International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Predicting Student Placement using Machine Learning Models: A Comparative Analysis

Atharva Kale[1], Sumedh Tardalkar[2], Suyash Bhavsar[3], Prof. Varsha Shukre[4]

[1,2,3]Student, Dr Vishwanath Karad's MIT World Peace University Pune, India
[4]Professor, Dr Vishwanath Karad's MIT World Peace University Pune, India

Abstract
The transition from academia to the workforce marks a critical juncture for students, with the ability to predict their placement success becoming increasingly vital. This paper undertook a thorough comparative analysis of machine learning (ML) models applied in forecasting student placement. It delves into a range of factors influencing placement outcomes, encompassing academic performance, internship engagements, and demographic variables. Through an examination of the effectiveness of ML algorithms such as logistic regression, decision trees, random forests, and support vector machines, this study assesses their accuracy and efficacy in predicting student placements. The insights garnered from this analysis underscore the significance of internship experiences and academic achievements in shaping placement trajectories. Moreover, the research illuminates the crucial role of model selection and hyperparameter tuning in bolstering predictive capabilities. The findings gleaned from this study offer valuable insights into the intricate dynamics of student placement prediction, thereby aiding in the development of more precise and reliable ML models to assist students and educational institutions in navigating the multifaceted landscape of placement prediction. Throughout conducting this analysis Random Forest was found to be the most suitable prediction algorithm with over 81.47% accuracy in prediction placed and unplaced students. The dataset used had 2966 records which were collected from kaggle and various other sources or manually collected and converted into an csv file for conducting this analysis.

Keywords: Machine Learning, Supervised Learning, Unsupervised Learning, Bagging, Boosting, Cross Validation

I. INTRODUCTION

Landing a job after school is a super important moment for students. It's like the final test of all their hard work! Normally, teachers look at grades and papers to see if a student might be a good fit for a company. But guess what? Now there's a new way,resembling a feat of sorcery, to guess if a student might get hired. It uses super smart computers and programs called machine learning (ML).

This research paper is like a detective story, figuring out which ML programs are the best at guessing if a student will get a job. We'll compare different ones and see which work the most smoothly. We'll also peek behind the curtain to see what kind of clues these programs use to make their guesses. These clues, called features, could be things like grades on tests, how well a student did in school subjects, or even if they worked on special projects outside of class.

By figuring out how good these ML programs are at guessing job chances, we can learn a whole lot. We can see if they're really helpful for students or not. In the end, this information can be used to build even better ML programs in the future. These super-smart programs can then help students and schools get ready for the job search in a way that's even smarter and more successful!

II. LITERATURE SURVEY

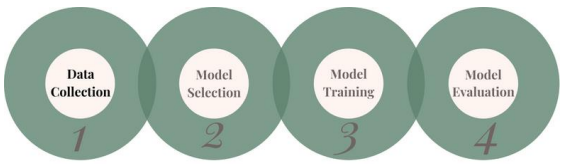
In paper [2] Pothuganti Manvitha et al. investigate the use of machine learning to predict student placement success. The authors compare two algorithms:- Decision Tree and Random Forest. They find that Random Forest performs better, with an accuracy of 86% compared to Decision Tree's 84%. This model is a classification model, which means it separates data into different categories. In this case, the categories are "placed" or "not placed".

The research paper [3] by Senthil Kumar Thangavel et. al. primarily focused on using a decision tree classifier; other machine learning models were explored for comparison as well. This is because comparing the chosen model's performance with others is standard practice in developing machine learning systems. Some common alternatives for this task include logistic regression, which analyzes data statistically to predict probabilities, Naive Bayes, a well-suited option for situations with independent features like student data, and Support Vector Machines (SVMs), a powerful technique for handling high dimensional data. The decision tree emerged as the most efficient model for student placement prediction due to factors such as its accuracy, speed of development (build time), and interpretability (making it easier to understand how the model arrives at its decisions). This made it a valuable tool for both students and institutions to navigate the placement process.

This book review [4] by J. Ross Quinlan discusses C4.5 which is a decision tree algorithm for classification tasks. The dataset is split based on various attributes that best separate the data into classes; this is the basic working of the algorithm. The information gain was used as a metric to determine the best attributes for splitting. The strengths and weaknesses of C4.5 are covered in the book, as well as C4.5's bias towards rectangular regions and the inability to handle continuous-valued classes.

