Optimizing Talent Acquisition: A Data-Driven Approach to Predicting Job Applicant Employability

Team #5: Ziyu Chen, Xiao Luo, Xinyuan Qian, Yu Wang, Kunyi Xia

Introduction

In the quest to revolutionize recruitment, our project leverages data-driven techniques to enhance the accuracy of employment decisions. We utilize machine learning to analyze a dataset rich in candidate attributes, aiming to identify key factors influencing employability. This approach addresses the biases in traditional hiring, offering a more objective and efficient methodology for employers. Key findings reveal the significant impact of 'Computer Skills' and 'TypeScript' on hiring outcomes, using models like XGBoost and Logistic Regression. The project thus serves dual purposes: aiding employers in refining their recruitment strategies and guiding job seekers in aligning their skills with market needs.

Problem Definition

1. Objective from the Employer's Perspective:

The primary objective of this project is to develop a predictive model that assists employers in the preliminary screening of job applicants. The model aims to identify candidates who are most likely to be suitable for employment, thereby enabling companies to focus their interviewing efforts on the most promising applicants. This approach is intended to streamline the recruitment process by efficiently filtering out candidates who are unlikely to be selected for the position.

To minimize missed qualified candidates, is to focus on the recall value as the metrics. In this scenario, a false negative occurs when a qualified candidate is incorrectly predicted as 'not employable' by the model. High recall ensures that the model correctly identifies as many

qualified candidates as possible, reducing the risk of overlooking suitable applicants. Missing out on good candidates can be a significant loss for a company, both in terms of talent acquisition and the potential value these individuals could bring to the organization.

2. Objective from the Applicant's Perspective:

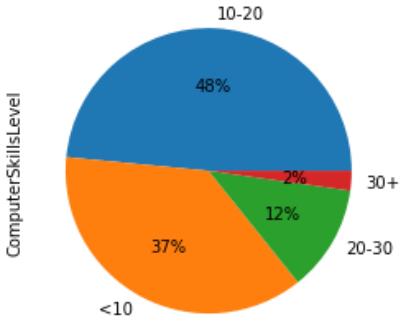
In addition to aiding employers, the model is designed to provide valuable insights for job seekers. Another key goal is to identify and quantify the impact of various features on the likelihood of employment. By understanding which attributes most significantly influence the classification decision, applicants can better understand how to enhance their profiles and qualifications to increase their chances of being hired.

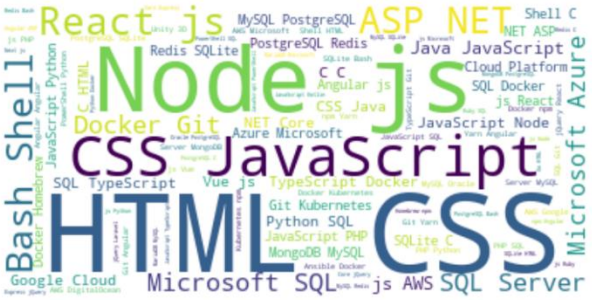
Background

In the competitive job market, students and professionals are keen to align their skills with employers' evolving demands. Our project applies machine learning to employment data, aiming to identify key employability factors. This initiative is designed to help job seekers, especially data science candidates, to enhance their marketability and assist employers in efficient recruitment, reflecting the dynamic needs of the industry.

Dataset and Exploratory data analysis

We selected a real-world dataset about the employability of job applicants from Kaggle (AyushTankha, 2023), originally collected from a variety of sources including job portals, career fairs, and online applications. It covers a broad spectrum of industries, job positions, and qualifications to ensure data reliability. This dataset contains over 73000 samples and each sample has 13 attributes and target variable of employed or not. Considering most of these features are categorical with few categories, we decide to use hard encoder to encoding categorical variables, i.e., transfer Other to 0, NoHighEr to 1, Undergraduate to 2, Master to 3, PhD to 4. The country column has too many different countries which may impose unknown effect on the model, so we drop country feature in our model and consider the dataset as a global job market.





To get insights on the distribution of data for a specific column, we draw several pie charts. The pie chart (upper left graph) represents the computer skills level of individuals within the dataset, categorized into four groups based on the number of skills. The largest segment, comprising 48% of the dataset, has a skill range of 10 to 20.

We then calculated the correlation matrix for the training dataset to gain insights into the relationships between the features and the target variable 'Employed'. This also helps future feature selection when training models.

