Hardik Polamarasetti

DSA 469

Big Data and Big Data Processing Systems

3 May 2024

Implementing a Content-Based Document Search Engine with Machine Learning

Introduction

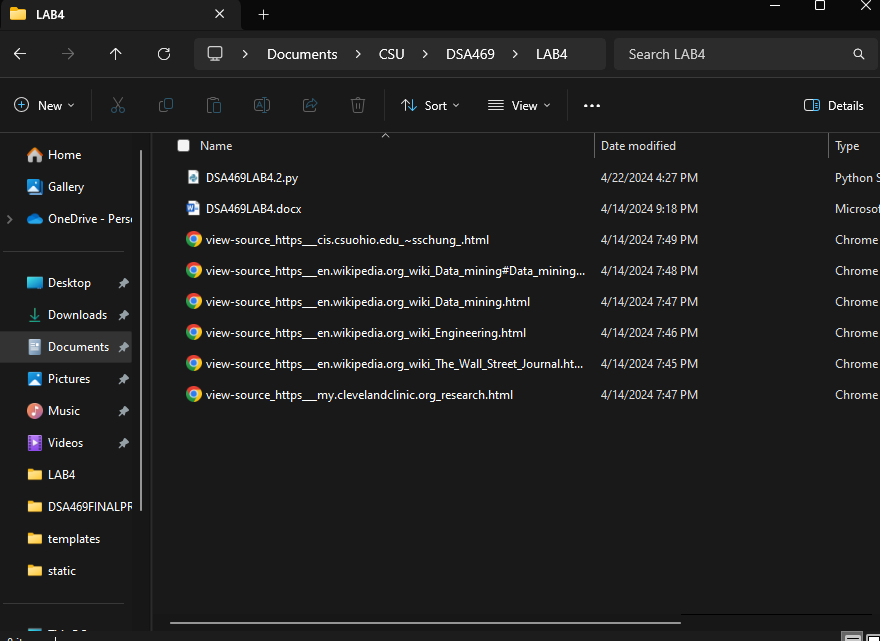
The objective of this project was to develop a content-based document search engine capable of analyzing and retrieving HTML documents based on user queries. This search engine utilizes machine learning techniques to enhance the accuracy and relevance of search results, making it a powerful tool for navigating large datasets of documented information.

Project Overview

The project focused on creating a user-friendly interface for uploading, storing, and retrieving HTML documents. A text processing pipeline was implemented to extract text and vectorize it from HTML documents. The main basis of the search algorithm is based on TF-IDF (Term Frequency-Inverse Document Frequency) and cosine similarity to rank documents based on their relevance to user queries.

Data Description and Collection

HTML documents that the user inputs form the basis of the dataset. These documents were preprocessed to extract clean text, such as removing HTML tags using BeautifulSoup. The text data was then vectorized using the TfidfVectorizer from Scikit-learn, transforming the text into a numerical format suitable for similarity computations.

