Matt Nelson

Final Report:
Literary Award Winnings Analysis

Problem Statement

There has long been a debate about nepotism within the literary community. Depending on who you know or what graduate program you attended, you may or may not succeed as a writer. Whether it leans to the academic side dealing with professorships and tenure, or to the publishing side for book deals and publicity, many authors are left wondering why not them. Juliana Spahr, Stephanie Young, and Claire Grossman set out to statistically analyze one route of success for writers, literary awards, through a racial lens. They found that there was a correlation between elite education and winners of major prizes, and, within the past few years, 80% of recipients had a graduate degree. The problem is if a graduate degree makes it more likely for you to win an award, is it worth the cost of tuition to go?

By using the same data set, made available through the Post45 Data Collective at Emory University, I created a machine learning model to help someone calculate their expected earnings based on features like specific MFA and elite college status, but also not having an MFA, if they wrote poetry or prose, and if they were male or female among a few other features.

After expanding and then reducing to 20 features, I tuned my Random Forest Regressor model and achieved an R^2 score of around 70%. This process can be used for anyone thinking about going into an MFA and wondering if they too could surpass the average hardscrabble life of an author.

Data Wrangling

Rows, columns, what dropped, null values, final shape

The raw data set had 7133 entries with 18 features, but this dataset included both judges and winners of literary awards of at least $10,000. I quickly found that there were around 1.5x more judges than authors (4268 judges: 2776 winners). For my purposes, the winner of literary awards was more important than who the judge was so I

split the data frame into authors and judges. From there, I started looking for duplicates within the author dataframe and cleaning the few that I found. I also noticed that within the “elite_institution” feature, although over 20% were connected to Harvard upon first look, the observations also included instances of multiple elite institutions.

I found that Iowa was by far the higher education institute with the most literary prizes (Figure 1), which fact was held across winners and judges with normalized values (Figure 2). 31% of the prizes were for Poetry, 25% were for prose, and 44% had no or were unrelated to genre.

