**Assignment - Text Analysis Script Documentation**

**Approach to the Solution**

The solution is designed to perform sentiment analysis on articles retrieved from a list of URLs present in the excel sheet. The key steps involved in the solution are as follows:

* Loading Dependencies and Data:
  + Import the necessary libraries, including pandas, requests, BeautifulSoup, and nltk.
  + Download the required NLTK data packages.
  + Load the stop words, positive words, and negative words into sets for efficient lookup.
* Text Extraction:
  + Now using BeautifulSoup define a function to extract the article title and text from the target div of the webpage accessed through the URL provided.
  + Clean the text by tokenizing it and removing stop words.
* Sentiment Analysis:
  + Define functions to calculate various sentiment metrics, such as the positive score, negative score, polarity score, and subjectivity score.
  + Additional functions are defined to calculate the average sentence length, percentage of complex words, fog index, syllable count, personal pronouns, and average word length.
  + The above scores have been calculated by formulas and specification provided in Text Analysis document.
* Processing URLs:
  + Read the input Excel file containing the URLs.
  + For each URL, extract the article text, clean the text, and calculate the sentiment metrics.
  + Save or update the results to an output Excel file and the extracted text to individual text files.

**How to Run the .py File to Generate Output**

* Install Required Libraries:
  + Make sure you have the following Python libraries installed. You can install them using pip:
  + pip install pandas requests beautifulsoup4 nltk .
  + the requirement.txt file has also been added to install the required libraries and dependencies.
* Download NLTK Data:
  + Ensure that the necessary NLTK packages are downloaded. You can manually download them using the following code snippet:
  + import nltk
  + nltk.download('punkt')
  + nltk.download('stopwords')
* Prepare Directory Structure:
* Verify that the directory structure matches the paths specified in the script. The paths to the stop words and master dictionary files should be correctly set in the script.
* In the python file the relative bath has already been written so that the program can run independent of the system facing no issue of directory to a particular file.

* Run the Script:
  + Save the script as `text \_analysis.py` and execute it using Python:
  + python text\_analysis.py.
* View the Results:
  + The extracted articles will be saved in folder name ExtractedArticles as text files with the `URL\_ID` as the filename.
  + The analysis results will be updated or saved in an Excel file named Output.xlsx.

**Required Dependencies**

* The following Python libraries are required to run the script:
* pandas: Used for reading input Excel files and writing output results.
  + requests: Used for fetching web content from URLs.
  + beautifulsoup4: Used for parsing HTML content.
  + nltk: Used for natural language processing tasks, such as tokenization and stop word removal.
* You can install all the required libraries using the following command:
* pip install pandas requests beautifulsoup4 nltk.
* The requirement.txt file is also saved to download all the libraries used in this project.

This documentation provides all the necessary information to understand the approach, run the script, and ensure that all dependencies are met.