IEOR 4572 Data Analytics Group Project

Book Recommendation System

Group: Synergy

Youtian Guo (yg2488)
Zhiyi Guo (zg2249)
Sili Wang (sw3159)
Xintong Zhou (xz2555)
Jiachen Zou (jz2790)

May 1st, 2017

Introduction

In this project, we aim to build a web application that allows users to search for books and recommends books based on the search. We retrieved the original dataset from BookCrossing Community. For each ISBN in the data, we did web scraping to extract detailed information for each ISBN from Amazon.com, including price, customer reviews, category, etc. To recommend books, we built k-nearest-neighbor models based on book and user attributes. In our website, we visualized all the analytical results and more importantly, we used Flask to build web framework and to realize the interactions between users and the database by navigating through HTML pages.

Database and Summary Statistics

From Amazon.com, we scraped information of book name, author, introduction, language, category, tags, number of pages, sales price, publisher, ranking, customer rating, and customer reviews, based on ISBN. We stored the data in a JSON file, processed it with python, and stored the data in MongoDB. Our database contains a total of 17406 books with over 1890000 customer reviews. There are 36 different categories, ranging from business, education, science to literature, romance, and hobbies. The rating distribution shows that most of our books received ratings of 4+ stars from customers. We have over 13400 5-star books in our database.

We were also interested in how user traits are associated with book purchases. We extracted city information from our user data and obtained the corresponding latitudes and longitudes from a public database GeoLite, which contains all the geographical coordinates of the world cities. In the user heatmap, we demonstrated the distribution of worldwide user locations. We see that users mainly concentration on Northeast and West coast of the United States. A closer examination of the age distribution among book categories revealed that most readers are young adults and middle-aged people. In categories such as Health, Self-help, and Technology, there are more middle-aged readers than young readers, while in categories such as Politics, Romance, and Textbooks, there are significantly more young readers than older readers.

Web Applications

We used Flask to incorporate features that allow users to interact with our database and get book information in their needs. A user may input a specific book name, or ISBN, or author name under the “Search” tab. If the book or author exists in our database, the web app will return the detailed information of the matching book. The user will see a profile picture of the book of interest, and more information such as author name, category, language, customer rating, publisher, and introduction. The web page also displays word cloud generated from customer reviews of the book so that the user can immediately get a sense of what people are saying most of the book. In addition, we implemented the K-nearest-neighbor algorithm to compute the most similar books, with modeling features of book category, language, etc. Following the basic information of a book, a user will see similar books and choose the ones that may interest them for further exploration. If a book that the user searches does not exist in our database, the web app will return 404 error page and redirect the user to the home or search page.

If the user is not sure what specific books to look at but only know his/her preferences on category and language, the “Filter” tab offers the option to display a grand list of books that belong to the category and language the user chooses. On the result page, the user may click on any one of the books for further information. When browsing the detailed book information, the users can again see a list of similar books generated by our k-nearest-neighbor algorithm.

To have more personalized recommendations, we made use of the known user information and implemented k-nearest-neighbor modeling again to find the most similar users based on their traits. An existing user (whose purchase history is in our database) may input the user ID under the “About you” tab and get a list of recommendations based on the purchase history of the most similar reading peers. If a user does not have a user ID, he/she may simply pick a location and see what people around that location are reading. Our interactive regional map shows the locations of all neighboring users as well as the number of users in each neighboring location. Furthermore, the user will see a word cloud demonstrating the buzzwords about what users are reading in a specific region and will have a thought about what kind of books people around him/her is reading and then go back to the search page to explore.

Finally, if the user would like to randomly browse books to read, he/she may go to the “Recommendation” tab and browse under each general topic of interest. The user can click on “I Want It” under any book and get detailed information about that book as well as a list of similar books to explore.

Conclusion

Our book recommendation web application provides a wealth of information on books and intelligently interact with users based on their book search needs. Areas of improvement include increasing the database of book and user information, allowing for searches based on partial information, as well as user login portal.