**GOODREADS BOOK EXPLORER**

**CSCI-E-108 FINAL INDEPENDENT PROJECT PROPOSAL**

**By Ana Raicar**

**Summer 2024**

**EXECUTIVE SUMMARY**

Ernest Hemingway once said, "There is no friend as loyal as a book”. I am an avid reader and even in this digital era, I take immense pleasure in reading. One of the challenges readers always struggle with is discovering quality recommendations. Goodreads is my go-to source for finding new recommendations, but I’ve found myself with some bizarre and unexpected recommendations, which are not related to my reading or rating history, every now and then. There are reports from other users facing similar challenges.

Goodreads has its own recommendation engine which combines multiple proprietary algorithms, analyzing 20 billion data points to better predict which books people will want to read next. It maps out the connections between books by looking at how often they appear on the same bookshelves and whether they were enjoyed by the same people. However, these recommendations often fall short, representing a missed opportunity for Goodreads, which is owned by Amazon. Better recommendations would entice a user to buy those books on Amazon, thereby driving revenue.

Driven by sheer curiosity and the excitement to explore some of the techniques learnt in class, I'm going to develop a book recommendation system. I'm eager to assess its performance, the recommendations it proposes and learn about the challenges involved in build recommender systems.

**DATA SOURCE**

While I originally planned on grabbing the data from <https://mengtingwan.github.io/data/goodreads> (recommended as a possible data source in class), I faced some memory issues and wasn’t able to download all of it. Therefore, I looked for an alternative data source and found a Github user who has collected and saved the data in his public repo - <https://github.com/zygmuntz/goodbooks-10k/tree/master> .

Here is a summary:

* Number of unique books: 10,000 (dataset includes the most popular by ratings)
* Number of unique users: 53,424
* Number of unique ratings: 5,976,478

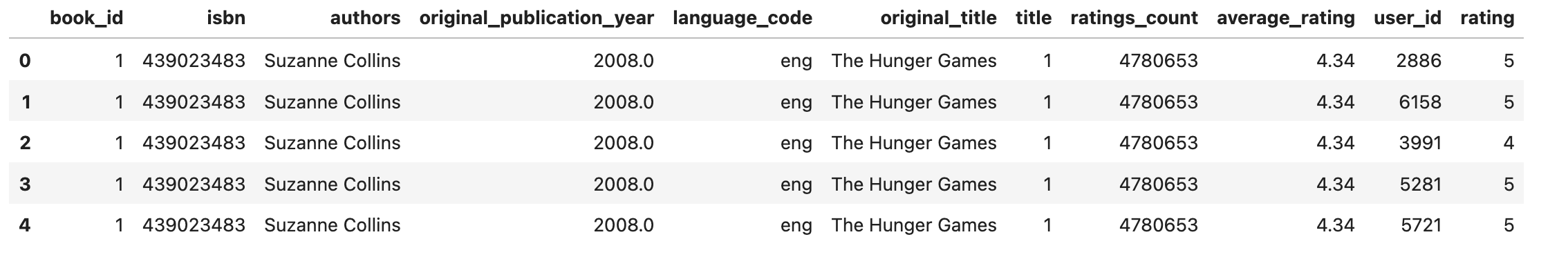
**DATA PREPARATION**

The books and ratings datasets I will be using include the following variables:

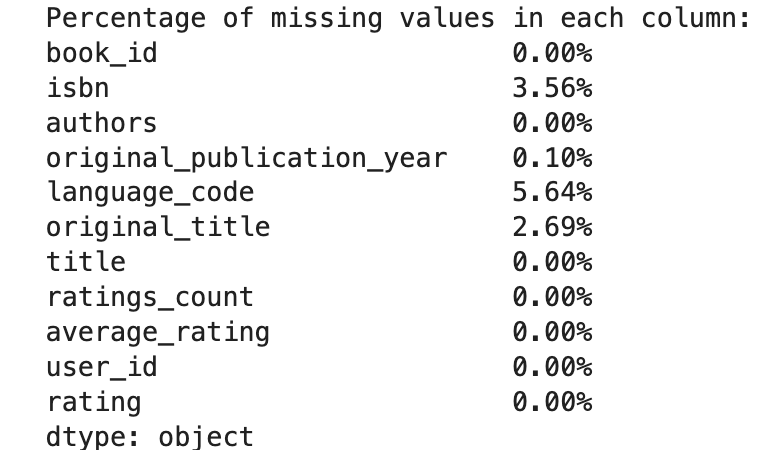
A screenshot of a computer

Description automatically generated

For my exploratory data analysis and recommender, I selected some of the interesting and relevant variables and merged the datasets together.



