



CLASSIFICATION OF PLACEMENT READY CANDIDATES FOR PROGRAMMING PROFILE USING MACHINE LEARNING

¹Prabha Siddhesh Kadam, ²Girish Tere

¹Assistant Professor, ² Assistant Professor

¹Department of Computer Science,

¹Kirti M. Doongursee College, Mumbai, India

Abstract: Educational institutes use their students records and able to achieve significant improvement in teaching learning. Students' placement is one of the processes has its own significance in building reputation of the institute. Machine learning techniques proves useful to predict the placement possibilities of the students. Ensemble machine learning algorithms used to predict the placement domain for students at the institutes. Student's records of size 550 used to classify for job title programmer. Outcomes shows 92.60% accuracy with Random Forest classifiers. The model classifies samples among three classes skilled, need to improve and another domain.

Index Terms - Machine learning, placement prediction, ensemble classifiers, supervised learning.

I. INTRODUCTION

Many times, especially in self-financed course student faculty ratio was poor. Mentoring many students at a time and find out their Strong areas and weakness in study domain is tremendous task where number of faculties appointed are less. Due to pandemic and otherwise also ICT (Information Communication Technology) plays a vital role in education domain. Both face-to-face and virtual models promote the information and communication technology, in both teaching-learning and academic management processes.[7] Reputation of the institute is judged based on the placement record. Placement of undergraduate students in the industry with good package is a challenge for the institutes. New students, who inquire admission in undergraduate courses of the institutes do inquire about the placement records of the earlier batches.[4]

In this system, authors focus on Computer Science undergraduate students at Mumbai University. The system learns from the academic record of the student about the past years, starting from 12th and all 3 years of the Undergraduate Course. Based on the learning's the system predict the programming job profile best suited for the student or not.

In Computer Science undergraduate courses there are many tracks that the student would consider for getting placed in the industry. Based on the programming courses and practical performance in which the student has excelled, the system will predict the future job profile for the student. The algorithm considers various parameters using the previous year's student's historical data and predict the placement possibilities of the current students and focus on increasing the placement percentage of the institute.

This system will help to detect students eligible to place as a programmer by using student's record sets. System will classify students with good programming skills. By taking corrective measure student can be groom for the placement drives and possibility of increase in campus placements.

The flow of the work divided into seven sections. The previously done work by researchers included in the chapter two literature review. The objectives of the study mentioned in the chapter three. Research method has its own significance, included in the chapter four. Results and conclusion inferred from the work mentioned in the chapter five. Section six talks about the future scope while the section seven holds the references.

II. LITERATURE REVIEW

Machine learning techniques brings new revolution in the teaching learning domain. The challenging task of classifying students and predict the successes or drop out is possible and became easy. Only need for this process in student's records.[8] Placement of student is one of such area, where faculty struggle to search right candidate for placement position. Grooming the student for the placement drive is also a mentor's responsibility where this mechanism helps him to classify the student in some specific group not only depends on students' interest but judging his skills. Researcher Ajay Kumar Pal and Saurabh Pal (2013) used WEKA as a tool for data analysis on the students earlier and current academic records. Authors found relation between the academic performance of the students and the placement. Authors preferred the supervised learning mechanism for classification task. Implementation of Naïve Bias, MLP, and tree base algorithm C4.5 makes it clearer that on sample size of 65 records Naïve bias

