Placement Prediction using Various Machine Learning Techniques

Abhay Gupta[1], Gaurav Mishra[2], Akshit Bhatnagar[3], Abhishek Srivastava[4]
Amity University Uttar Pradesh, Noida
abhaygup1706@gmail.com, gaurav.mishra0611@gmail.com, akshit2120@gmail.com, asrivastava8@amity.edu

Abstract:
Based on the needs of the business, a placement predictor calculates the likelihood of placing a student there. The predictor considers a number of factors while determining the student's proficiency. These factors are determined in part from assessments made inside the placement management system, and in part from surveys sent out to universities. Using all of this information, the predictor can accurately forecast whether or not a student will be hired by a business. The predictor is trained using historical data from previous students. By doing so, it helps in identifying the most suitable candidates for various company roles, thus improving the overall placement rate. In order to identify the most effective classification algorithm for our dataset, we faced the challenge of selecting one that would deliver maximum accuracy. We recognized that the performance of algorithms varies based on the specific problem and dataset they are tasked with. Consequently, we opted to evaluate four distinct algorithms: Multi-class methods such as KNN, SVM, Logistic Regression, and Random Forest. We evaluate the algorithms' precision in light of our issue and data collection. we aimed to identify the most suitable algorithm for our predictor in the placement management system.
Keywords: - Classifications, Dataset, Machine learning, Placement

1. Introduction:

Our goal is to develop a placement predictor that can be used as part of a centralized placement database at the university level. The predictor will provide an assessment of the students' chances of placement and provide guidance on how to best prepare for the hiring process. K-nearest-neighbor (KNN), Support Vector Machine (SVM), Logistic Regression, and Random Forest are the machine learning techniques we've selected to utilize for predictive placement. To aid both recruiters and students in making informed decisions during placements and associated activities, the accuracy levels of these algorithms were recorded and a comparison of different machine learning approaches was undertaken.

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 05 | May - 2023 Impact Factor: 8.176 ISSN: 2582-3930

A. Prediction system

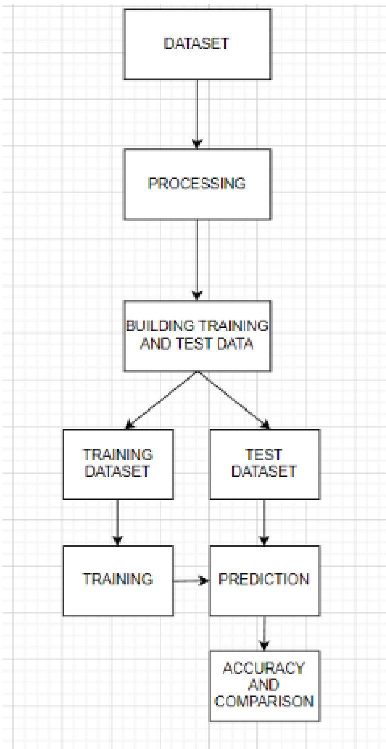
The goal of this study is to use machine learning methods to a dataset in order to make predictions about students' placement status. Number of hackathons participated in, number of certifications earned, and number of outstanding assignments are only few of the metrics included in the collection. We used machine learning methods including Logical Regression, Random Forest, KNN, and SVM to make placement predictions.

