Predicting Book Ratings Using Machine Learning

A Study of Feature Engineering and Model Performance on Goodreads Data

Deon Saju, Leny Belamich, Pranav Prakash Nair

DataScience Tech Institute

Machine Learning With Python

September 18, 2024

1. Introduction

Nowadays, the sheer number of books available can make choosing the right one a daunting task. With an ever-growing pool of literary options, ratings and reviews from readers offer valuable insights into the quality and popularity of books. These ratings help readers navigate their choices and select books that suit their preferences. However, manually sifting through reviews can be time-consuming, and not every book has extensive feedback.

The goal of this project is to develop a machine learning model that can predict the average rating of a book based on various features sourced from the Goodreads dataset. By predicting a book's rating, this model can support recommendation systems and content-based filtering, offering users personalized book suggestions. The potential applications of such a model extend to improving user experience in online bookstores, libraries, and reading apps by providing instant ratings for books with limited or no reviews.

2. Data Description

The dataset consists of information about books, including attributes such as:

- **bookID**: A unique identifier for each book.
- **title**: The book's title.
- **authors**: The names of the book's authors.
- **average_rating**: The target variable, representing the average rating of the book.
- **isbn** and **isbn13**: Unique book identifiers.
- **language_code**: The primary language of the book.
- **num_pages**: The number of pages in the book.
- **ratings_count**: The total number of ratings received.
- **text_reviews_count**: The number of text reviews.
- **publication_date**: The date the book was published.
- **publisher**: The publishing company.

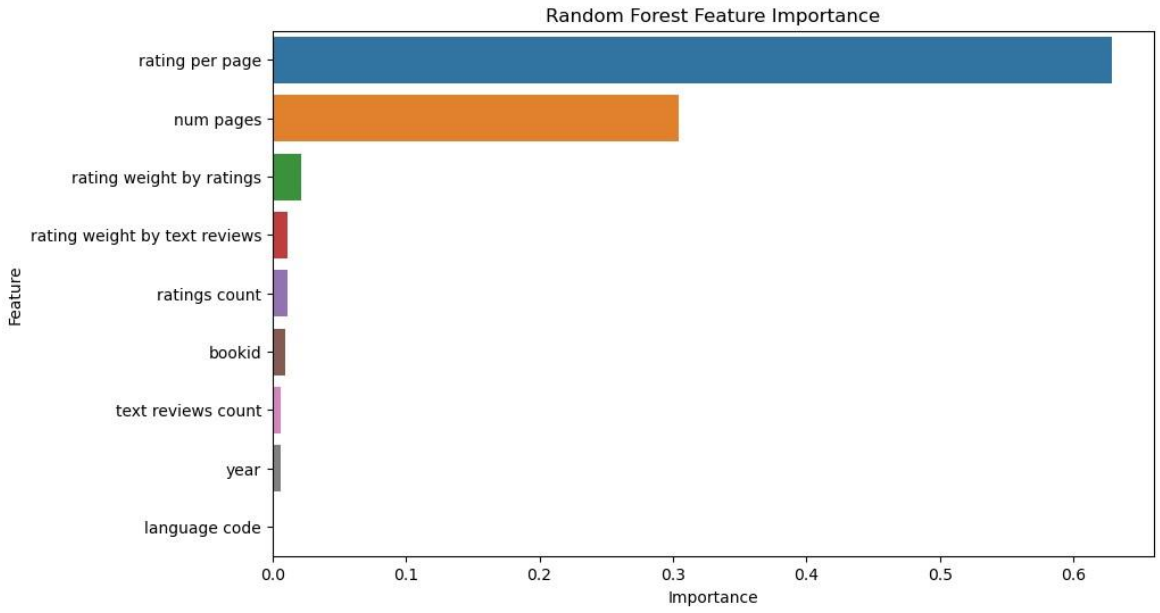
We aim to predict the average rating of a book based on these features.

There were no missing values initially identified in the dataset. However, when converting the publication date to a date format, we encountered two missing entries. After researching the correct publication dates online, we updated the dataset accordingly. We also standardized the column names for consistency and unified the language codes across the dataset. Additionally, we removed 76 books with 0 pages, as these were likely errors or audiobooks, which we deemed irrelevant to the analysis, given the total of over 11,000 entries. Finally, we added new features, “rating weight by

text reviews", "rating weight by ratings" and "rating per page" in our dataset. The first two features amplify the importance of books that have a high number of ratings and reviews, allowing us to assess whether increased engagement can improve the accuracy of rating predictions. The third feature could help reveal correlations between the length of the book and its quality.

