Adaptive Web Assignment 3

Santosh Bidve

sbidve@asu.edu | 1211219006

Summary

Collection of Data

- Implemented Crawler in Python using Beautiful soup, recursively traversing all the links from the main page
- Collected the output in different JSON files with title and content as keys.

Indexing the Content

- For indexing, loaded the data in Elasticsearch, it uses Apache Lucene as underlying implementation for indexing.
- Insert all the JSON files as bulk input with the index mapping done using parameters BM25 and English analyzer.

Web app

- You will find the list of 10 posts below with their respective recommendations which are queried from Elasticsearch. The AJAX recommendation API is implemented in Express.

Originality

- The interactive and response UI is developed in Materialize framework. Made use of the collapsible(accordion) element from it.
- Using NLTK library in python I have tokenized words, performed stemming, removed repeating words and characters, and extracted nouns using Regex, extracted leaves from each generated tree as list of meaningful words which is later used in data.js file.
- Performed indexing in Elasticsearch using similarity algorithm BM25, also applied the English analyzer in index mapping while building of indices.

- BM25 Algorithm: In information retrieval, BM25 (BM stands for Best Matching) is a ranking function used by search engines to rank matching documents according to their relevance to a given search query (term frequency). It uses probabilistic approach.

Instructions to run:

1. Go to “ElasticSearch/bin/” folder and run elasticsearch.exe.
2. Make sure the ElasticSearch service is running by opening http://localhost:9200.
3. In the project root directory, run → **npm start**
4. Open http://localhost:3000 and wait for a second to load the results.
5. The layout is an accordion, click on each post to see its top 10 recommendations.