Semantic Analysis and recommendation system for Book

Aishwarya Sinhasane[1]* , Anuj Mahajan[2]*, Shashwati Diware[3]*,Shubham Jambhale[4]*

Abstract

For both users and publishers/authors, the issue of sentiment analysis of book reviews and recommendations for books utilizing Amazon dataset is crucial. The objective of book review sentiment analysis is to automatically categorize a book review's sentiment as positive, negative, or neutral. Machine learning algorithms that have been trained on labeled review datasets are frequently used to do this. The task of book recommendation is to make book recommendations to users based on their preferences and previous actions. Sentiment analysis of book reviews can be helpful for book recommendations in a number of ways, including helping readers discover books they are likely to enjoy and assisting publishers and authors in understanding reader reactions to their works. Book recommendation systems can give users a more personalized and relevant experience, boost user engagement and satisfaction, and help users discover books that they might not have found otherwise by assessing the sentiment of book reviews and using this information to create better recommendations. Moreover, book recommendation systems can assist publishers and authors in more efficiently promoting their books by recommending books to readers based on their tastes and behavior. This results in higher sales, more money, and the ability to reach new audiences. Also, by examining the tone of user reviews and other user data, book recommendation algorithms can better understand user preferences and behavior. As a result, publishers and authors can design more specialized marketing strategies and new consumer-focused goods and services. Improving personalization, customer satisfaction, competitive advantage, innovation, and social impact can all be facilitated by finding a solution to the book review sentiment analysis and book suggestion conundrum.

Keywords

sentiment analysis — recommendation — book reviews

[1] *Luddy School of Informatics, Computing, and Engineering, Indiana University, Bloomington, IN, USA*

Contents

1. Problem and Data Description

Problem Description

- The problem of sentiment analysis of book reviews and recommendations for books using the Amazon dataset can be utilized to deliver more tailored and relevant book recommendations based on individual readers' likes and behavior.

- Book review sentiment analysis is the task of automatically classifying the sentiment of a book review as positive, negative, or neutral. This is often accomplished using machine learning algorithms that have been trained on labeled datasets of reviews. The algorithm's input is the review text, and its output is the sentiment label. Book review sentiment analysis can be used for a variety of purposes, including assisting users in finding books that they are likely to enjoy and assisting publishers and authors in understanding the response to their books.

- Book recommendation is the task of suggesting books to users based on their preferences and past behavior. This can be done using a variety of techniques, including collaborative filtering, content-based filtering, and hybrid models.

- Book review sentiment analysis can be useful for book recommendation in several ways. For example, the sentiment of a user's review can be used as a feature in a content-based filtering model, helping the algorithm understand the user's preferences more accurately. Sentiment analysis can also be used to filter out reviews that are not relevant to the user's preferences, such as reviews of books in genres that the user does not like.

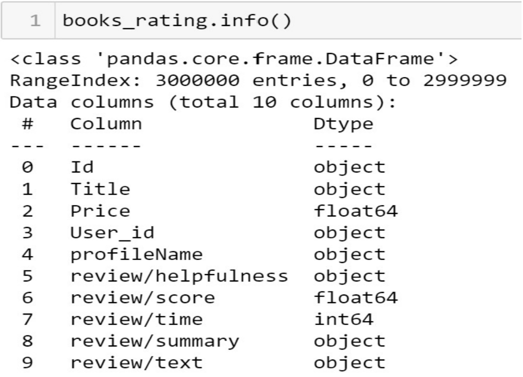
Data Description

- The data we obtain has 2 csv which are linked with each other using book titles.
- The 2 csv are Book Details and Book Reviews.
- Book Details consist of attributes like Title, Description, Author, Publisher, Published Date, Categories, and Rating Count.
- Majority of data in the dataset is categorical in nature. This CSV gives us insight into the book we are reviewing. It tells us who published the book at what time and who are the authors of the book. It also conveys which category the book belongs to and how many ratings it has received.
- The other CSV which is Book Rating is mostly focused on reviews received by different books. It has short summary of the review to analyze the review in short phrases which gives us important information from the overall review. This data is the main source for our sentimental analysis. It also shows helpful review ratings.

