**Approach to the Solution:**

1. **Data Extraction:**
    - Utilized the '**requests'** library to fetch content from the provided URLs.
    - Employed '**BeautifulSoup'** for HTML parsing and extracted article text from paragraphs.
2. **Text Analysis:**
    - Performed sentiment analysis using positive and negative word dictionaries.
    - Calculated readability metrics such as average sentence length, percentage of complex words, and fog index.
    - Extracted information like average word length, personal pronouns, etc.
3. **Handling Additional Stopwords:**
    - Integrated various stop words lists (e.g., StopWords_Auditor.txt, StopWords_Currencies.txt) for comprehensive text cleaning.
4. **Output Structure:**
    - Followed the structure outlined in "Output Data Structure.xlsx" for the final output.

**How to Run the .py File:**

1. **Dependencies:**
    - Ensure you have the necessary dependencies installed. Run the following command to install them:

pip install requests beautifulsoup4 nltk pandas syllables

pip install syllables

2. **Execution:**
    - Place the .py file, input data file (Input.xlsx), and additional stopwords files in the same directory.
    - Run the .py file in a Python environment:

python assignment.py

3. **Output:**
    - The script will generate an output file (Output.xlsx or Output.csv) with the computed metrics for each URL.

**Additional Notes:**

- Ensure an internet connection for web scraping.
- Verify that all files are in the same directory to avoid file path issues.