Data Collection Report

By Novella Alvina and Steven Tasmin

Project: Book Recommendation System
Database source: https://www.gutenberg.org/ (book detail) and https://zenodo.org/record/2422561#.Y-1LZOzMK3I (book content already in the form of a list of tokenized words without including the unnecessary header)
Book details: title, author, language, categories, released date, number of downloads, book content

How do you obtained your data?

Web scraping is done using requests and beautifulsoup packages from html source code. Requests module functions to get the html source code from the url, then once it's converted into text, it can be made into a soup object with BeautifulSoup module which enables many features like finding a certain tag in the html source code that filters the information that we aim to obtain using find() or findAll() or clean the html tags in the line using get_text().

How Large is your data?

Our dataset currently consists of 69,962 books from various languages, including books that are using non-alphabetic words and characters such as Chinese. To narrow our focus, we plan to filter our dataset to include only books that use alphabetic words and characters. Based on our estimations, we expect that there will be approximately 60,000 books in alphabetical languages that meet these criteria.

In what format are you storing your data?

For data management, 2 dictionaries are implemented. The main dictionary stores the key as a book identifier with the book details as its value. The book details itself is also a dictionary which holds the information of title, author, language, etc mentioned above. This is particularly useful when converting the book details into a csv file producing a table for data frame type in our Jaccard Similarity and Clustering algorithm implementation.

Sample Book Dictionary for each data

{'37106': {'title': 'Little Women; Or, Meg, Jo, Beth, and Amy', 'author': 'Alcott, Louisa May, 1832-1888', 'language': ['English'], 'subject': ['Autobiographical fiction', 'Young women -- Fiction', 'Sisters -- Fiction', 'Domestic fiction', 'Family life -- New England -- Fiction', 'New England -- Fiction', 'Bildungsromans', 'Mothers and daughters -- Fiction', 'March family (Fictitious characters) -- Fiction'], 'released date': '2011-08-16', 'download number': 151924, 'context': ['illustration', 'little', 'women', 'meg', 'jo', 'beth', 'and', 'amy', 'louisa', 'alcott', 'little', 'women']}}

Did you need to process the original data to get it into an easier, more compressed format (e.g., convert from one format to another one)?

From the website's bookshelf page, requests.get() enables us to obtain html source code and beautifulsoup transformed it into an manageable object by the feature='html.parser'. Parsing is done by filtering from html tag <a> with class=link and get attribute href for individual book webpage address as text. Do the same process for each webpage, filter for each detail itemprop respectively, namely "creator" for author and store the values in appropriate data type, such as strings and list(for more than 1 item). Regarding the data pre-process, once the source code is converted into a manageable soup object, there are only minor process done, namely adjusting the data into appropriate data type, like date for released date, etc.