2. Data Preprocessing & Exploratory Data Analysis

2.1 Handling Missing Values

Handling Missing Values and Data Preprocessing

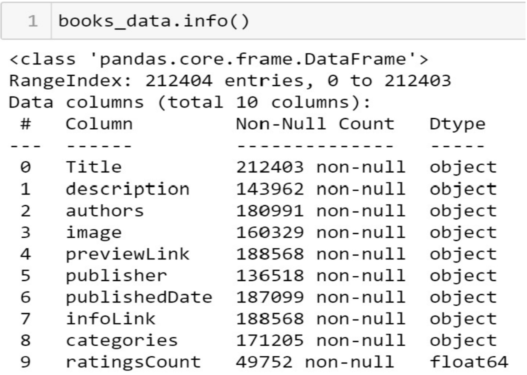


```
books_rating.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3000000 entries, 0 to 2999999
Data columns (total 10 columns):
 #   Column              Dtype  
---  ------              -----  
 0   Id                  object 
 1   Title               object 
 2   Price               float64
 3   User_id             object 
 4   profileName         object 
 5   review/helpfulness  object 
 6   review/score        float64
 7   review/time         int64  
 8   review/summary      object 
 9   review/text         object 
```

Figure 1. Book Rating Information

- Book Rating csv consists of above columns and info
- It consists of only 3 numerical columns which are Id, review/score, and review/time. Other all the columns are categorical.

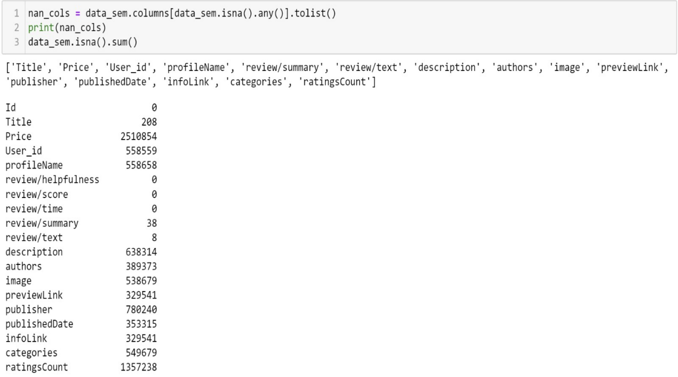


```
books_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 212404 entries, 0 to 212403
Data columns (total 10 columns):
 #   Column         Non-Null Count   Dtype  
---  ------         --------------   -----  
 0   Title          212403 non-null  object 
 1   description    143962 non-null  object 
 2   authors        180991 non-null  object 
 3   image          160329 non-null  object 
 4   previewLink    188568 non-null  object 
 5   publisher      136518 non-null  object 
 6   publishedDate  187099 non-null  object 
 7   infoLink       188568 non-null  object 
 8   categories     171205 non-null  object 
 9   ratingsCount   49752 non-null   float64
```

Figure 2. Book Data Information

- Book Data csv consists of the above columns and info
- It consists of only 1 numerical column which is ratingsCount. Other all the columns are categorical.



```
nan_cols = data_sem.columns[data_sem.isna().any()].tolist()
print(nan_cols)
data_sem.isna().sum()
```

```
['Title', 'Price', 'User_id', 'profileName', 'review/summary', 'review/text', 'description', 'authors', 'image', 'previewLink',
'publisher', 'publishedDate', 'infoLink', 'categories', 'ratingsCount']

Id                          0
Title                     208
Price                 2510854
User_id                558559
profileName            558658
review/helpfulness          0
review/score                0
review/time                 0
review/summary             38
review/text                 8
description            638314
authors                389373
image                  538679
previewLink            329541
publisher              780240
publishedDate          353315
infoLink               329541
categories             549679
ratingsCount          1357238
```

