International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 9 (2019) pp. 2188-2191

Campus Placement Prediction Using Supervised Machine Learning Techniques

Pothuganti Manvitha

Department of Information Technology
Sreenidhi Institute of Science and Technology

Neelam Swaroopa

Department of Information Technology
Sreenidhi Institute of Science and Technology

Abstract:

Placement of students is one of the most important objective of an educational institution. Reputation and yearly admissions of an institution invariably depend on the placements it provides it students with. That is why all the institutions, arduously, strive to strengthen their placement department so as to improve their institution on a whole. Any assistance in this particular area will have a positive impact on an institution's ability to place its students. This will always be helpful to both the students, as well as the institution. In this study, the objective is to analyse previous year's student's data and use it to predict the placement chance of the current students. This model is proposed with an algorithm to predict the same. Data pertaining to the study were collected form the same institution for which the placement prediction is done, and also suitable data pre-processing methods were applied. This proposed model is also compared with other traditional classification algorithms such as Decision tree and Random forest with respect to accuracy, precision and recall. From the results obtained it is found that the proposed algorithm performs significantly better in comparison with the other algorithms mentioned.

Keywords: Classification, Decision tree, Random forest

1. INTRODUCTION

Placements are considered to be very important for each and every college. The basic success of the college is measured by the campus placement of the students. Every student takes admission to the colleges by seeing the percentage of placements in the college. Hence, in this regard the approach is about the prediction and analyses for the placement necessity in the colleges that helps to build the colleges as well as students to improve their placements [1].

In Placement Prediction system predicts the probability of a undergrad students getting placed in a company by applying classification algorithms such as Decision tree and Random forest. The main objective of this model is to predict whether the student he/she gets placed or not in campus recruitment. For this the data consider is the academic history of student like overall percentage, backlogs, credits. The algorithms are applied on the previous years data of the students.

2. RELATED WORKS

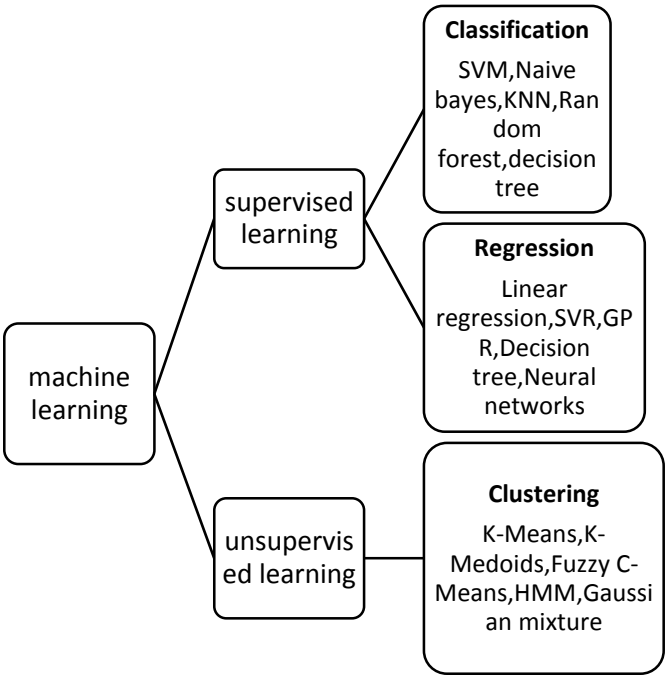

Figure 1. Machine learning algorithms

From the above mentioned machine learning models Supervised learning is used in this paper.

2.1 Taxonomy

Ajay Shiv Sharma, Swaraj Prince, Shubham Kapoor, Keshav Kumar used Logistic regression technique on their college placement dataset which got 83.33% of accuracy [2].

Jai Ruby, Dr. K. David used ID3, J48, REP Tree, NB Tree, MLP, Decision Table Classification techniques on the placement dataset collected from their college. The results had shown that ID3 predicted well among them with an accuracy of 82.1% [3].

Ankita A Nichat, Dr. Anjali B Raut used C4.5 classification technique on the placement dataset which was collected from their college which got 80% of accuracy [4].

Oktariani Nurul Pratiwi used J48, Simple cart, kstar, SMO, NaiveBayes, OneR classification techniques on the data gathered from their high school. The results had shown that J48 and Simple Cart predicted well among them with an accuracy of 79.61% [5].

Ajay Kumar Pal and Saurabh Pal collected the data for the study and analysis of the student's educational performance basically for training and placement. The authors used different classification algorithm and used WEKA data mining tool [6]. They concluded that naive Bayes classification model is the better algorithm based on the placement data with found accuracy of 86.15% and overall time taken to build the model is at 0 sec. As compared with others Naïve Bayes classifier had lowest average error i.e. 0.28.

Ravi Tiwari and Awadhesh Kumar Sharma built the prediction model to improve the placement of the students [7]. They used WEKA as the data mining tool to build the model using random tree algorithm. They also used ID3, Bayes Net, RBF network, J48, algorithms on the student data set. They resolved that the RT (Random Tree) algorithm is more accurate with 73% for the classification/prediction of the model. The accuracy using ID3 and J48 is 71%. Bayes Net is 70%

3. METHODOLOGY:

The whole approach is depicted by the following flowchart.

