INFORMATION RETRIEVAL (CS F469) ASSIGNMENT 1

Domain Specific Information Retrieval System

Members

Nikhil Joshi 2015A7PS0179H

Nikhil Madaan 2015A7PS0283H

Bhavesh Gawri 2015A7PS0116H

GETTING DATASET:

The dataset is from wikipedia articles in both english and arabic language. Dataset contains more than 100000 articles in .txt format. The dataset obtained was in compressed version which was further extracted into individual articles.

Functionality Implemented:

* Searching top 10 article based on query given
* Comparing Similarity Between two Articles.
* Wilcard Query
* Term Auto-completion

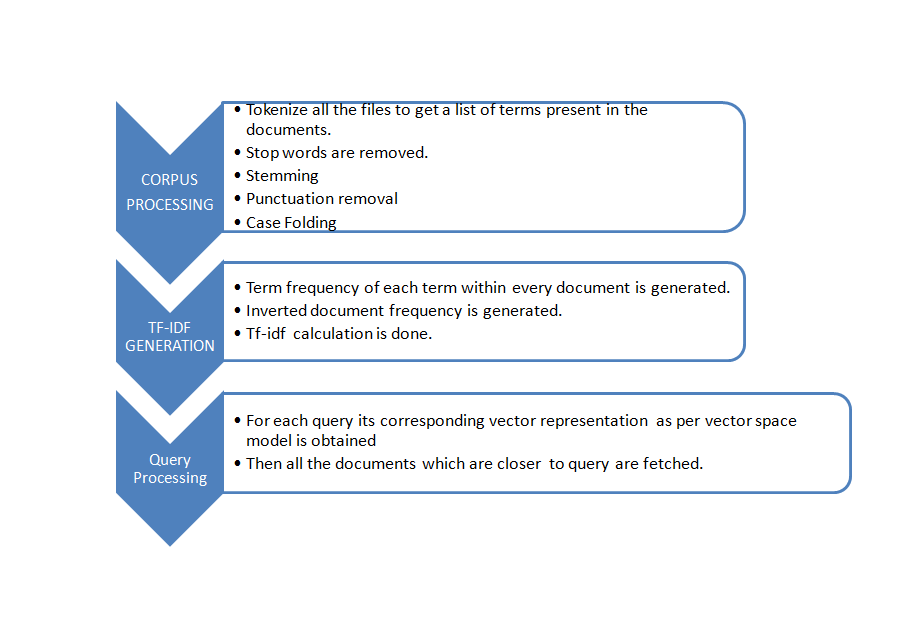
MAJOR DATA STRUCTURES USED

* Trie:

Different Trie were used to implement functionality of wildcard query and auto complete. Different trie were made for all corresponding english letter alphabets and symbols

* list in python – List in python were used to manipulate data.
* Dictionary in python - Dictionary in python is equivalent to Hash tables

DESIGN ARCHITECTURE:



BRIEF NOTES ON CODE EXECUTION:

* Input from compressed wikipedia articles and extracted using wikiextractor tool
* Above uncompressed file were futher processed to obtainn indivisual articles
* These Individual articles were tokenized, removing of stop words, stemming , punctuation removal and finally case folding were respectively implemented.
* Parallely we calculate and save tf of corresponding term in a document
* generating inverted index was the next process which was further followed by calculation of idf.
* File corresponding to tf-idf and after normalization is obtained into a dictionary object
* Input Query is processed and vector corresponding to query is generated and cosine similarity is generated with respect to all the documents
* Auto correction and wild card queries were implemented using trie.
* Further tf-idf data were used to calculate similarity between given documents and document in corpus.

RUNNING TIME

* For Preprocessing: 5hrs

Includes time for extacting uncompressed articles, tokenizing , stemming , removal of stop words etc. and finally calculating tf-idf and obtaining inverted index.

* For Search Retrieval: 0.8 sec