Figure 3. Missing values in dataset

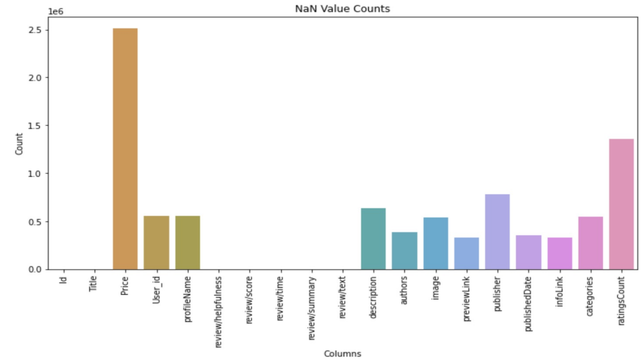


Figure 4. Missing values in dataset using histogram

- We have merged both the datasets (books_data and books_rating) based on Title and above are all the columns we have obtained.
- Fig 3 shows all the missing values in the merged dataset.
- It consists of multiple columns for which we have missing values. Price has the most missing values followed by RatingsCount.

- Fig 4 shows the missing values count using a histogram.

- As part of data pre-processing we have also removed the columns which doesn't have much impact on semantic analysis and which are irrelevant along with a huge amount of null values. Those columns are : [Price, Description, image, publishedDate, and publisher]

- The presence of missing or invalid data in a dataset can severely impact the accuracy of any analysis or modeling conducted on that dataset. One common approach to handling this issue is to replace any missing or invalid values with a standard value, such as "unknown". This approach can be particularly useful when dealing with categorical data, such as categories, authors, infolink, and previewlink, which are essential for recommendation tasks.

- While there are many statistical techniques available for replacing missing values, these methods may not be appropriate for all datasets or contexts. For example, some statistical techniques may assume that the missing data are randomly distributed or that the missing values are related to other variables in the dataset. However, in many cases, the reasons for missing data are not random or easily discernible.

- Replacing missing or invalid data with the value "unknown" can help ensure that the dataset remains consistent and accurate. This approach provides a standard value for all instances of missing or invalid data, which can help prevent inconsistencies in the dataset and ensure that any analysis or modeling conducted on the dataset is accurate and unbiased.

Categories

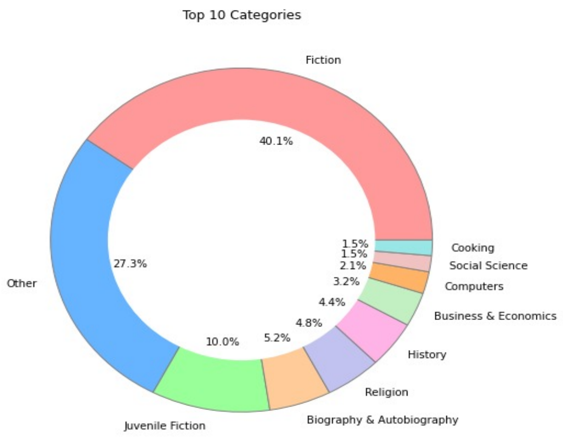


Figure 5. Different Categories distribution

- Data has a lot of empty values in categories. As part of data preprocessing, we have replaced the empty values in the categories column using 'Other' to make it more meaningful

- Fig 5 shows the top 10 different categories of books along with 'others'.

- From the data we can see that the 'Fiction' category has the maximum frequency followed by 'Other'.

Authors

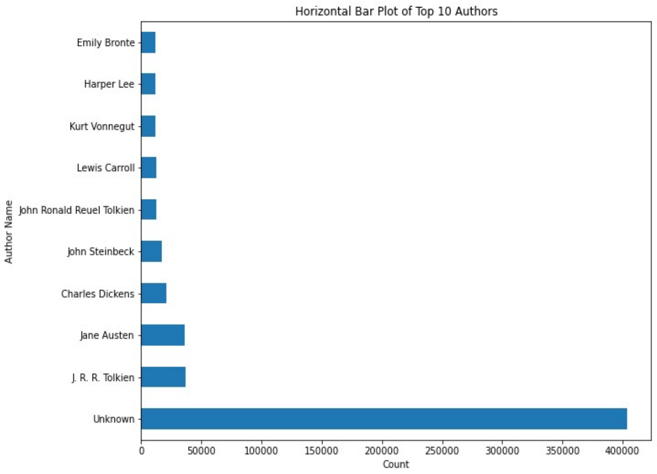


Figure 6. Different Categories distribution

- Data has a lot of empty values in Author. As part of data preprocessing, we have replaced the empty values in the categories column using 'Unknown' to make it more meaningful.