Next, I took a look at the data for missing values. A few variables with missing values were identified but since they will be used for EDA purposes only, it is not an issue. All the books have a corresponding user(reader) and rating.



**EXPLORATORY DATA ANALYSIS (EDA)**

Here are a few insights which were derived from the data exploration process:

1. Based on all the ratings given by the universe of users, we can see that most of them give a rating of 4 or 5.

A graph with blue and white bars

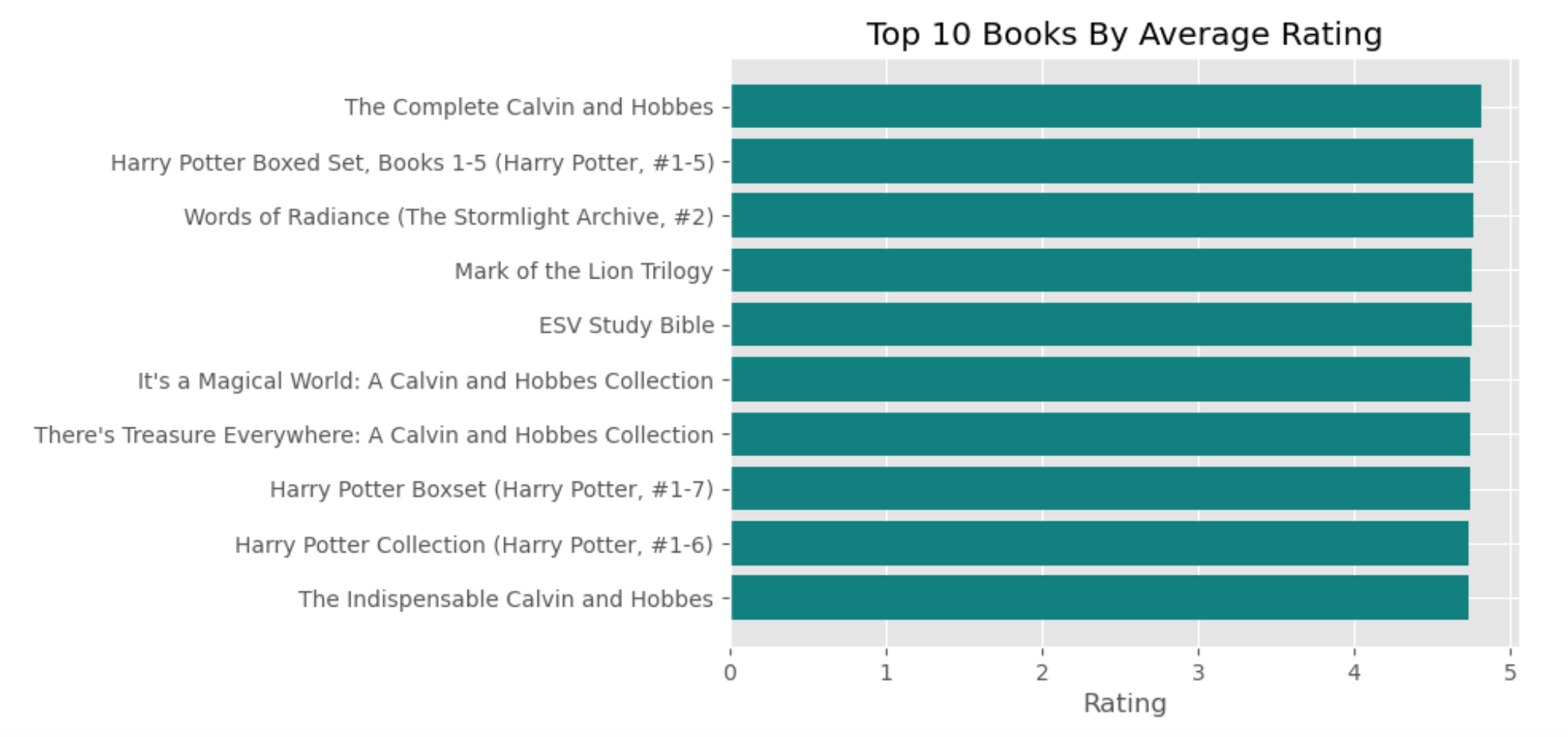
Description automatically generated

1. From the distribution below, we can see that most of the users rate 75-150 books. The average number of books rated by a user is 112.