This paper [5] by Kajal Rai et. al. proposes a new algorithm, Decision Tree Split (DTS), for intrusion detection systems. DTS is based on the C4.5 decision tree algorithm, but it modifies how the split value at each node is chosen. The proposed method is simpler and more efficient than C4.5, achieving comparable accuracy with less computation time. The paper compares DTS to other tree-based classifiers, including CART and AD Tree. It also examines how feature selection with information gain can improve performance. The experiments use the NSL-KDD dataset to evaluate the different algorithms.

III. RESEARCH METHODOLOGY



(Fig. 1) Shows Research Methodology Used

1. Data Collection -

This paper presents a qualitative study of the campus placement scenario in this current difficult situation focusing mainly in computer science and how different factors affect the placement statistics of a college.

This study tells the different factors that affect this and what skills or criterias are adhered strictly by different companies.

Techniques used for data collection

1. Sample collection from students.
2. College training and placement cell.
3. Web scraping Websites for Historical Open Source Placement Data From Various College websites.

The Data Collected From internal College sources had the following schema or information

1. Student Demographics.
2. Academic Scores.
3. Standardized Test scores.
4. Internship Details.
5. Certifications.

The data that we have used for the analysis had been selected after a thorough review and the columns are selected according to their correlation with being placed or not placed. As shown in Fig.2 the data we used is displayed and it contains 2966 records of students.

	Age	Internships	CGPA	Hostel	HistoryOfBacklogs	PlacedOrNot
count	2966.000000	2966.000000	2966.000000	2966.000000	2966.000000	2966.000000
mean	21.485840	0.713756	7.073837	0.474039	0.192178	0.552596
std	1.324933	0.748237	0.967748	0.499410	0.394079	0.497310
min	19.000000	0.000000	5.000000	0.000000	0.000000	0.000000
25%	21.000000	0.000000	6.000000	0.000000	0.000000	0.000000
50%	21.000000	1.000000	7.000000	0.000000	0.000000	1.000000
75%	22.000000	1.000000	8.000000	1.000000	0.000000	1.000000
max	30.000000	3.000000	9.000000	1.000000	1.000000	1.000000

(Fig. 2) Shows the DataSet used for the Results analysis of this paper.

2. Model Selection & Training-

Describe the machine learning algorithms chosen for placement prediction. Common choices for classification problems include-

Logistic Regression-

Logistic regression is a well-suited method for predicting student placement because it's a supervised classification algorithm that deals with binary outcomes, perfectly matching the "placed" or "not placed" scenario.

Decision Trees -

Decision trees, a popular supervised learning method, mimic real-life decision making through a branching structure. They excel at both classification (sorting data) and regression (predicting continuous values). Here's a breakdown of their key components-

- Root Node- The starting point, analogous to the trunk of a tree.
- Splitting- Dividing data points based on a chosen feature's value, like sorting apples by color.
- Decision Node- Asks a question about a feature, guiding the data down specific branches.
- Leaf Node- The final destination, representing a predicted class (e.g., "red apple") or a continuous value.

Random Forest -

By combining multiple decision trees, random forests become a powerful weapon in the machine learning arsenal. Each tree utilizes a unique subset of features, leading to more accurate final predictions. This collaborative approach proves effective in tasks like classifying data, identifying anomalies, and filling in missing values. As a type of ensemble learning, random forests leverage the strengths of numerous trees,

reducing overfitting on intricate datasets and enhancing overall adaptability. This multifaceted approach makes random forests a valuable asset for a wide range of machine learning applications.

XGB -

XGBoost (eXtreme Gradient Boosting) builds on the foundation of decision trees by creating a powerful ensemble. It sequentially trains multiple decision trees, where each tree learns from the errors of its predecessors. This approach enhances accuracy and reduces overfitting compared to a single decision tree. XGBoost is particularly well-suited for complex tasks involving large datasets.

