testing throughout the software development lifecycle of the project. During the life cycle of the product iterations were built simultaneously providing efficient and quality output.

Implementation Plan is comprised of following major steps:

1. Gathering Data
2. Data Preparation
3. Choosing a model
4. Training
5. Evaluation
6. Hyper-parameter tuning
7. Prediction

9.1 Gathering Data:

This step is very important because the quality and quantity of data that you gather will directly determine how good your predictive model can be. In this case, data we collected consisted student's marks across all semester.

9.2 Data Preparation:

Data preparation, where we load our data into a suitable place and prepare it for use in our machine learning training. This is also a good time to do any pertinent visualizations of your data, to help you see if there are any relevant relationships between different variables you can take advantage of, as well as show you if there are any data imbalances. This step comprised of converting data from different formats to Excel and perform different data visualisation techniques to get the insights about the features.

We'll also need to split the data in two parts. The first part, used in training our model, will be the majority of the dataset. The second part will be used for evaluating our trained model's performance. We don't want to use the same data that the model was trained on for evaluation.

9.3 Choosing a model:

The next step in our workflow is choosing a model. There are 60+ predictive modelling algorithms to choose from. We must understand the type of problem and solution requirement to narrow down to a select few models which we can evaluate. The algorithms we considered –

1. Logistic Regression
2. Support Vector Machine (SVM)
3. K-Nearest Neighbour (KNN)
4. Decision Tree
5. Artificial Neural Network (ANN)

After getting confidence score from each model we ranked our evaluation of all the models to choose the best one for our problem. We decided to go forward with ANN because it can handle much more variability as compared to traditional models.

9.4 Training:

In this step, we will use our data to incrementally improve our model's ability to predict the probability of a student being placed. The training model consists of Weights (W) and biases (b) where weights is nothing but a collection of features.

The training process involves initializing some random values for W and b and attempting to predict the output with those values. We can compare our model's predictions with the output that it should produced, and adjust the values in W and b such that we will have more correct predictions. This process then repeats. Each iteration or cycle of updating the weights and biases is called one training "step".

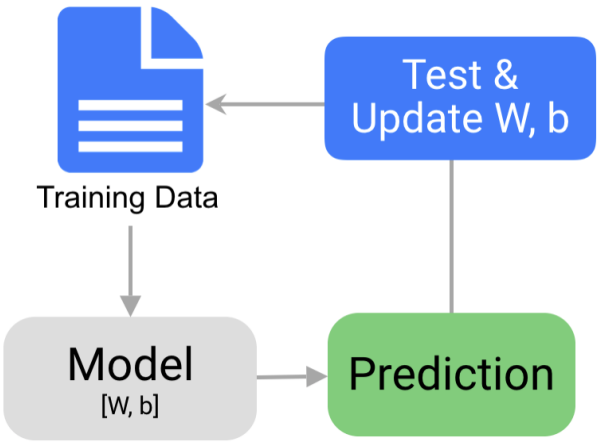


Figure 9-1: Training Model

9.5 Evaluation:

Once training is complete, it's time to see if the model is any good, using evaluation. This is where that dataset that we set aside earlier comes into play. Evaluation allows us to test our model against data that has never been used for training. This metric allows us to see how the model might perform against data that it has not yet seen.

9.6 Parameter Tuning:

Further improvement of the model is done using Parameter tuning. There were a few parameters we implicitly assumed when we did our training, and in this step we go back and test those assumptions and try other values.

One example is how many times we run through the training dataset during training. We can "show" the model our full dataset multiple times, rather than just once. This leads to higher accuracies.

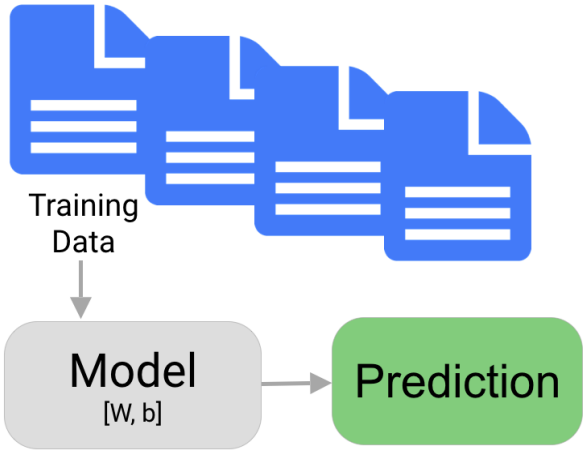


Figure 9-2: Parameter Tuning

Another parameter is "*learning rate*". This defines how far we shift the line during each step, based on the information from the previous training step. These values all play a role in how accurate our model can become, and how long the training takes.

9.7 Prediction:

Machine learning is using data to answer the questions. So Prediction, or inference, is the step where we get to answer some questions. In this step we used our model to predict probability of a student getting placed.

10. Design Details

10.1 Context Level Diagram

The Context Diagram shows the system under consideration as a single high-level process and then shows the relationship that the system has with other external entities (systems, organizational groups, external data stores, etc.). A Context Diagram and a DFD for that matter provides no information about the timing, sequencing, or synchronization of processes such as which processes occur in sequence or in parallel. Therefore it should not be confused with a flowchart or process flow which can show these things.

The Context Level Diagram shows the working of our system. All the three modules viz. Student, Corporate end and College is connected with our placement system. The scores of student would be available to college and the companies as well. The college can mentor the students in proper direction by looking at the student's performance and will also send notifications regarding the companies coming and the job profile they need. The company can directly look at the full profile of the student.

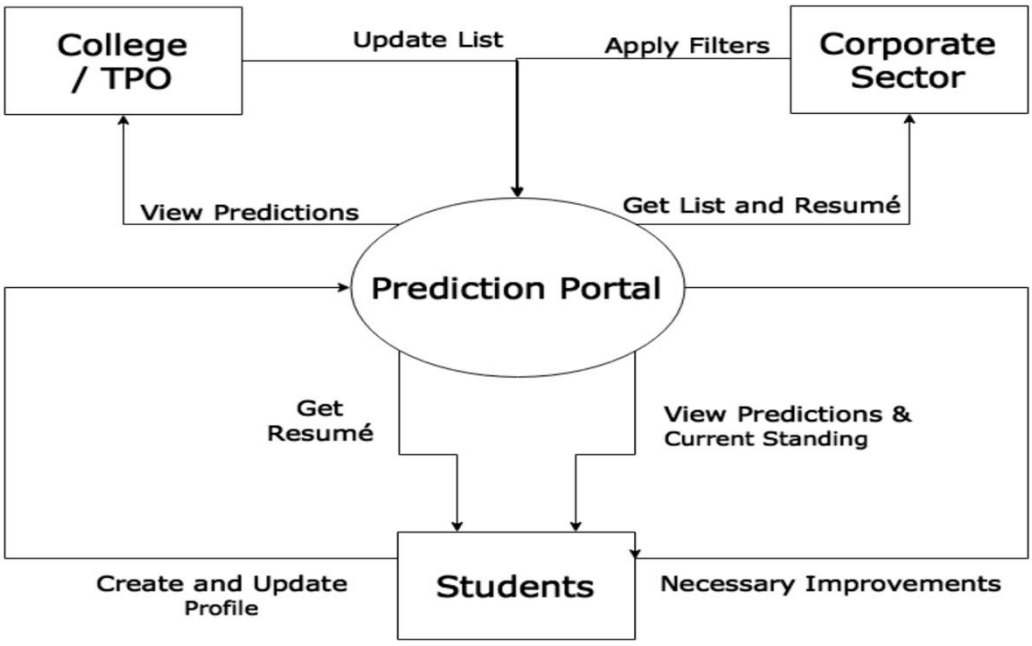


Figure 10-1: Context Level Diagram

10.2 Data Flow Diagram

10.2.1 DFD Level 1

A two-dimensional diagram, explains how data is processed and transferred in a system. The graphical depiction identifies each source of data and how it interacts with other data sources to reach a common output. Individuals seeking to draft a data flow diagram must

1. Identify external inputs and outputs
2. Determine how the inputs and outputs relate to each other
3. Explain with graphics how these connections relate and what they result in.

The DFD diagram gives the data which is being accessed by various entities. The diagram gives brief information about how the student, college and company is going to be validated. It also gives information about the various roles each entity has.

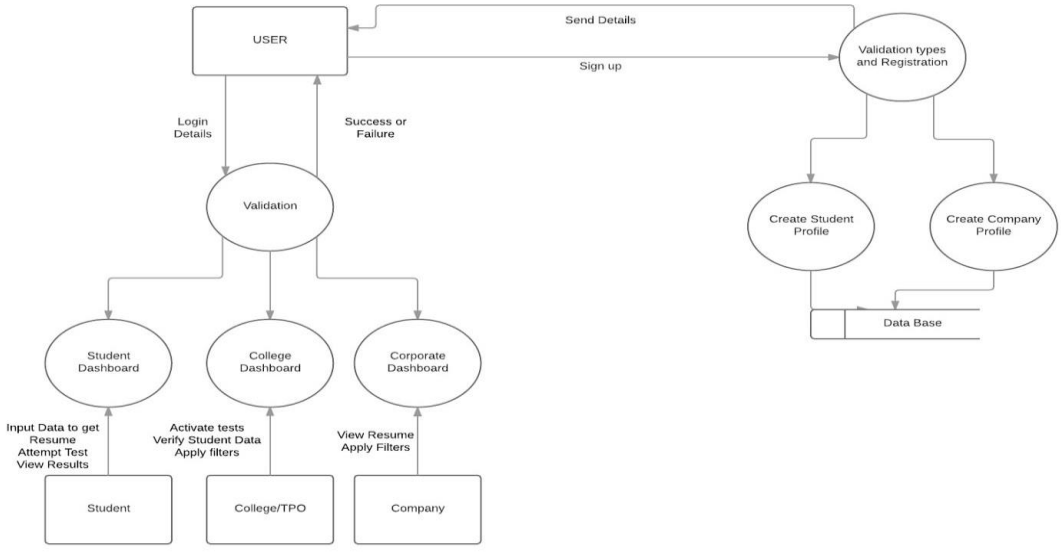


Figure 10-2: DFD Level 1

10.2.2 DFD Level 2

Here we have divided the level 2 diagram in two parts. The first one gives information about the work of the Training and Placement Officers. He will activate test modules and send notification to students. The second diagram gives information about the role of student's work. The student will be attempt tests and a prediction will be made based on the academic scores and aptitude test results.

