**INSTRUCTIONS**

**Introduction:**

Top of FormTextual analysis is a multifaceted and indispensable method employed in various fields to gain a deeper understanding of written or spoken texts. Its purpose extends beyond mere comprehension, delving into the realms of interpretation, critique, and uncovering underlying meanings. Through systematic examination of linguistic, structural, and contextual elements, textual analysis aims to unveil the layers of significance embedded within a text.

The scope of textual analysis is vast and adaptable, as it can be applied to diverse types of texts, including literature, speeches, advertisements, legal documents, social media posts, and more. It provides a toolkit for researchers, scholars, and analysts to decipher the intentions of authors, the cultural and historical contexts shaping the text, and the potential impact on various audiences.

**Framework used**: Jupyter Notebook

**Input Used**: Input.xlsx which provided URL\_ID and URL

**Extracted Output**: With the reference of given output\_data\_structure.xlsx, output had been generated accordingly.

**Functionality:**

The functionality of conducting textual analysis from multiple URLs involves systematically extracting, processing, and analyzing text content from various web sources to derive insights, patterns, and valuable information.

**Data Collection**: URLs were provided in the Input.xlsx file to use.

**Web Scraping**: Automated tools or scripts can be used to extract the text content from the URLs. This involves navigating to the web page, retrieving the HTML content, and then parsing and extracting the textual components from the HTML structure.

**Text Pre-processing**: The extracted text often contains HTML tags, advertisements, navigation menus, and other irrelevant elements. Text pre-processing is performed to clean and normalize the text, removing non-textual elements and formatting inconsistencies.

**Removal of Stopwords & Punctuations:** It is a text pre-processing technique where provided stopwords files that carry little contextual meaning are eliminated from a text document. This helps streamline the analysis process by focusing on more meaningful words that contribute to the core content. By removing stopwords, the resulting text retains its essential information and context, making it easier to identify significant terms, patterns, and themes during subsequent text analysis tasks. Punctuations mentioned in the provided Text Analysis file had been removed

**Tokenization**: The cleaned text is divided into individual words or tokens. Each token is used for whether it exist in positive and negative words of given text file. Then the dictionary had been created for the positive and negative words for the every URL links.

**Sentiment Analysis**: Sentiment analysis can be performed to determine the emotional tone of the text. This involves classifying the text as positive, negative, or neutral, which can be valuable for gauging public opinion, customer feedback, and brand perception.

**Keyword Extraction**: By identifying and extracting positive and negative keywords, you can understand the main themes and topics discussed in the text. This aids in categorizing and summarizing the content.

**Textstat**: It is a Python library designed to provide a range of useful text statistics and readability measures for analyzing text documents. It offers a simple and convenient way to extract insights about the complexity, readability, and structure of textual content. By using Textstat, developers and researchers can assess the difficulty level of a text, estimate the reading age required to understand it, and gain a deeper understanding of its linguistic characteristics.

The library offers functionalities such as word count, sentence count, average word length, and syllable count, which are crucial for assessing text complexity. These measures are valuable for tailoring text to specific audiences and improving communication effectiveness. Textstat's user-friendly interface and comprehensive metrics make it an essential tool for anyone seeking to perform text analysis, assess writing quality, or enhance content readability in Python-based projects.

**Dataframe**: Creating a Dataframe from all extracted contents is a common task in data analysis and text processing, especially when you have collected and pre-processed textual data from various sources. A Dataframe is a structured data object provided by libraries like Pandas in Python, which allows you to organize, manipulate, and analyze tabular data efficiently.

**Insight Generation**: By combining the results of various analyses, you can generate valuable insights about trends, opinions, emerging topics, and potential areas for further exploration.Incorporating textual analysis from multiple URLs can provide a comprehensive view of how certain topics, sentiments, and themes are distributed and discussed across the web. It's a powerful approach for understanding public discourse, market trends, research topics, and more, within the vast realm of digital content.

**PS:** I have submitted the Python file, output excel file in required format and Instructions document where I mentioned the overview of processes that took part in this tast. I faced many errors while doing this task, so the coding part had been crafted according to my laptop condition and limitations.

**Best Regards,**

**Dinesh Kumar D**

**For contact: +91 9360162075**

**E-mail Address: dinesh.duvd@gmail.com**