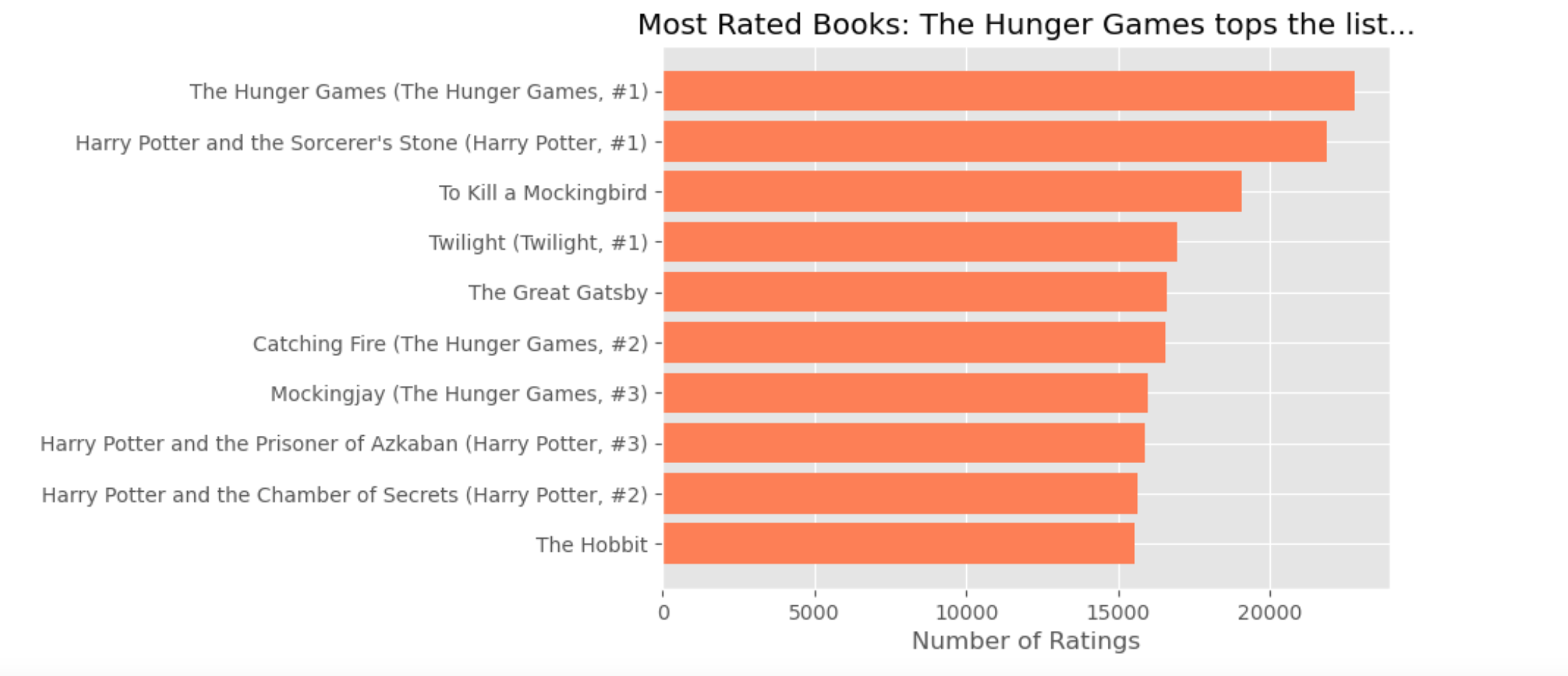
A graph of rating

Description automatically generated

1. 87% of the books in the dataset are in English
2. “The Complete Calvin and Hobbes” is the best book based on the average rating.



