**Data Science Assignment**

**Submitted to**

**BlackCoffer**

**BY**

Name : **Ujjwal Singh**  
Bhubaneswar  
ujjwalsingh31122000@gmail.com  
+916204303143  
linkedin.com/in/ujjwal-singh  
github.com/US3112

|  |  |  |
| --- | --- | --- |
| **Table of Contents** | | |
| 1 | | Data Extraction |
| 2 | | Sentiment Analysis |
|  | 2.1 | Cleaning using Stop Words Lists |
|  | 2.2 | Creating a Dictionary of Positive and Negative Words |
|  | 2.3 | Extracting Sentiment Variables |
| 3 | | Analysis of Readability |
|  | 3.1 | Gunning Fog Index Calculation |
|  | 3.2 | Average Sentence Length |
|  | 3.3 | Percentage of Complex Words |
| 4 | | Linguistic Features |
|  | 4.1 |  |
|  | 4.2 | Average Number of Words Per Sentence |
|  | 4.3 | Complex Word Count |
|  | 4.4 | Word Count |
|  | 4.5 | Syllable Count Per Word |
|  | 4.6 | Personal Pronouns |
|  | 4.7 | Average Word Length |
| 5 | | Running the Script |
| 6 | | Dependencies |

# **Data Extraction**

### **Objective**

To extract the main content (title and article text) from each URL listed in the Input.xlsx file and save it in individual text files, named according to the URL\_ID.

### **Methodology**

* **Library Used:** requests, BeautifulSoup
* **Steps:**

1. Read URLs from Input.xlsx.
2. Send an HTTP request to each URL.
3. Parse the HTML content using BeautifulSoup.
4. Extract the article's title (<h1> tag) and the main body of the article (inside <p> tags).
5. Save the extracted content in the TitleTexts directory with the file name as URL\_ID.txt.

# **Sentimental Analysis**

The objective is to assess the sentiment of each article by calculating sentiment scores derived from the frequency of positive and negative words. This sentiment analysis process classifies the text as positive, negative, or neutral, particularly tailored for financial content.. It consists of steps:

## **Cleaning using Stop Words Lists**

The Stop Words Lists (found in the folder StopWords) are used to clean the text so that Sentiment Analysis can be performed by excluding the words found in Stop Words List.

## **Creating a dictionary of Positive and Negative words**

The Master Dictionary (found in the folder MasterDictionary) is used for creating a dictionary of Positive and Negative words. We add only those words in the dictionary if they are not found in the Stop Words Lists.

## **Extracting Sentiment Variables**

We convert the text into a list of tokens using the nltk tokenize module and use these tokens to calculate the 4 variables described below:

· **Positive Score:** The count of positive words in the text.

· **Negative Score:** The count of negative words in the text.

· **Polarity Score:** Indicates the overall sentiment of the text. Calculated as:

Polarity Score = (Positive Score−Negative Score) / (Positive Score+Negative Score+0.000001)​

· **Subjectivity Score:** Reflects the degree of personal opinion in the text. Calculated as:

Subjectivity Score = (Positive Score+Negative Score) / (Total Words after cleaning+0.000001)​

# **Analysis of Readability**

Analysis of Readability is calculated using the Gunning Fox index formula described below.

**Average Sentence Length** = the number of words / the number of sentences

**Percentage of Complex words** = the number of complex words / the number of words

**Fog Index** = 0.4 \* (Average Sentence Length + Percentage of Complex words)

# **Linguistic Features**

### **Objective :**

### To analyze the text's structural and linguistic features.

### **Average Number of Words Per Sentence :**

The formula for calculating is:

**Average Number of Words Per Sentence =** the total number of words / the total number of sentences

### **Complex Word Count**

* **Definition:** The number of words in the text that contain more than two syllables.

### ****W**ord Count**

* **Definition:** The total number of words in the text after cleaning.
* **Methodology:**

1. Remove stop words and punctuation.
2. Count the remaining words.

### **Syllable Count Per Word**

* **Objective:** Count the average number of syllables per word.
* **Methodology:**

1. Count vowels in each word.
2. Handle exceptions like words ending with "es" or "ed."

### **Personal Pronouns**

* **Objective:** Count the occurrence of personal pronouns (e.g., I, we, my, ours, us).
* **Methodology:** Use regex to count occurrences, ensuring that "US" (as in the country) is not included.

### **Average Word Length**

* **Objective:** Compute the average length of words in the text.
* **Calculation:** Average Word Length = Sum of the Total Number of Characters in Each Word / Total  Number of Words​

# **Running the Script**

* Steps to Execute the Python Script:

1. **Preparation:** Ensure all input files (Input.xlsx, Output Data Strctures, MasterDictionary, StopWords) are available in the same directory as the script.
2. **Execution:** Run the script in a Python environment like Google Colab or any local Python environment.
3. **Output:** The script will generate the following outputs:

* Extracted text files in the TitleTexts directory.
* An output CSV file named Output\_Data.csv containing the computed variables.

# **Dependencies**

* **Python 3.x**
* **Libraries:** requests, BeautifulSoup, pandas, nltk, re
* Installation :

1. install the required Python libraries using pip:
2. Code -> pip install requests beautifulsoup4 pandas nltk