gave 86.15 percent accuracy. The attributes like student's performance in the seminar, laboratory work, soft skills, MCA results are used in the binary classification to predict whether student will get placed or not. [1] Researcher Prabha Kadam and Girish Tere (2021) explore the relation between the grades scored at entry point and final score in the Batchelor course.[7] Author uses ensemble approach to classify the learners into three categories depends upon there learning capability. Researchers build hybrid model using bagging concept and Random Forest algorithm for the classification of records, which returned the 87.50% accuracy.[6] N. Kumaran et. al (2021) implements machine learning Fast Tree algorithm using visual studio plugin model builder for placement prediction. Regression algorithms implementation gave R squared value 0.8806 at fifth iteration. [4] The researcher Pothuganti Manvitha and Neelam Swaroopa (2019) used Random Forest and achieved accuracy 86% for Campus Placement Prediction. The sample size 1000 and variables used were credit, backlogs, whether placed or not, Batchelor of Technology percentage. Authors used preprocessing techniques to clean data but not implemented the class balancing techniques to enhance performance and avoid bias.[5] Kachi Anvesh et al. (2019) developed an automatic student analysis and placement prediction system using machine learning algorithms. SVM (Support Vector Machines) gave more accuracy with 90.3, XG Boost with 88.33, Logistic regression with 86.3 percent accuracy for sample size 20,000 with 56 attributes considered under study. Binary classification performed using 80-20 split without considering the hyperparameter tuning.[3] Irene Treasa Jose et.al (2020) applied multiple machine learning model and perform the comparative study based on efficiency. Author used KNN, Random Forest, SVM (Support Vector Machines) and Logistic regression algorithm in this study. This study was based on the test score, hackathon participation, no of certification done by the students along with the score in the programming score and CGPA (Cumulative Grade Point Average). The SVM reached to the maximum accuracy value for small data set.[2]

III. OBJECTIVE

The objectives of proposed study are as follows –

1. To study the right data split that could help in creating the classification model for classifying students for programming jobs, using small dataset size.
2. To perform comparative study of multiple model design using machine learning algorithms for prediction of students having good programming skills.
3. To propose a near-to-precise prediction model to detect most suited student for programming job profile in computer science course.
4. To evaluate performance of proposed technique.

IV. RESEARCH METHODOLOGY

This study is limited for undergraduate computer science students within Mumbai region for data size 550. The samples collected from survey using google platform. Total twenty attributes used under the study in which the score of the student from its first semester and second semester programming subjects gets consider as well as the entry level marks also consider for the study. In the literature it was seen that entry level marks can contribute towards the final score of the students at the end of the course.[7] Subjects like Physics, Mathematics and English in the higher secondary schools considered in model building.

The study was performed in two parts, first with the selected 20 attributes and afterward removing the attributes, marks received at the higher secondary level. The grades obtained in semester one and semester two in the programming subjects considered under the study. The subject namely Programming with Python-I, Programming with Python-II, Programming with C, Linux, Data Structures, Practical component and internal evaluation of each subject were the 16 attributes. The three other attributes considered under the study were the entry level marks of the student, he scored in the subject of mathematics, physics, and English. The total 20 attributes, including the target variable profile make the data set complete

Tree base classifier J48, REP Tree applied on data as well as Ensembled Tree Base Algorithm. Random Forest used for classification of records. Bagging approach implemented using REP Tree as a base learner, boosting mechanism also explored in the study.

The primary dataset divided into two parts training data set and testing data set, where the training data set pusses 70% of samples from the record set while the testing data set have remaining 30% records. Another approach of classification of records using cross-validation (10-fold) also explored in this work to produce the best model. In the pre-processing stage class balancer also applied on the data set to balance the imbalance classes and avoid bias outcome. The data analysis performed using WEKA Tool.

J48: It is an algorithm which based on the value of information gain. The algorithm classifies samples using one attribute value at a time, thus known as univariate type algorithm. Usually, the tree structure generates using this algorithm was seen small compared to the other tree base approaches. This was achievable due the divide and conquer mechanism implemented while building the procedure of this algorithm.[1][7]

REP Tree: This algorithm also counted as univariate algorithm and works with the information gain. This algorithm build model in the fastest manner and allows to reduce variance. Reduced Error Pruning Tree able to overcome the known problem of overfitting.[7]

Random Forest: It is one of the favourite algorithms of researcher for classification of categorical data is Random Forest. Random Forest is nothing, but a collection of multiple decision trees constructed with some randomness. Bootstrap mechanism allows the experiment to replace the record back to the pool of training records and reuse it for building the next tree. Result generates by performing averaging operation on the output generates in the testing phase. [2][5][6][7]

Bagging: An ensemble approach which works with base learner and classify samples using random subsets of records. The name bagging suggest that records are retrieved and place back to the pool, so that can be reused again for generating another tree.[6]

