Python Assignment

Introduction to Software Systems

April 10, 2023

1 Task 1

This task will teach you how to use Python to extract data from websites and create visualizations by plotting the data on graphs.

1. Write a Python script to scrape the top 1000 highest-grossing movies of all time from the IMDB website. After scraping the data, save it to a file named 'q1text.txt' for further analysis. You must also use the data obtained to plot the following graphs:

   (a) A bar graph of the genre to frequency of movies belonging to that genre.

   (b) A line graph of the top 100 movies and their gross value.

2. Create a python program that first prints 100 movies in the descending order of their Imdb rating (if two movies have the same Imdb rating, they must be sorted based on increasing runtime). Additionally, this program should enable the user to filter the list of top 1000 movies based on the following criteria.

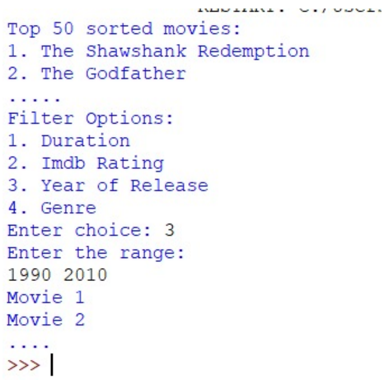
   (a) Duration

   (b) Imdb Rating

   (c) Year of Release

   (d) Genre

   Take an input from the user on which of the criteria they want to filter based on. For the first 3, the user must be prompted a range(i.e. 2 values), and the entries that come within the range must be displayed. For the genre, the user must be prompted to enter a genre and all the movies belonging to that genre must be printed.



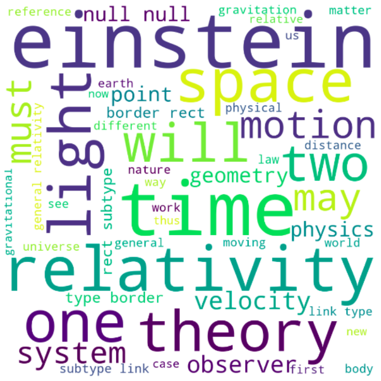
```
Top 50 sorted movies:
1. The Shawshank Redemption
2. The Godfather
.....
Filter Options:
1. Duration
2. Imdb Rating
3. Year of Release
4. Genre
Enter choice: 3
Enter the range:
1990 2010
Movie 1
Movie 2
....
>>> |
```

Note: You are encouraged to use any Python libraries of your choice to complete this task. Additionally, we recommend that you refer to the documentation of the libraries you are using to understand how they work and how to implement them in your code. This will help you develop your skills in working with external libraries and will prepare you for real-world scenarios where you will need to work with unfamiliar libraries.

2 Task 2

In this task, you will learn how to manipulate text data in Python. The objective is to create a word cloud from a given text file by removing common English stopwords. Stopwords are words that occur frequently in a language, such as 'and', 'the' and 'a'. By removing these words, we can focus on the more meaningful words in the text.

A word cloud is a visual representation of text data where the most frequently occurring words are displayed in larger font sizes and are usually placed in the centre of the image. It is a way to visually summarize the most important or common words in a text. For example, if the data is on Einstein and the theory of relativity, you would get a word cloud that looks like this:



Write a Python script to perform the following tasks.

1. First, you must read the two files 'sh.txt' and 'stopwords.txt'. The 'sh.txt' file contains the text that will be used to generate the word cloud. The 'stopwords.txt' file contains a list of English stopwords, one per line. Remove the stop words from the text in 'sh.txt' and create a word cloud with the leftover words.
2. Next, print the most common word in sh.txt (after removing the stop words).
3. Finally, print average word length of the words in sh.txt .

Note: For this task, you must manually remove the stopwords from the given story. You are not allowed to use any libraries or pre-built functions for this task, except for the generation of the word cloud. This will help you understand the process of stopword removal and develop your skills in working with text data in Python. Once you have removed the stopwords, you can use a library of your choice to generate the word cloud.

3 Submission Format

Your submission should consist of the following files.

1. q1.py : Task 1.1, i.e. scraping the website, and saving the data, and printing the graphs.
2. q2.py : Task 1.2, i.e. sorted list of movies, and filter system.
3. q3.py : Task2, i.e. the word cloud, most common word, and average word length.
4. q1text.txt
5. graph1.png
6. graph2.png
7. wordcloud.png

The submission will be automatically downloaded directly from GitHub classroom, so push your progress to your repository. **Note: It is imperative that you adhere to the following submission format for your assignment. Failure to do so may result in a loss of points.**