After we found that ComputerSkills play a dominant role in the model, we decide to come back to dig deeper into the HaveWorkedWith column. For better comparing different machine learning models in the next step and research the impact of specific skills on the results, we first extract a list of skills from the HaveWorkedWith column. We also obtained a word cloud graph (upper right graph) to get insights on what skills are most popular. Then we count the occurrences of each skill to identify the ten most common skills and update the dataset by marking the presence (1) or absence (0) for each individual binary columns.

After that, we split the dataset into three separate subsets for the purposes of machine learning: training, validation, and testing. The ratios for the split are set as follows: 70% for training, 15% for validation, and 15% for testing.

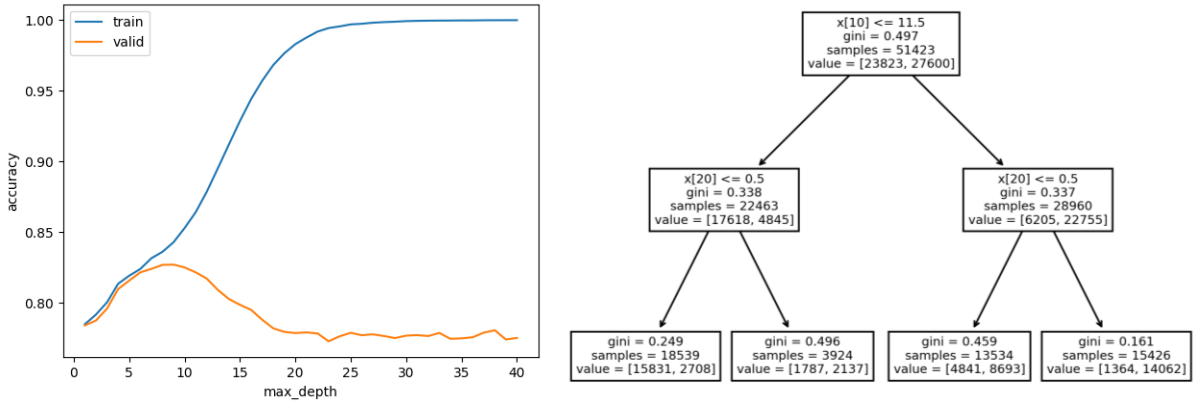
Methods and Models

(1) **Decision tree**

The most obvious advantage of using a decision tree is that it does not discriminate between categorical and numerical values of features (Sivaram & Ramar, 2010), since our dataset contains both. Also, the model is flexible since we can limit the height of the tree to

prevent overfitting. A potential concern is that since there are so many factors acting on the result, it might be difficult to build a tree with a reasonable height.

Using the extended table with the top 10 skills, we test different depths of trees and plot the corresponding accuracy on the training set and validation set. The figure nicely demonstrates the overfitting when the tree gets deeper; In fact, the training accuracy goes all the way to 100% since we have an abundance of features, but for the validation set the accuracy peaks at 82.72% when the depth is 8, then falls again. An example of the tree is also shown.



Decision trees also provide a measurement of how discriminating a particular feature is: The closer to the root it appears, the more information is gained by dividing on this feature, hence the feature is more important. As is shown in the tree graph, the root splits on the "ComputerSkills" feature, which is the number of skills an applicant has. Then it continues to split on the acquisition of skills including TypeScript, JavaScript, SQL and Git. Based on this analysis it indicates that the number of skills is the most important feature to affect the hire ability, and those previously mentioned skills are the most valuable to have.

(2) Logistic regression

The logistic regression model, tailored for our binary classification objective, has demonstrated its efficacy, particularly in cases where there's a linear or nearly linear relationship between features and the response variable. Our dataset indicates a significant trend: applicants with longer coding experience, whether in professional or non-professional capacities, tend to have higher admission odds. This observation underscores a potential linear correlation between coding tenure and admission likelihood, making logistic regression a fitting choice for our analysis.

In our initial model, we focused on four predictors—ComputerSkills, Age, MainBranch, and EdLevel—identified from the correlation matrix as having the most substantial impact on employment status. This model achieved a 78% accuracy on the test set. The recall metrics, at 78% for the test set, underscore the model's proficiency in accurately identifying relevant cases.

The second model, however, showed a moderate accuracy level—68% on the test set—but excelled in recall, with scores of 86%. This implies a strong ability to minimize false negatives, although at the cost of overall accuracy.

In the third iteration, we experimented by excluding the ComputerSkills feature. This adjustment resulted in a slight improvement, with the test set exhibiting accuracies of 77% and a recall rate of 79%. This suggests that while ComputerSkills is a significant predictor, its absence doesn't substantially diminish the model's effectiveness as we also have separate skills present in the model.