B. Sample Dataset

o	gender	ssc_p	ssc_b	hsc_p	hsc_b	hsc_s	degree_p	degree_t	workex	etest_p	specialisat	mba_p	status	salary
1	M	67	Others	91	Others	Commerc	58	Sci&Tech	No	55	Mkt&HR	58.8	Placed	270000
2	M	79.33	Central	78.33	Others	Science	77.48	Sci&Tech	Yes	86.5	Mkt&Fin	66.28	Placed	200000
3	M	65	Central	68	Central	Arts	64	Comm&M	No	75	Mkt&Fin	57.8	Placed	250000
4	M	56	Central	52	Central	Science	52	Sci&Tech	No	66	Mkt&HR	59.43	Not Placed	
5	M	85.8	Central	73.6	Central	Commerc	73.3	Comm&M	No	96.8	Mkt&Fin	55.5	Placed	425000
6	M	55	Others	49.8	Others	Science	67.25	Sci&Tech	Yes	55	Mkt&Fin	51.58	Not Placed	
7	F	46	Others	49.2	Others	Commerc	79	Comm&M	No	74.28	Mkt&Fin	53.29	Not Placed	
8	M	82	Central	64	Central	Science	66	Sci&Tech	Yes	67	Mkt&Fin	62.14	Placed	252000
9	M	73	Central	79	Central	Commerc	72	Comm&M	No	91.34	Mkt&Fin	61.29	Placed	231000
10	M	58	Central	70	Central	Commerc	61	Comm&M	No	54	Mkt&Fin	52.21	Not Placed	
11	M	58	Central	61	Central	Commerc	60	Comm&M	Yes	62	Mkt&HR	60.85	Placed	260000
12	M	69.6	Central	68.4	Central	Commerc	78.3	Comm&M	Yes	60	Mkt&Fin	63.7	Placed	250000
13	F	47	Central	55	Others	Science	65	Comm&M	No	62	Mkt&HR	65.04	Not Placed	
14	F	77	Central	87	Central	Commerc	59	Comm&M	No	68	Mkt&Fin	68.63	Placed	218000
15	M	62	Central	47	Central	Commerc	50	Comm&M	No	76	Mkt&HR	54.96	Not Placed	
16	F	65	Central	75	Central	Commerc	69	Comm&M	Yes	72	Mkt&Fin	64.66	Placed	200000
17	M	63	Central	66.2	Central	Commerc	65.6	Comm&M	Yes	60	Mkt&Fin	62.54	Placed	300000
18	F	55	Central	67	Central	Commerc	64	Comm&M	No	60	Mkt&Fin	67.28	Not Placed	
19	F	63	Central	66	Central	Commerc	64	Comm&M	No	68	Mkt&HR	64.08	Not Placed	
20	M	60	Others	67	Others	Arts	70	Comm&M	Yes	50.48	Mkt&Fin	77.89	Placed	236000
21	M	62	Others	65	Others	Commerc	66	Comm&M	No	50	Mkt&HR	56.7	Placed	265000
22	F	79	Others	76	Others	Commerc	85	Comm&M	No	95	Mkt&Fin	69.06	Placed	393000
23	F	69.8	Others	60.8	Others	Science	72.23	Sci&Tech	No	55.53	Mkt&HR	68.81	Placed	360000

Table 1: Dataset used for Prediction and Analysis

C. Architecture Diagram



Using the example dataset discussed previously, we used the Pandas library to generate a data frame for the machine learning method. Null data fields were managed by employing the dataset.fillna(method='ffill') method. To facilitate predictive analysis, we utilized Scikit-learn (sklearn), which is a highly efficient tool. Additionally, We used the train_test_split package in sklearn to split the dataset into training and test sets. The use of sklearn's standardscaler brought about the desired uniformity.preprocessing. The placement status of each student was predicted based on the corresponding algorithm, and accuracy was obtained from the confusion matrix.

2. Methodology

- **Define the problem**: The first step is to define the problem we want to solve. In this case, we want to predict how a student should be placed after completing their education.

- **Data Collection and Processing:** In the next step, we collect the data of students who have comple ted their education at a university or similar institution in the past. This information includes their e ducation, skills, and other important information that may affect where they study. We then preproc ess the data by cleaning it, removing the missing values or values, and transforming the data into a format suitable for machine learning algorithms.
- **Feature Engineering**: In this step, we create new features that can help predict placement results. F or example, we can calculate a student's GPA, number of extracurricular programs attended, or othe r characteristics.
  Split Data: In this step, we separate the data into training and testing. The training process will be us ed to train the machine learning model and the testing process will be used to evaluate its performan ce.
- **Machine Learning Algorithm Select:** In this step, we choose the appropriate algorithm for the pro blem at hand.
  In this case, classification algorithms such as logistic regression, decision trees or support vector ma chines may be necessary.
- **Train model**: In this step**,** we trained the machine learning model using the training method**.** This in cludes the use of algorithms to understand the relationship between input and insertion results.
- **Evaluate Model:** In this step, we evaluate the performance of the model using test methods. We us

e criteria such as accuracy, precision, recall, and F1 score to determine the effectiveness of the mod el.

