Book Recommendation System

Vikas Chaudhary

Data Science Trainee

AlmaBetter, Bengaluru

Abstract

Recommendation Systems are becoming essential part of businesses across the domain. They are widely used in video sharing platforms, OTTs, short-video platforms from last few years. Recently, other businesses including books stores (online-offline) have also started embracing recommendation systems to increase their outreach and better customer experience.

Keywords: Python, EDA, Feature Engineering, ML, Unsupervised, k Nearest Neighbor.

Problem Statement

With the plethora of options available for a reader to choose the book it is required to have some well functional system that can assist the reader to choose the next book and get back to the platform.

About Dataset

The project has 3 datasets containing data about Books, Users, Rating as given below:

1. books_df: It has more than 2.7 lakh rows and 8 columns with the following information:
    - ISBN – Contain ISBN number of book.
    - Book-Title – Renamed as ***'title'***, contains the title of the book.
    - Book-Author – Renamed as ***'author'***, contains book's author name.
    - Year-Of-Publication – Renamed as ***'year'***, contains year of publication.
    - Publisher – Renamed as ***'publisher'***, contains name of publisher.
    - Image-URL-S
    - Image-URL-M

- Image-URL-L
    - The above three contain cover image of book. They were deleted.

2. 'user_df': It has more than 2.7 lakh rows and 3 columns with the following information:
    - User-ID – Contain the ID number of user
    - Location – It has address of the user that includes city, province/state, country.
    - age – contains age of the user
3. 'rating_df': It has more than 11.4 lakh rows and 3 columns with the following information:
    - User-ID – Contain the ID number of user
    - ISBN - Contain ISBN number of book
    - Book-Rating – Renamed as ***'rating'***, contains rating out of 10 given to the book

The final dataframe 'final_rating' was created after merging the three dataframes on 'ISBN'. It has 4.8 lakh rows and 8 columns which are:

'user_id', 'ISBN', 'rating', 'title', 'author', 'year', 'publisher', 'number_of_ratings'

Introduction

As per the Market Analysis Report by Grand View Research, the global books market size was valued at USD 138.35 billion in 2021 and is expected to expand at a compound annual growth rate (CAGR) of 1.9% from 2022 to USD 164.22 billion in 2030.

As per Mordor Intelligence, the Recommendation Engine market was valued at USD 2.12 billion in 2020, and it is expected to reach USD 15.13 billion by 2026, registering a CAGR of 37.46% during the period of 2021-2026.

As the number of publications is increasing and every book has a possible global reach it is necessary for book sellers (both online and offline) to adopt recommendation system to increase their reader base.

Tools Used

Since it is an Unsupervised Machine Learning (ML) project, the following Python libraries are required:

- *numpy:* Numpy is the core library for scientific computing in Python. It provides a high-performance multidimensional array object and tools for working with these arrays.
- *pandas:* Pandas is an open-source library that is built on top of the NumPy library. It is a Python package that offers various data structures and operations for manipulating numerical data and time series.
- *seaborn:* Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.
- *matplotlib:* Matplotlib is probably the most used Python package for 2D graphics. It provides both a quick way to visualize data from Python and publication-quality figures in many formats.
- *Scipy:* SciPy provides algorithms for optimization, integration, interpolation, eigenvalue problems, algebraic equations, differential equations, statistics and many other classes of problems.
- *Sklearn:* Scikit-learn (Sklearn) is the most useful and robust library for machine learning in Python. It provides a selection of efficient tools for machine learning and statistical modeling including classification, regression, clustering and dimensionality reduction via a consistence interface in Python. This library, which is largely written in Python, is built upon NumPy, SciPy and Matplotlib.

Approach Used

The project was built on the three dataset, every dataset was explored meticulously using different methods to carve the final dataset that is required for better exploration and finally algorithm implementation. The following steps were involved:

1. Data Preparation
2. EDA and Feature Engineering
3. Model Implementation

Data Preparation

In the ‘book_df’ out of 8 features, 5 features were kept. 2-4 null values were also present in this dataset so those rows were deleted. The 5 features including ISBN, title, author, year, publisher. All these features contain some information about the book.

In the ‘user_df’ all 3 features were required, ‘age’ has a lot of null values and a lot of outlier i.e. age above 100 years. The primary objective of this dataset is to know the trend and distribution of age feature so it was performed on the correct values present in the dataset. Other 2 features are ISBN and location, the country, province/state and city were extracted from location.

In the ‘rating_df’, none of the 3 features contains any null values, it contains the information about the ratings given to a book by a user.

All the three dataframes were merged on ‘ISBN’ to create ‘final_rating’ with 8 columns.

EDA and Feature Engineering

While, exploring every feature plenty of inaccuracies displayed including the dissimilar entries in author column, ‘0’ is present ‘year’, data-type of ‘year’ etc. All the inaccuracies ‘year’ were removed but in ‘author’ they were removed upto some extend.