3. Methodology

In this project, we implemented four machine learning models—Random Forest Regressor, XGBoost, AdaBoost Regressor and Linear Regression—to predict the average rating. The initial step involved feature selection, where we dropped certain features like bookid, title, authors, isbn, isbn13, publication date, and publisher. These were textual data deemed irrelevant for the regression task. The remaining features were used for model training and evaluation. To understand which features were most influential in predicting the target variable, we used the feature importance scores generated by the Random Forest model. The plot below illustrates the relative importance of each feature used in the model.



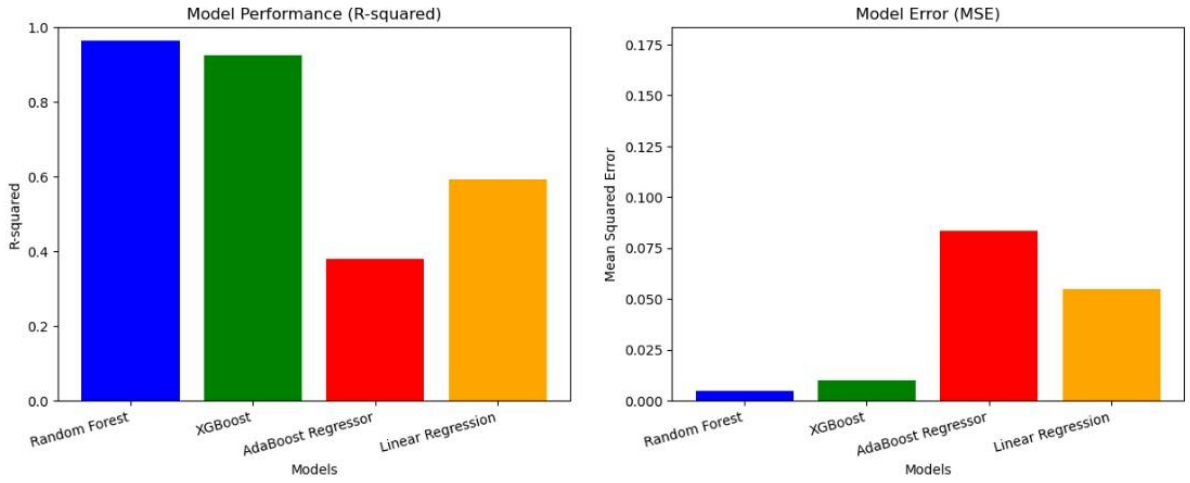
Each of the four models was selected based on their ability to handle different types of data relationships. Random Forest Regressor, an ensemble method based on decision trees, was chosen for its capability to model non-linear relationships. XGBoost, a gradient-boosting algorithm, was selected due to its high performance in complex regression tasks. We also employed AdaBoost Regressor, which iteratively improves model performance by focusing on previous errors. Linear Regression served as a baseline to compare against the more advanced models.

The dataset was split into a training set (80%) and a validation set (20%) to ensure robust model evaluation. The models were assessed using different metrics such as Mean Squared Error (MSE) or R-squared (R^2) metrics.

4. Results

Among the four models, Random Forest Regressor produced the best results, achieving an R^2 score of 0.96, indicating a very strong fit and accurate predictions. XGBoost followed closely with an R^2 score of 0.92, demonstrating solid performance but slightly less accurate than Random Forest.