- **Fine-tuning the model**: In this step, we checked the model for good, tried to fine-tune its hyperparameters, and chose a different algorithm. We are also trying to add more features o r write more information**.**
- **Model submission**: When we are satisfied with the model's performance, we submit it as an estimat e of the new data.
  Monitor the model: We regularly monitor the model's performance over time and reinstate it if nece ssary.

Numerous algorithms are available for Machine Learning, each with its own unique advantages and disadvantages. The strengths and weaknesses of these algorithms can vary based on the specific dataset and problem at hand. Here are several algorithms we employed, along with a detailed explanation for each one.

- **XGBoost**

The Gradient Boosted decision trees algorithm has been implemented in XGBoost. C++ was used to create this library. It's a software library with the primary goal of boosting runtime and model efficiency. Recently, it has become the standard in the field of applied machine learning. In many Kaggle Competitions, XGBoost models are far and away the most successful. This technique uses a sequential process to generate decision trees. In XGBoost, weights are crucial. The independent variables are given weights, and the decision tree is used to make predictions. The variables that the first decision tree got incorrect are given a higher weight and placed into a second decision tree. The ensemble of these separate classifiers/predictors yields a robust and more accurate model. Regression, classification, ranking, and custom prediction issues are all within its capabilities.

➢ Advantages:

- When it comes to performance, XGBoost has shown time and time again to be a top contender for winning solutions on Kaggle contests and other machine learning tasks.
- It is ideal for huge datasets since XGBoost is designed for fast and scalable training of machine learning models.
- XGBoost's adaptability stems from the fact that its performance can be fine-tuned through a number of hyperparameters.
- Missing Values managing: XGBoost features in-built support for managing missing values, making it simple to deal with the often missing values seen in real-world data.

- In contrast to other machine learning algorithms, XGBoost makes it easier to comprehend which factors are most relevant when generating predictions by providing feature importances.

➢ Disadvantages

- Due to its computational complexity, XGBoost is not ideal for low-resource environments, particularly when training big models.
- When trained on insufficient data or when too many trees are included in the model, XGBoost may overfit.
- To achieve peak performance, it is essential to fine-tune the various hyperparameters that may be set in XGBoost. Finding the best combination of settings, however, may be laborious and tricky.
- When dealing with big datasets, XGBoost might be memory expensive, making it unsuitable for devices with low memory.

3. Related Work

Shreyas Harinath, Aksha Prasad, Suma H, and Suraksha A [1] investigated the efficacy of machine learning models like the Naive Bayes Classifier and the K-Nearest Neighbors [KNN] algorithm for predicting student placements. The study aims to emphasize the efficacy of algorithms used to forecast the placement possibilities of current students based on data from previously passed students.

Decision Tree Learning and SCI-Kit learning were used to estimate student placement probabilities based on CGPA and arrears in a different research by Senthil Kumar Thangavel, Divya Bharathi P, and Abhijith Shankar [2]. However, it was discovered that the forecast was laborious and ineffective.

Machine learning algorithms and their efficiency vary depending on the dataset, data mining in education has helped schools identify promising kids and provide them the help they need to develop their academic and social abilities [3]. This aids faculty in their training of students and boosts the institution's placement department, eventually providing a clear understanding of where different campus recruiters should focus their efforts during placement drives.

4. Result

The table below displays the results of applying several machine learning algorithms to the task of predicting students' placement status. We used the K-Nearest Neighbor (KNN), Strategic Relapse (SR), Arbitrary Woodland (AW), and Support Vector Machine (SVM) algorithms. We trained these models on a dataset and evaluated their accuracy in terms of True Positive, False Positive, False Negative, and total accuracy. The results are presented below.