AdaBoost: A Boosting algorithm falls in the category of ensemble learner. It works well on noisy data and able to minimize the problem of overfitting since it bases on judgment stumps. Iterative approach and weights assignment concepts of the algorithm allows to boost the outcome.[3]

V. RESULTS AND DISCUSSION

The 550-sample space used in this study but after applying the class balancing each class holds 183 values. The distribution of classes before balancing was 441 of type need to improve, 71 skilled, 38 other domains. The tree base algorithms return good accuracy but bias towards the class having large samples. This leads towards misclassification of records in other class, thus the class balancing is necessary in multiclass classification. The comparative study was performed using the Tree base algorithms, J48, REP Tree as well as ensemble learner algorithms also use to build different models. The famous Random Forest, a tree base ensemble learning mechanism gave the maximum accuracy 92.60% among these models compare to the AdaBoost and Bagging. The data set divided into two-part training and testing, where the 70-30 split was used for building the model and study also with the 10-fold cross validation. The outcome of first approach, considering all 20 attributes for the study has given below in the figure 1.

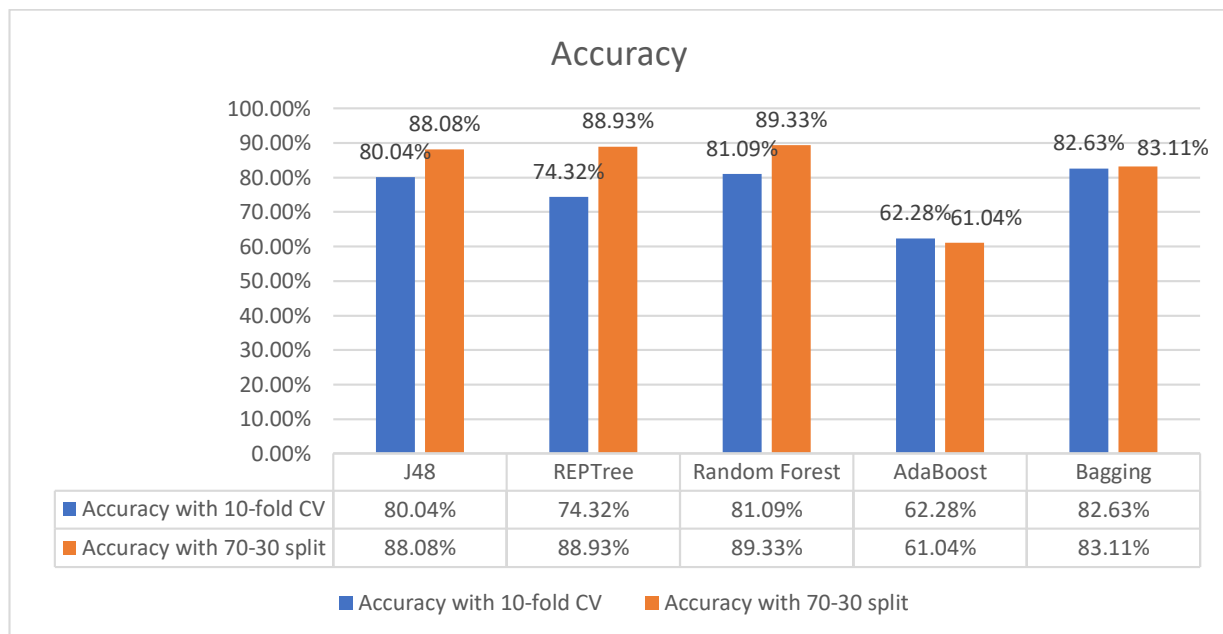


Figure 1. Accuracy of models using all attributes considered under study.

The results obtained in the first approach of model building suggest the 70-30 data split gives more accuracy compared to the 10-fold CV for this data set. Since the data set holds 550 samples 70% samples in the training suits more for this study. In this case random forest algorithm gave accuracy 89.33% while the REP Tree returns 88.93% and J48 gave 88.08%. The Bagging approach also perform well and able to reach the accuracy value 83.11%.

The rank of the feature variables seen in the study listed in the table 1. Information Gain Ranking Filter used to rank the attributes.