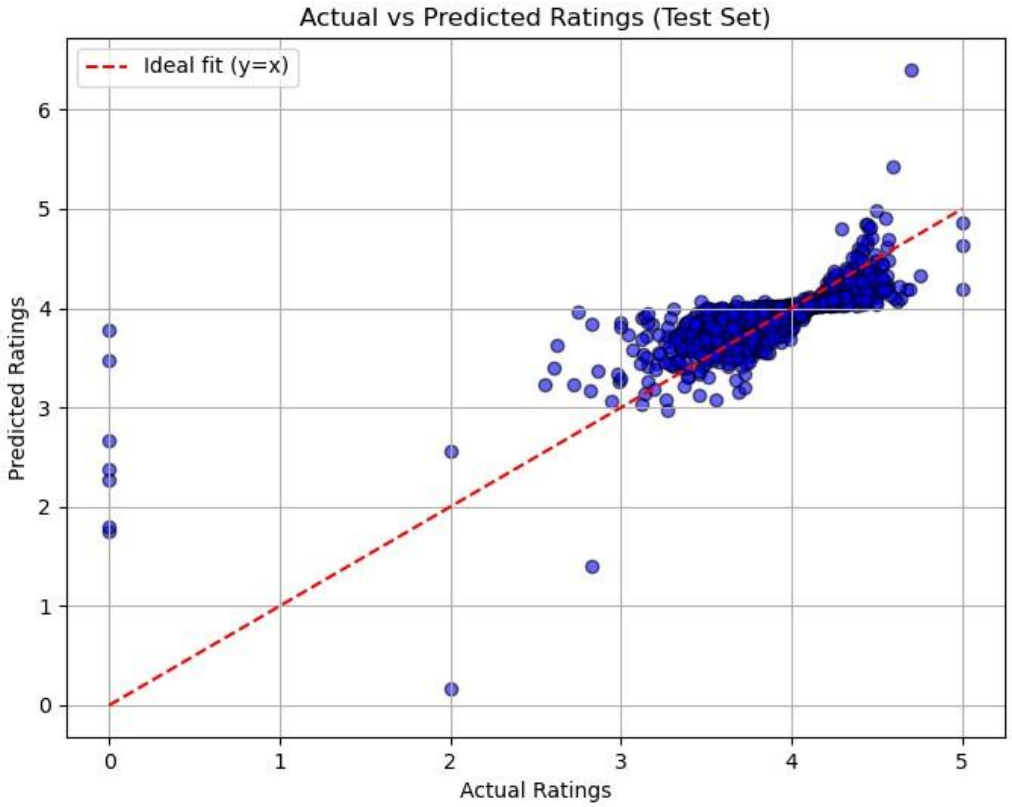
Linear Regression yielded an R^2 of 0.59, significantly lower than the ensemble models, highlighting its limitations in capturing the complexity of the data. AdaBoost Regressor performed the worst, with an R^2 of 0.38, suggesting that it struggled to generalize well to this dataset. The following plots display the R^2 and MSE values for each model.



In summary, Random Forest outperformed the other models, capturing intricate patterns in the data effectively, while XGBoost provided competitive results. Linear Regression and AdaBoost, although less effective, still offered valuable insights into the limitations of simpler models when handling more complex data.

5. Discussion

The results of the model evaluation clearly indicate that Random Forest performed significantly better. With the highest R-squared value and the lowest Mean Squared Error, it was the most effective model for predicting the target variable. To illustrate its performance, this plot compares the actual ratings from the test set with the model's predicted ratings.



The red dashed line represents the ideal fit. Points that fall closer to this line indicate better predictions, while points further from the line signify prediction errors.

6. Conclusion

In this project, we aimed to predict book ratings using various machine learning models, including Random Forest, XGBoost, AdaBoost Regressor, and Linear Regression. After evaluating the models using both R-squared and Mean Squared Error, the Random Forest model emerged as the most effective, achieving the highest predictive accuracy with an R-squared of 0.96 and a MSE of 0.005. XGBoost also performed well, while Linear Regression and AdaBoost struggled to capture the complexity of the data.

Our analysis of feature importance further reinforced the strength of the Random Forest model, with rating per page and number of pages standing out as the most influential predictors of book ratings. This finding not only validates the model's performance but also offers valuable insights into the key factors influencing user ratings.

Despite the promising results, there are still opportunities for improvement. Future work could involve refining feature selection, or incorporating new features, such as a genre attribute, which could significantly enhance the model's ability to predict average ratings. Additionally, addressing the outliers observed in the Actual vs. Predicted Ratings plot could help further improve model accuracy.

Overall, this study demonstrates the power of ensemble methods like Random Forest and XGBoost for predictive tasks in complex datasets and highlights the importance of careful feature selection in driving model performance.