**Overview**

This project aims to extract, analyze, and compile metrics from online articles. The process involves reading article URLs from an Excel file, fetching each article's content via HTTP requests, and performing sentiment analysis and readability assessment on the extracted text. The final results, including various textual metrics for each article, are saved to an Excel file for further analysis or reporting.

**Approach:**

**Data Collection**

1. \*\*URL Loading\*\*: The list of article URLs is loaded from an `Input.xlsx` Excel file.

2. \*\*Content Extraction\*\*: For each URL, an HTTP GET request is made to fetch the article's web page. The title and body text of the article are extracted using BeautifulSoup for HTML parsing.

**Data Preprocessing**

The extracted text is cleaned and tokenized to remove punctuation, convert to lowercase, and exclude stop words, preparing it for analysis.

**Analysis**

1. \*\*Sentiment Analysis\*\*: Calculates positive and negative scores based on occurrences of words from predefined positive and negative word lists.

2. \*\*Readability Assessment\*\*: Evaluates readability using metrics such as the Fog Index, average sentence length, and the percentage of complex words.

3. \*\*Feature Extraction\*\*: Additional textual features like syllable count per word, personal pronouns count, and average word length are calculated.

**Compilation and Output**

The calculated metrics for each article are compiled into a pandas DataFrame and outputted to an Excel file named `Output\_Result.xlsx`.

**How to Run**

To execute the script and generate the output file, follow these steps:

1. \*\*Ensure Dependencies Are Installed\*\*: The script requires Python 3.x and several third-party libraries. Install them using pip:

```bash

pip install requests beautifulsoup4 pandas nltk textblob

```

Additionally, run the following in your Python environment to download necessary NLTK data:

```python

import nltk

nltk.download('punkt')

```

2. \*\*Prepare Input File\*\*: Place your `Input.xlsx` file in the same directory as the script. This file should contain a column of URLs to be analyzed.***(Make sure all the files like Stopwords Folder, positive and negative words file,are in the same directory , our code exist)***

3. \*\*Run the Script\*\*: Execute the script in your terminal or command prompt:

```bash

python your\_script\_name.py

```

Replace `your\_script\_name.py` with the actual name of your Python script file.

4. \*\*Check Output\*\*: Upon successful execution, the `Output\_Result.xlsx` file will be generated in the same directory, containing the analysis results.

**Dependencies Required**

- Python 3.x

- requests: For fetching web pages.

- BeautifulSoup (bs4): For HTML parsing.

- pandas: For data manipulation and saving Excel files.

- nltk: For text processing.

- textblob: For additional text processing and readability metrics.

Ensure all these libraries are installed in your Python environment before running the script.