Table 1. Ranked Features

0.2269 3 FII C	0.0899 16 FII PRACTICAL USCSP203
0.2112 5 FII Python-II	0.0859 14 FII PRACTICAL USCSP201
0.2029 9 FII Database Structures	0.0747 13 FI PRACTICAL USCSP103
0.1824 7 FII Linux	0.0682 12 FI PRACTICAL USCSP102
0.1665 4 FII C_Internal	0.0675 11 FI PRACTICAL USCSP101
0.1604 1 FI Python-I	0.0622 15 FII PRACTICAL USCSP202
0.1392 6 FII Python-II_ Internal	0 17 12th standard marks: [PHYSICS]
0.1033 2 FI Python-I_ Internal	0 18 12th standard marks: [MATHEMATICS]
0.098 8 FII Linux_ Internal	0 19 12th standard marks: [ENGLISH]
0.0968 10 FII Database Structures _ Internal	Selected attributes: 3,5,9,7,4,1,6,2,8,10,16,14,13,12,11,15,17,18,19: 19

The above table shows that entry level marks ranked as zero, has less significance in the study. In the next approach only 17 attributes considered while building model. The results of the next approach where we removed the three attributes, the entry level marks are given in the figure 2.

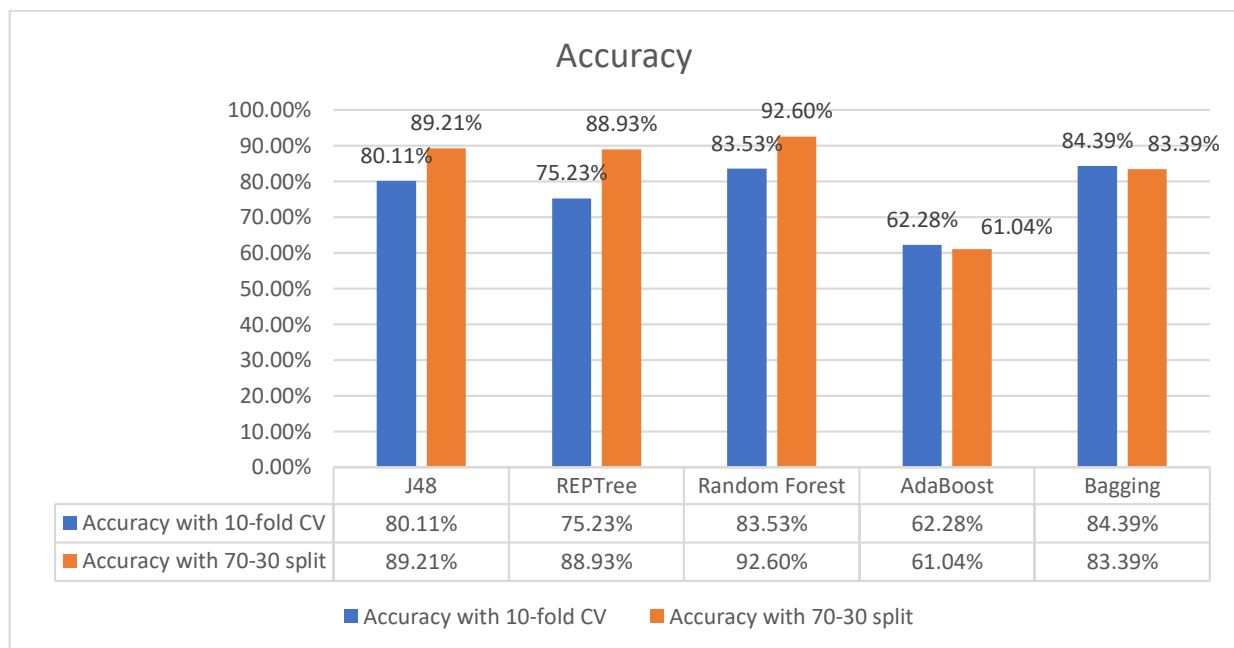


Figure 2. Accuracy of models using selected attributes considered under study.