- Fig 6 shows the top 10 different Authors.

- From the data we can see that most of the data consist of an 'Unknown' author.

Pre-Processing for review/text:

- We have performed preprocessing on text data, specifically reviews of the books, to prepare it for analysis and recommendation tasks. We are using the NLTK a popular library for text processing and analysis in Python.

- The first step in text preprocessing is to remove any URLs that may be present in the input text. This is done using regular expressions (re library) to match and replace any string starting with "http" or "www". URLs are usually irrelevant for the task of book recommendation, so removing them helps to simplify the data.

- The next step is to remove any unwanted characters such as punctuation marks, which are also irrelevant to the analysis. This is done by using another regular expression that matches any character that is not a word character or whitespace. The result is a string that contains only words and spaces.

- After that, we are removing any Twitter-specific symbols such as the @ and # symbols. Again, these symbols are not important for the analysis and can be safely removed.

- The next step is to convert all text to lowercase. This is done to standardize the text so that the same word in different cases is treated as the same word during analysis.

- The most important step in this preprocessing is to remove stop words and lemmatize the remaining words. Stop words are commonly used words in a language (such as ”the”, ”a”, ”an”, ”and”, etc.) that are not meant for analysis. Removing them helps to focus on the more relevant words in the text. NLTK library provides a set of predefined stop words for the English language, which are loaded into the stop words object.

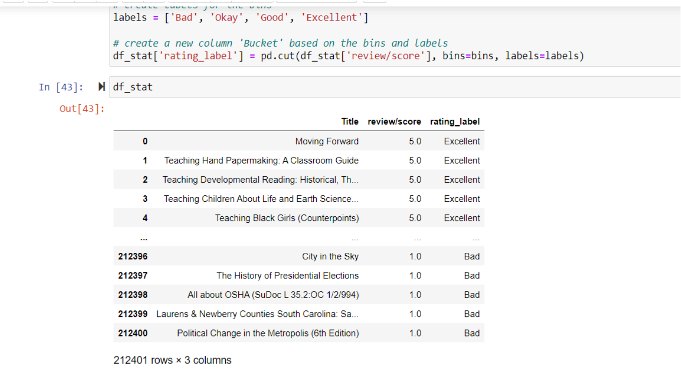
- We are using the word tokenize function from NLTK to split the text into individual words and then the lemmatizer object to lemmatize each word. Lemmatization is the process of reducing a word to its base or root form so that words with the same meaning but different forms are treated as the same word during analysis.

- The output is a cleaned and standardized version of the review text, which can be used for sentiment analysis and book recommendation.

2.2 Exploratory Data Analysis

- Calculating the average score rating for a book from a reviews dataset is a useful way to get an overall idea of how well-received a book is among readers.Breaking down the distribution of average ratings into categories such as ”bad,” ”okay,” ”good,” and ”excellent” provides an even clearer picture of how readers are responding to the book.

- We are using a pie chart to visualize the distribution of average rating categories, as it allows for a quick and easy comparison of the number of reviews falling into each category. By displaying the data in this way, patterns and trends can become more apparent and insights can be gained into the success of the book.



```
# create labels for the bins
labels = ['Bad', 'Okay', 'Good', 'Excellent']

# create a new column 'Bucket' based on the bins and labels
df_stat['rating_label'] = pd.cut(df_stat['review/score'], bins=bins, labels=labels)
```

```
In [43]:  df_stat
```

```
Out[43]:
                                                Title  review/score  rating_label
0                                      Moving Forward           5.0     Excellent
1         Teaching Hand Papermaking: A Classroom Guide           5.0     Excellent
2       Teaching Developmental Reading: Historical, Th...       5.0     Excellent
3         Teaching Children About Life and Earth Science...     5.0     Excellent
4                   Teaching Black Girls (Counterpoints)        5.0     Excellent
...                                                   ...       ...           ...
212396                                    City in the Sky       1.0           Bad
212397                The History of Presidential Elections     1.0           Bad
212398          All about OSHA (SuDoc L 35.2:OC 1/2/994)        1.0           Bad
212399  Laurens & Newberry Counties South Carolina: Sa...      1.0           Bad
212400          Political Change in the Metropolis (6th Edition) 1.0          Bad

212401 rows × 3 columns
```