Graphs and charts were plotted to get insight into the dataset. These Graphs including:

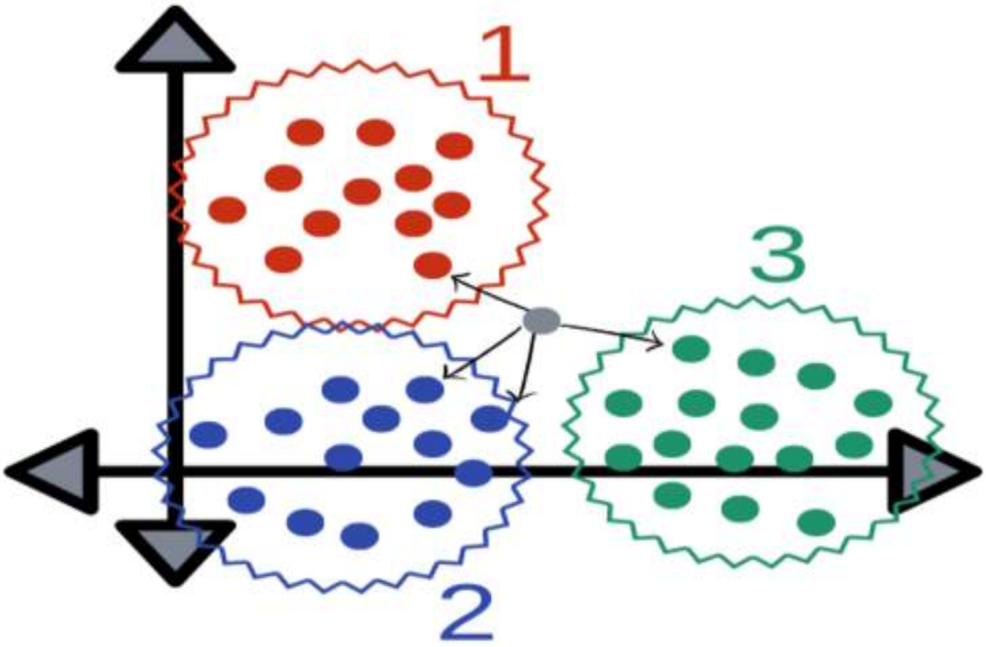
- Line chart to know the age of the users.
- Donut chart to know the origin of users form The USA VS rest of the world.
- Pie chart to know the origin of users from top 10 nations VS rest of the world.
- Bar Graph to know the number of users from top 10 nations.
- Bar Graph to know the number of users from top 10 provinces/states from across the world.
- Bar Graph to know the number of users from top 10 cities from across the world.
- Joint plot to know the distribution of ratings with the number of ratings.
- Bar graph for top 10 authors mentioned in the dataframe.
- Horizontal Bar graph for top 10 books mentioned in the dataframe.
- Horizontal Bar graph for top 10 books received 10 star ratings.
- Horizontal Bar graph for authors to write highest number of books.
- Horizontal Bar graph for average ratings for top 10 author that includes 0 as ratings.
- Horizontal Bar graph for average ratings for top 10 author that excludes 0 as ratings.
- Line graph to know the distribution of average ratings including 0 as rating.
- Line graph to know the distribution of average ratings excluding 0 as rating.

- Horizontal bar graph to know the top-10 publishers with the number of published books.
- Line graph to know the number of books published every year.

Finally, two columns including ‘user_id’ that contains the ID of the user and ‘title’- that contains the book title were required for model implementation.

Model Implementation

k-NN or k-NearestNeighbors algorithm has been used to build the model. KNN is a model that classifies data points based on the points that are most similar to it. It uses test data to make an “educated guess” on what an unclassified point should be classified as.

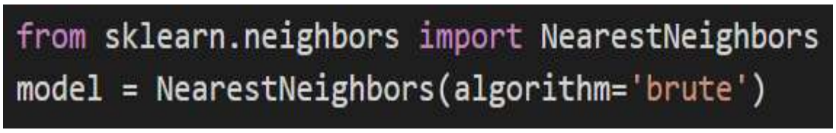


In the above plot the point will be classified as either red or green based on its distance from each group of points. The most common way to find this distance is the Euclidean distance.

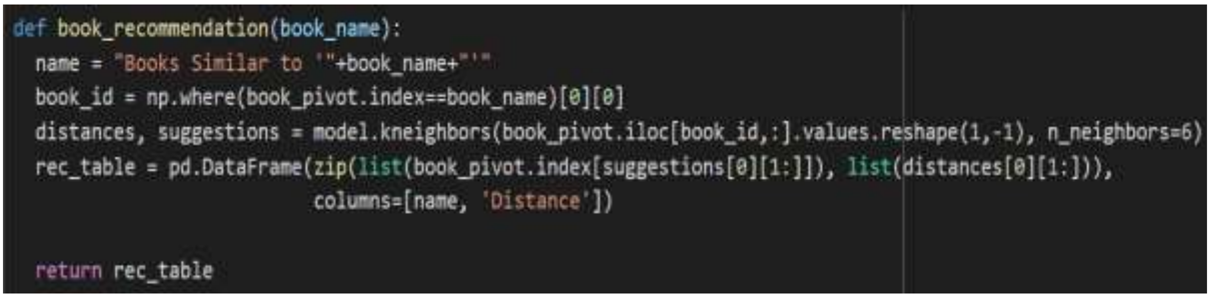
$$\begin{aligned} d(\mathbf{p},\mathbf{q}) = d(\mathbf{q},\mathbf{p}) &= \sqrt{(q_1-p_1)^2+(q_2-p_2)^2+\cdots+(q_n-p_n)^2} \\ &= \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}. \end{aligned}$$

It acts as a uniform interface to three different nearest neighbors algorithms: BallTree, KDTree, and a brute-force algorithm based on routines in sklearn.metrics.pairwise.