The experiment gave highest accuracy with Random Forest 92.60% where again 70-30 split wins. The other algorithms J48, REP Tree and Bagging return the accuracy 89.21%, 88.93% and 83.39% respectively. It was seen that boosting mechanism does not work well in any case. The multiclass classification details for Random Forest with 70-30 split and after performing class balancing using ClassBalancer in WEKA is given in the table 2.

Table 2. Detailed accuracy by class

TP (True Positive) Rate	FP (False Positive) Rate	Precision	Recall	F-Measure	Class
100.00%	0.80%	98.20%	100.00%	99.10%	Skilled
97.80%	10.70%	85.30%	97.80%	91.20%	Need to improve
77.80%	0.40%	98.80%	77.80%	87.00%	Another domain.
92.60%	4.60%	93.40%	92.60%	92.40%	Weighted Average

The results given in the table 2 shows that false positive rate for skilled class prediction 0.80% while the true positive rate is 100%. The F-measure term values 99.10% for class skilled and 91.20% for class need to improve, says that classification model suites for predicting the students with programming skills for placement drive in the institute. It seen that entry level marks make the difference in accuracy of model by almost 3%. The Tree structures generated by the REP Tree and J48 also not considered those attributes towards classification of the records. The study will highlight on machine learning techniques useful in the prediction model for small data sets. Even recommend candidates who need help to improve their knowledge in the programming subjects.

6. FURTHER SCOPE

This study will help students and college administrators for student's placement and grooming. Grooming of selected candidates will be easier and more subject specific in this scenario which will help to increase number of campus placements. There are multiple subject domains in the course and detailed study is possible for each domain from placement perspective. Large sample size will help to improve the prediction capability of the algorithms. Modelling may not be perfect and there may be probability of error (misclassification), which can be resolved using advanced techniques.

7. REFERENCES

- [1]. Ajay Kumar Pal, Saurabh Pal, "Classification Model of Prediction for Placement of Students," I.J.Modern Education and Computer Science, 2013, 11, 49-56, DOI: 10.5815/ijmecs.2013.11.07
- [2]. Irene Treasa Jose, Daibin Raju, Jeebu Abraham Aniyankunju, Joel James, Mereen Thomas Vadakkal, "Placement Prediction using Various Machine Learning Models and their Efficiency Comparison," International Journal of Innovative Science and Research Technology, Volume 5, Issue 5, May – 2020, ISSN No: -2456-2165
- [3]. Kachi Anvesh, B. Satya Prasad, V. Venkata Sai Rama Laxman, B. Satya Narayana, "Automatic Student Analysis and Placement Prediction using Advanced Machine Learning Algorithms," International Journal of Innovative Technology and Exploring Engineering (IJITEE), Volume-8 Issue-12, October 2019, ISSN: 2278-3075
- [4]. N. Kumaran, V. Sankaranarayanan, V. Somasundaram, T. Srinivasan, "Placement Prediction and Training Tool Using Machine Learning (ML.NET)," International Journal for Innovative Research in Multidisciplinary Field, Volume - 7, Issue - 4, Apr – 2021, ISSN: 2455-0620
- [5]. Pothuganti Manvitha, Neelam Swaroopa, "Campus Placement Prediction Using Supervised Machine Learning Techniques," International Journal of Applied Engineering Research Volume 14, Number 9 (2019) pp. 2188-2191, ISSN 0973-4562
- [6]. Prabha Kadam, Girish Tere, "An Ensemble Approach for Early Detection of At-Risk Students," GIS Science Journal, Volume 8, Issue 6, 2021, ISSN No: 1869-9391
- [7]. Prabha Kadam, Girish Tere, "A Study to Classify Students Records based on Academic Performance using Tree Base Classifiers.," International Journal of advanced and Innovative Research, Volume 8, Issue 2 (III) April - June 2021, ISSN: 2394-7780
- [8]. Prabha Kadam, Vinod Vaze, "Early Detection of Prone to Failure Student Using Machine Learning Techniques," Bioscience Biotechnology Research Communications (Special issue on Emerging Research on Management, Sciences and Technology) Volume 14(05) 2021 ISSN No: 0974-6455