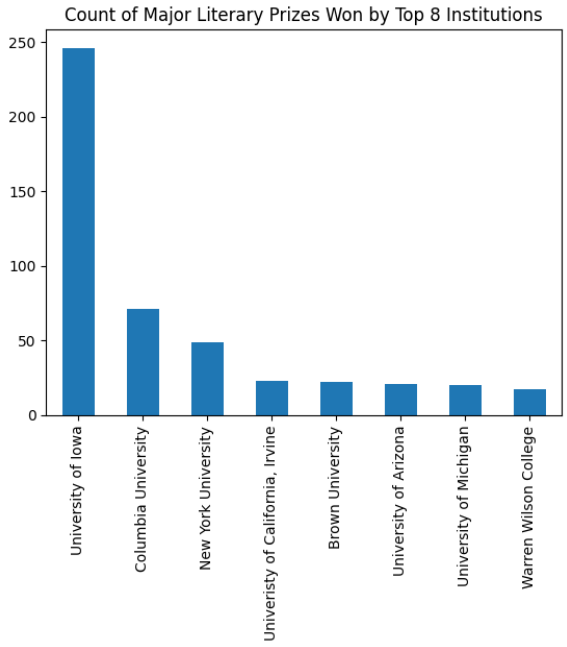

Figure 1: Frequency Count of Literary Awards Won by Top 8 Institutions

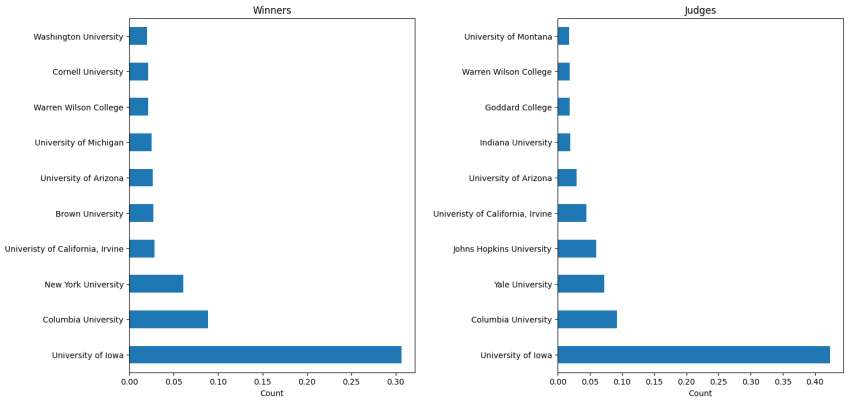

Figure 2: Standardized Winner and Judge Counts for Top 10 Institutions

Exploratory Data Analysis

While beginning the EDA process, I realized I still needed to deal with null values. My first step was to replace any instances of ‘None’ with NaN. Later, I removed the ‘stegner’ feature, which only had 124 non-null values out of 2776, and replaced any missing data in the ‘mfa_degree’ with ‘no_mfa’, and encoded ‘graduate_degree’ to either be 1 for degree or 0 for no degree instead of ‘graduate’ and NaN. After some cursory research, I also cleaned up a few records where the gender was unknown or there were duplicate entries with authors having different elite institutions.

To fix the problem of the ‘elite_instituion’ feature having values with multiple colleges, I iterated through those observations and split them into three features since the most colleges any one person had was three.

Finally, I created a new dataframe grouped by author name that summed the amount of awards they won, as well as the prize_amount. The five authors with the most money were John Keen ($1,010,000), Claudia Rankine ($945,000), Adrienne Rich ($920,000), A.R. Ammons ($900,000), and Edward P. Jones ($835,000). The first three of these top five have affiliations with elite institutions, and Keen, Rankine, and Jones all have MFA degrees. The MFA institution that has accrued the most money is by far the University of Iowa with $13,394,000. If we take every MFA program not in the top 10 and sum their awards, they clock in at $13,268,800 (Figure 3).

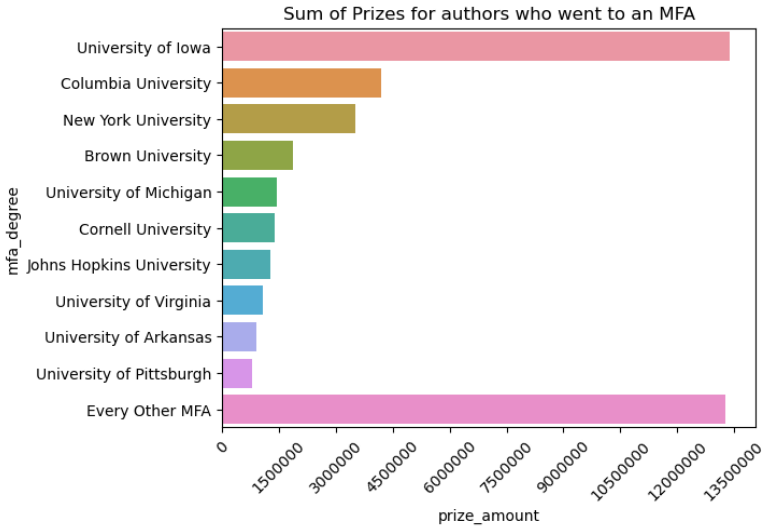

Figure 3: Top 10 Sum of Award Winnings for MFA Programs plus all other MFA Programs combined

For my numerical features, I created a heatmap of correlation and found that the highest amount of correlation (0.57) was between the number of awards and the sum prize amount (Figure 4).