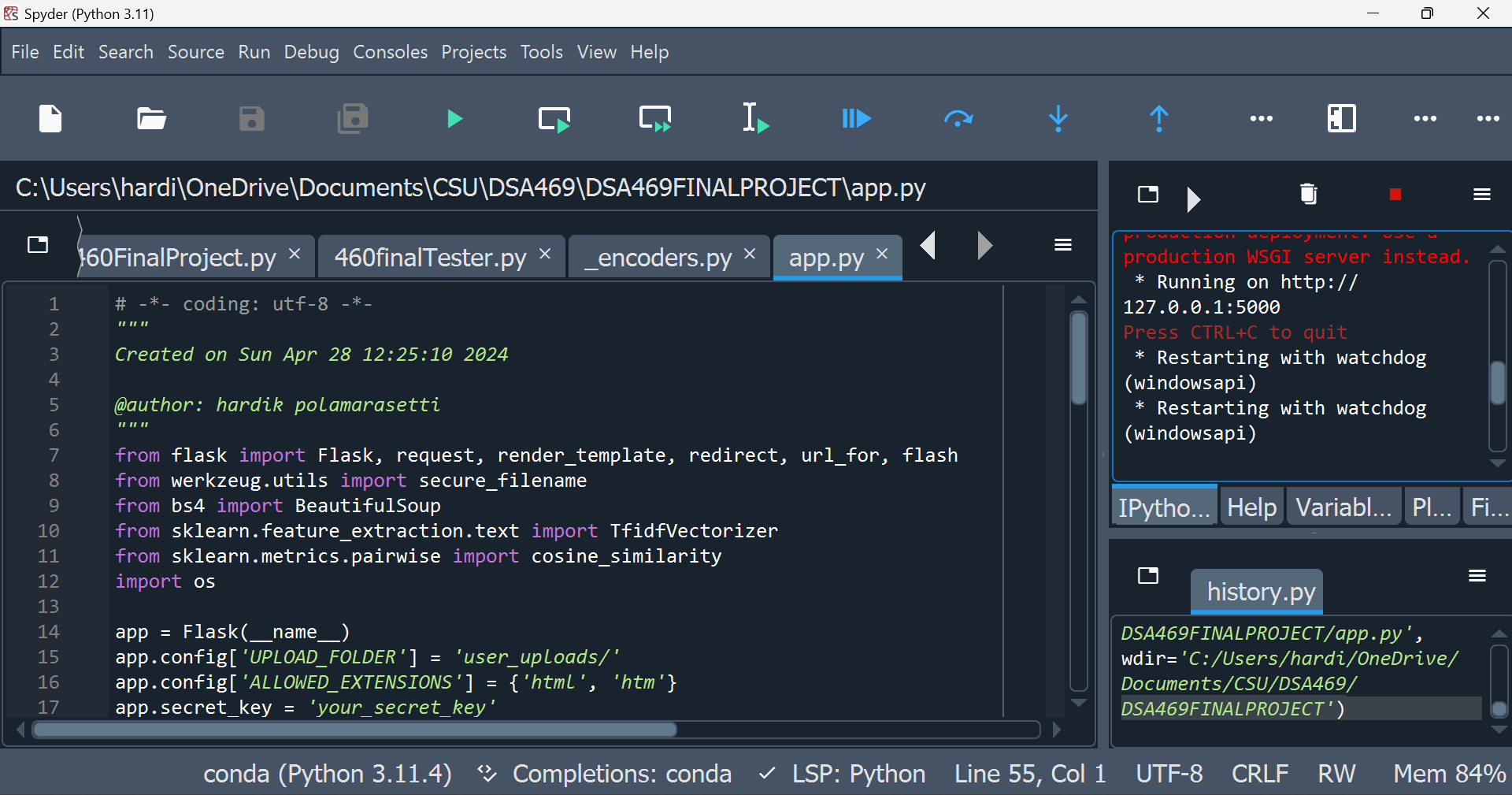


Goals of the Project

The primary goal was to provide a user friendly, efficient search engine that allows users to retrieve documents based on content relevance quickly. This involves processing and indexing documents in such a way that they can be searched through efficiently, proving the engine's capability to handle big data with accuracy and speed.

