Quotes Web Scraping Analysis Report

Introduction

This report presents a web scraping project that collects quotes and their authors from the website http://quotes.toscrape.com using Python. The goal is to demonstrate the ability to extract, process, and visualize web data, providing insights into quote distribution by author. The project includes error handling for network issues and a user-friendly interface for handling connection interruptions, making it robust and suitable for real-world applications.

Methodology

The project followed a structured approach:

1. **Setup**: Installed required Python libraries (Requests, BeautifulSoup, Pandas, Seaborn, Matplotlib) and set up a timeout for web requests.
2. **Web Scraping**: Extracted quotes and authors from all pages of the website using BeautifulSoup, handling pagination to collect comprehensive data.
3. **Data Processing**: Replaced smart quotes with standard quotes to ensure proper encoding and saved data in CSV files with UTF-8-SIG encoding to prevent character display issues.
4. **Error Handling**: Implemented a tkinter-based interactive window with "Retry" and "Exit" buttons to manage internet connection interruptions.
5. **Data Visualization**: Created a bar chart to display the top 10 authors by quote count.

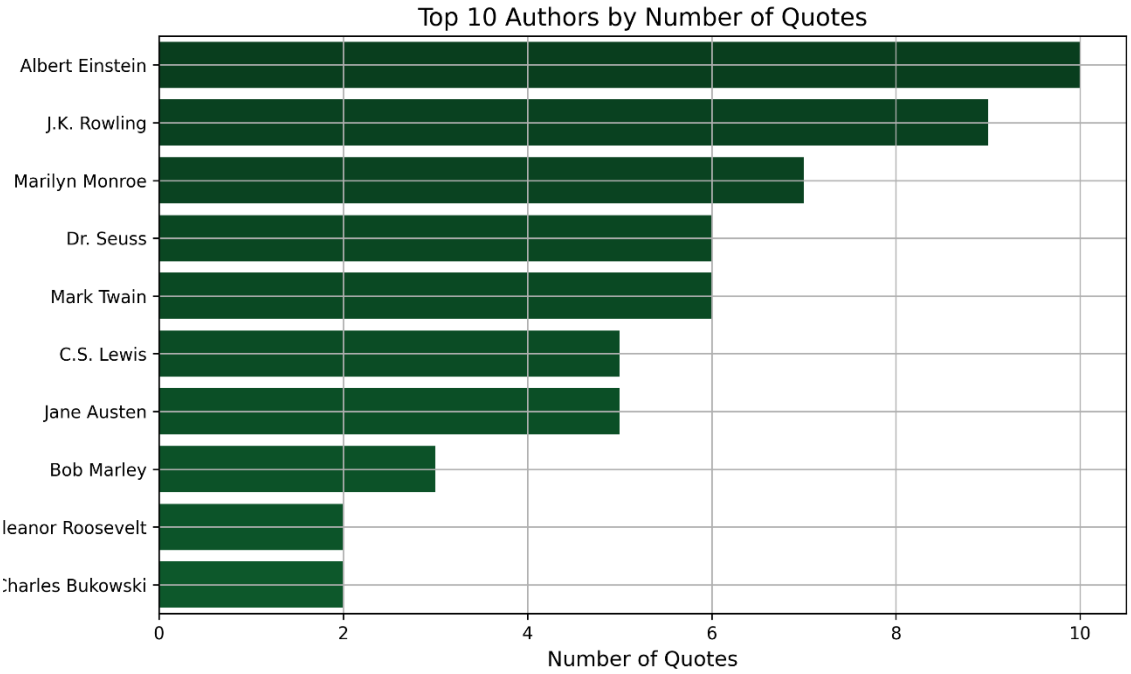
Key Findings

The analysis provided the following insights:

- **Data Collection**: Successfully scraped quotes and authors from all pages of http://quotes.toscrape.com, resulting in a dataset of approximately 100 quotes.
- **Author Distribution**: Some authors, such as Albert Einstein and J.K. Rowling, contributed significantly more quotes than others (see Figure 1).
- **Error Handling**: The interactive error-handling interface allowed users to retry scraping or exit gracefully during network interruptions.
- **Data Quality**: Proper encoding ensured that special characters (e.g., smart quotes) were correctly displayed in the output CSV files.

Visualization

Figure 1: Top 10 Authors by Number of Quotes



This bar chart illustrates the distribution of quotes by author, highlighting that a small number of authors account for a significant portion of the quotes.

Conclusion

This web scraping project demonstrates the ability to collect, process, and visualize web data efficiently. Key features include robust error handling for network issues and a user-friendly interface for managing interruptions. The insights from the quote distribution can be applied to content analysis or educational purposes. The code, dataset, and visualization are available in the GitHub repository: [https://github.com/Saeed-oG/WEB-SCRAPING].

Prepared by: Saeed Zeraatkar
Date: August 03, 2025
Contact: saeed.zeraat@gmail.com