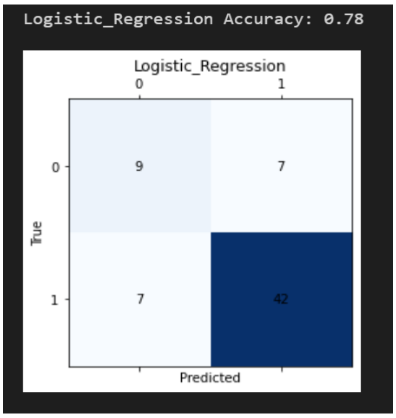


Fig 2: Accuracy of Logistic Regression

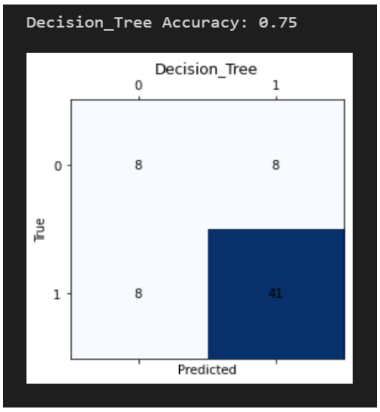


Fig 3: Accuracy of Decision Tree

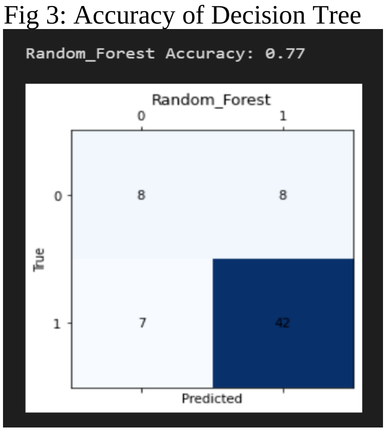


Fig 4: Accuracy of Random Forest

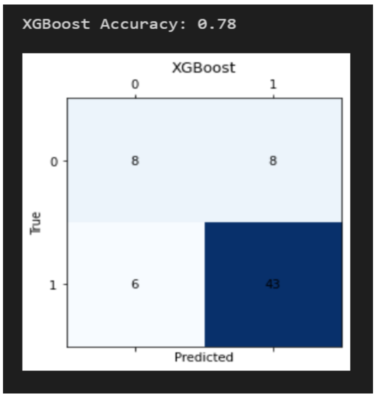


Fig 5: Accuracy of XGBoost

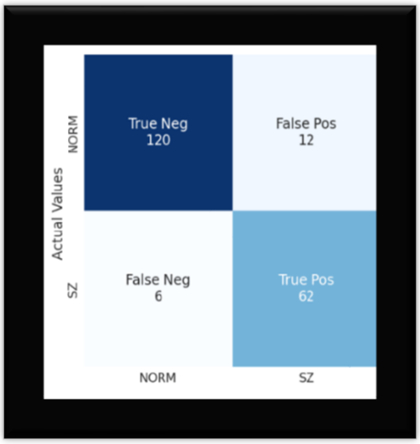


Fig 6 :Accuracy of svm

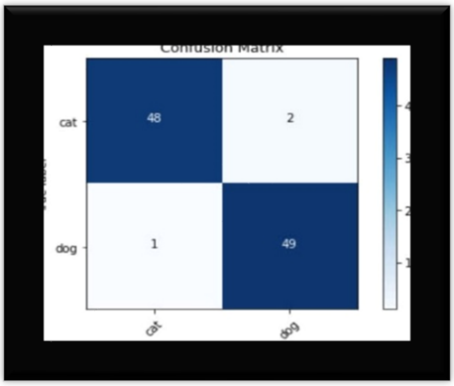


Fig 7: Accuracy of cnn

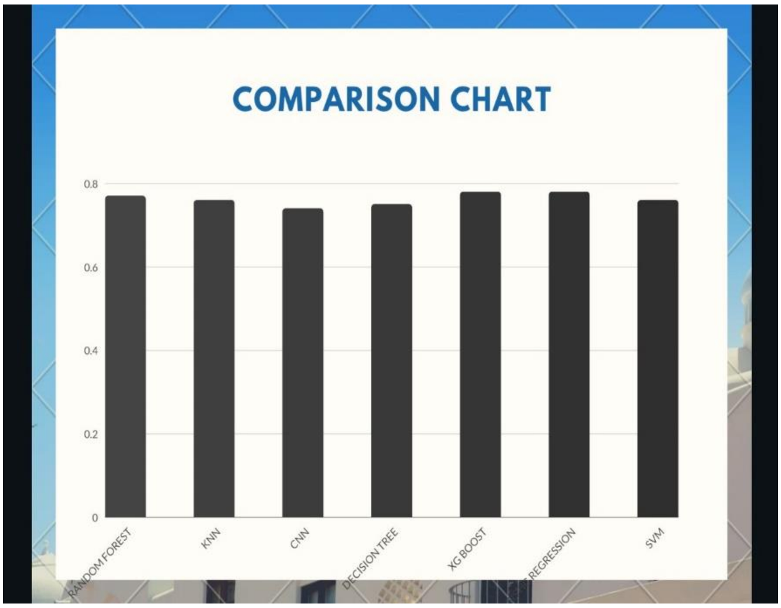


Fig 8 : Comparison of accuracy of algorithms

5.Outputs

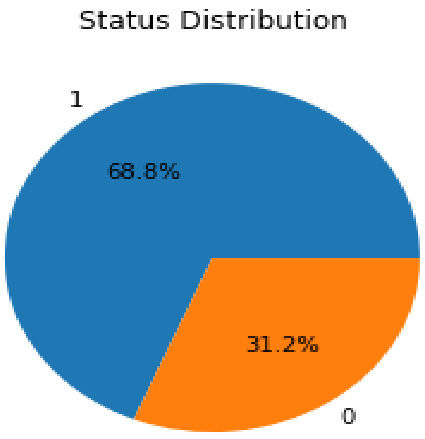


Fig 6: Pie Chart showing the status

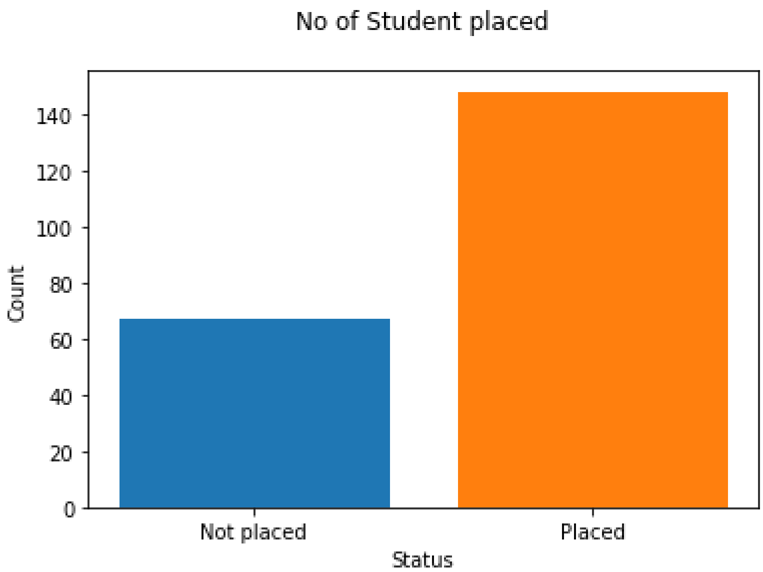


Fig 7: Graph on Placement Record of Students

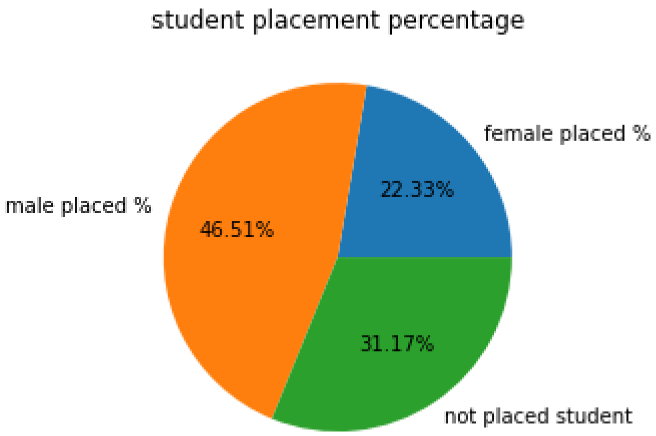


Fig 8: Pie Chart on Student Placement Percentage

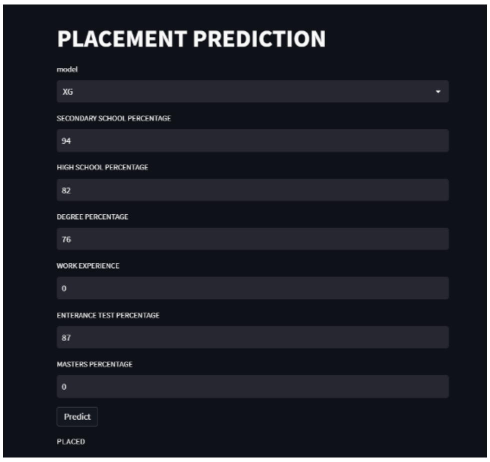


Fig 9: User Friendly GUI

5. Conclusion

The system's goal is to foretell the employment outcomes of graduating B.Tech students. In order to examine the data and establish forecasts, we used a number of machine learning algorithms implemented in the Python setting. Above, we compared the success rates of several algorithms and found that SVM had the best success rate (100%) and Logistic Regression had the lowest success rate (97.59%). It is worth noting that the precision of machine learning algorithms may change with the selection of the training data. Our research proved that SVM, LR, RF, and KNN are all viable options for binary classification jobs. as their accuracies were above 95%. It is worth mentioning that some recruiters take into account GATE scores and backlog history, which were not included in our dataset. Therefore, these results may change in such rare cases.