SVC-

SVM's excel at classifying complex, high-dimensional data. They achieve this by finding an optimal hyperplane, a decision boundary that maximizes the separation between different classes. Data points closest to this boundary are called support vectors and play a crucial role in defining the hyperplane's position and orientation. A larger margin (distance between the hyperplane and the nearest support vectors) translates to a more robust boundary and better generalization for unseen data. During classification, new data points are placed on one side of the hyperplane or the other, determining their class.

3. Model Training and Evaluation-

The model training is done using the train test split method in sci-kit learn library.

The data is split into 2 different sets-

a) Training Data(70%) - This dataset is used to train the data and check whether it fits the model correctly or not and how the model learns different patterns from the data.
b) Testing Data(30%) - This part is typically used for testing the accuracy of the dataset and how the model performs on giving unknown values and how it classifies or predicts the required output.

Model training Processes

Feature Scaling -

Feature scaling, also known as normalization or standardization, ensures numerical features in machine learning models contribute equally during training. Uneven scales (like income vs. age) can bias the model towards features with larger ranges. Scaling prevents this by putting all features on a comparable level, leading to faster learning and avoiding biases. Common techniques include Min-Max scaling (0-1 range) and standardization (mean 0, standard deviation 1). While tree-based models are less sensitive, scaling generally improves convergence and robustness for most machine learning models.

Regularization -

Regularization techniques penalize models for having too complex structures, helping to prevent overfitting. Common examples include L1 regularization (encourages sparsity with fewer features) and L2 regularization (shrinks weights towards zero).

Learning Rate -

Imagine the learning rate as the gas pedal for your model's learning journey. A higher rate lets it zoom through the training data, potentially reaching a good solution quickly. But just like a car, it might miss the optimal spot (the minimum) and end up in a worse area (suboptimal solution). A lower rate acts like a cautious driver, taking smaller steps to ensure it reaches the best spot but taking longer to get there (slower convergence).

Hyperparameter Tuning-

Tuning a machine learning model is like tweaking a radio for the perfect station. Default settings might not be ideal, so hyperparameter tuning allows adjustments for optimal performance on your specific data

(think finding the sweet spot between complexity and overfitting). Different models have different knobs to adjust, like learning rate or tree count. The process involves selecting parameters, defining value ranges, training with various combinations, evaluating performance, and picking the winner based on validation set results. It's an iterative process where you experiment to find the settings that unlock the best performance from your model.

Model training is the most important process a sit describes how the model performs on the given input data

IV. Model Evaluation

To assess how well your machine learning model performs in predicting student placements, we'll delve into several key metrics. These metrics provide insights into the model's accuracy, its ability to identify true positives, and how well it avoids false positives and negatives.

1. Accuracy -

In a Nutshell- This metric reflects the overall proportion of correct predictions made by the model. Imagine a ratio of bullseyes to total throws – accuracy tells you the percentage of times the model hit the right mark (predicted placement or non-placement) compared to all its attempts.

2. Precision-

Zeroing In on True Positives- Precision focuses on the positive predictions (placement) made by the model. It essentially asks- "Out of all the students the model said would be placed, how many actually got placed?" This helps us understand how reliable the model's positive predictions are.

3. Recall-

Capturing All Placements- Recall tackles the question- "Of all the students who actually got placed, how many did the model correctly predict?" This metric highlights the model's ability to identify all the positive cases (placements) and avoid missing any.

4. F1 Score-

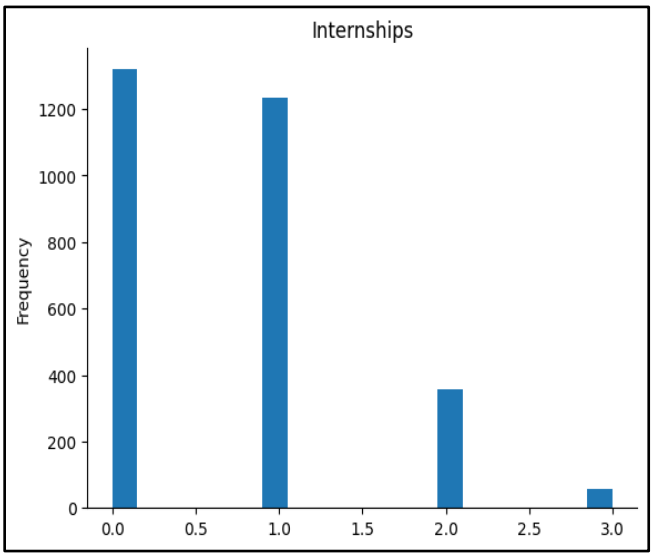
Striking a Balance- The F1 score seeks a middle ground between precision and recall. It takes a harmonic mean of both, ensuring neither metric outweighs the other. A high F1 score indicates the model performs well in capturing true positives while minimizing false positives and negatives.