10.2.3 DFD Level 2 for College End

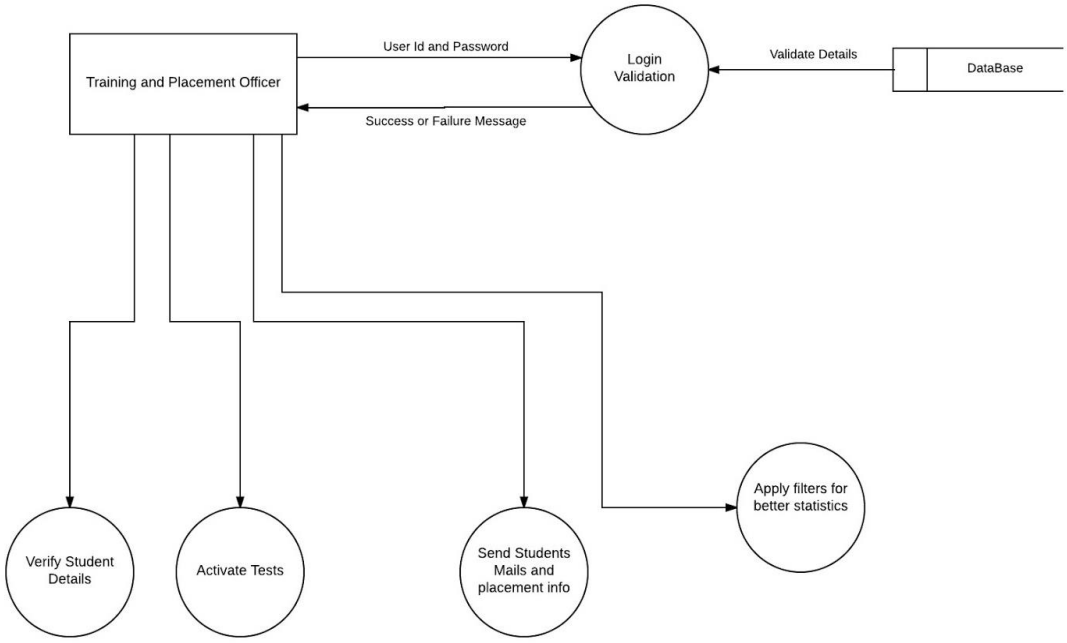


Figure 10-3: DFD Level 2 for TPO

10.2.4 DFD Level 2 for Student

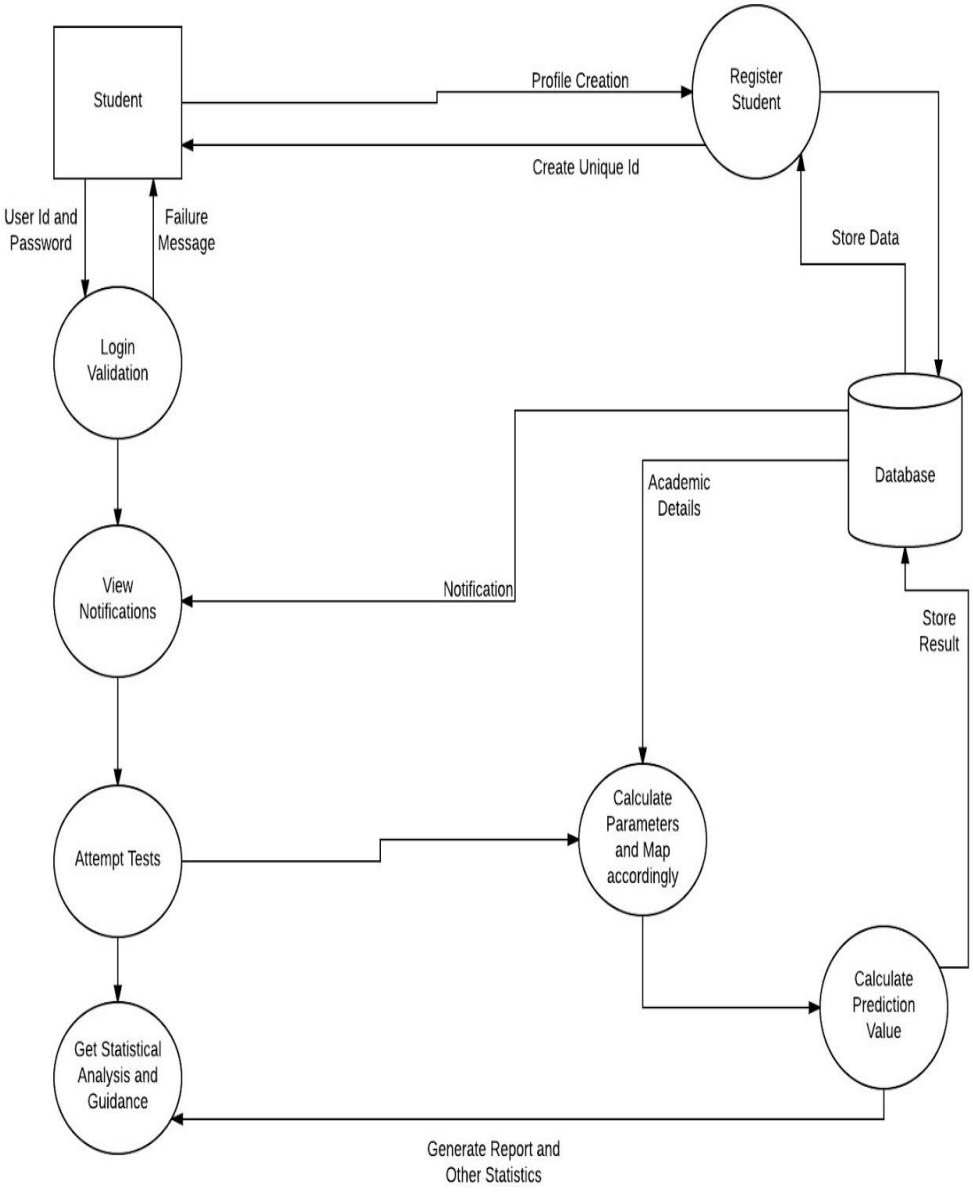


Figure 10-4: DFD Level 2 for Student

10.3 Sequence Diagram

In our sequence diagram, the process with which the student has to go is being mentioned. First the student will register and create a new profile for himself. In the profile, student will enter his academic scores and internship details and other parameters. Then the student will attempt aptitude tests and student will be evaluated. Then a final prediction will be made considering all the parameters. The prediction will tell what are chances of a student getting placed.

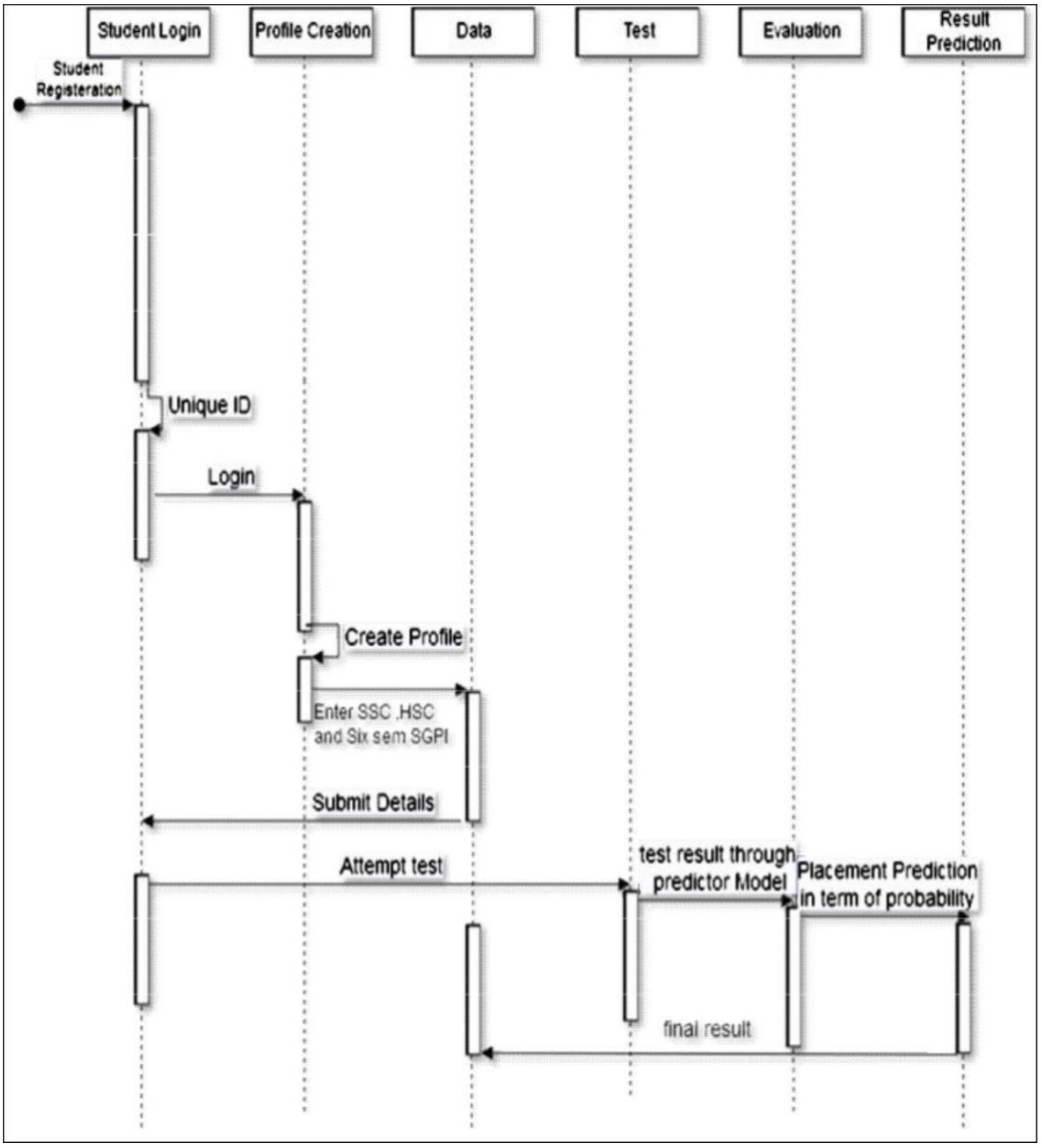


Figure 10-5: Sequence Diagram

10.4 E – R Diagram

There are three main entities: Student, TPO / College, and Company. The students maintain and manage their profile and gets their resume. Students can also get their predicted package range and how to improve profile which depends on the student's profile. TPO / College validates every student's profile and pushes notifications to every student about upcoming companies. TPO also defines series of quizzes that students take in order to improve their profile which also helps in evaluating the student's profile. The companies define placement parameters according to their company policy which helps in filtering students; also, gets questions that can be asked relevant to the student.