The fourth model was developed using Recursive Feature Elimination (RFE) for a more nuanced feature selection. The RFE analysis pinpointed four critical predictors, leading to a model that demonstrated consistent accuracy—76% for both test and validation sets.

In conclusion, based on the performance metrics, the second logistic regression model appears to be the most effective, particularly due to its high recall rate of 86%.

In addition, we checked the coefficients of the model as the feature importance. The following table compares the feature importance given by the second and third model. The result of third model gives a top five computer skills, which again tells us that the total number of computer skills are the most important.

	with ComputerSkills		without ComputerSkills	
Rank	Feature	ABS Coefficient	Feature	ABS Coefficient
1	ComputerSkills	0.12326	TypeScript	1.9579
2	YearsCode	0.03023	JavaScript	0.98781
3	YearsCodePro	0.01511	SQL	0.89785
4	EdLevel	0.00897	MySQL	0.48183
5	TypeScript	0.00868	HTML/CSS	0.47379

(3) RandomForest

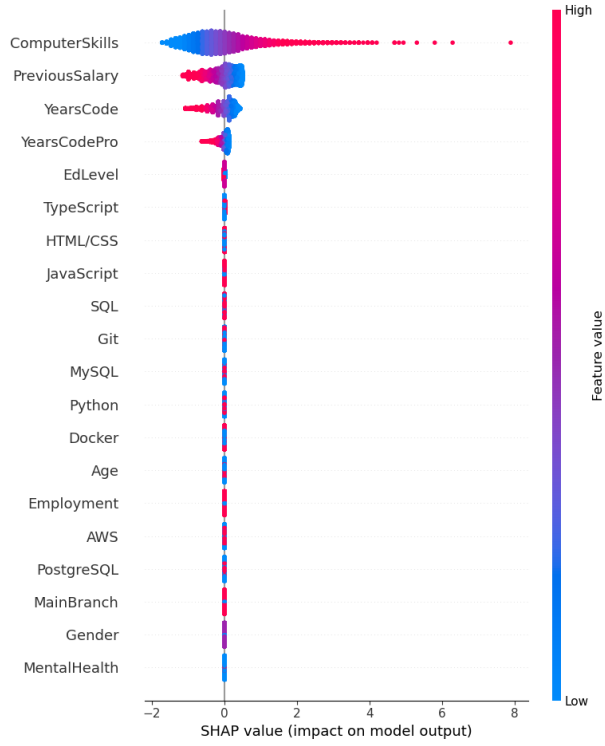
Random Forest, an ensemble of decision trees, inherently reduces the risk of overfitting—a common challenge with individual decision trees, especially when they grow

deep. This quality ensures more reliable and generalizable results. Furthermore, Random Forest typically exhibits higher accuracy (see Conclusions), as it aggregates predictions from numerous trees, capturing more nuances in the data than a single tree could. We also checked the feature importance given by the model which is based on how much each feature decreases the impurity in the nodes of these trees.

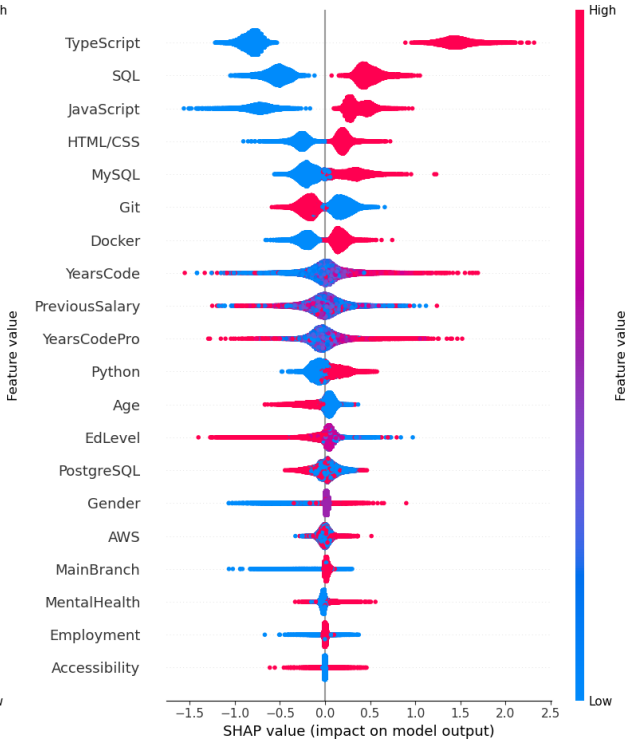
(4) XGBoost

We have incorporated XGBoost (EXtreme Gradient Boosting) due to its superior predictive accuracy, often surpassing other machine learning algorithms. XGBoost operates on a gradient boosting framework, constructing a series of decision trees sequentially. Each subsequent tree in the series is fine-tuned to address the errors made by its predecessors, enhancing the model's overall predictive power. The standard XGBoost model gives a recall value of 0.86.