Figure 7. Distribution of Average Review Score

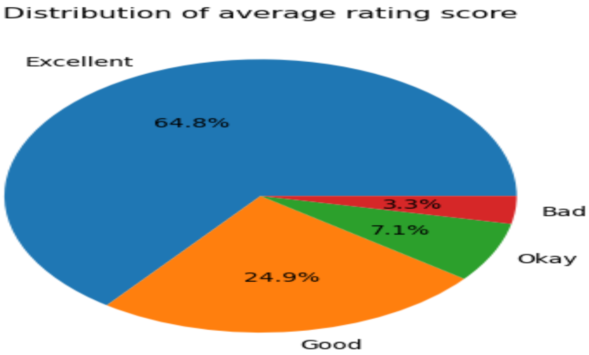


Figure 8. Distribution of Average Review Score

- When analyzing book reviews, it is important to consider the category of the book being reviewed. Reviews for books in different categories can have varying language, tone, and content.

- The category of a book can also help in determining the target audience for a particular book. For instance, young adult novels typically target teenagers, while children’s books are aimed at young children. Therefore, knowing the category of a book can help in recommending it to the appropriate audience.

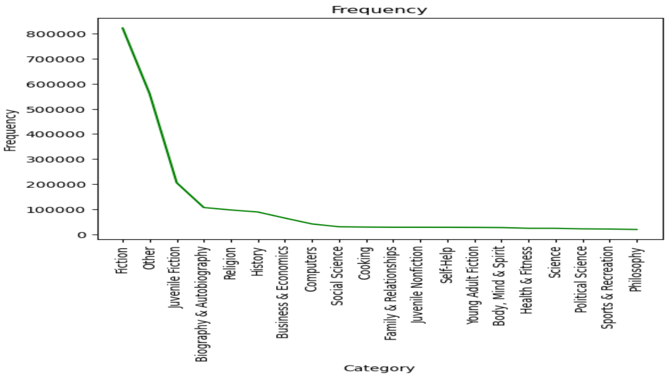


Figure 9. Categories Distribution

- The line chart above depicts which category has the maximum count of reviews. This information can be used to gain insights into the reading preferences of a particular population or to identify gaps in a library or bookstore's collection.

- We can also observe that there are many null values for categories that are represented by other category.