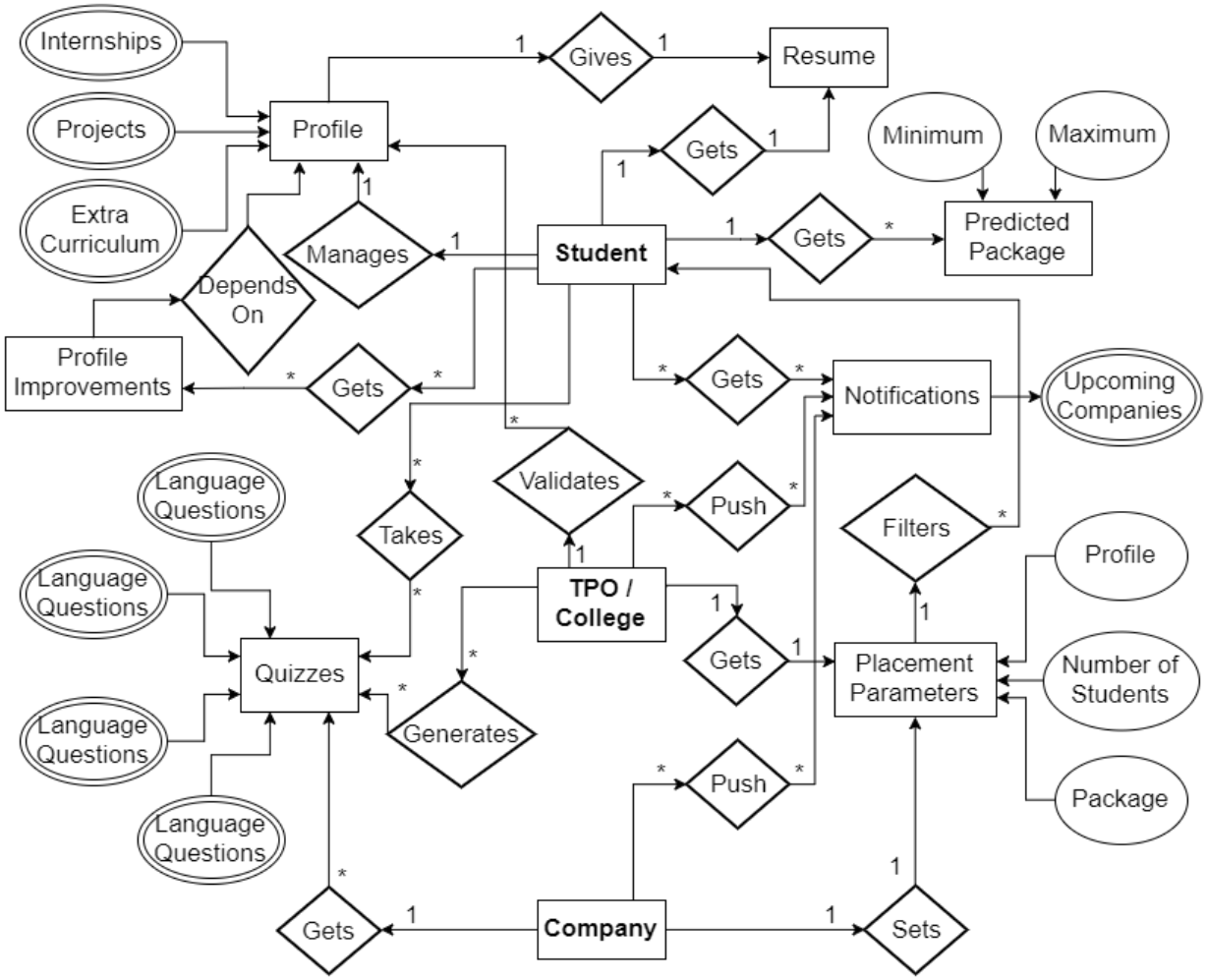


Figure 10-6: E - R Diagram

10.5 Use Case Diagram

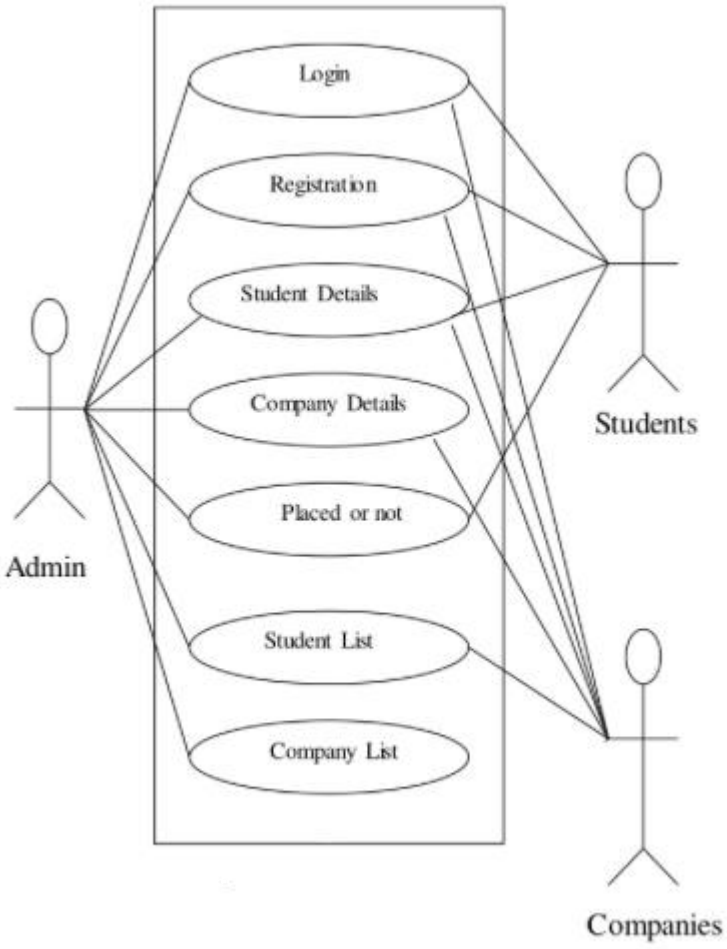


Figure 10-7: Use Case Diagram

10.6 Control Flow Diagram

The control flow diagram starts with the student entity where student will have to register first to use the system. After doing successful login student will get personalise dashboard where he will enter various educational details and download his resume. The user can read, edit and search through this information. Also, he will be able to apply for upcoming placement session through the dashboard. After entering the details, student will give different tests. The system will perform calculations by using this data and store it on a secure server. The prediction of each student will be shown in the form of graphs. This information will be validated by the Training and Placement Office (TPO).

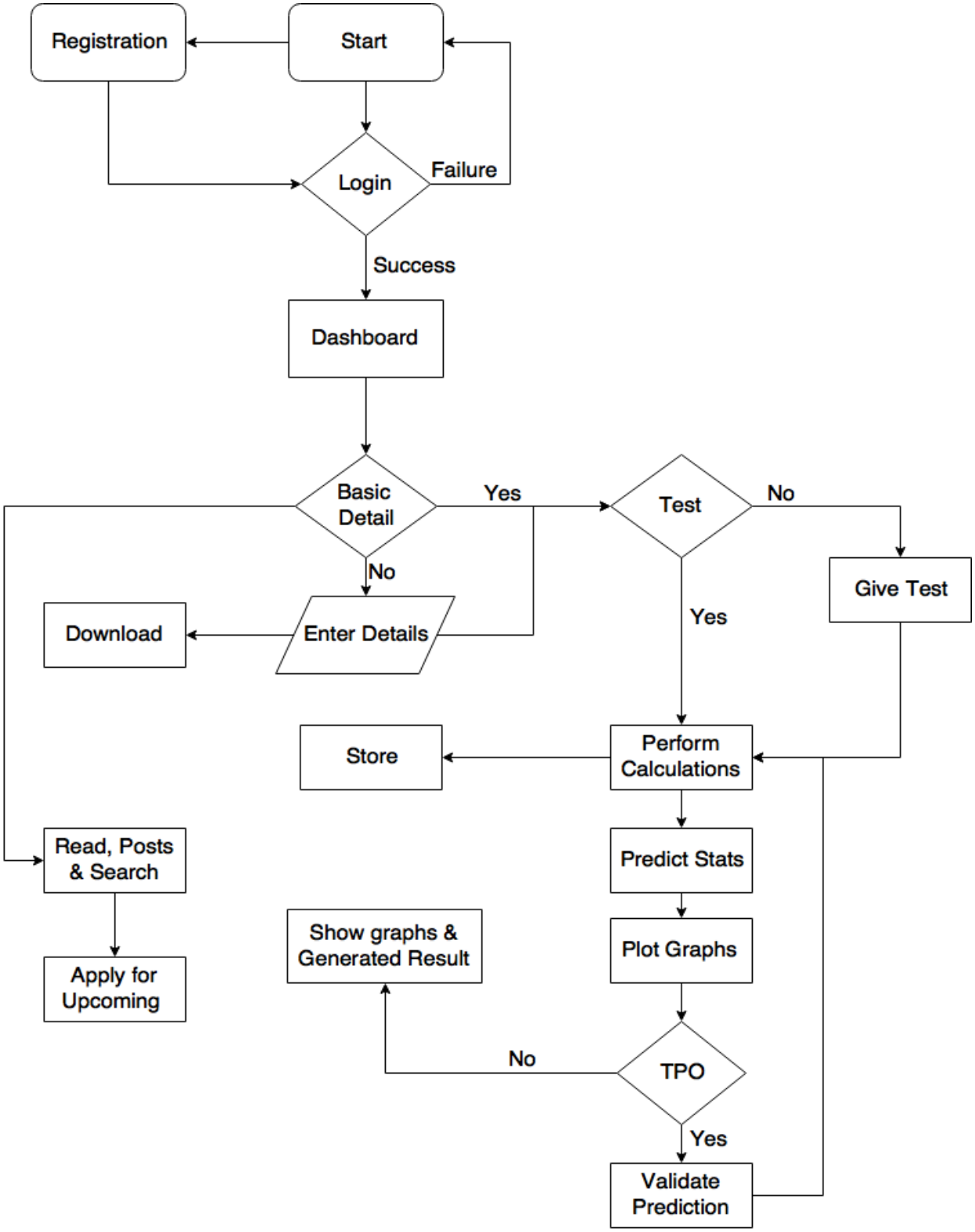


Figure 10-8: Control Flow Diagram

11. Implementation and Experimental Setup

11.1 Operational Requirements

11.1.1 Technical Environment

The system is web based therefore always has a real time database. The database further is used for analysis and prediction. Real time database provides data availability to student as well as the college anytime.

11.1.2 System Integration

The system must be able track all the records of a student through irrespective of different seat number. The data will only be updated after approval by admin since protecting it against data manipulation.

11.1.3 Portability Requirements

The system must work in different environments. It works on different operating systems since it is a web browser-based system.

11.2 Hardware and Software Requirements

11.2.1 Hardware Requirements

1. 1GB of RAM
2. Processor: i3 or Higher
3. Internet Connection: 512 Kb/s or above.
4. Screen Resolution: 1020 x 768 (or above)
5. Disk Storage for Database : 10 GB

11.2.2 Software Requirements

1. Browser (preferably Chrome or Firefox)
2. Operating System : Windows, Linux, OSX.
3. Text Editor : Sublime / Visual Studio Code / Atom
4. Development Environment: SQL, Node.JS, Python 3.0, PostgreSQL, Numpy, Pandas, Scikit-Learn, Plotly.js, Matplotlib, Seaborn

11.3 Dataset and Connectivity

The Models we have trained are based on the previous data of the CMPN branch of Atharva College of Engineering. We have taken the placement data and all the results from the 2015 Passout batch to 2018 Passout batch. The data includes: All the University gazettes from Sem 3 to Sem 8. The gazettes include all the details like:

Internal Assessment marks,

Grades,

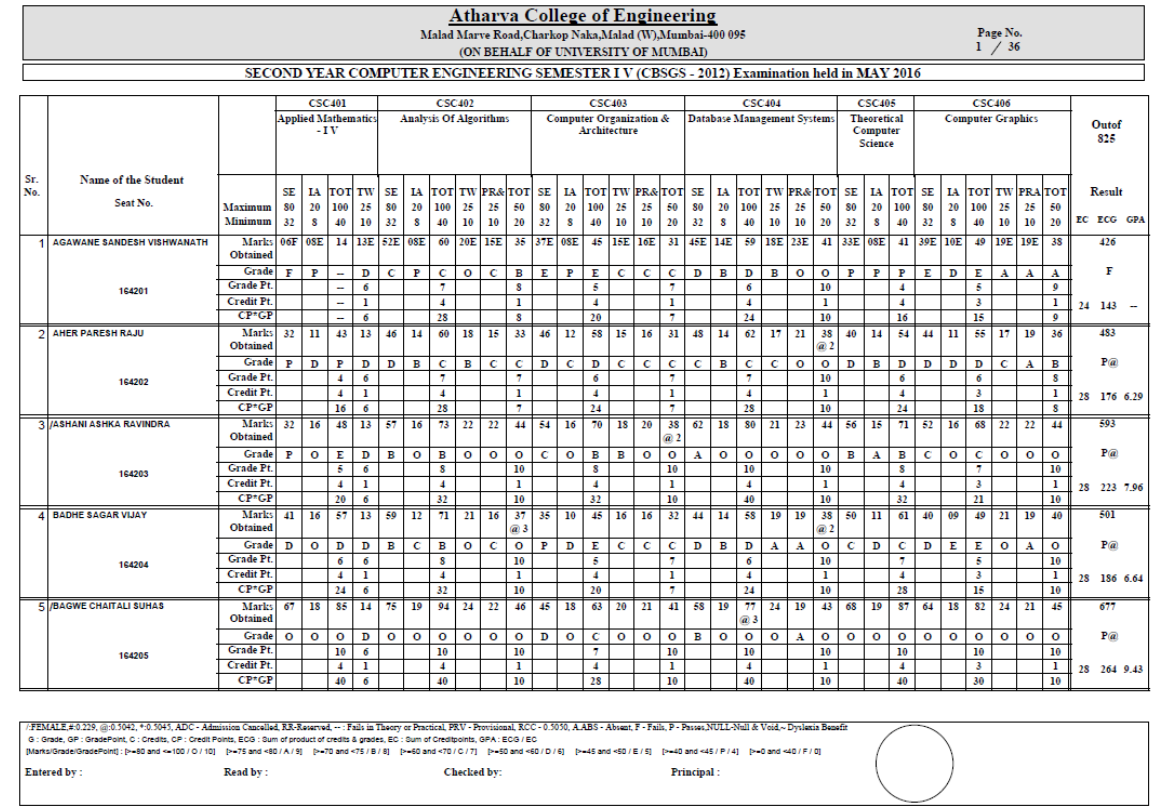
Pointers,

Practical Marks,

Termwork Marks and

Semester Theory Marks.

The Data we worked on was in Pdf format which we converted into excel using python scripts and then imported the excel sheets as pandas dataframe where in we analysed merged and performed transformations upon the data to get insights and make it ready for the model training. The data consisted of only quantitative parameters so we had to infer qualitative parameters based on those quantitative parameters. By keeping seat number and Name as a primary key we merged student records and got one single track record of the student which was basically the entire history of the student in the college. We merged the previous placement results in the same so as to train our model in the right manner, thus creating a complete and reliable centralised database. These processed data were stored in the excel formats which can be easily converted to MS-SQL format. The database is tied with the helper functions using cufflinks so that whenever the data changes the graphs reflect the desired changes. We have added the transformation of data images below.