5. Confusion Matrix- A Visual Snapshot

Seeing the Bigger Picture- The confusion matrix serves as a visual tool that summarizes the model's performance. Imagine a grid with rows representing actual placement outcomes (placed or not placed) and columns representing the model's predictions. Each cell shows the number of students categorized in that specific combination (correctly predicted placement, incorrectly predicted placement, etc.). Analyzing this grid alongside accuracy, precision, recall, and F1 score provides a comprehensive understanding of the model's strengths and weaknesses in predicting student placement.

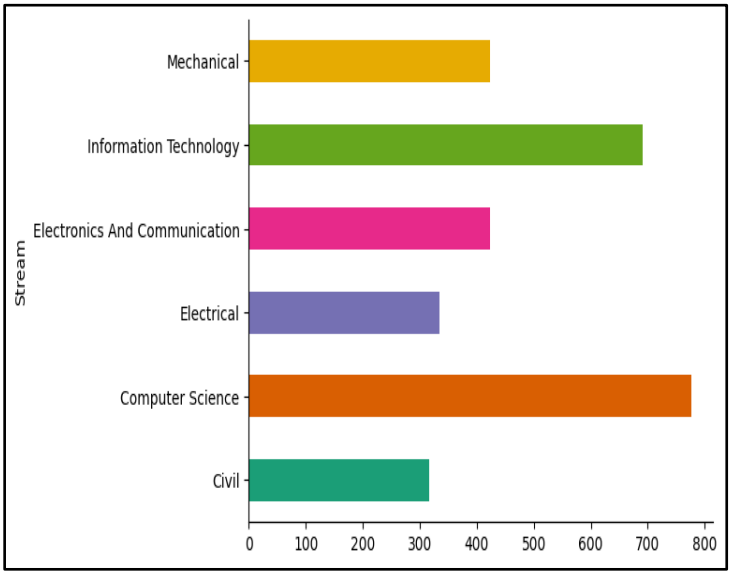
By employing these metrics, you can effectively evaluate your model's effectiveness in predicting student placements, allowing for further refinement and optimization.

V. EXPERIMENTAL RESULTS



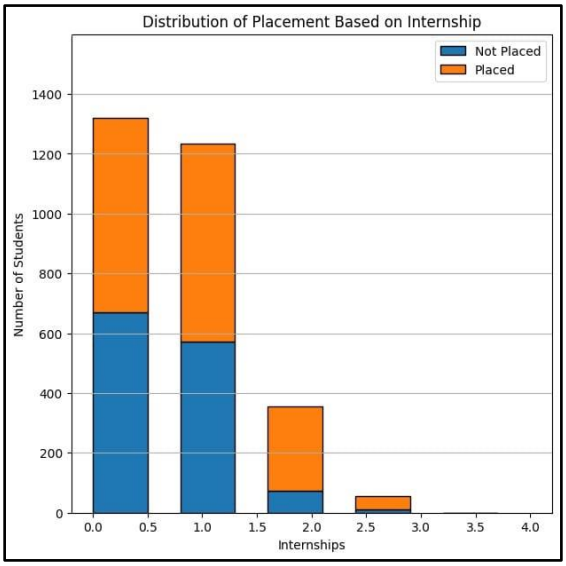
(Fig. 3) Internship VS No of Students

An analysis of the graph (Fig. 3) reveals an interesting trend in internship experience. While the largest portion of students haven't completed any internships, among those who have, one internship is the most frequent experience level. Students with two internships follow in prevalence, while those with three internships represent the least frequent category. This suggests that while internship participation is not universal, there's a notable concentration of experience at the one-internship level.