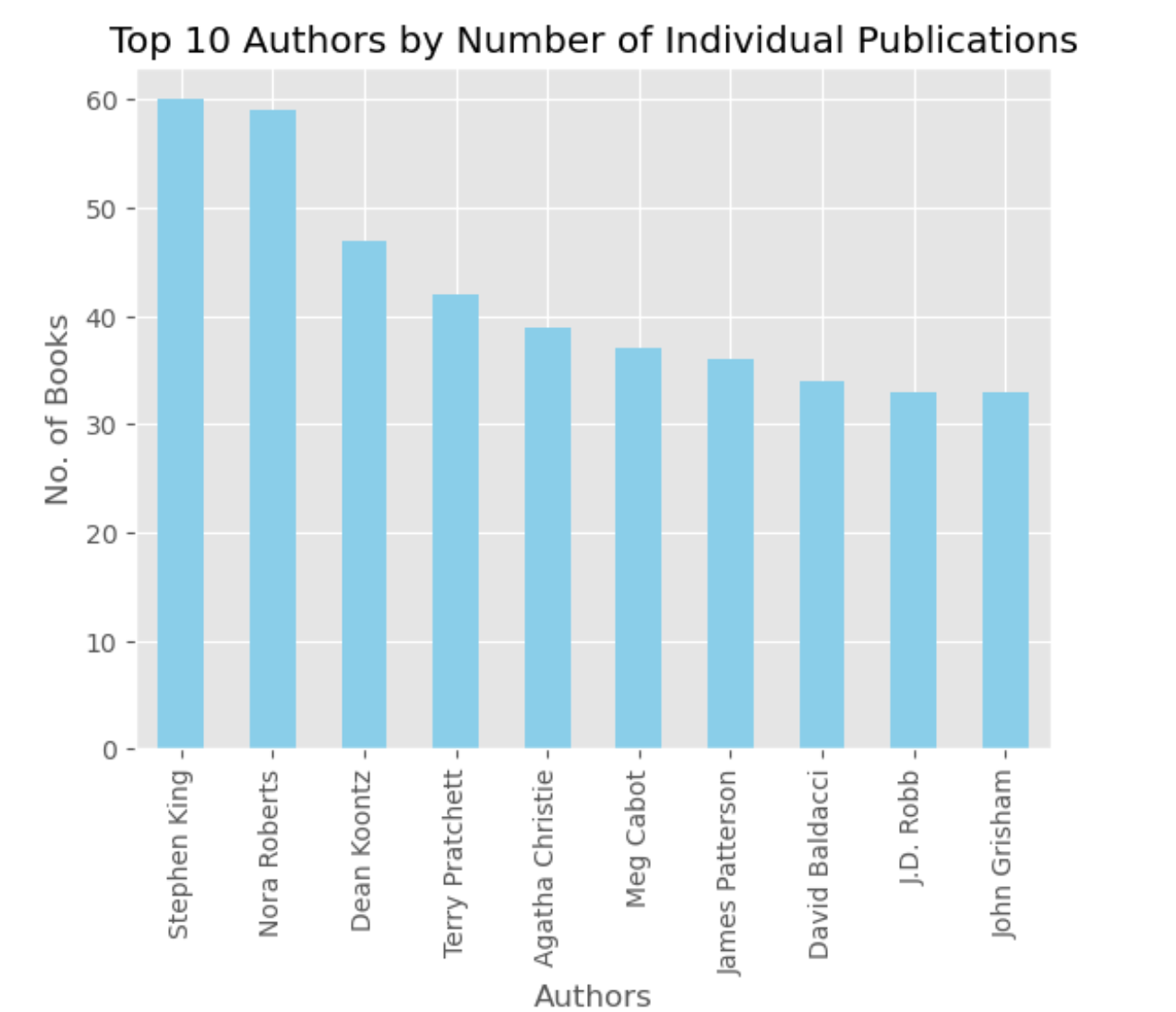
1. “The Hunger Games” is the most rated book, it was 22.8K times.



