ECOR 1042

Project Report

Interactive Google Books Dataset Analyzer in Python

Submitted by

T004

Morgan Pleet, 101224360

Suren Kulasegaram, 101220595

Zane Labonte-Hagar, 101239002

Jeremy Hornung, 101219284

10/12/2021

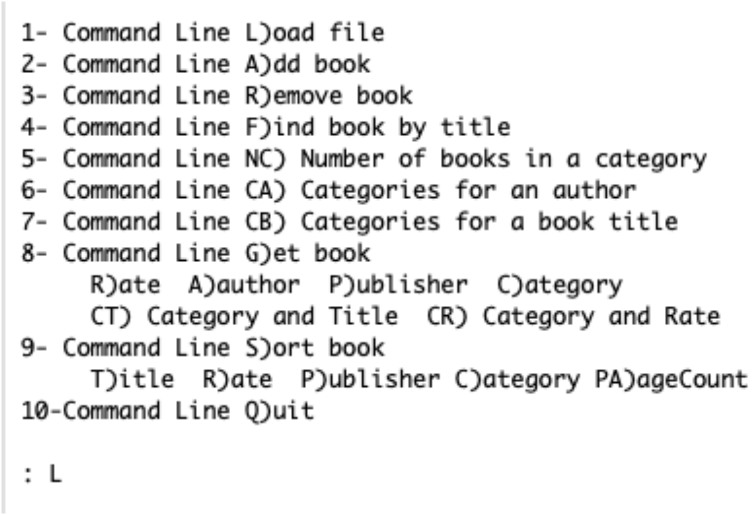
Carleton University,
Faculty of Engineering and Design

1 CONTENT

2 THE PROBLEM STATEMENT:

Given a set of data, organizing and retrieving desired information is often a tedious and frustrating process which may present various problems when data must be manipulated in a timely fashion. Whether the program is used for the purpose of a large-scale business or simply for personal use, when the internet is the primary source of information and a data overload is inevitable, an interactive data analyzer ensures both simplicity and ease of access for the organization of a mass collection of data.

The creation of an interactive program in which data can be grouped, classified and readily available offers the user a valuable tool to improve the efficiency of data analysis. Provided with a set of commands that allow for a given file to be loaded, and thus a dataset to be analyzed in depth, both the time-complexity of organizing large data files and the error prone nature of sorting and retrieving data are avoided. The following figure illustrates the commands available to the user.



```
1- Command Line L)oad file
2- Command Line A)dd book
3- Command Line R)emove book
4- Command Line F)ind book by title
5- Command Line NC) Number of books in a category
6- Command Line CA) Categories for an author
7- Command Line CB) Categories for a book title
8- Command Line G)et book
     R)ate  A)author  P)ublisher  C)ategory
     CT) Category and Title  CR) Category and Rate
9- Command Line S)ort book
     T)itle  R)ate  P)ublisher C)ategory PA)ageCount
10-Command Line Q)uit

: L
```

Figure 1. Example of user input to begin analyzing the data in a given CSV file – Command Line L)oad File

Upon loading the given file, the user will proceed to enter commands, until they wish to quit in which they will enter ‘q’ or ‘Q’ when prompted, and the program will terminate. All possible commands are shown in Figure 1 above.

3 THE PROJECT GOAL

The objective of the project is to develop an interactive program that will allow the user to organize and manipulate a CSV file of their choosing, consisting of various books and their respective book information [1]. The program will prompt the user to load the given file, and subsequently retrieve, search, and sort the data as the user commands.

4 THE PROJECT DESIGN:

The final program will consist of 27 functions which make up the four fundamental steps listed below:

1. A user interface will display a menu of the possible commands the user can make to analyze the provided data. The user will be prompted to enter a command of their choosing, however, until they select the Load File command, none of the other commands, except Quit, will be executed. Upon entering the Load File command, the user is prompted to enter a file name. The provided CSV file must contain a book’s number in the file, to illustrate, the first book in the file will have a “1”, the second will have a “2”, then that is followed by the title, author, rating, publisher, page count, genres, and language, in that exact order. In total there should be eight elements per line.
2. The provided file will be read and processed into a dictionary by a function. The data is organized such that the dictionary’s keys are the genres of the books, and the values to each key are a list of all books that are a part of that genre. Each element of the lists contains another dictionary where the keys are the title, author, rating, publisher, page count, and language, with values as the corresponding information to those keys. The function will return the dictionary and will be assigned to a variable which will hold the organized data for further use.
3. After a file has been loaded, the user can select any of the data analyzing commands, including Load File, to load another file to be analyzed. A function exists to perform each of the available commands. When the user selects a command, they will be prompted to enter the required information, either through another function or an input statement, which are necessary arguments for the command they have selected. The chosen function will then output the organized or refined data to the user’s screen, with the information they desired.

4. The user interface will display the menu between each command initiated by the user. Shall the user wish to quit at any time, entering the Quit command will end the program. The program cannot be terminated if another command is in progress.

5 THE PROJECT PROCESS

There will be 27 functions that need to be developed individually and combined to work as a fully functional program. The functions will be built by a group of four. The first task that must be completed is the function of Step Two as detailed above. The function from Step Two, that reads and processes the data of the given file, is the backbone of the program. All other functions use its processed data to perform their tasks.

Steps One, Three, and Four consist of creating functions that fall into three categories: searching and modifying the data, sorting the data both alphabetically and numerically, and processing information from the user through the interactive user interface. A total of 12 functions will be developed to search and/or modify the data. To sort such data in different manners, another six functions will be created. To read and respond to the information input by the user, another six functions will be built within the interactive user interface, which would be the main menu, or the functions that collect information to be passed as arguments to the search, modify, and sorting functions. The functions that will be used to perform the tasks of the user input within the interactive program will be placed into their own respective files and imported into the interface script.

After the function for Step Two has been completed, each member of the team will build three functions for the search and/or modify category and test three functions made by another group member. Functions will be tested using a CSV file, containing 200 book entries, which is read and processed using the function in Step Two as input. Group members will ensure that the actual output of the function matches the expected output. These functions will be placed in their own file and imported into the user interface script.

Next, the functions for sorting will be created. Two members will build two functions and test one, while the other two members will test two functions and build one. The testing process here is the same as mentioned above. These functions will also be placed in their own file and imported to the user interface script.

Subsequently, the script for the user interface will be developed. Each group member will be responsible for creating the functions that respond to a quarter of the available commands. All commands will then be refactored to work collectively as the complete menu of commands of Figure 1, as a fully functional interactive user interface. As mentioned above, the functions for the user interface are designed to receive user input and process it accordingly. The user interface calls upon three modules which consist of all functions developed prior to the user interface. The user interface testing will ensure that the correct function is called, and the output matches what is expected, when its respective command is entered by the user as input.

Finally, all the files of code are collected into a folder named T004_data_analyzer, and final review is done to ensure that all functions work together as intended. Following the necessary refactoring, the program is set for distribution. The timeline for the Interactive Google Books Dataset Analyzer in Python project is expected to be five weeks.

6 TEAM CONTRIBUTIONS

M. Pleet	Problem Statement, Project Goal
S. Kulasegaram	Project Design, Project Process
J. Hornung	Review, Format
Z. Labonte-Hagar	Review, Format

7 REFERENCES

[1] "ECOR 1042 Project Description." (2021, December 6). *Brightspace*. [Online]. Available: https://brightspace.carleton.ca/d2l/le/content/90081/viewContent/2296333/View.