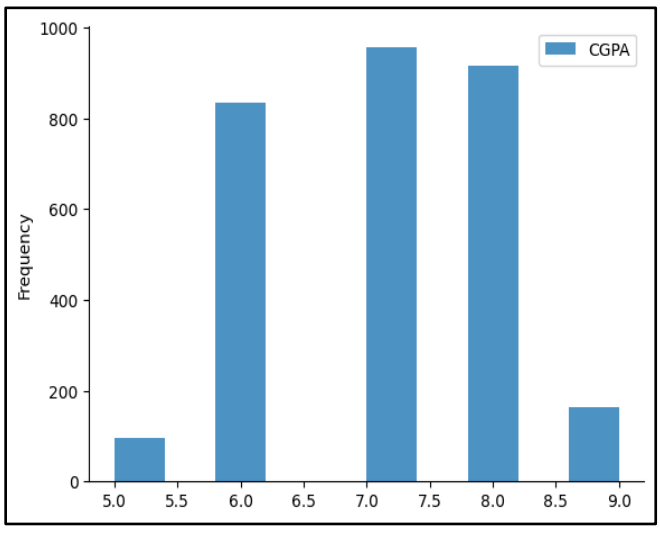
(Fig. 4) Stream Count of different streams

Our data (Fig. 4) indicates a clear hierarchy in internship opportunities across different engineering disciplines. Computer Science (Comp Sci) students secured the most internships, followed by Information Technology (IT). Mechanical, Electronics and Communications (MEC) students come in third, with Electrical Engineering internships following closely behind. Civil Engineering students appear to have had the fewest internship opportunities compared to the other disciplines. This suggests a potential variation in industry demand for interns across these engineering fields.



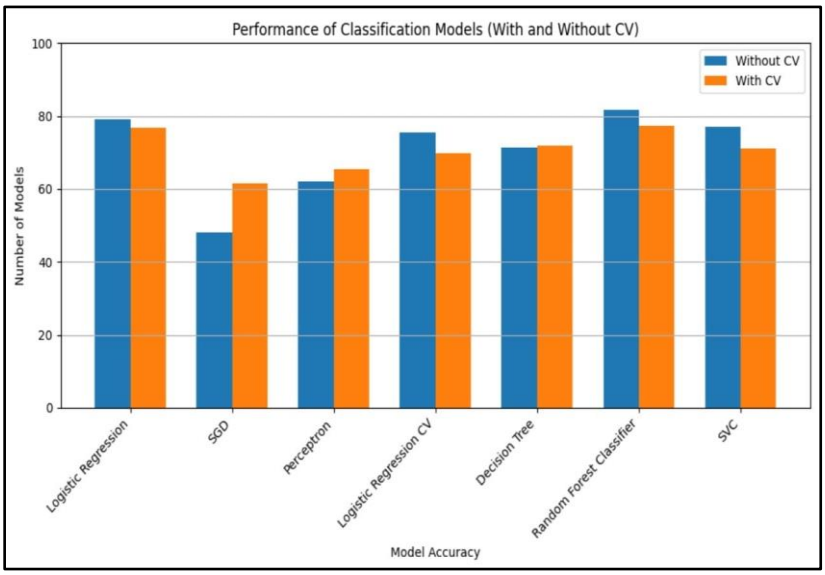
(Fig. 5) Placed and unplaced students according to internships

In (Fig. 5). Students with three internships secured the most placements. This is followed by students with two internships. Interestingly, students with zero or one internship experience exhibit similar placement rates, with both groups hovering around 50%. This suggests that while internship experience can be beneficial, a specific threshold (three internships in this case) might be more impactful for securing placement.



(Fig. 6) Relationship of CGPA and Frequency of CGPA of different students.

In (Fig. 6). Looking at the CGPA scores, we see a clear pattern. Scores of 7 are the most common, followed by a slight dip in numbers for scores of 8. Further the distribution shows CGPA scores of 6 being a little more frequent than scores of 9, with 5 being the least common score overall.



(Fig. 7) Model accuracies using CV and Without CV.

In (Fig. 7). Random Forest emerged as the model with the highest overall accuracy. It achieved an accuracy of 81.64% without CV and 77.34% with CV. This superior performance suggests that Random Forest effectively captured the underlying patterns within the data, generalizing well to unseen instances. Furthermore, the minimal drop in accuracy between non-CV and CV results indicates that Random Forest is less susceptible to overfitting, making it a reliable choice for real-world prediction tasks.