Atharva College of Engineering

Malad Marve Road,Charkop Naka,Malad (W),Mumbai-400 095

(ON BEHALF OF UNIVERSITY OF MUMBAI)

Page No. 1 / 36

SECOND YEAR COMPUTER ENGINEERING SEMESTER I V (CBSGS - 2012) Examination held in MAY 2016

Sr. No.	Name of the Student / Seat No.		CSC401 Applied Mathematics - I V				CSC402 Analysis Of Algorithms					CSC403 Computer Organization & Architecture					CSC404 Database Management Systems					CSC405 Theoretical Computer Science			CSC406 Computer Graphics					Outof 825		
		Maximum / Minimum	SE 80 32	IA 20 8	TOT 100 40	TW 25 10	SE 80 32	IA 20 8	TOT 100 40	TW 25 10	PR& 25 10	TOT 50 20	SE 80 32	IA 20 8	TOT 100 40	TW 25 10	PR& 25 10	TOT 50 20	SE 80 32	IA 20 8	TOT 100 40	TW 25 10	PR& 25 10	TOT 50 20	SE 80 32	IA 20 8	TOT 100 40	SE 80 32	IA 20 8	TOT 100 40	TW 25 10	PRA 25 10	TOT 50 20	Result EC ECG GPA
1	AGAWANE SANDESH VISHWANATH	Marks Obtained	06F	08E	14	13E	52E	08E	60	20E	15E	35	37E	08E	45	15E	16E	31	45E	14E	59	18E	23E	41	33E	08E	41	39E	10E	49	19E	19E	38	426
		Grade	F	P	--	D	C	P	C	O	C	B	E	P	E	C	C	C	D	B	D	B	O	O	P	P	P	E	D	E	A	A	A	F
	164201	Grade Pt.			--	6			7			8			5			7			6			10			4			5			9	
		Credit Pt.			--	1			4			1			4			1			4			1			4			3			1	24 143 --
		CP*GP			--	6			28			8			20			7			24			10			16			15			9	
2	AHER PARESH RAJU	Marks Obtained	32	11	43	13	46	14	60	18	15	33	46	12	58	15	16	31	48	14	62	17	21	38 @ 2	40	14	54	44	11	55	17	19	36	483
		Grade	P	D	P	D	D	B	C	B	C	C	D	C	D	C	C	C	C	B	C	C	O	O	D	B	D	D	D	D	C	A	B	P@
	164202	Grade Pt.			4	6			7			7			6			7			7			10			6			6			8	
		Credit Pt.			4	1			4			1			4			1			4			1			4			3			1	28 176 6.29
		CP*GP			16	6			28			7			24			7			28			10			24			18			8	
3	/ASHANI ASHKA RAVINDRA	Marks Obtained	32	16	48	13	57	16	73	22	22	44	54	16	70	18	20	38 @ 2	62	18	80	21	23	44	56	15	71	52	16	68	22	22	44	593
		Grade	P	O	E	D	B	O	B	O	O	O	C	O	B	B	O	O	A	O	O	O	O	O	B	A	B	C	O	C	O	O	O	P@
	164203	Grade Pt.			5	6			8			10			8			10			10			10			8			7			10	
		Credit Pt.			4	1			4			1			4			1			4			1			4			3			1	28 223 7.96
		CP*GP			20	6			32			10			32			10			40			10			32			21			10	
4	BADHE SAGAR VIJAY	Marks Obtained	41	16	57	13	59	12	71	21	16	37 @ 3	35	10	45	16	16	32	44	14	58	19	19	38 @ 2	50	11	61	40	09	49	21	19	40	501
		Grade	D	O	D	D	B	C	B	O	C	O	F	D	E	C	C	C	D	B	D	A	A	O	C	D	C	D	E	E	O	A	O	P@
	164204	Grade Pt.			6	6			8			10			5			7			6			10			7			5			10	
		Credit Pt.			4	1			4			1			4			1			4			1			4			3			1	28 186 6.64
		CP*GP			24	6			32			10			20			7			24			10			28			15			10	
5	/BAGWE CHAITALI SUHAS	Marks Obtained	67	18	85	14	75	19	94	24	22	46	45	18	63	20	21	41	58	19	77 @ 3	24	19	43	68	19	87	64	18	82	24	21	45	677
		Grade	O	O	O	D	O	O	O	O	O	O	D	O	C	O	O	O	B	O	O	O	A	O	O	O	O	O	O	O	O	O	O	P@
	164205	Grade Pt.			10	6			10			10			7			10			10			10			10			10			10	
		Credit Pt.			4	1			4			1			4			1			4			1			4			3			1	28 264 9.43
		CP*GP			40	6			40			10			28			10			40			10			40			30			10	

/FEMALE,#:0.239, @:0.5042, *:0.5045, ADC - Admission Cancelled, RR-Reserved, -- : Fails in Theory or Practical, PRV - Provisional, RCC - 0.5050, AABS - Absent, F - Fails, P - Passet,NULL-Null & Void.~ Dyslexia Benefit

G : Grade, GP : GradePoint, C : Credits, CP : Credit Points, ECG : Sum of product of credits & grades, EC : Sum of Creditpoints, GPA : ECG / EC

[Marks/Grade/GradePoint] : [>=80 and <=100 / O / 10] [>=75 and <75 / B / 8] [>=60 and <70 / C / 7] [>=50 and <60 / D / 6] [>=45 and <50 / E / 5] [>=40 and <45 / P / 4] [>=0 and <40 / F / 0]

Entered by : Read by : Checked by: Principal :

Figure 11-1: Result Gazet for Data Cleaning

Atharva Educational Trust's

ATHARVA COLLEGE OF ENGINEERING, MALAD (W)

Approved by AICTE, New Delhi, DTE, Mumbai,

Affiliated to University of Mumbai, ISO Certified- 9001:2008

Department of Computer Engineering

Academic Year 2015-2016

PLACEMENT RECORD FOR THE ACADEMIC YEAR 2015-2016

Sr. No.	NAME OF THE STUDENT	SELECTED COMPANY
1	Abhishek Ravishanka Tripathi	TECH MAHINDRA
2	Aditya Deepak Nandiwdekar	TECH MAHINDRA
3	Akash Arabinda Mukherjee	ACCENTURE
4	Akash Nivrutti Yede	ACCENTURE
5	Akshay Balu Medge	ACCENTURE
6	Akshay Sandeep Mhatre	TECH MAHINDRA
7	Amit Anilkumar Menon	ACCENTURE
8	Amit Yadav Shivprasad Yadav	INFOSYS
9	Amol Shyamsunder Govekar	ACCENTURE
10	Apurva Sudhir Patil	ACCENTURE
11	Arjun Dilipbhai Vekariya	ACCENTURE
12	Ashokchand Hikmatchand Thakur	INFOSYS
13	Darshan Ashok Gangar	TECH MAHINDRA
14	Deepak Mewalal Sharma	ACCENTURE
15	Forum Manish Bhayani	INFOSYS
16	Janvi Girish Shukla	TECH MAHINDRA
17	Jaydeepgiri Vasantgiri Goswami	INFOSYS
18	Jyoti Ghanshyam Daga	ACCENTURE

Figure 11-2: Placement Data for Cleaning

Out[4]:
click to unscroll output; double click to hide

	SEAT NO.	NAME OF THE CANDIDATE	CN	Unnamed: 3	Unnamed: 4	Unnamed: 5	Unnamed: 6	Unnamed: 7	ADS	Unnamed: 9	...	Unnamed: 28	Unnamed: 29	EVS	Unnamed: 31	Unnam
0	NaN	NaN	TH	NaN	TW	NaN	P	NaN	TH	NaN	...	P	NaN	TH	NaN	
1	NaN	MAX	100	NaN	25	NaN	50	NaN	100	NaN	...	25	NaN	50	NaN	
2	NaN	MIN	40	NaN	10	NaN	20	NaN	40	NaN	...	10	NaN	20	NaN	
3	235201	ADMANE PRADEEP VIDNYANRAO	50	NaN	20	NaN	41	NaN	42	NaN	...	19	NaN	28	NaN	
4	235202	AHIRE NAMRATA ASHOK	41	NaN	22	NaN	44	NaN	41	NaN	...	21	NaN	31	NaN	
5	235203	AMIN RISHITA VIJAYKUMAR	44	NaN	23	NaN	41	NaN	50	NaN	...	20	NaN	25	NaN	
6		AMRE PRITAM														

Figure 11-3: Pandas Dataframe used for Cleaning

	CN	CN_TW	CN_Practical	ADS	ADS_TW	ADS_Practical	MP	MP_TW	MP_Practical	TCS	TCS_TW	WE
count	166.000000	169.000000	169.000000	166.000000	169.000000	169.000000	163.000000	169.000000	169.000000	161.000000	169.000000	163.000000
mean	53.722892	21.467456	42.112426	47.795181	20.520710	39.100592	41.220859	21.041420	21.384615	41.888199	21.213018	48.613497
std	13.579229	1.721833	3.050043	8.761844	1.942901	2.953297	12.340351	1.943717	1.349603	16.340897	2.418101	12.625824
min	4.000000	12.000000	35.000000	8.000000	16.000000	18.000000	3.000000	12.000000	16.000000	3.000000	16.000000	13.000000
25%	45.000000	20.000000	40.000000	41.000000	19.000000	38.000000	33.000000	20.000000	20.000000	32.000000	19.000000	40.000000
50%	54.000000	21.000000	42.000000	47.500000	20.000000	39.000000	42.000000	21.000000	22.000000	41.000000	22.000000	50.000000
75%	62.000000	23.000000	44.000000	54.000000	22.000000	40.000000	48.000000	23.000000	22.000000	51.000000	23.000000	58.500000
max	84.000000	24.000000	48.000000	69.000000	23.000000	47.000000	76.000000	24.000000	23.000000	89.000000	24.000000	72.000000

Figure 11-4: Cleaned Dataframe

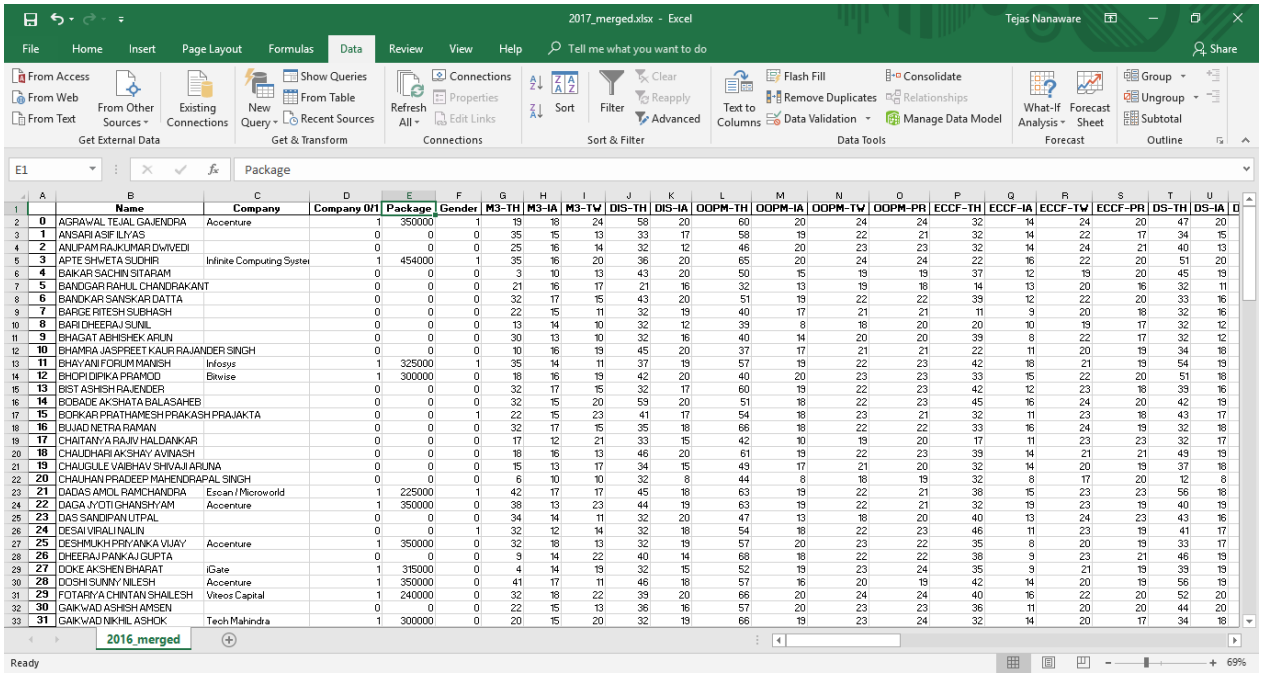


Figure 11-5: Cleaned and Merged Data

11.4 Simulation and Working Environment

For the project to be simulated on a completely new system. The system needs to have python version 3 and all the compatible libraries for the same. The working environment for the project expects:

- The data to be present in the form of gazettes which are in the Mumbai University format.
- The person working with the system should be aware of all the subjects that are there in the Mumbai University Syllabus

11.5 Gantt Chart

Gantt Chart is the timeline which is predefined in the project to make sure things are falling in place and the team has a deadline to match. The following Gantt chart is based on our project keeping the aim of publishing two papers and making the synopsis. We have selected the following time line as it gives us ample amount of time to review for our project and have a clear idea of further proceedings.