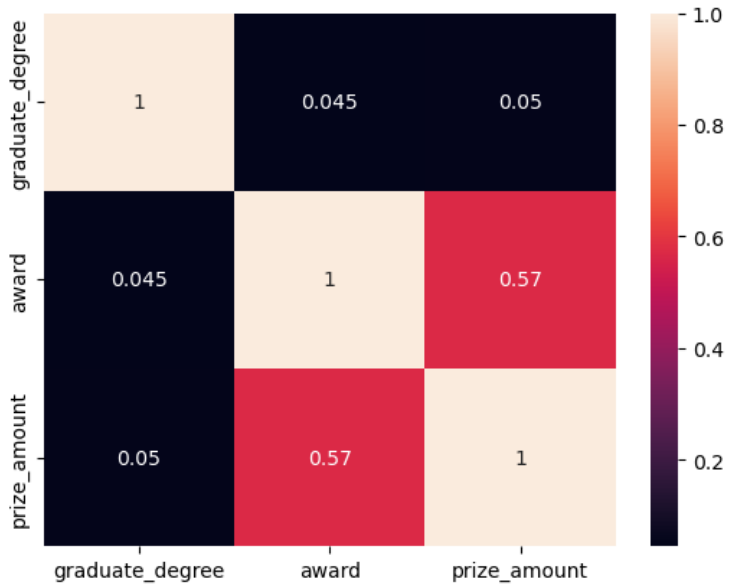

Figure 4: Heatmap between Graduate Degree, Number of Awards, and Sum of Prize Amounts

In-depth Analysis

To pre-process my data, I used one-hot encoding to split the genre feature into poetry, prose, and no genre features, but I ended up dropping any observation with no genre. I also split the award type into book and career. I scaled my number of awards and prize award features using the Standard Scaler method.

After splitting the data into 80% training and 20% testing sets, I created my first graph using a simple Linear Regression machine learning model looking at scaled literary awards won vs scaled prize money won over a career (Figure 5). The coefficient of determination between the predictor and the target was 0.31. I thought that maybe it would be higher since one could assume the more awards you win, the more money you'd make. The range of award purses went from a minimum of $10,000 up to $625,000, and respectively, for those price ranges, there were 1069 winners compared to 20.

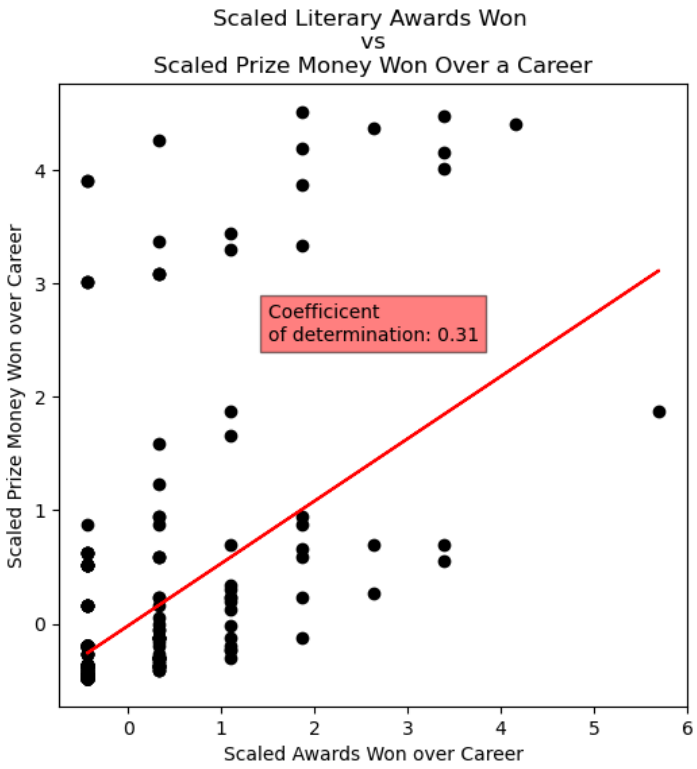

Figure 5: Scaled Literary Awards Won vs Scaled Prize Money Won

Model Selection

I decided to try out four different regression models: Random Forest Regressor, a Decision Tree Regressor, Extra Tree Regressor, and XGBoost Regressor. My initial Random Forest Regressor had 100 estimators and had a mean squared error of 0.71 and an R^2 score of 30.0% (Figure 6). The R^2 score seemed low, so I decided to try a simple Decision Tree.

