Instructions and Documentation for Sentiment Analysis Solution

Problem Statement

We are tasked with performing sentiment analysis on multiple URLs. The analysis involves:

- Scraping content from websites
- Calculating sentiment scores (positive, negative, polarity, and subjectivity)
- Computing readability metrics (average sentence length, fog index, syllables per word, and more)

The results are saved into an Excel file.

Approach

1. **Data Loading**
    - Read a CSV file containing URLs for scraping
    - CSV is hosted online and accessed directly via a link
2. **Web Scraping**
    - Use `requests` library to retrieve web pages
    - Employ `BeautifulSoup` library to extract article title and content
3. **Text Processing**
    - Tokenization: Break content into words and sentences
    - Stop Words: Exclude irrelevant words using custom and NLTK stop words list
    - Spell Checking: Utilise the pyspellchecker library to correct spelling mistakes in the text. This involves checking each word and replacing it with its corrected form if necessary.
    - Punctuation Removal: Remove punctuation from the text to ensure that only clean words are analysed.
4. **Sentiment Analysis**
    - Count occurrences of positive and negative words
5. **Readability Metrics**
    - Calculate average sentence length, fog index, etc.
6. **Personal Pronoun and Word Metrics**
    - Count personal pronouns
    - Calculate average word length based on cleaned text
7. **Export**

    - Compile results for each URL into a DataFrame
    - Export as an Excel file

Dependencies

Before running the code, install the following Python libraries:

pip install beautifulsoup4 requests selenium pyspellchecker nltk pandas numpy openpyxl

Key Libraries:

- `pandas`: Data manipulation and Excel export
- `requests`: Fetch HTML content from URLs
- `beautifulsoup4`: HTML parsing and content extraction
- `nltk`: Tokenization and stop words handling
- `pyspellchecker`: Correct misspelled words
- `openpyxl`: Save DataFrame to Excel file
    -

Run the Notebook File in Google Colab

1. **Download** the `.ipynb` file
2. **Open Google Colab**
    - Visit Google Colab
3. **Upload the Notebook**
    - Click **File** > **Upload notebook**
    - Choose the downloaded `.ipynb` file
4. **Run the Notebook**
    - Click the play button on each code cell to execute
5. **Export the Output**
    - Add this code to download the result:
      ```
      from google.colab import files
      files.download('article_scores.xlsx')
      ```

Key Assumptions

1. Stop words and sentiment dictionaries from provided URLs are correct and sufficient
2. HTML structure of websites remains constant (title in `<h1>`, content in `<div class='td-post-content tagdiv-type'>`)