1	Analysis Phase
1.1	Study of existing system
1.2	Study of discussion and research papers
1.3.1	Problem definition
1.3.2	Scope
1.3.3	Feasibility
1.4	Defining the problem
1.5	Fixing the scope of the project
1.6	Feasibility analysis
1.7	Requirement analysis
1.8	Project estimation
2	Design Phase
2.1	Developing algorithms of various modules
2.2	Developing data flow diagrams of the system
3	Coding
3.1	Coding algorithm
3.2	Coding module
4	Testing
5	Documentation

Table 11-1: Work Breakdown Structure

11.5.1 TimeLine Chart

Work Task	Wk 1	Wk 2	Wk 3	Wk 4	Wk 5	Wk 6	Wk 7	Wk 8	Wk 9	Wk 10	Wk 11	Wk 12
1.1	■											
1.2		■	■									
1.3.1				■								
1.3.2 1.3.3					■							
1.4						■	■					
1.5								■	■			
1.6										■		
1.7											■	
1.8												■

Table 11-2: Analysis Phase

Work Task	Wk 13	Wk 14	Wk 15	Wk 16	Wk 17	Wk 18	Wk 19	Wk 20	Wk 21	Wk 22	Wk 23	Wk 24
2.1												
2.2												
3												
4												
5												

Table 11-3: Design Phase

12. Testing

12.1 Testing

System testing is a critical phase implementation. Testing of the system involves hardware device implementation and debugging of the computer programs and testing information processing procedures. Testing can be done with text data, which attempts to stimulate all possible conditions that may arise during processing. If structured programming Methodologies have been adopted during coding the testing proceeds from higher level to lower level of program module until the entire program is tested as unit. The testing methods adopted during the testing of the system were unit testing and integrated testing.

12.1.1 Unit Testing

Unit testing focuses first on the modules, independently of one another, to locate errors. This enables the tester to detect errors in coding and logical errors that is contained within that module alone. Those resulting from the interaction between modules are initially avoided. In this methodology of the testing every single element in the project pipeline has been individually tested.

1. Data Cleaning Script - Whether the excel exportation is correct or not.
2. Dimensionality Reduction - Whether the parameters have their co-relations maintained
3. Exploratory Data Analysis - Whether the captured dependancies and graphs are actually efficient
4. Prediction - Whether the prediction is accurate or not

12.1.2 Integration Testing

Integration testing is a systematic technique for constructing the program structure while at the same time to uncover the errors associated with interfacing. The objective is to take unit- tested module and build a program structure that has been detected by designing. It also tests to find the discrepancies between the system and its original objectives. Subordinate stubs are replaced one at time actual module. Tests were conducted at each module was integrated. On completion of each set another stub was replaced with the real module. The entire project pipeline was treated as one and the entire process from insertion of data to training the model was done and the interactions between them were captured.

12.1.3 Functional Testing

The testing of functionalities or individual features in a module is known as functional testing. In this methodology of testing we have incorporated various parameters to be supplied to the modules to check the outcome. Functional testing was done on graph generation and student tracking. Also checking whether the centralised record created was right or not. All the functionalities were tested before integrating them in a pipeline.

12.1.4 Performance Testing

Expected Results:

- The Accuracy was expected to be somewhere between 80 - 85 percent
- The Classification was expected to be the best in Decision Tree.

Observed Results:

- The Accuracy for the best model was 98 percent.
- The Best model for this classification was ID3 Decision Tree

12.1.5 Load Stress Testing

Expected Result:

- Response time will be affected by the number of data points and sample size.
- The introduction of the more number of data points should not break the model.

Observation

- The speed of query execution was fine even when the more students were added

12.2 Test Cases

Test cases are the scenarios where we try to evaluate the performance of the system and test all the modules in different aspects and scenarios. The different models we have tuned for the guidance of placement are all classification models and it is a multi-class classification. These models need to be evaluated based on how they perform when they are exposed to outliers or average data or data from different sigma ranges. There are a set of tests that need to be performed while doing the same.

The test cases for the same that we have considered are as follows:

1. False Positives and False Negatives :

An evaluation of how many unplaced candidates were suggested to be placed and how many placed candidates were predicted to be unplaced. These two scenarios are known as False Positives and False Negatives, where in the negative results are predicted to be positive and the positive results are predicted to be negative.

2. Confusion Matrix :

A 2 x 2 matrix with True Positive, False Positive, True Negative, False Negatives, which helps in the evaluation of the actual accuracy rather than the hoax accuracy

3. Accuracy Paradox :

The numbers mentioned in the accuracy are just numbers it is on how many false negatives and positives are present in the different models, we find the actual accuracy rather than the numeric accuracy

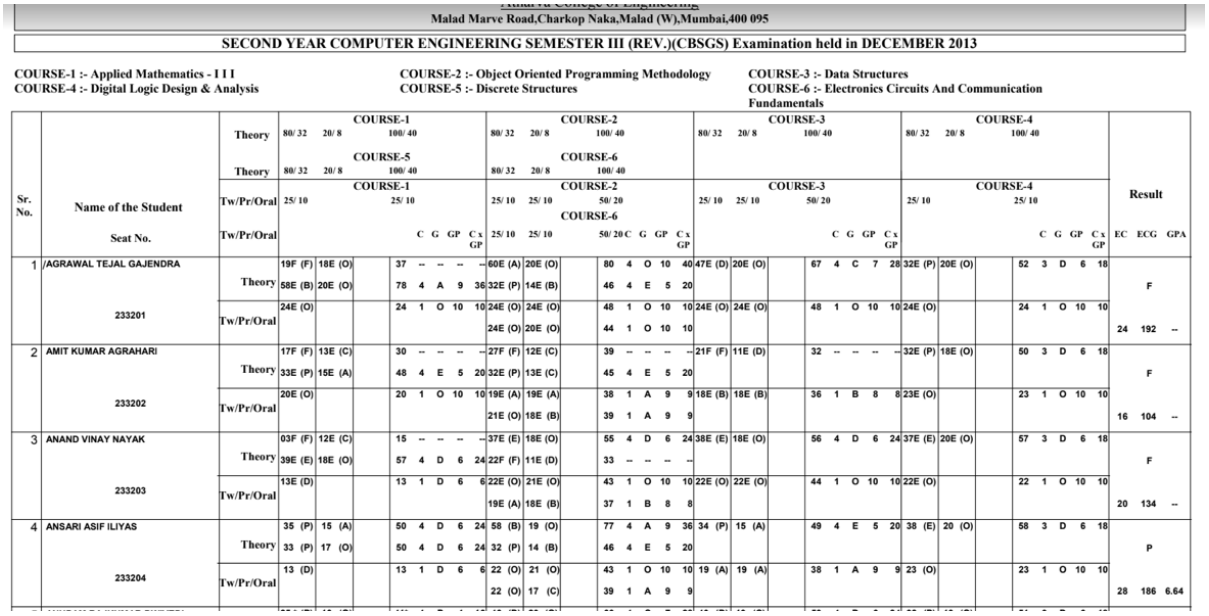
4. CAP Curve and Analysis :

Cumulative Accuracy Profile (CAP) Curve—It is a more robust method to assist our machine model. The idea here is to compare the model to a random scenario and evaluate the results.

13. Result and Analysis

13.1 Data Cleaning Results –

Data cleaning is the process of detecting and correcting inaccurate records from a database and refers to identifying incomplete, incorrect, inaccurate or irrelevant parts of the data and then replacing, modifying, or deleting the dirty or coarse data. Data cleansing may be performed interactively with data wrangling tools, or as batch processing through scripting.



Malad Marve Road,Charkop Naka,Malad (W),Mumbai,400 095

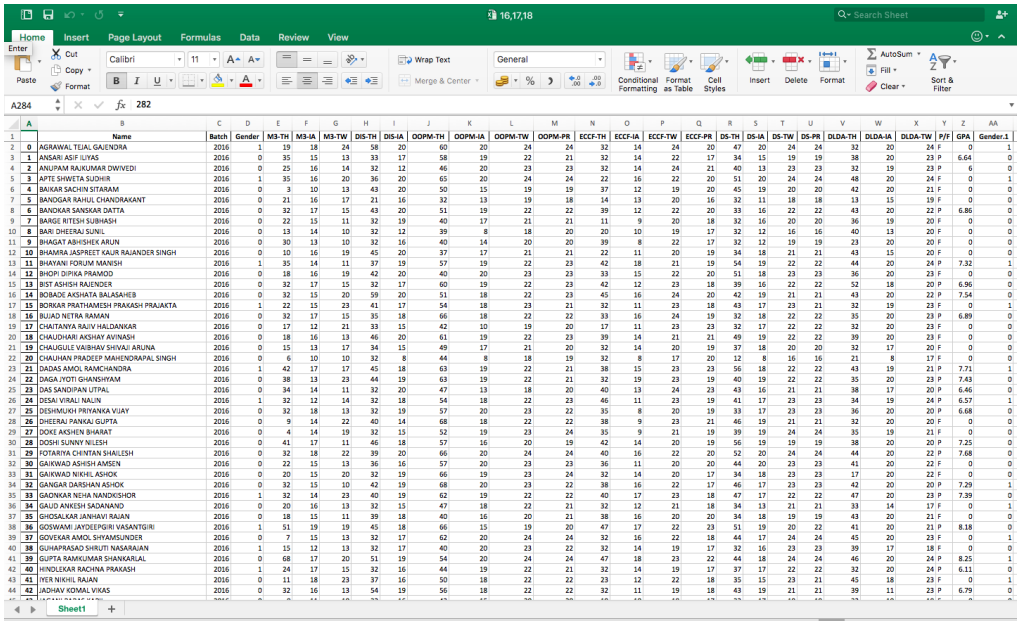
SECOND YEAR COMPUTER ENGINEERING SEMESTER III (REV.)(CBSGS) Examination held in DECEMBER 2013

COURSE-1 :- Applied Mathematics - I I I
COURSE-2 :- Object Oriented Programming Methodology
COURSE-3 :- Data Structures
COURSE-4 :- Digital Logic Design & Analysis
COURSE-5 :- Discrete Structures
COURSE-6 :- Electronics Circuits And Communication Fundamentals