Fast computation of nearest neighbors is an active area of research in machine learning. The most naive neighbor search implementation involves the brute-force computation of distances between all pairs of points in the dataset: for N samples in D dimensions, this approach scales as $O[DN^2]$. Efficient brute-force neighbors searches can be very competitive for small data samples. However, as the number of samples grows, the brute-force approach quickly becomes infeasible. In the classes within sklearn.neighbors, brute-force neighbors searches are specified using the keyword algorithm = 'brute'



```
from sklearn.neighbors import NearestNeighbors
model = NearestNeighbors(algorithm='brute')
```



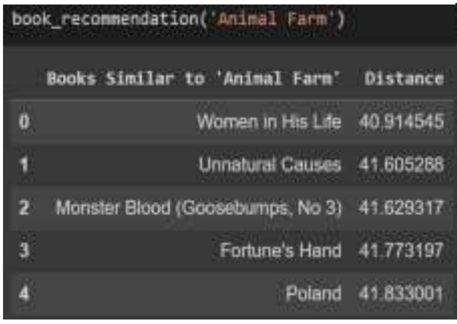
```
def book_recommendation(book_name):
  name = "Books Similar to '"+book_name+"'"
  book_id = np.where(book_pivot.index==book_name)[0][0]
  distances, suggestions = model.kneighbors(book_pivot.iloc[book_id,:].values.reshape(1,-1), n_neighbors=6)
  rec_table = pd.DataFrame(zip(list(book_pivot.index[suggestions[0][1:]]), list(distances[0][1:])),
                           columns=[name, 'Distance'])

  return rec_table
```

Challenges Faced

- Three different datasets were required to make the final dataset. All three were large datasets.
- There are several books with multiple authors but they are different books with common book titles.
- The name of some authors with different publishers has the different format as some entries have a middle name or short form of name.
- There are many 0 in rating columns that reduces the average ratings when considered as actual rating.
- Some books have less number of ratings resulting in unfair rating distribution.
- Contrary to the above point the books with the higher number of ratings will have an unfair advantage in the final recommendation system as compared to those with less number of ratings.

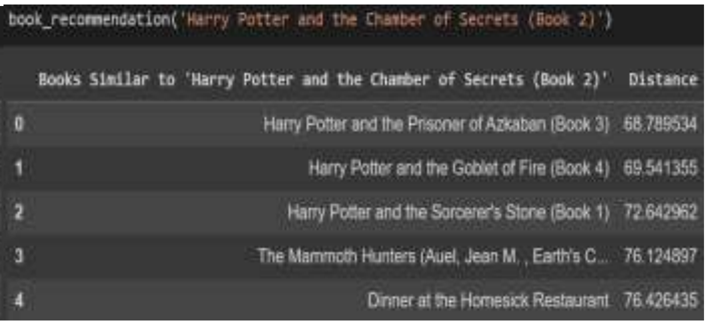
Conclusion

- Model Performance



```
book_recommendation('Animal Farm')
```

	Books Similar to 'Animal Farm'	Distance
0	Women in His Life	40.914545
1	Unnatural Causes	41.605288
2	Monster Blood (Goosebumps, No 3)	41.629317
3	Fortune's Hand	41.773197
4	Poland	41.833001



```
book_recommendation('Harry Potter and the Chamber of Secrets (Book 2)')
```

	Books Similar to 'Harry Potter and the Chamber of Secrets (Book 2)'	Distance
0	Harry Potter and the Prisoner of Azkaban (Book 3)	68.789534
1	Harry Potter and the Goblet of Fire (Book 4)	69.541355
2	Harry Potter and the Sorcerer's Stone (Book 1)	72.642962
3	The Mammoth Hunters (Auel, Jean M., Earth's C...	76.124897
4	Dinner at the Homesick Restaurant	76.426435

- Majority of the users are young.
- North America and Europe are dominating the reader base.
- The USA among the countries, California among states/province and London in cities have the maximum number of readers.
- The average rating stood at 1.8 considering 0 as rating and when 0 is excluded the average rating increased to 7.8.
- The dataset has the collection of book from 1920 to 2010 with higher number of books are published in 2002.
- It has user to rate more than 13 thousand books.
- ‘Stephen King’, ‘Nora Roberts’ are the most famous authors in terms of the average number of ratings received and the number of book wrote by them.
- ‘Harry Potter’ and ‘The Lord of the Rings’ series have received the highest number of 10 ratings.

Reference:

1. GeeksforGeeks
2. https://scikit-learn.org/
3. Analytics Vidhya
4. Stack Overflow
5. https://towardsdatascience.com/
6. Youtube, etc