Networking Project 2 Report

Sumana Endreddy, Srujana Endreddy, Yashika Y
CSC 360-02

Explanation of our overall code:

This Python script represents an innovative approach to understanding online consumer behavior, particularly in the context of product reviews on Amazon. Utilizing Selenium, a sophisticated web automation tool, the script is designed to extract and analyze data from product review pages, offering insights into the authenticity and potential biases of these reviews.

The process begins with the initialization of Selenium WebDriver for Chrome, accomplished through the `init_driver` function. This WebDriver serves as an automated version of the Chrome browser, enabling the script to interact with and navigate through web pages just as a human user would, but in an automated and programmatically controlled manner.

Central to the script's functionality is the `scrape_reviews` function. This function is tasked with visiting a specified product URL on Amazon and collecting data from the reviews present on the page. To ensure accuracy and efficiency, the function employs explicit waits, known as `WebDriverWait`, to allow web elements to fully load and become interactive. The data it collects includes key details such as the names of the reviewers, the URLs of their profiles, and the unique IDs of the reviews they have posted.
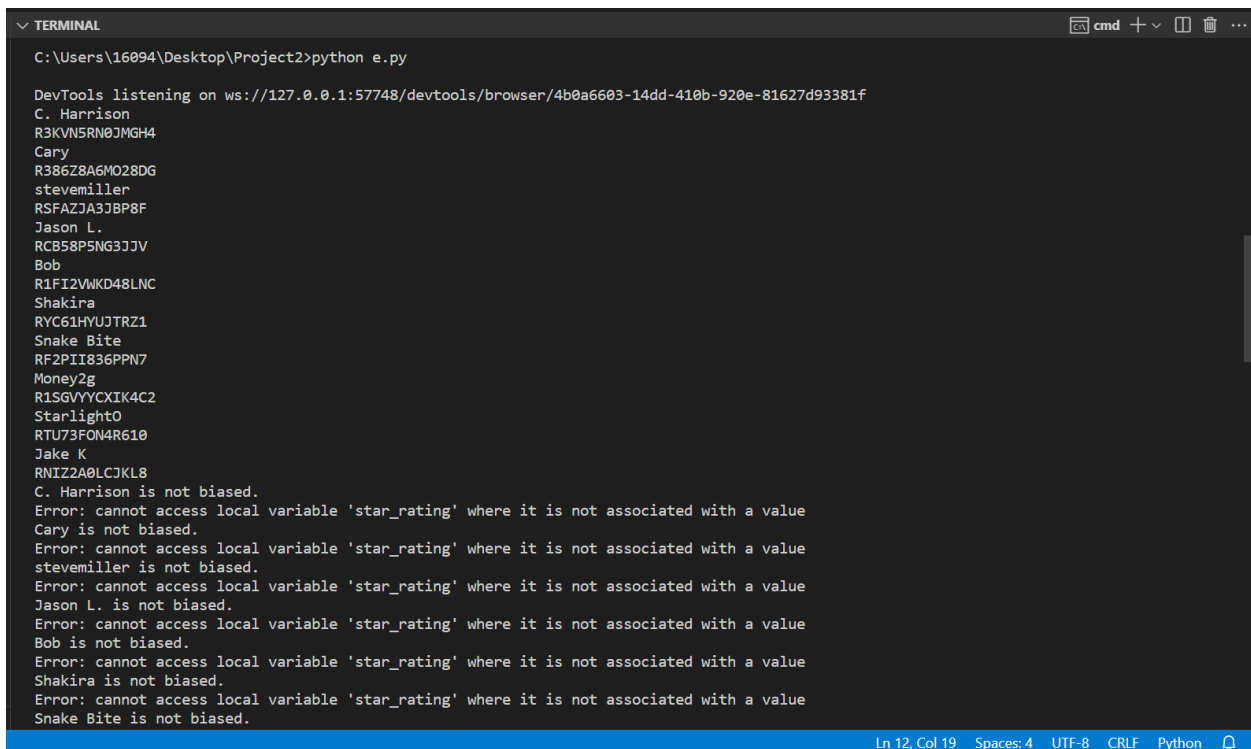
The script then goes deeper with the `scrape_reviewer_profiles` function. This part of the script scans the data gathered from individual reviews, focusing on the profiles of the reviewers themselves. It examines the distribution of star ratings each reviewer has given across all their reviews, searching for patterns that might indicate bias. For example, if a reviewer predominantly gives extreme ratings (either 1 star or 5 stars), this could suggest a lack of objectivity in their reviews.

Building on this analysis, the script proceeds to calculate an average score from the reviews deemed to be unbiased. This stage is vital as it works to remove any extreme scores and potential biases in the overall rating, leading to a more truthful and balanced understanding of the product's acceptance among customers.

The coordination of these functions and the overall execution flow of the script are controlled by the main function.This function specifies the target product URL, initiates the WebDriver, and sequentially activates the review scraping and profile analysis functions. Upon completing these tasks, it responsibly closes the WebDriver. The script is designed to execute as a standalone program, as indicated by the `if __name__ == "__main__":` condition. In this instance, the script is applied to a specific Amazon product, the Bose QuietComfort.

In summary, this script is an example of how web scraping and data analysis can be applied using tools like Selenium to navigate and interact with dynamic web content. It is particularly adept at identifying potential biases in online reviews, offering valuable insights into their reliability , which is increasingly crucial in the digital marketplace where consumer opinions heavily influence purchasing decisions.

Terminal output and its explanation: This program is a python script that uses selenium library to scrape the reviews from the Amazon website (headphones is the product that we used) and analyzes the reviewer bias based on their given star ratings. To calculate the bias, we used a scale that we thought was most appropriate in this situation. During the execution of the program, the program checks the previous reviews made by the reviewer in their profile. If that particular reviewer gave all 5 stars or 0 stars for previous products then it means that the reviewer is biased. Otherwise, if that reviewer gave mixed rating stars for the previous products then it means that the reviewer is not biased. In the screenshot below, it first displays the reviewer's name followed by their ID and informs whether the reviewer is biased or not. In our example, the output displays that the reviewer is not biased because he/she gave mixed star reviews for the previous products that they reviewed. Based on the bias analysis, it is likely that headphones product is not biased.


```
TERMINAL                                                                    cmd + ∨ □ 🗑 ⋯

C:\Users\16094\Desktop\Project2>python e.py

DevTools listening on ws://127.0.0.1:57748/devtools/browser/4b0a6603-14dd-410b-920e-81627d93381f
C. Harrison
R3KVN5RN0JMGH4
Cary
R386Z8A6MO28DG
stevemiller
RSFAZJA3JBP8F
Jason L.
RCB58P5NG3JJV
Bob
R1FI2VWKD48LNC
Shakira
RYC61HYUJTRZ1
Snake Bite
RF2PII836PPN7
Money2g
R1SGVYYCXIK4C2
Starlight0
RTU73FON4R610
Jake K
RNIZ2A0LCJKL8
C. Harrison is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
Cary is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
stevemiller is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
Jason L. is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
Bob is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
Shakira is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
Snake Bite is not biased.

                                            Ln 12, Col 19   Spaces: 4   UTF-8   CRLF   Python  🔔
```

```
Error: cannot access local variable 'star_rating' where it is not associated with a value
Money2g is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
StarlightO is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
Jake K is not biased.
Error: cannot access local variable 'star_rating' where it is not associated with a value
```