Sr. No.	Name of the Student / Seat No.		COURSE-1	COURSE-2	COURSE-3	COURSE-4	COURSE-5	COURSE-6	Result
		Theory	80/32 20/8 100/40	80/32 20/8 100/40	80/32 20/8 100/40	80/32 20/8 100/40	80/32 20/8 100/40	80/32 20/8 100/40	
		Tw/Pr/Oral	25/10 25/10	25/10 25/10 50/20	25/10 25/10 50/20	25/10 25/10			
			C G GP C x GP	C G GP C x GP	C G GP C x GP	C G GP C x GP			EC ECG GPA
1	/AGRAWAL TEJAL GAJENDRA	Theory	19F (F) 18E (O) 37 -- -- -- -- 58E (B) 20E (O) 78 4 A 9 36	60E (A) 20E (O) 80 4 O 10 40 32E (P) 14E (B) 46 4 E 5 20	47E (D) 20E (O) 67 4 C 7 28	32E (P) 20E (O) 52 3 D 6 18			F
	233201	Tw/Pr/Oral	24E (O) 24 1 O 10 10	24E (O) 24E (O) 48 1 O 10 10 24E (O) 20E (O) 44 1 O 10 10	24E (O) 24E (O) 48 1 O 10 10	24E (O) 24 1 O 10 10			24 192 --
2	AMIT KUMAR AGRAHARI	Theory	17F (F) 13E (C) 30 -- -- -- -- 33E (P) 15E (A) 48 4 E 5 20	27F (F) 12E (C) 39 -- -- -- -- 32E (P) 13E (C) 45 4 E 5 20	21F (F) 11E (D) 32 -- -- -- --	32E (P) 18E (O) 50 3 D 6 18			F
	233202	Tw/Pr/Oral	20E (O) 20 1 O 10 10	19E (A) 19E (A) 38 1 A 9 9 21E (O) 18E (B) 39 1 A 9 9	18E (B) 18E (B) 36 1 B 8 8	23E (O) 23 1 O 10 10			16 104 --
3	ANAND VINAY NAYAK	Theory	03F (F) 12E (C) 15 -- -- -- -- 39E (E) 18E (O) 57 4 D 6 24	37E (E) 18E (O) 55 4 D 6 24 22F (F) 11E (D) 33 -- -- -- --	38E (E) 18E (O) 56 4 D 6 24	37E (E) 20E (O) 57 3 D 6 18			F
	233203	Tw/Pr/Oral	13E (D) 13 1 D 6 6	22E (O) 21E (O) 43 1 O 10 10 19E (A) 18E (B) 37 1 B 8 8	22E (O) 22E (O) 44 1 O 10 10	22E (O) 22 1 O 10 10			20 134 --
4	ANSARI ASIF ILIYAS	Theory	35 (P) 15 (A) 50 4 D 6 24 33 (P) 17 (O) 50 4 D 6 24	58 (B) 19 (O) 77 4 A 9 36 32 (P) 14 (B) 46 4 E 5 20	34 (P) 15 (A) 49 4 E 5 20	38 (E) 20 (O) 58 3 D 6 18			P
	233204	Tw/Pr/Oral	13 (D) 13 1 D 6 6	22 (O) 21 (O) 43 1 O 10 10 22 (O) 17 (C) 39 1 A 9 9	19 (A) 19 (A) 38 1 A 9 9	23 (O) 23 1 O 10 10			28 186 6.64

Figure 13-1: Representation of Data in Initial Format

Here, we used different scripts for data cleaning purposes. After cleansing, a data set was consistent with other similar data sets in the system.



	Name	Batch	Gender	M3-TH	M3-IA	M3-TW	DIS-TH	DIS-IA	OOPM-TH	OOPM-IA	OOPM-TW	OOPM-PR	ECCF-TH	ECCF-IA	ECCF-TW	ECCF-PR	DS-TH	DS-IA	DS-TW	DS-PR	DLDA-TH	DLDA-IA	DLDA-TW	P/F	GPA	Gender.1
0	AGRAWAL TEJAL GAJENDRA	2016	1	19	18	24	58	20	60	20	24	24	32	14	24	20	47	20	24	24	32	20	24	F	0	1
1	ANSARI ASIF ILIYAS	2016	0	35	15	13	33	17	58	19	22	21	32	14	22	17	34	15	19	19	38	20	23	P	6.64	0
2	ANUPAM RAJKUMAR DWIVEDI	2016	0	25	16	14	32	12	46	20	23	23	32	14	24	21	40	13	23	23	32	19	23	P	6	0
3	APTE SHWETA SUDHIR	2016	1	35	16	20	36	20	65	20	24	24	22	16	22	20	51	20	24	24	48	20	24	F	0	1
4	BAIKAR SACHIN SITARAM	2016	0	3	10	13	43	20	50	15	19	19	37	12	19	20	45	19	20	20	42	20	21	F	0	0
5	BANDGAR RAHUL CHANDRAKANT	2016	0	21	16	17	21	16	32	13	19	18	14	13	20	16	32	11	18	18	13	15	19	F	0	0
6	BANDKAR SANSKAR DATTA	2016	0	32	17	15	43	20	51	19	22	22	39	12	22	20	33	16	22	22	43	20	22	P	6.86	0
7	BARGE RITESH SUBHASH	2016	0	22	15	11	32	19	40	17	21	21	11	9	20	18	32	16	20	20	36	19	20	F	0	0
8	BARI DHEERAJ SUNIL	2016	0	13	14	10	32	12	39	8	18	20	20	10	19	17	32	12	16	16	40	13	20	F	0	0
9	BHAGAT ABHISHEK ARUN	2016	0	30	13	10	32	16	40	14	20	20	39	8	22	17	32	12	19	19	23	20	20	F	0	0
10	BHAMRA JASPREET KAUR RAJANDER SINGH	2016	0	10	16	19	45	20	37	17	21	21	22	11	20	19	34	18	21	21	43	15	20	F	0	0
11	BHAYANI FORUM MANISH	2016	1	35	14	11	37	19	57	19	22	23	42	18	21	19	54	19	22	22	44	20	24	P	7.32	1
12	BHOPI DIPIKA PRAMOD	2016	0	18	16	19	42	20	40	20	23	23	33	15	22	20	51	18	23	23	36	20	23	F	0	0
13	BIST ASHISH RAJENDER	2016	0	32	17	15	32	17	60	19	22	23	42	12	23	18	39	16	22	22	52	18	20	P	6.96	0
14	BOBADE AKSHATA BALASAHEB	2016	0	32	15	20	59	20	51	18	22	23	45	16	24	20	42	19	21	21	43	20	22	P	7.54	0
15	BORKAR PRATHAMESH PRAKASH PRAJAKTA	2016	1	22	15	23	41	17	54	18	23	21	32	11	23	18	43	17	23	21	32	19	23	F	0	1
16	BUJAD NETRA RAMAN	2016	0	32	17	15	35	18	66	18	22	22	33	16	24	19	32	18	22	22	35	20	23	P	6.89	0
17	CHAITANYA RAJIV HALDANKAR	2016	0	17	12	21	33	15	42	10	19	20	17	11	23	23	32	17	22	22	32	20	23	F	0	0
18	CHAUDHARI AKSHAY AVINASH	2016	0	18	16	13	46	20	61	19	22	23	39	14	21	21	49	19	22	22	39	20	23	F	0	0
19	CHAUGULE VAIBHAV SHIVAJI ARUNA	2016	0	15	13	17	34	15	49	17	21	20	32	14	20	19	37	18	20	20	32	17	20	F	0	0
20	CHAUHAN PRADEEP MAHENDRAPAL SINGH	2016	0	6	10	10	32	8	44	8	18	19	32	8	17	20	12	8	16	16	21	8	17	F	0	0
21	DADAS AMOL RAMCHANDRA	2016	1	42	17	17	45	18	63	19	22	21	38	15	23	23	56	18	22	22	43	19	21	P	7.71	1
22	DAGA JYOTI GHANSHYAM	2016	0	38	13	23	44	19	63	19	22	21	32	19	23	19	40	19	21	21	43	20	22	P	7.43	0
23	DAS SANDIPAN UTPAL	2016	0	34	14	11	32	20	47	13	18	20	40	13	24	23	43	16	21	21	38	17	20	P	6.46	0
24	DESAI VIRALI NALIN	2016	1	32	12	14	32	18	54	18	22	23	46	11	23	19	41	17	23	23	34	19	24	P	6.57	1
25	DESHMUKH PRIYANKA VIJAY	2016	0	32	18	13	32	19	57	20	23	22	35	8	20	19	33	17	23	23	36	20	20	P	6.68	0
26	DHEERAJ PANKAJ GUPTA	2016	0	9	14	22	40	14	68	18	22	22	38	9	23	21	46	19	21	21	32	20	20	F	0	0
27	DOKE AKSHEN BHARAT	2016	0	4	14	19	32	15	52	19	23	24	35	9	21	19	39	19	24	24	35	19	21	F	0	0
28	DOSHI SUNNY NILESH	2016	0	41	17	11	46	18	57	16	20	19	42	14	20	19	56	19	19	19	38	20	20	P	7.25	0
29	FOTARIYA CHINTAN SHAILESH	2016	0	32	18	22	39	20	66	20	24	24	40	16	22	20	52	20	24	24	44	20	22	P	7.68	0
30	GAIKWAD ASHISH AMSEN	2016	0	22	15	13	36	16	57	20	23	23	36	11	20	20	44	20	23	23	41	20	22	F	0	0
31	GAIKWAD NIKHIL ASHOK	2016	0	20	15	20	32	19	66	19	23	24	32	14	20	17	34	18	23	23	17	20	22	F	0	0
32	GANGAR DARSHAN ASHOK	2016	0	32	15	10	42	19	68	20	23	22	38	16	22	17	46	17	23	23	42	20	20	P	7.29	1
33	GAONKAR NEHA NANDKISHOR	2016	1	32	14	23	40	19	62	19	22	22	40	17	23	18	47	17	22	22	47	20	23	P	7.39	0
34	GAUD ANKESH SADANAND	2016	0	20	16	13	32	15	47	18	22	21	32	12	21	18	34	13	21	21	33	14	17	F	0	1
35	GHOSALKAR JANHAVI RAJAN	2016	0	18	15	11	39	18	40	16	20	21	38	16	20	20	34	18	19	19	43	20	21	F	0	0
36	GOSWAMI JAYDEEPGIRI VASANTGIRI	2016	1	51	19	19	45	18	66	15	19	20	47	17	22	23	51	19	20	22	41	20	21	P	8.18	0
37	GOVEKAR AMOL SHYAMSUNDER	2016	0	7	15	13	32	17	62	20	24	24	32	16	22	18	44	17	24	24	45	20	23	F	0	1
38	GUNAPRASAD SHRUTI NASARAJAN	2016	1	15	12	13	32	17	40	20	23	22	32	14	19	17	32	16	23	23	39	17	18	F	0	0
39	GUPTA RAMKUMAR SHANKARLAL	2016	0	68	17	20	51	19	54	20	24	24	47	18	23	22	44	18	24	24	46	20	24	P	8.25	1
40	HINDLEKAR RACHNA PRAKASH	2016	1	24	17	15	32	16	44	19	22	21	32	14	19	17	37	17	22	22	32	20	24	P	6.11	0
41	IYER NIKHIL RAJAN	2016	0	11	18	23	37	16	50	18	22	22	23	12	22	18	35	15	23	21	45	18	23	F	0	1
42	JADHAV KOMAL VIKAS	2016	0	32	16	13	54	19	56	18	22	22	32	11	19	18	43	19	21	21	39	11	23	P	6.79	0

Figure 13-2: Merged Data

13.2 Exploratory Data Analysis Results

Exploratory data analysis (EDA) is an approach to analyzing data sets to summarize their main characteristics, often with visual methods. Numerous data visualization techniques are used here to find meaningful insights in the dataset.

```
In [8]: g = sns.FacetGrid(sem_3_2015_merge, col='Remarks')
        g.map(plt.hist, 'Company', bins=5)

Out[8]: <seaborn.axisgrid.FacetGrid at 0x11bcedb50>
```

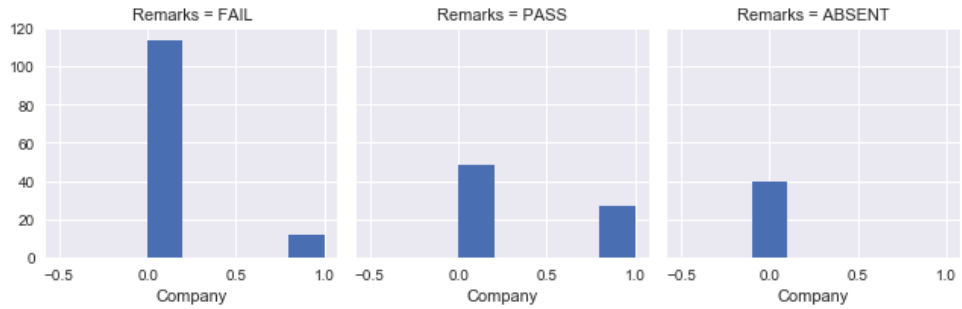


Figure 13-3: Comparison of Fail/Pass Students With Placement

From the graph we can conclude that unplaced student contains high percentage of FAIL students. This validates our assumption that failing of a student in certain exam has direct impact on his placement.