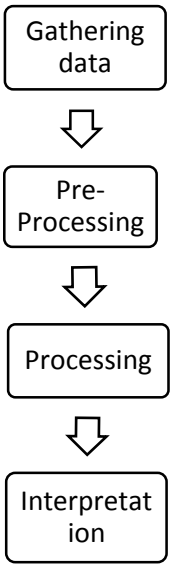

Figure 2. Flow chart of the technique

3.1 Data gathering

The sample data has been collected from our college placement department which consists of all the records of previous years students. The dataset collected consist of over 1000 instances of students.

3.2 Pre processing

Data pre processing is a technique that is used to convert raw data into a clean dataset. The data is gathered from different sources is in raw format which is not feasible for the analysis.

Pre-processing for this approach takes 4 simple yet effective steps.

3.2.1 Attribute selection

Some of the attributes in the initial dataset that was not pertinent (relevant) to the experiment goal were ignored. The attributes name, roll no, credits, backlogs, whether placed or not, b.tech % ,gender are not used.The main attributes used for this study are credit , backlogs , whether placed or not, b.tech %.

3.2.2 Cleaning missing values

In some cases the dataset contain missing values . We need to be equipped to handle the problem when we come across them. Obviously you could remove the entire line of data but what if you're inadvertently removing crucial information?after all we might not need to try to do that. one in every of the foremost common plan to handle the matter is to require a mean of all the values of the same column and have it to replace the missing data.

The library used for the task is called Scikit Learn preprocessing. It contains a class called Imputer which will help us take care of the missing data.

3.2.3 Training and Test data

Splitting the Dataset into Training set and Test Set

Now the next step is to split our dataset into two. Training set and a Test set. We will train our machine learning models on our training set, i.e our machine learning models will try to understand any correlations in our training set and then we will test the models on our test set to examine how accurately it will predict. A general rule of the thumb is to assign 80% of the dataset to training set and therefore the remaining 20% to test set.

3.2.4 Feature Scaling

The final step of data pre processing is feature scaling.

But what is it?

It is a method used to standardize the range of independent variables or features of data.

But why is it necessary? A lot of machine learning models are based on Euclidean distance. If, for example, the values in one column (x) is much higher than the value in another column (y), (x2-x1) squared will give a far greater value than (y2-y1) squared. So clearly, one square distinction dominates over the other square distinction. In the machine learning equations,

the square difference with the lower value in comparison to the far greater value will almost be treated as if it does not exist. We do not want that to happen. That is why it's necessary to transform all our variables into the same scale. There are several ways of scaling the data. One way is called Standardization which may be used. For every observation of the selected column, our program will apply the formula of standardization and fit it to a scale.

$$Xs = \frac{X - mean}{s.d.}$$

$$Xs = \frac{X - mean}{max - min}$$

$$Xs = \frac{X - min}{\max - min}$$

3.3 Processing

Processing in this paper's sense is applying different algorithms to the data to find the best results

3.3.1 ID3 algorithm

The decision tree technique involves constructing a tree to model the classification process. Once a tree is built, it is applied to each tuple in the database and results in classification for that tuple. The following issues are faced by most decision tree algorithms:

- Choosing splitting attributes
- Ordering of splitting attributes
- Number of splits to take
- Balance of tree structure and pruning
- Stopping criteria

The ID3 algorithm is a classification algorithm based on Information Entropy, its basic idea is that all examples are mapped to different categories according to different values of the condition attribute set; its core is to determine the best classification attribute form condition attribute sets. The algorithm chooses information gain as attribute selection criteria; usually the attribute that has the highest information gain is selected as the splitting attribute of current node, in order to make information entropy that the divided subsets need smallest . According to the different values of the attribute, branches can be established, each branch to create other nodes and branches until all the samples in a branch belong to the same category. To select the splitting attributes, the concepts of Entropy and Information Gain are used.[8]

A. Entropy

Given probabilities p1, p2, …, ps, where ∑pi = 1, Entropy is defined as

$$H(p1, p2, \ldots, ps) = \sum - (pi \log pi)$$

Entropy finds the amount of order in a given database state. A value of H = 0 identifies a perfectly classified set. In other words, the higher the entropy, the higher the potential to improve the classification process.

B. Information Gain

ID3 chooses the splitting attribute with the highest gain in information, where gain is defined as difference between how much information is needed after the split. This is calculated by determining the differences between the entropies of the original dataset and the weighted sum of the entropies from each of the subdivided datasets. The formula used for this purpose is:

$$G(D, S) = H(D) - \sum P(Di)H(Di)$$

3.3.2 Random forest

The random forest algorithm can also be thought of as an ensemble method in machine learning. The input to a random forest algorithm is a dataset consisting of records, with attributes. Random subsets of the input are created. On each of the random subset created, a decision tree will be constructed. The final class of a test record will be decided by the algorithm which uses the majority vote technique. Random forest algorithm makes use of the out of bag error technique.