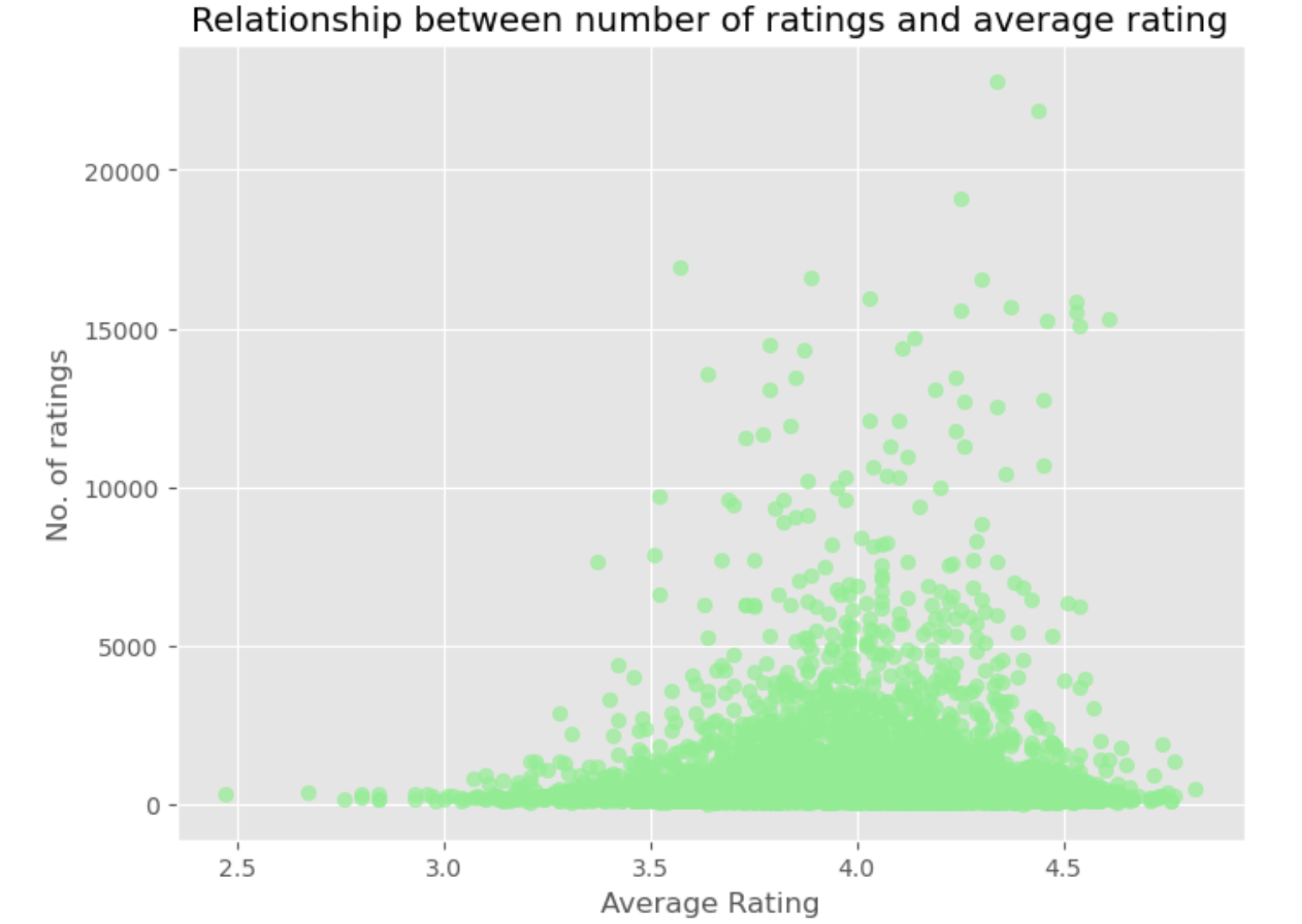
1. Distribution of books by publication year – majority of books in the dataset were published post 2000



1. Out of the 10k books in the dataset, Stephen King has authored the most books – 61 individual books and he co-authored an additional 36 books.



1. The scatter plot below shows the relationship between the average rating and the number of ratings a book received. We can observe that as the number of ratings given to a book goes up, the average rating gets closer to 4.



Results of this initial data exploration suggest that the data is well-structured and of great quality. I will continue exploring this data a bit more and then proceed to building the recommender model.

**MODELING PROCESS AND METHODOLOGY**

Baseline Model: To deal with the cold start problem, I will build a baseline model which will recommend all the books to all the users, ignoring their reading or rating history. Recommendations from this model will be given to new users who have no history with the platform.

Collaborative Filtering Model: The next step will be building a collaborative filtering model which will recommend books based on the highest similarity between user-user or item or item. For this dataset, since the number of items is less than the number of users, an item-based model makes the most sense as user preferences tend to change over time. As part of this process, the following steps would have to be taken:

* Build a User Item Matrix: This would represent interactions between users and the items (ratings). However, a threshold for items and users would have to be decided on. For example, we would want to include only those users who rated at least 50 books.
* Measuring Item Similarity: Leverage measures like Pearson and Cosine similarity to calculate pairwise similarities between items based on user interactions. The rational here is if users tend to rate or interact with two items similarly, their similarity score will be higher.
* Calculate the implied rating for items the user hasn’t interacted with yet
* Make recommendations: Recommend items to the target user that are similar to items they have already interacted with positively.
* Model Evaluation: Assess model performance using RMSE
* Explore other techniques like Content-based filtering, KNN based or an SVD model, if time permits