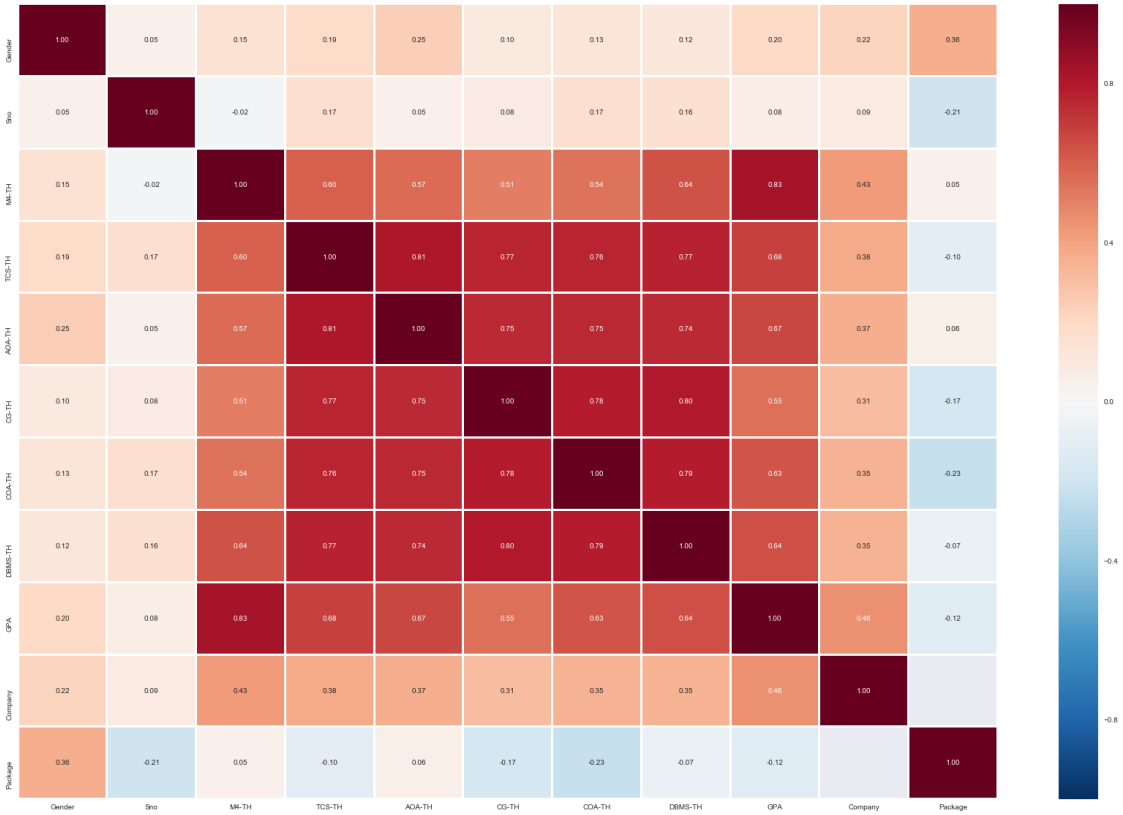


Figure 13-4: Correlation of Each Subject With One Another

The above correlation graph provides insights about importance of each subject from placement perspective. Insights like these proved to be useful in selection of features for model prediction.

13.3 Student Tracking

Data visualization proves to be useful for tracking of different students. Using cleaned data, it becomes easier to track progress of each student over each semester. Understanding of the data becomes easier when it is represented in the form of graphs.

```
In [25]: df2018_Theory = df2018_copy.filter(regex='TH')
         rowx = df2018_Theory[df2018_Theory.index == "JHA ABHISHEK SHASHINATH"]
         rowx.iloc[0].iplot(kind="bar")
```

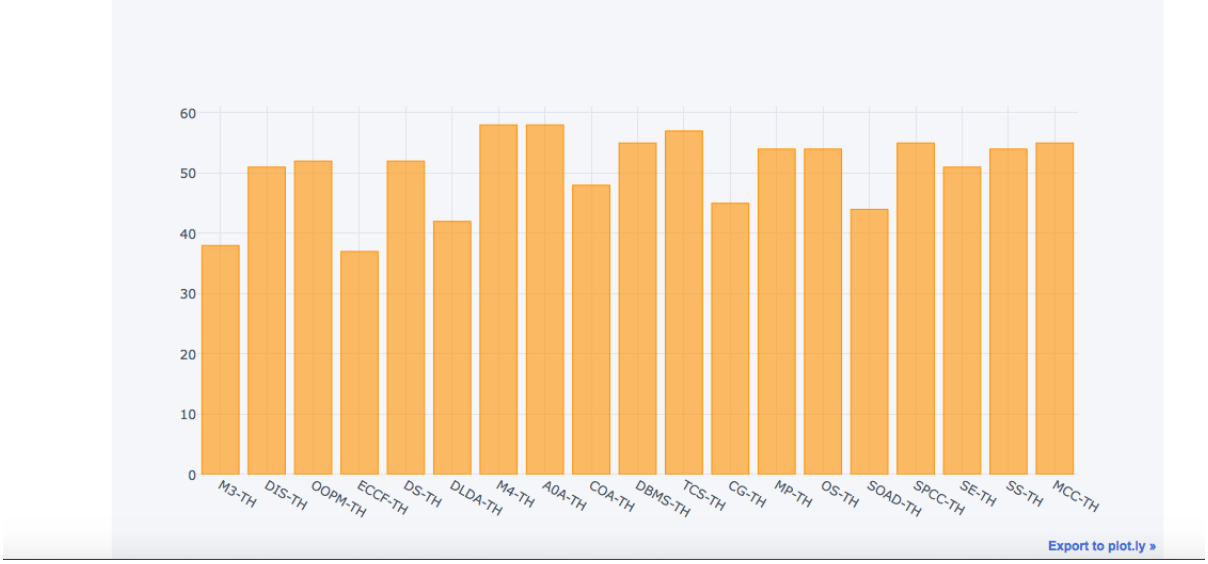


Figure 13-5: Subject Wise Marks of A Student

```
In [23]: df2018_GPA = df2018_copy.filter(regex='GPA')
         df2018_GPA[df2018_GPA.index == "JHA ABHISHEK SHASHINATH"].iplot(kind="bar")
```

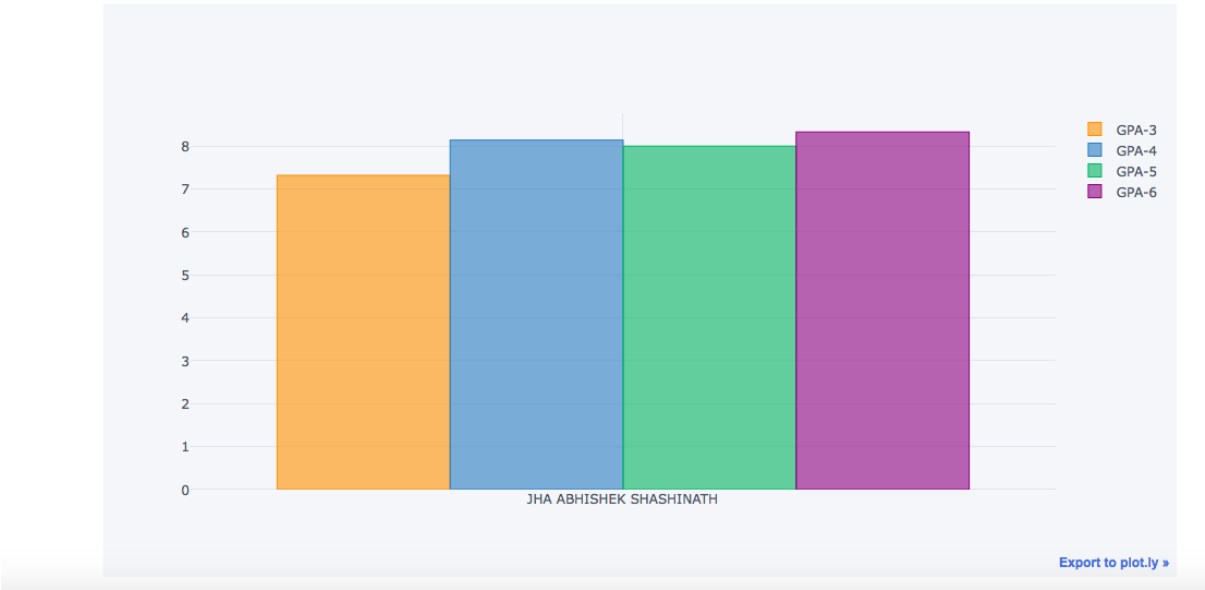


Figure 13-6: Semester Wise GPA Progression

13.4 Models and Their Accuracy Checking

Availability of numerous models for prediction makes it difficult to select appropriate model for your system. To overcome the challenge we trained our dataset on different models to check their accuracy and suitability for our system.

13.4.1 Logistic Regression –

```
In [6]: # Logistic Regression

        logreg = LogisticRegression()
        logreg.fit(X_train, Y_train)
        Y_pred = logreg.predict(X_test)
        acc_log = round(logreg.score(X_train, Y_train) * 100, 2)
        acc_log

Out[6]: 66.67
```

Figure 13-7: Logistic Regression

Logistic Regression provided accuracy of 66.67% after training it on our dataset. This accuracy is way lower for any system.

13.4.2 k- Nearest Neighbor (KNN) -

```
In [8]: # KNN

        knn = KNeighborsClassifier(n_neighbors = 281)
        knn.fit(X_train, Y_train)
        Y_pred2 = knn.predict(X_test)
        acc_knn = round(knn.score(X_train, Y_train) * 100, 2)
        acc_knn

Out[8]: 52.48
```

Figure 13-8: k - Nearest Neighbor

KNN provided accuracy lesser than the logistic regression. As the number of n_neighbors increased this accuracy showed further reduction in percentage.

13.4.3 Decision Tree –

```
In [9]: # Decision Tree

        decision_tree = DecisionTreeClassifier()
        decision_tree.fit(X_train, Y_train)
        Y_pred3 = decision_tree.predict(X_test)
        acc_decision_tree = round(decision_tree.score(X_train, Y_train) * 100, 2)
        acc_decision_tree

Out[9]: 100.0
```

Figure 13-9: Decision Tree

Decision Tree proved to overfit the dataset with its 100% accuracy.

13.4.4 Support Vector Machine -

```
In [7]: # Support Vector Machine

        svc = SVC()
        svc.fit(X_train, Y_train)
        Y_pred1 = svc.predict(X_test)
        acc_svc = round(svc.score(X_train, Y_train) * 100, 2)
        acc_svc

Out[7]: 99.29
```

Figure 13-10: Support Vector Machine

Again, usage of Support Vector Machine proved to over fit the dataset.

Overfitting of data provides a great accuracy but in the long run proves to be inefficient.

14. Advantages and Limitations

14.1 Advantages

1. Models have been trained on real dataset
2. To solve the issue for training models a centralised database was created
3. Result analysis easier
4. College can get easy insights and generate graphs easier.
5. By training multiple models we have identified the pros and cons of all the algorithms giving the freedom to find the best possible algorithm and leverage the best parts of all the algorithms.
6. The model being trained on actual student data captures the anomalies in real scenario and adapts to it over time.
7. Classification problem can be converted to a regression problem by predicting salaries instead of whether being placed or not thus gaining best of both worlds.

- Some models have accuracy as high as 99 percent.

14.2 Limitations

1. The data on which the model was trained on was of the previous few years and hence there were limitations that had occurred while training the model.
2. The project for now doesn't have a front end to be accessed from, and adding the front end will give the students to access insights and basically the scripts will come to life
3. The problem that we have treated is primarily a classification and there are better ways to convert this problem into a regression problem and solve it efficiently.

15. Applications and Future Scope

15.1 Applications

This project can find its applications in various places:

1. For students who want to look at their overall growth and understand what their strengths and weaknesses are
2. For students to get a fair idea about the subjects they are going to face and how difficult will it be
3. The college can find the insights of all the population and see the growth of individual people as well.
4. Focus on the curriculum and see the changes that occur.
5. Train students as per the latest trends in technology
6. The company can have a look at this insight for the recruitment procedure as well.
7. Whenever there is an educational leap, this project can be turned into an application.