Each tree is constructed using the following algorithm:

1. Let the number of training cases be N, and the number of variables in the classifier be M.
2. We are told the number m of input variables to be used to determine the decision at a node of the tree; m should be much less than M.
3. Choose a training set for this tree by selecting N times with replacement from all N available training cases (i.e. take a bootstrap sample). Use the rest of the cases to estimate the error of the tree, by predicting their classes.
4. For each node in the tree, randomly choose m variables on which to base the decision at that node. Calculate the best split based on these m variables in the training set.
5. Each tree is fully grown and not pruned (as may be done in constructing a normal tree classifier).

4. RESULTS AND DISCUSSION

The data set used for is further splitted into two sets consisting of two third as training set and one third as testing set. Among

the two algorithms applied random forest shown the best results. The efficiency of the two approaches is compared in terms of the accuracy.

The accuracy of the prediction model/classifier is defined as the total number of correctly predicted/classified instances. Accuracy is given by using following formula:

Accuracy= (TP+TN/TP+FN+FP+TN)∗100

where TP, TN, FN, FP represents the number of true positives, true negative, false negative and false positive cases.

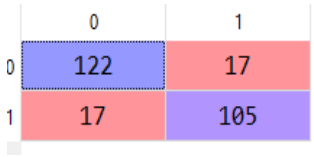

Figure 3. confusion matrix of random forest algorithm

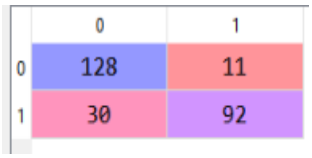

Figure 4. confusion matrix of decision tree algorithm

Table 1. Comparision of the performances of various algorithms

Algorithms	Accuracy	Precision	Recall
Random forest	86%	0.877	0.87
Decision Tree	84%	0.92	0.81

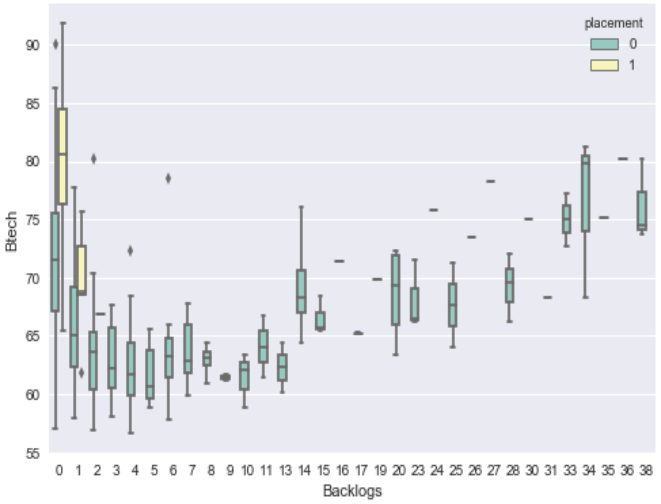

Figure 5. Boxplot representation of the data

5. CONCLUSION

The campus placement activity is incredibly a lot of vital as institution point of view as well as student point of view. In this regard to improve the student's performance, a work has been analyzed and predicted using the classification algorithms Decision Tree and the Random forest algorithm to validate the approaches. The algorithms are applied on the data set and attributes used to build the model. The accuracy obtained after analysis for Decision tree is 84% and for the Random Forest is 86%. Hence, from the above said analysis and prediction it's better if the Random Forest algorithm is used to predict the placement results.

REFERENCES

[1]. Mangasuli Sheetal B, Prof. Savita Bakare "Prediction of Campus Placement Using Data Mining Algorithm-Fuzzy logic and K nearest neighbour" International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 6, June 2016 .

[2]. Ajay Shiv Sharma, Swaraj Prince, Shubham Kapoor, Keshav Kumar "PPS-Placement prediction system using logistic regression" IEEE international conference on MOOC,innovation and Technology in Education(MITE), December 2014.

[3]. Jai Ruby, Dr. K. David "Predicting the Performance of Students in Higher Education Using Data Mining Classification Algorithms - A Case Study" International Journal for Research in Applied Science & Engineering Technology (IJRASET) Vol. 2,Issue 11,November 2014.

[4]. Ankita A Nichat, Dr.Anjali B Raut "Predicting and Analysis of Student Performance Using Decision Tree Technique" International Journal of Innovative Research in Computer and Communication Engineering V0l. 5, Issue 4, April 2017.

[5]. Oktariani Nurul Pratiwi "Predicting Student Placement Class using Data Mining" IEEE International Conference 2013.

[6]. Ajay Kumar Pal and Saurabh Pal, "Classification Model of Prediction for Placement of Students", I. J. Modern Education and Computer Science, 2013, 11, 49-56

[7]. Ravi Tiwari and Awadhesh Kumar Sharma, "A Data Mining Model to Improve Placement", International Journal of Computer Applications (0975 – 8887) Volume 120 – No.12, June 2015

[8]. Ms.sonal patil, Mr.Mayur Agrawal, Ms.Vijaya R. Baviskar "Efficient Processing of Decision Tree using ID3 & improved C4.5 Algorithm", International Journal of Computer Science and Information Technologies, Vol. 6 (2) , 2015, 1956-1961