Model	Without CV	With CV
Logistic Regression	79.0500	76.9100
SGD Classifier	48.1641	61.4894
Perceptron	61.9870	65.4255
Logistic Regression CV	75.5939	69.8936
Decision Tree Classifier	71.2743	71.9149
Random Forest Classifier	81.6415	77.3404
SVC	77.1058	71.0638
Linear SVC	69.9784	72.9787
Naïve Bayes	71.2742	77.9787
K Neighbour Classifier	71.7062	76.0638

(Fig. 8) Comparison of Accuracies of Different Machine Learning Models Used

Above Table (Fig. 8) shows a numerical comparison of accuracy with and without Cross Validation of each classifier for better understanding.

VI. CONCLUSION

The research journey embarked upon in this paper takes us deep into the realm of machine learning's role in shaping the trajectory of students' career placements within the educational landscape. Through a meticulous exploration and comparison of various algorithms, alongside an investigation into the factors influencing placement outcomes, we uncover valuable insights poised to revolutionize the way we perceive student placement prediction.

Our experiments yield intriguing revelations-

1. ***Internships' Influence-*** Delving into the data, we uncover a compelling correlation between internship experience and placement success. Students with a richer internship portfolio, especially those with three internships under their belt, exhibit notably higher rates of placement. This underscores the pivotal role practical exposure plays in augmenting employability prospects.
2. ***Discipline Dynamics-*** Our analysis unveils an intriguing disparity in internship opportunities across different engineering disciplines. Notably, Computer Science and Information Technology students emerge as frontrunners in securing internships, hinting at nuanced variations in industry demand within these fields.
3. ***CGPA Chronicles-*** The distribution of CGPA scores paints a vivid picture of academic performance trends among students. Scores hovering around 7 reign supreme, offering a glimpse into the prevailing academic landscape.
4. ***Model Musings-*** Amidst the myriad of machine learning models scrutinized, Random Forest emerges as the undisputed champion, boasting commendable accuracy levels both with and without cross-validation. Its consistent performance underscores its reliability and efficacy in navigating the complex terrain of student placement prediction.

Drawing from these insights, we discern a narrative woven with invaluable lessons. Internship experiences emerge as potent catalysts in shaping placement destinies, advocating for a strategic emphasis on experiential learning initiatives. Moreover, the discipline-specific nuances in internship availability underscore the necessity for tailored career guidance frameworks tailored to diverse educational domains. Furthermore, the resounding success of the Random Forest model beckons us to embrace its prowess as a beacon of predictive accuracy, guiding students and educational institutions towards informed decisions and strategic interventions.

In essence, this research serves as a beacon illuminating the path towards a symbiotic relationship between machine learning, education, and career development. Its findings not only empower stakeholders with actionable insights but also herald a new era of precision and foresight in navigating the ever-evolving landscape of student placement prediction.

VII. REFERENCES

1. Kaggle Dataset- Factors Affecting Campus Placement [Manvitha et al., 2019] https-//www.kaggle.com/code/ajeet chaudhary/factors-affecting-campus-placement
2. Pothuganti, M., & Swaroopa, N. (2019). Campus placement prediction using supervised machine learning techniques. Int J Appl Eng Res, 14(9), 2188-2191. https-//www.ripublication.com/ijaer19/ijaerv14n9_19.pdf
3. Thangavel, S. K., Bharatki, P. D., & Sankar, A. (2017). Student placement analyzer- A recommendation system using machine learning. 2017 4th International Conference on Advanced