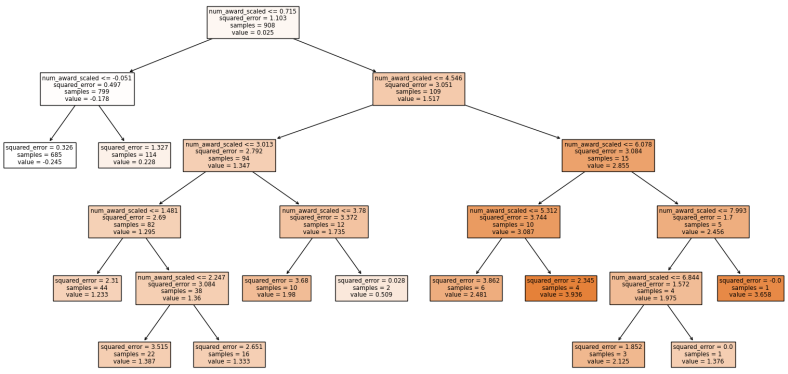

Figure 6: First Random Tree Regressor

The Decision Tree Regressor was out of the box, and it definitely overfit with a MSE of 0.0002 and an R^2 score of 99.98%. I decided to try an out-of-the-box Extra Tree Regressor because of the lower processing cost and quicker turn around than the Random Forest. I was most interested in finding the most important features. After my initial finding of the Extra Tree Regressor, I found that the top five features were 'prize_amount-scaled', 'prize_amount', 'award', 'num_award_scaled', and 'Type_career' (Figure 7). This was of course wrong because it had both the scaled and unscaled features of amounts and number of awards. I went back and dropped the unscaled features and re-split my testing and training data and did another top five features (Figure 8). This info about top features was what I was looking for, but the MSE (0.87) and R^2 score (14.4%) seemed to be lacking.

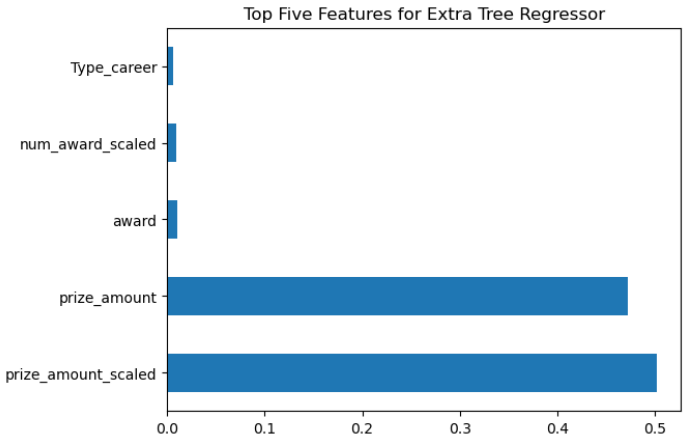

Figure 7: First Top Five Features for Extra Tree Regressor

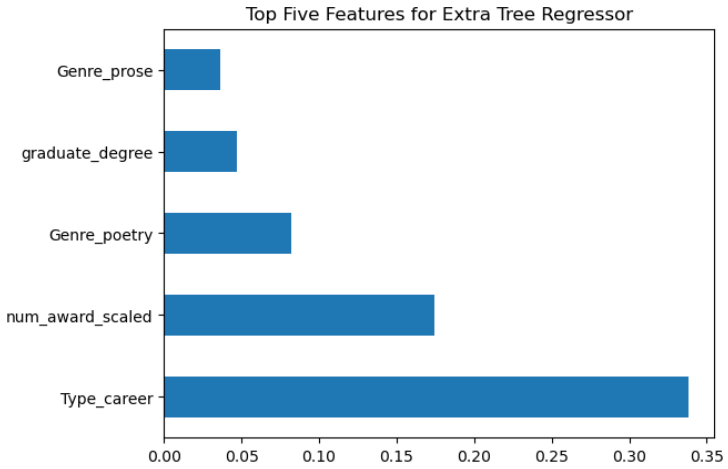

Figure 8: Second Top Five Features for Extra Tree Regressor

That said, I did look closer into the split between career awards and book

awards. There were 718 book awards with a sum total of $12,960,000 and there were 2058 career awards with a sum total of $123,359,800. It makes sense that career awards are so important due to the fact that its sum is almost 10 times more than the sum of book award winnings.

I wanted to see if that tracked with the Random Forest Regressor. Using the newly split data, with 100 estimators, my model had an MSE score of 0.77 and an R^2 score of 24.2%. The most important feature was also the career type, but included new ones like poetry, number of awards scaled, graduate degree and then prose (Figure 9).

