**Project Documentation**

**Project Overview**

This project is a Python-based application designed to extract and analyze text content from URLs provided in an Excel file. The tool processes each URL, extracts the article content, saves it as .txt files, and calculates key metrics like sentiment scores using predefined word dictionaries. The results are merged with the input data and saved in a CSV file. Additionally, the application has been deployed using **Streamlit**, making it user-friendly and accessible through a web interface.

**Instructions**

**1. Approach to the Solution**

* **Input Handling**: The application takes an Excel file as input, which contains a list of URLs and their corresponding IDs.
* **Content Extraction**: For each URL, the article content is extracted using the *Newspaper3k* library and saved as a .txt file in the output folder.
* **Text Preprocessing**: The extracted articles are preprocessed by removing stopwords and unnecessary characters.
* **Metrics Calculation**: Using positive and negative word dictionaries, the tool calculates sentiment metrics such as positive score, negative score, and polarity score.
* **Output Generation**: The calculated metrics are merged with the original input data and saved as a new CSV file.
* **Streamlit Deployment**: The project is deployed on Streamlit to provide an interactive interface for users to upload files and view results.

**2. How to Run the Application**

**Running Locally**

1. **Set Up the Environment**:
   * Ensure Python 3.x is installed on your system.
   * Install the required dependencies listed in the requirements.txt file using: pip install -r requirements.txt
2. **Run the Application**:
   * Navigate to the project directory.
   * Start the Streamlit app by running: streamlit run app.py
   * This will open the Streamlit interface in your default web browser.
3. **Using the Application**:
   * Upload the Excel file containing URLs via the Streamlit interface.
   * The application will process the URLs, extract content, calculate metrics, and display the results.
   * Download the output CSV file directly from the interface.

**Output Structure**

* Extracted articles are saved in the output folder as .txt files.
* The final metrics are saved as a CSV file named extraction\_to\_output.csv.

**3. Dependencies Required**

The following libraries and modules are required for the project:

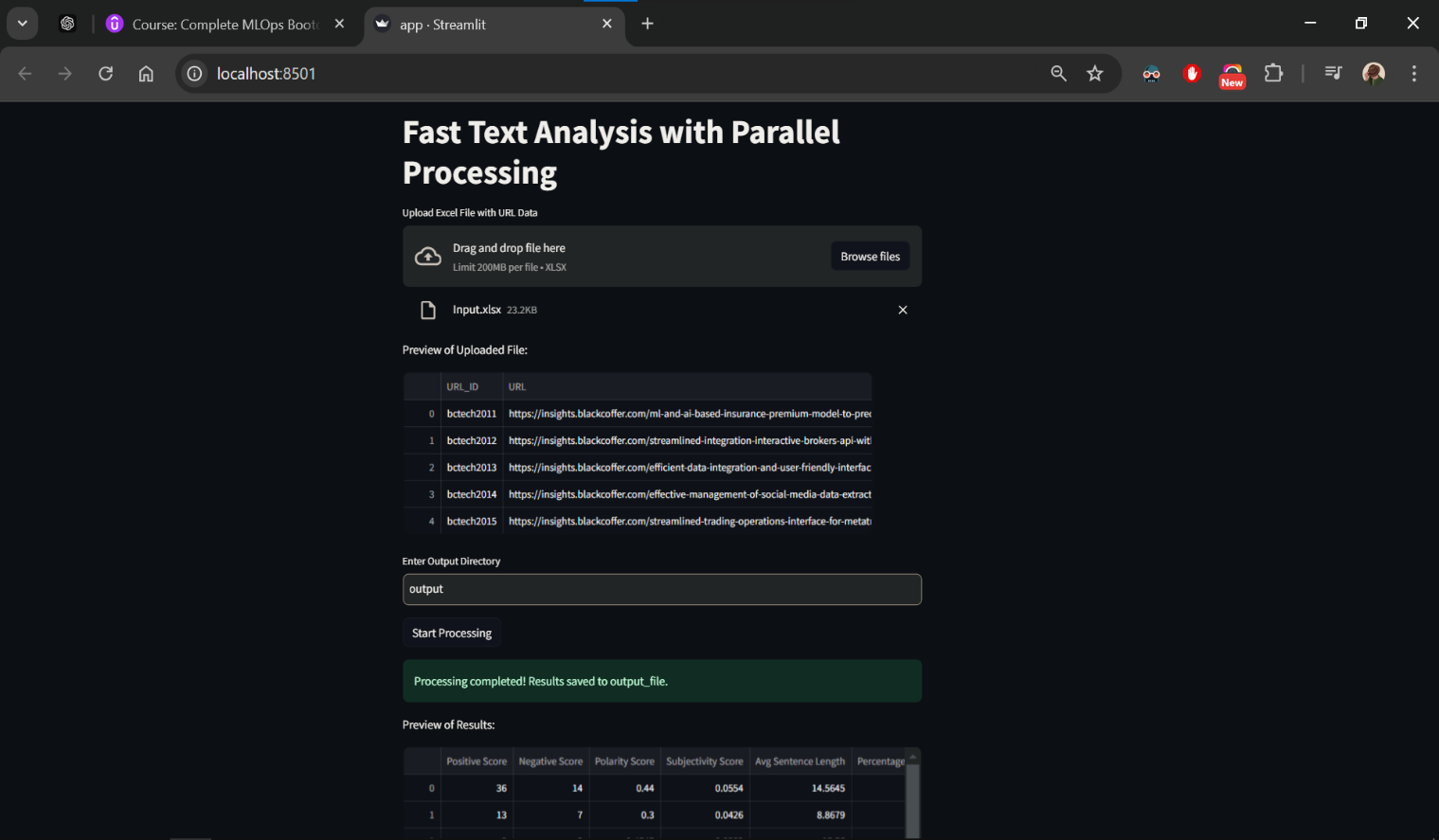
* **os**: For file and directory handling.
* **pandas**: For data manipulation and merging input with metrics.
* **tqdm**: For progress bars during processing.
* **openpyxl**: For handling Excel files.
* **newspaper3k**: For article content extraction.
* **nltk**: For text preprocessing and stopwords.
* **Streamlit**: For deploying the application as a web app.
* **concurrent.futures**: For parallel processing of URLs.
* **logging**: For error handling and logging events.

Ensure all dependencies are installed by running: pip install -r requirements.txt

**Key Features**

* **Automated Content Extraction**: Extracts articles from a list of URLs efficiently.
* **Sentiment Analysis**: Calculates various metrics like positive and negative scores.
* **Streamlit Deployment**: Provides an interactive web interface for easy usage.
* **Scalable Design**: Handles large datasets with multithreading support.

### application ui



Thank you 😊

Name : SHINU RATHOD

Gmail : [shinukrathod0143@gmail.com](mailto:shinukrathod0143@gmail.com)

LinkedIn : <www.linkedin.com/in/shinu-rathod-608871216>

GitHub : <https://github.com/SHINU4RATHOD>