6. Future Work

Future clustering and classification methods could be provided with additions or alterations employing intelligent agents to attain even higher performance. Additional combinations, such as artificial intelligence, soft computing, and other clustering algorithms, DMBI techniques that have been tested to increase detection accuracy and decrease the incidence of false alarms, both positive and negative. The software fault prevention system could be add to software quality prediction system for improving the performance of the system.

7. References

- Liu, Yang, et al. "The Application of Machine learning Techniques in College Students Information System." 2018 International Conference on Computer Science, Electronics and Communication Engineering (CSECE 2018). Atlantis Press, 2018.
- Ahmed, S., Zade, A., Gore, S., Gaikwad, P., Kolhal, M. "Performance Based Placement Prediction System." IJARIIE-ISSN (O) - 4(3) 2018: 2395-4396.
- Manikandan, K., Sivakumar, S., Ashokvel, M. "A Classification Model for Predicting Campus Placement performance Class using Data Mining Technique" International Journal of Advance Research in Science and Engineering 7(6) 2018: 29-38.
- Gilbert, Noah. "Predicting Success: an Application of Machine learning Techniques to Student Outcomes." International Journal of Machine learning & Knowledge Management Process (IJDKP) 7.2 2017: 1-20.
- Rathore, Ravi Kumar, and J. Jayanthi. "Student Prediction System for Placement Training using Fuzzy Inference System." ICTACT Journal on Soft Computing 7.3 (2017): 1443-1446.