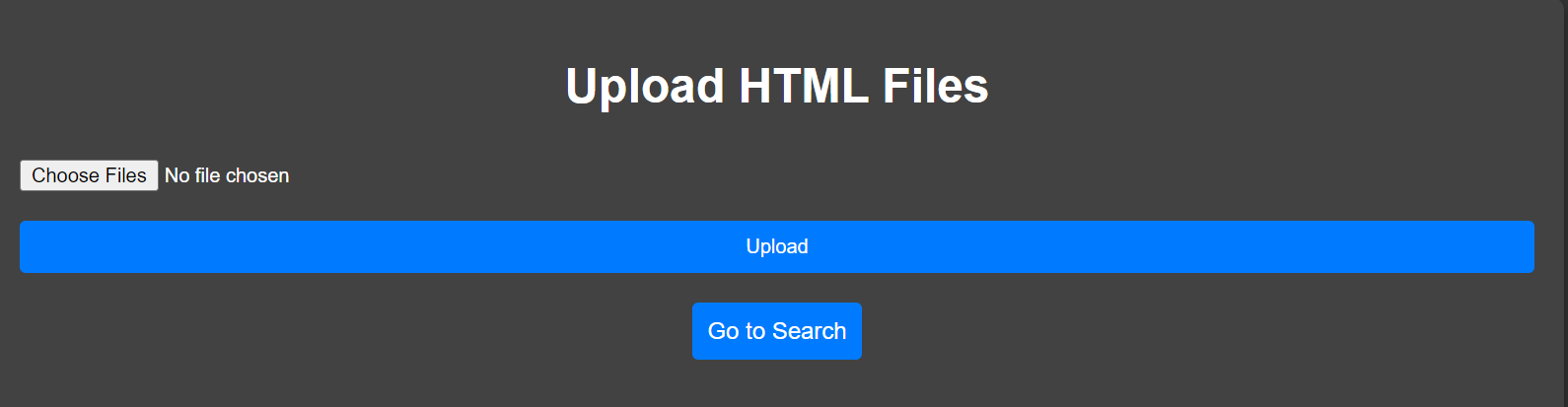
System Design and Implementation

The system was designed with a front end implemented in HTML and CSS to manage user interactions and display results. The backend was developed using Flask and handles data processing, including the uploading of documents, and interacts with the search engine to fetch and display results based on the queries submitted.

****

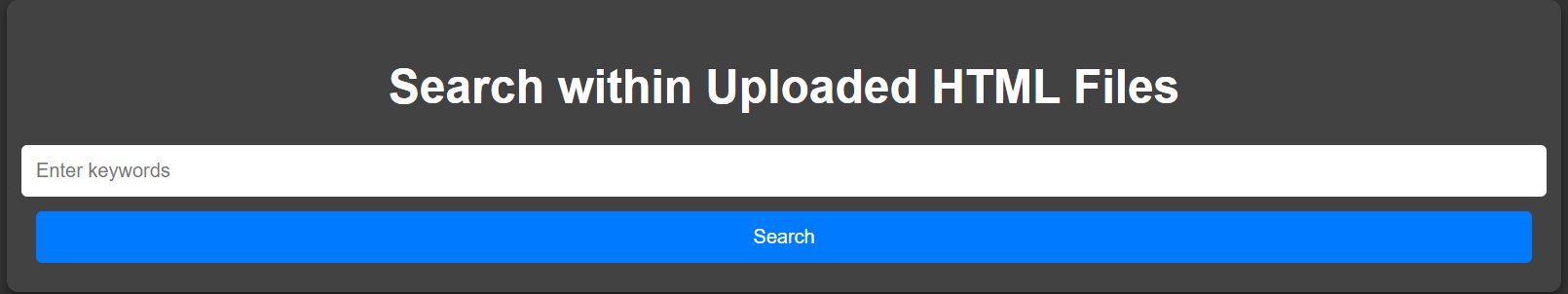
Data Preprocessing and Vectorization

Data preprocessing involved using BeautifulSoup to extract text from HTML files, which were then processed through a pipeline that included cleaning steps such as removing special characters and stop words, and applying TF-IDF to vectorize the text. This process was crucial for reducing noise in the data and focusing on meaningful content, which is essential for effective search results.