(5) Feature Importance using SHAP



SHAP Results from
Second Logistic Regression Model



SHAP Results from XGBoost Model Dropping
ComputerSkills

SHAP (SHapley Additive exPlanations) is a Python-based tool designed to interpret machine learning model outputs, drawing inspiration from cooperative game theory ((Lundberg &

Lee, 2017)). It provides an additive explanatory model where each feature contributes as a 'player' in the prediction. SHAP values quantify each feature's contribution to a particular prediction, indicating whether the effect is positive or negative on the predicted outcome.

In our project, we utilize SHAP to go beyond traditional feature importance, gaining a deeper understanding of how each feature influences individual predictions. For instance, a positive SHAP value for a feature indicates an increase in the likelihood of a positive outcome, such as employability, while a negative value suggests a decrease. SHAP's visualizations offer a 1nuanced view, displaying each feature's Shapley value on the x-axis and aligning features along the y-axis based on their importance. The color coding further helps to discern how variations in feature values affect predictions. This approach enables us to not only identify key features but also understand the direction and magnitude of their impact, thereby enhancing the interpretability and transparency of our predictive models.

From the upper left graph, the SHAP summary given by the 2nd Logistic Regression model, we can see that high ComputerSkills have positive impact on the model, while higher value of PreviousSalary impose negative impact on the prediction result. Intriguingly, our findings challenge conventional assumptions: a higher 'Previous Salary' surprisingly corresponds to a lower likelihood of employment. This insight suggests that factors such as salary expectations or perceived overqualification could be at play, impacting an applicant's chances of being hired. From the upper right graph, the SHAP summary given by XGBoost model with the ComputerSkills column dropped, we find almost all the separate skills impose significant positive impact on the decision, except the Git skill having negative effect, which is hard to explain. Also, the previous coding experience is also remarkably important.

Conclusions

The summary of results from four models are given in the following table:

Model	Recall	Accuracy	1st feature	2nd feature
DecisionTree	85%	83%	ComputerSkills	TypeScript
Logistic Regression	86%	68%	ComputerSkills	YearsCode

RandomForest	85%	84%	ComputerSkills	TypeScript
XGBoost	86%	84%	ComputerSkills	TypeScript

Model Selection and Performance:

Our comprehensive analysis led us to compare various predictive models, with a specific focus on achieving high recall values. Notably, both Logistic Regression and XGBoost models demonstrated impressive recall scores of 0.86. However, upon evaluating overall performance metrics, we recommend adopting the XGBoost model for predictive tasks. This recommendation is rooted in XGBoost's superior accuracy score, marking it as the most reliable model among those tested.

Insights on Feature Importance:

A critical aspect of our analysis involved understanding feature importance across different models. A consistent observation emerged, highlighting the prominence of 'Computer Skills' in influencing admission decisions. This feature was ranked as the most significant by three out of the four models evaluated, underscoring its paramount importance. Additionally, 'TypeScript' was identified as the second most influential feature. From these findings, two key insights emerge for job seekers, particularly those in computer science and related fields:

Firstly, the paramount importance of computer skills suggests that acquiring a broader range of such skills could significantly bolster a candidate's employability. Continuous learning and skill development, therefore, should be a focal point for aspirants in this competitive job market.

Secondly, another significant revelation is the prominence of 'TypeScript' as the second most influential feature. This highlights the growing importance of specific technical skills in the job market, particularly in the tech industry. TypeScript, a modern language building on JavaScript, is evidently a valuable skill set, resonating with current industry trends and demands.

In summary, our analysis through advanced predictive modeling techniques provides actionable insights for job seekers while offering a robust tool for employers to streamline their recruitment process. The findings accentuate the importance of skill acquisition and professional experience in the dynamic landscape of employment.

References

AyushTankha. (2023, July 10). *70K+ job applicants data (human resource)*. Kaggle. https://www.kaggle.com/datasets/ayushtankha/70k-job-applicants-data-human-resource/

Lundberg, S., & Lee, S.-I. (2017, November 25). *A unified approach to interpreting model predictions*. arXiv.org. https://arxiv.org/abs/1705.07874

Sivaram, N., & Ramar, K. (2010). Applicability of clustering and classification algorithms for recruitment data mining. *International Journal of Computer Applications, 4*(5), 23–28. https://doi.org/10.5120/823-1165