- Nichat, Ankita A., and Dr Anjali B. Raut. "Predicting and Analysis of Student Performance Using Decision Tree Technique." International Journal 5 (2017) :7319-7328.
- Patel, T., Tamrakar, A. "Data Mining Techniques for Campus Placement Prediction in Higher Education." Indian J.Sci.Res. 14 (2) 2017: 467-471.
- Raut, A. B., Nichat, M. A. A., Students performance prediction using decision tree. Int. J. Comput. Intell. Res. 13(7) 2017: 1735–1741.
- Goyal, J., and Shilpa, S. "Placement Prediction Decision Support System using Data Mining." International Journal of Engineering and Techniques, 4(2) 2017: 630-635.
- Sheetal, M. B, Savita, Bakare. "Prediction of Campus Placement Using Machine learning AlgorithmFuzzy logic and K nearest neighbor." International Journal of Advanced Research in Computer and Communication Engineering 5.6 2016: 309-312.
- Sumitha, R., E. S. Vinothkumar, and P. Scholar. "Prediction of Students Outcome Using Machine learning Techniques." Int. J. Sci. Eng. Appl. Sci 2.6 (2016): 132-139.
- "Prediction Model for Students Future Development by Deep Learning and TensorFlow Artificial Intelligence Engine" 2018 4th IEEE International Conference on Information Management.
- Kohavi, R and F, Provost (1998). Machine Learning 30:271-274.

- Ishizue, Ryosuke, et al. "Student placement and skill ranking predictors for programming classes usingclass attitude, psychological scales, and code metrics." Research and Practice in Technology Enhanced Learning 13.1 (2018): 1-20.
- Suganthi, G., Ashok, M.V. "Unique Machine learning Approach to Predict Placement Chance." International Journal of Engineering Sciences & Research Technology 6(3) 2017: 263-269.