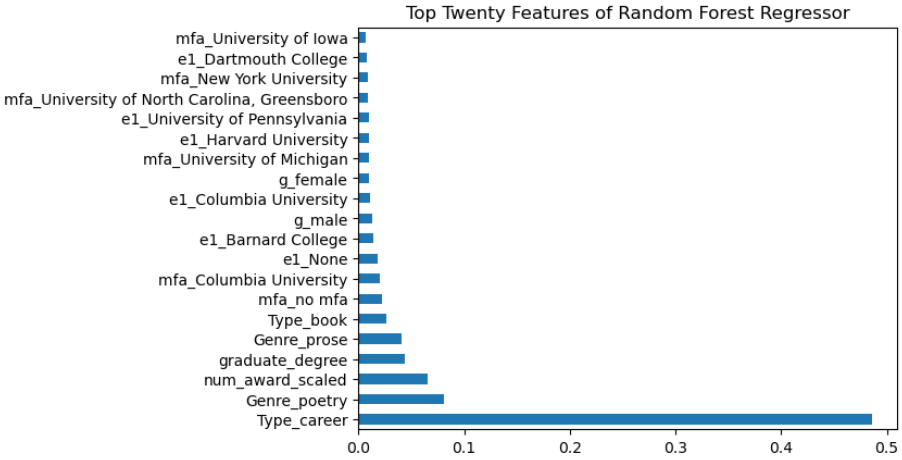

Figure 9: Top Twenty Features of Random Forest Regressor featuring career type awards as the number one feature.

My final model was an XGBoost Regressor with an objective of ‘reg:squarederror’ and 100 estimators. It had an MSE of 0.87 and an R^2 score of 14.5%. It wasn’t much better than the Extra Tree Regressor, but it also had a different set of important features including more elite connections and MFA programs (Figure 10).

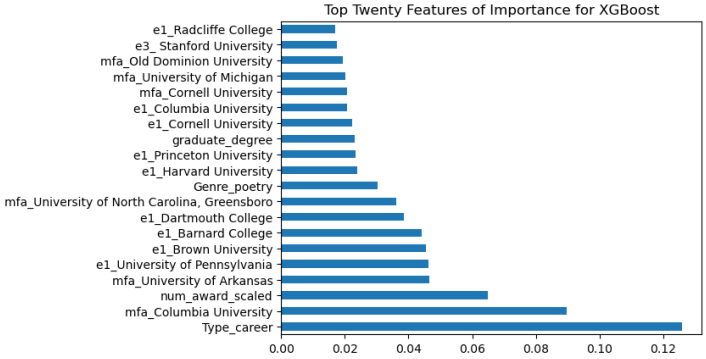

Figure 10: Top Twenty Features of XGBoost Regressor

I decided to tune the Random Forest using first cross validation to see which set of features gave the best R^2 score. It turned out that between using the top 5, top 10, top 20, and all features, using the top 20 had the best cross validation score of 0.66. I then used GridSearchCV to find the hyper parameters with the highest R^2 score.

Takeaways

The best model had a max depth of 10, a minimum samples leaf of 2, a minimum samples split of 20 which provided a MSE of 0.69 and a R^2 score of 31.7%. Looking at the residuals plot, there's a large cluster near the y=0 line, but there are many data points on both sides of the line which means the predictions were both above and below the ground truth (Figure 11). The feature importance top five scores were career type award (0.71), poetry genre (0.07), num of awards scaled (0.04), graduate degree (0.02), and prose genre (0.01) (Figure 12).