15.2 Future Scope

1. When the research work and the different models combined together and given a web application to be accessed the project will be of use to all the people.
2. The continual additions of data will cause the models to work more efficiently but fine tuning of the models with more qualitative parameters is necessary.
3. The models are exposed to the risk of overfitting in the future and hence the parameters can change their co relations and hence retraining the models is important.
4. A complete application where in a centralised database and the internship data of the student can be kept for mining processes to better understand the students and provide valuable guidance to students.
5. The project is open for using the models that can be available to the mankind in near future (Capsule Networks).

16. Conclusion

As we have seen throughout our studies, that the problem statements we have approached are student, college, and corporate centric. The solution to all of these problem statements, is based on the model we are going to build, the output of which will be a number between 0-1, which will determine, the prediction of a student being placed. During this process, a lot of other dependent variables will be predicted which will help solve the problem statements.
The expected outputs of the system for student end, is the prediction about their placement, and the statistics of how they can fair well. College end will have the analysis of every student, and will have the opportunity to focus more on the improvement of students. Also because of the system, the college will have one platform to manage the data of the students, thus solving another issue. The corporates will be able to apply filters, compare students, and download resume of the students they're interested in, also they will get student related questions that they can ask, in the interview.

17. Acknowledgement

We would like to thank our project guide Prof. Mahendra Patil for his enormous cooperation and guidance. We have no words to express our gratitude for a person who wholeheartedly supported the project and gave freely of his valuable time while making this project. All the inputs given by him found a place in the project. The technical guidance provided by him was more than useful and made this a success. He has always been a source of inspiration for us. It was a great experience learning under such a highly innovative, enthusiastic and hard working professor.

We are also thankful to our Principal Dr. S.P. Kallurkar, and all the staff members of the Computer Department who have provided us various facilities and guided us throughout to develop this project idea.

We also thank Dr. Sameer Sahasrabuddhe for his expert guidance and approval for the project under IIT Bombay.

Finally, we would like to thank teachers of the college and friends who guided and helped us while we worked on this project.

18. References

1. Use of ID3 Decision Tree algorithm for Placement Prediction.

   Bhatt, H., Mehta, S., & D'mello, L. R. (2015). Use of ID3 Decision Tree Algorithm for Placement Prediction. *International Journal of Computer Science and Information Technologies*, 4785-4789.

2. Prediction of Final Result and Placement of Students using Classification Algorithm

   Naik, N., & Purohit, S. (2012). Prediction of Final Result and Placement of Students using Classification Algorithm. *International Journal of Computer Applications*, *56*(12).

3. A Placement prediction system using K-Nearest Neighbors Classifier

   Giri, A., Bhagavath, M. V. V., Pruthvi, B., & Dubey, N. (2016, August). A Placement Prediction System using k-nearest neighbors classifier. In *Cognitive Computing and Information Processing (CCIP), 2016 Second International Conference on* (pp. 1-4). IEEE.

4. Student Placement Prediction Using ID3 Algorithm.

   Namita Puri, Deepali Khot, Pratiksha Shinde, Kishori Bhoite, , , , ."Student Placement Prediction Using ID3 Algorithm", Volume 3, Issue III, International Journal for Research in Applied Science and Engineering Technology (IJRASET) Page No: , ISSN : 2321-9653, www.ijraset.com.

5. PPS - Placement Prediction System using Logistic Regression.

   Sharma, A. S., Prince, S., Kapoor, S., & Kumar, K. (2014, December). PPS—Placement prediction system using logistic regression. In *MOOC, Innovation and Technology in Education (MITE), 2014 IEEE International Conference on*(pp. 337-341). IEEE.

6. An Empirical Analysis of Classification Techniques for Predicting Academic Performance

   S.Taruna , Mrinal Pandey ,"An Empirical Analysis of Classification Techniques for Predicting Academic Performance" in 2014 IEEE International Advance Computing Conference (IACC).

7. Predicting students marks in hellenic open university

   Kotsiantis, Sotiris B., and Panayiotis E. Pintelas, "Predicting students marks in hellenic open university", in Advanced Learning Technologies, 2005. ICALT 2005. Fifth IEEE International Conference on, pp. 664-668. IEEE, 2005.

8. Applying logistic regression model to the examination results data

   Saha, Goutam, "Applying logistic regression model to the examination results data.,in Journal of Reliability and Statistical Studies 4, no.2(2011):1-13.

9. Improving accuracy of students' final grade prediction model using optimal equal width binning and synthetic minority over-sampling technique

Syed Tanveer Jishan, Raisul Islam Rashu, Naheena Haque and Rashedur M Rahman, "Improving accuracy of students' final grade prediction model using optimal equal width binning and synthetic minority over-sampling technique," in Decision Analytics (2015) 2:1 DOI 10.1186/s40165-014-0010-2(Springer Journel).

10. Prediction and Analysis for Students' Marks Based on Decision Tree Algorithm

Zhiwu Liu and Xiuzhi Zhang, "Prediction and Analysis for Students' Marks Based on Decision Tree Algorithm" in 2010 Third International Conference on Intelligent Networks and Intelligent Systems.

11. Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data

Carlos Márquez-Vera, Alberto Cano, Cristóbal Romero, Sebastián Ventura, "Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data" in Appl Intell (2013) 38:315–330 DOI 10.1007/s10489-012-0374-8.

12. Predicting student performance by using data mining methods for classification

Kabakchieva, Dorina. "Predicting student performance by using data mining methods for classification", in Cybernetics and Information Technologies 13, no. 1 (2013): 61-72.

13. Predicting Student' Performance using ID3 and C4.5 Classification Algorithms

Kalpesh Adhatrao, Aditya Gaykar, Amiraj Dhawan, Rohit Jha ,Vipul Honrao, "Predicting Student' Performance using ID3 and C4.5 Classification Algorithms", in International Journal of Data Mining & Knowledge Management Process (IJDKP) Vol.3, No.5, September 2013.

14. Predicting Student Performance using Artificial Neural Network Analysis

Van Heerden et. al designed a system for placement prediction in the University of Pretoria Medical School, using artificial neural networks, which had 99 input parameters out of which 80 parameters were qualitative. They tested their system on some students and found out that for those students where all the parameters were available the prediction of the system was almost 100% accurate, and a 90% accuracy was obtained when only qualitative parameters were used

19. Appendix

19.1 Concepts Related to Neural Networks

19.1.1 What are Neural Networks made of?

A typical neural network has anything from a few dozen to hundreds, thousands, or even millions of artificial neurons called units arranged in a series of layers, each of which connects to the layers on either side. Some of them, known as input units, are designed to receive various forms of information from the outside world that the network will attempt to learn about, recognize, or otherwise process. Other units sit on the opposite side of the network and signal how it responds to the information it's learned; those are known as output units. In between the input units and output units are one or more layers of hidden units, which, together, form the majority of the artificial brain. Most neural networks are fully connected, which means each hidden unit and each output unit is connected to every unit in the layers either side. The connections between one unit and another are represented by a number called a weight, which can be either positive (if one unit excites another) or negative (if one unit suppresses or inhibits another). The higher the weight, the more influence one unit has on another. (This corresponds to the way actual brain cells trigger one another across tiny gaps called synapses.)

19.1.2 How does a Neural Network learn?

Information flows through a neural network in two ways. When it's learning (being trained) or operating normally (after being trained), patterns of information are fed into the network via the input units, which trigger the layers of hidden units, and these in turn arrive at the output units. This common design is called a feedforward network. Not all units "fire" all the time. Each unit receives inputs from the units to its left, and the inputs are multiplied by the weights of the connections they travel along. Every unit adds up all the inputs it receives in this way and (in the simplest type of network) if the sum is more than a certain threshold value, the unit "fires" and triggers the units it's connected to (those on its right).

For a neural network to learn, there has to be an element of feedback involved—just as children learn by being told what they're doing right or wrong. In fact, we all use feedback, all the time. Think back to when you first learned to play a game like ten-pin bowling. As you picked up the heavy ball and rolled it down the alley, your brain watched how quickly the ball moved and the

line it followed, and noted how close you came to knocking down the skittles. Next time it was your turn, you remembered what you'd done wrong before, modified your movements accordingly, and hopefully threw the ball a bit better. So you used feedback to compare the outcome you wanted with what actually happened, figured out the difference between the two, and used that to change what you did next time ("I need to throw it harder," "I need to roll slightly more to the left," "I need to let go later," and so on). The bigger the difference between the intended and actual outcome, the more radically you would have altered your moves.

Neural networks learn things in exactly the same way, typically by a feedback process called backpropagation (sometimes abbreviated as "backprop"). This involves comparing the output a network produces with the output it was meant to produce, and using the difference between them to modify the weights of the connections between the units in the network, working from the output units through the hidden units to the input units—going backward, in other words. In time, back propagation causes the network to learn, reducing the difference between actual and intended output to the point where the two exactly coincide, so the network figures things out exactly as it should

19.1.3 How does Neural Network work in Practice?

Once the network has been trained with enough learning examples, it reaches a point where you can present it with an entirely new set of inputs it's never seen before and see how it responds. For example, suppose you've been teaching a network by showing it lots of pictures of chairs and tables, represented in some appropriate way it can understand, and telling it whether each one is a chair or a table. After showing it, let's say, 25 different chairs and 25 different tables, you feed it a picture of some new design it's not encountered before—let's say a chaise longue—and see what happens. Depending on how you've trained it, it'll attempt to categorize the new example as either a chair or a table, generalizing on the basis of its past experience—just like a human. That doesn't mean to say a neural network can just "look" at pieces of furniture and instantly respond to them in meaningful ways; it's not behaving like a person. Consider the example we've just given: the network is not actually looking at pieces of furniture. The inputs to a network are essentially binary numbers: each input unit is either switched on or switched off. So if you had five input units, you could feed in information about five different characteristics of different chairs using binary (yes/no) answers. The questions might be 1) Does it have a back? 2) Does it have a top? 3) Does it have soft upholstery? 4) Can you sit on it comfortably for long periods of time? 5) Can you put lots of things on top of it? A typical chair would then present as Yes, No, Yes, Yes, No or 10110 in

binary, while a typical table might be No, Yes, No, No, Yes or 01001. So, during the learning phase, the network is simply looking at lots of numbers like 10110 and 01001 and learning that some mean chair (which might be an output of 1) while others mean table (an output of 0).

19.1.4 What is Classification in Machine Learning?

In machine learning and statistics, classification is the problem of identifying to which of a set of categories (sub-populations) a new observation belongs, on the basis of a training set of data containing observations (or instances) whose category membership is known. An example would be assigning a given email into spam non-spam classes or assigning a diagnosis to a given patient as described by observed characteristics of the patient (gender, blood pressure, presence or absence of certain symptoms, etc.). Classification is an example of pattern recognition.

19.1.5 List of Common Machine Learning Algorithms

1. Linear Regression
2. Logistic Regression
3. Decision Tree
4. SVM
5. Naive Bayes
6. K - NN
7. K - Means
8. Random Forest
9. Dimensionality Reduction Algorithms
10. Gradient Boosting algorithms
11. GBM
12. XGBoost
13. LightGBM
14. CatBoost

20. Paper Publication Details

Title: Improving Predictions using Qualitative Parameters

Conference Name (Detail): International Journal of Research Publications in Engineering and Technology.

Date of Publication: August 2017

Journal Name: International Journal of Research Publications in Engineering and Technology.

Name of Authors: Prashant Mahajan[#1], Pratik Deshpande[#2], Tejas Nanaware[#3], Prof. Mahendra Patil[#4].

Affiliation of Authors:

#Department of Computer Engineering,

Atharva College of Engineering,

Mumbai, India

Title: Exploratory Data Analysis using Dimensionality Reduction

Conference Name (Detail) : International Conference on Innovative and Advanced Technologies in Engineering

Date of Publication: March 2018

Journal Name: IOSR

Name of Authors: Prashant Mahajan[#1], Pratik Deshpande[#2], Tejas Nanaware[#3], Ravi Chandak[#4] Prof. Mahendra Patil[#5].

Affiliation of Authors:

#Department of Computer Engineering,

Atharva College of Engineering,

Mumbai, India