****

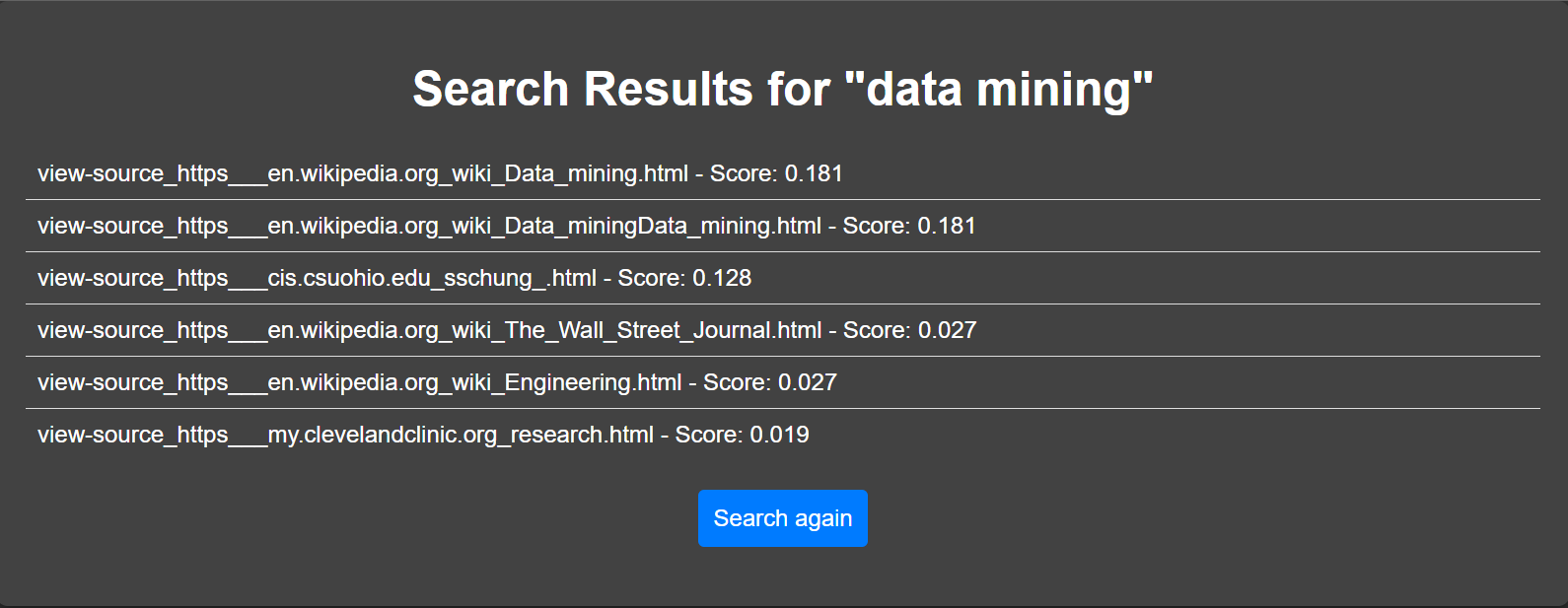
Search Mechanism and ML Implementation

The search mechanism utilized a TF-IDF model to transform queries into vector forms that could be compared with document vectors stored in the system. Cosine similarity scores were then calculated to determine the relevance of each document to the query, which allowed for ranking the documents by relevance.

****

Results and Evaluation

The system was evaluated based on its ability to accurately retrieve documents that are relevant to the user queries. Results were displayed to the user in a ranked order, providing a clear and effective overview of the most pertinent documents.

****

Challenges and Resolutions

One of the main challenges was optimizing the search engine to handle real-time queries efficiently. This was addressed by optimizing the document processing pipeline and ensuring that documents were only re-vectorized when new uploads were made, rather than on every search, which significantly improved performance.

Conclusion and Future Work

The project successfully demonstrated the application of machine learning techniques in creating a functional document search engine. Future enhancements will focus on scaling the system to handle a larger dataset, implementing more advanced ML models for better accuracy, and potentially extending the system to include real-time web scraping capabilities.

This comprehensive report reviews and analyzes the methodology, implementation, and outcomes of the project, providing a detailed overview of the system's capabilities and the approach used. The project not only showcased the practical application of big data processing techniques but also set a foundation for further development in content-based document retrieval systems.