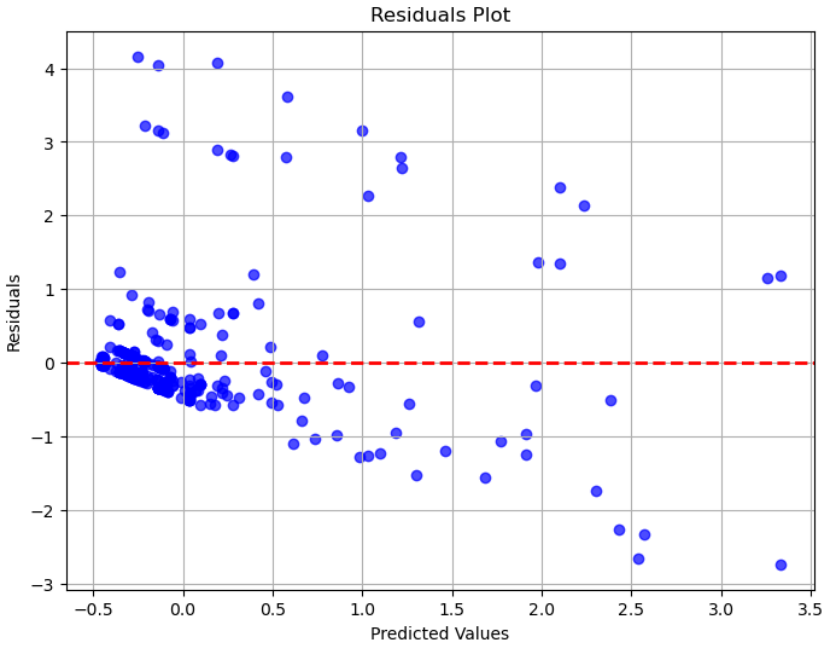

Figure 11: Residual plot of Predicted Values and their residuals.

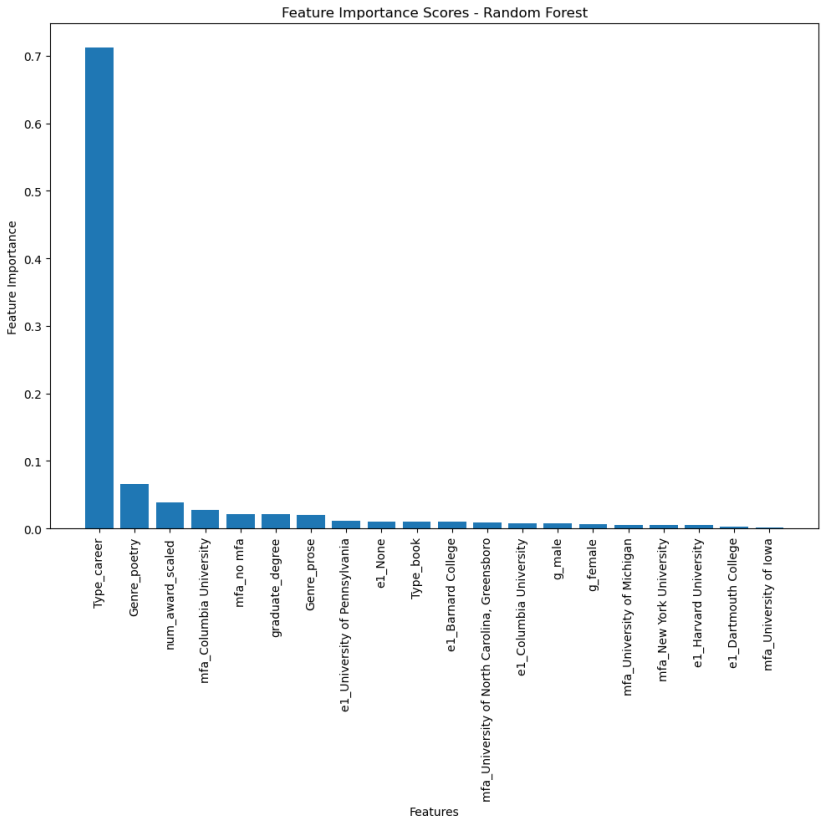

Figure 12: Feature Importance of Tuned Random Forest Regressor

In conclusion, the biggest determinant of whether or not you make a lot of money as an author through awards is career awards.

Future Research

That said, to further my research I would look into the specifics of the career awards: when did they begin, what are the demographics of the winners, are there features that have a strong correlation with career awards. I also would be interested in reverse engineering some unsupervised classification models to see if any machine learning model could accurately predict which MFA a person went to based on the number of awards won and an author's cumulative winnings.