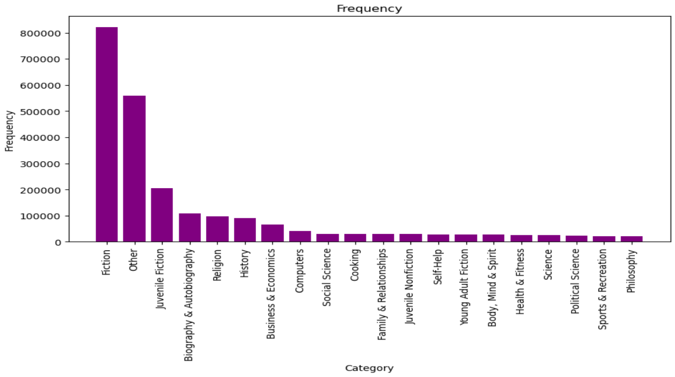


Figure 10. Categories Distribution

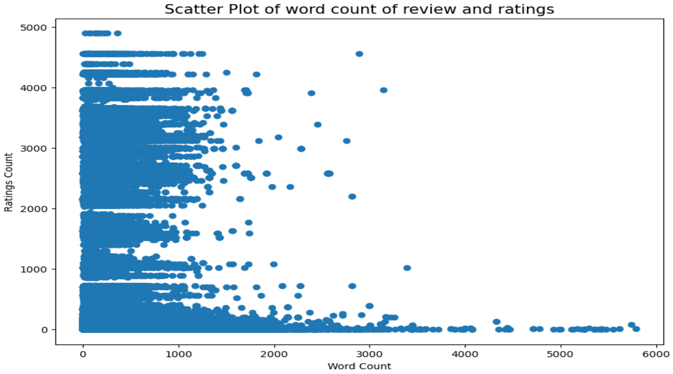


Figure 11. Scatter plot of Ratings Count vs Word Count

- This can help to understand the relationship between the length of a review (in terms of word count) and the number of ratings it has received. Additionally, it can be useful for sentiment analysis as longer reviews may have more in-depth analysis and opinions which can provide a more nuanced understanding of a book's reception. Furthermore, the plot can reveal any outliers or trends in the data.

- From the scatter plot, we can observe that there are some outliers in the data, which means there are some reviews that are very long and have a high number of ratings. These reviews could be either very positive or very negative, and they may have contributed to the overall popularity of the book.

- We also see that when the word count is less, there are a lot of records having fewer ratings. This could be because shorter reviews are less informative and may not provide enough details about the book, which could make it less popular among readers.

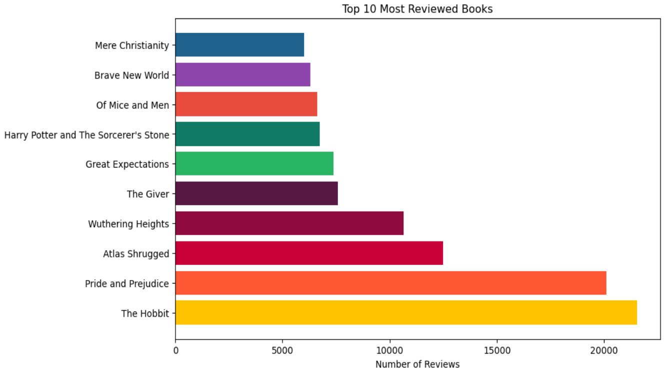


Figure 12. Top 10 Most Reviewed Books

- The above bar chart shows the top 10 most reviewed books that can be useful in making recommendations to readers who are looking for popular or highly-rated books. These books have already gained a lot of attention and interest, which makes them more likely to be enjoyed by a wider audience.

- Also, by analyzing the reviews of these books, we can identify the common themes, topics, and characters that readers enjoy or dislike, and use this information to recommend similar books to readers who have enjoyed these popular books.

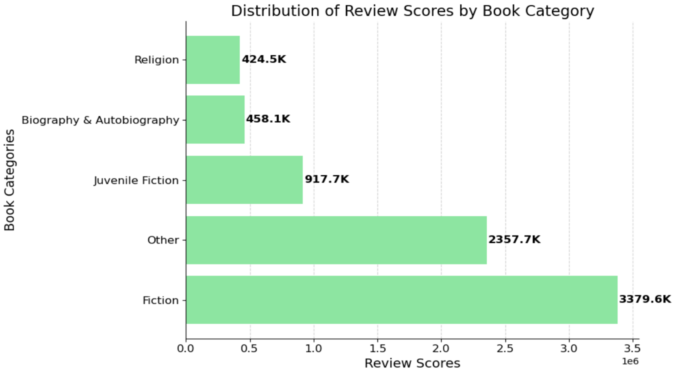


Figure 13. Categories and its average review score

- By analyzing the distribution of review scores for different book categories, we can identify which categories are more likely to receive positive or negative reviews. For example, if we observe that books in the "Fiction" category have a higher average review score compared to books in the "Biography & Autobiography" category, this can suggest that readers generally prefer fiction over non-fiction or biographical books.

- If a reader enjoys books in a particular category that tends to have high review scores, we can recommend other books in that category with similar characteristics.

Github Repository Link : https://github.com/aishwaryavijaysinhasane/Book-Recommendation-System/tree/master

3. Algorithm and Methodology

Add subsections if needed.

4. Experiments and Results

5. Deployment and Maintenance

6. Summary and Conclusions

Acknowledgments

References