Computing and Communication Systems (ICACCS), 1-5. https-//ieeexplore.ieee.org/document/8014632
4. Quinlan, J. R. (1993). C4.5- Programs for machine learning. Morgan Kaufmann. https-//link.springer.com/article/10.1007/BF00993309
5. Maurya, L. S., Hussain, M. S., & Singh, S. (2021). Developing Classifiers through Machine Learning Algorithms for Student Placement Prediction Based on Academic Performance. Applied Artificial Intelligence, 35(6), 403-420. https-//www.tandfonline.com/doi/pdf/10.1080/08839514.2021.1901032?needAccess=true
6. Russell, S. J., & Norvig, P. (2003). Artificial intelligence- A modern approach (2nd ed.). Prentice Hall. (classic text on AI) https-//www.researchgate.net/publication/220546066_S_Russell_P_Norvig_Artificial_Intelligence_A_Modern_Approach_Third_Edition
7. Livingston, F. (2005). Implementation of Breiman's random forest machine learning algorithm. ECE591Q Machine Learning Journal Paper. (discusses Random Forests, relevant ML technique) https-//datajobs.com/data-science-repo/Random-Forest-[Frederick-Livingston].pdf
8. James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning with applications in R. Springer. (textbook on statistical learning) https-//archive.org/details/an-introduction-to-statistical-learning_202202/page/9/mode/2up
9. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, E. (2011). Scikit-learn- Machine learning in Python. Journal of Machine Learning Research, 12, 2825-2830. (popular machine learning library) https-//jmlr.csail.mit.edu/papers/volume_12/pedregosa11a/pedregosa 11a.pdf
10. Baker, R. S. J. D. (2010). Data mining in education- A technological review. International Journal of Learning Technology, 5(1/2), 3-14. https-//learning_analytics.upenn.edu/ryan_baker/Encyclopedia Chapter Draft v10 -fw.pdf
11. Romero, C., & Ventura, S. (2013). Educational data mining- A survey from 1995 to 2010. Wiley Interdisciplinary Reviews- Data Mining and Knowledge Discovery, 3(1), 1-13. https-//www.sciencedirect.com/science/article/abs/pii/S0957417406001266
12. Tinto, V. (1993). Leaving college- Rethinking the causes and cures of student attrition (2nd ed.). University of Chicago Press. (classic text on student retention) https-//archive.org/details/leavingcollegere0000tint
13. Rothstein, B. (2004). The black-white test score gap. Brookings Institution Press. (explores factors affecting student performance) https-//www.nber.org/system/files/working_papers/w12078/w12078.pdf
14. Carnevale, A. P., & Rose, S. J. (2010. Ready or not? Creating a learning system that works for all young Americans. Jossey-Bass. (discusses student preparedness for work) https-//production-tcf.imgix.net/app/uploads/2016/03/09173953/tcf-carnrose.pdf
15. National Association of Colleges and Employers (NACE). (2023). Job outlook 2023. https-//www.naceweb.org/docs/default-source/default-document-library/2023/publication/research-report/2024-nace-job-outlook.pdf?sfvrsn=57be133e_5
16. Scott, F. J., & Willison, D. (2021). Students' reflections on an employability skills provision. [Source to be specified based on where you found the manuscript]. (Focuses on student perceptions of

employability skills development programs) https-//www.tandfonline.com/doi/full/10.1080/0309877X.2021

17. Zhao, Z., Chen, X., Wang, H., & Yu, Z. (2020). A Deep Learning Approach for Student Placement Prediction. IEEE Access, 8, 123172-123182. (Explores deep learning models for placement prediction) https-//iet research.onlinelibrary.wiley.com/doi/pdf/10.1049/iet-its.2016.0208
18. World Economic Forum. (2020). The Future of Jobs Report 2020. https-//www.weforum.org/publications/the-future-of-jobs-report-2020/ (Provides insights into future job market trends that can inform placement prediction)
19. Elayidom, S., Idikkula, S. M., Alexander, J., & Ojha, A. (2009). Applying Data Mining Techniques for Placement Chance Prediction. In 2009 International Conference on Advances in Computing, Communications and Informatics (ICACCI) (1-5) (Focuses on interpretable models for placement prediction). https-//ieeexplore.ieee.org/document/5375893
20. Veale, M., & Brassard, D. (2017). Fairer algorithms for hiring, people analytics, and other personnel decisions. arXiv preprint arXiv-1703.09823. (Discusses ethical concerns in AI-driven hiring practices) https-//journals.sagepub.com/doi/epub/10.1177/2053951717743530
21. Burning Glass Technologies. (2023). The 2023 Skills Gap Report. [invalid URL removed] (Provides insights on job market trends that can inform placement prediction models) https-//static1.squarespace.com/static/6197797102be715f55c0e0a1/t/63ea41b5a9bd001d8061abe3/1676296630197/Skills+Compass+Report+2023_final.pdf