TEAM MEMBERS:

G.Sai Divya(S20180010059)
K.Keerthana(S20180010084)

INFORMATION RETRIEVAL

PROBLEM STATEMENT:

The aim of the project is to collect various sports articles from thousands of websites and retrieving the top 10 documents most relevant to the query.

DATASET:

Collection of 1000 sports articles based on seed URL.

TASKS PERFORMED:

1.Web Crawling

2.Creating Inverted Index

3.Searching

4.Ranking

1.Web Crawling:

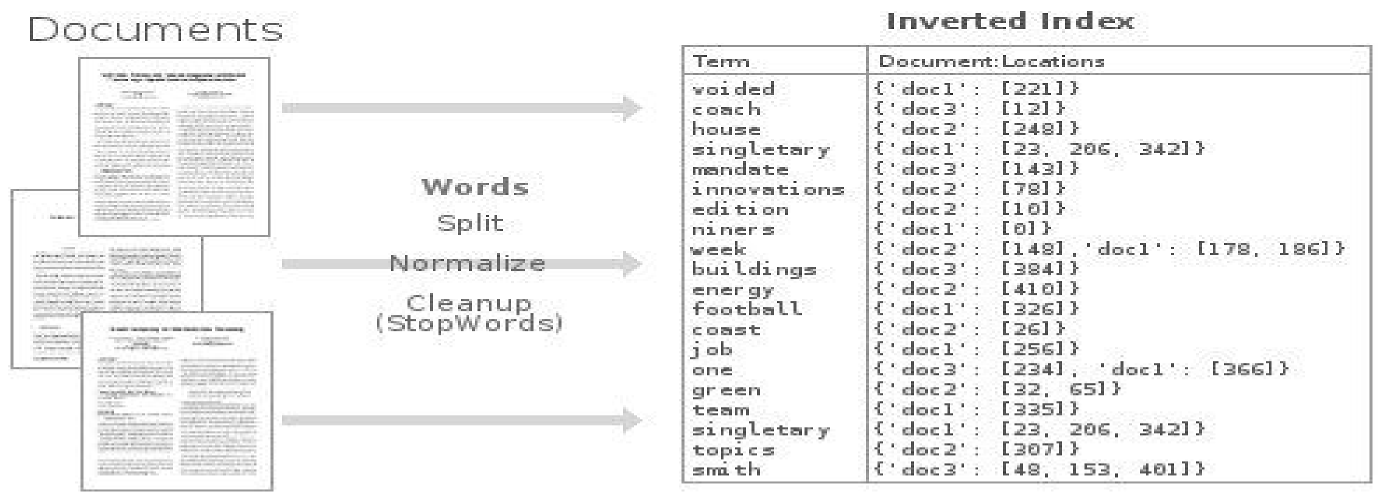
In this task, we provide seed URL = https://en.wikipedia.org/wiki/Sport

Using the BFS algorithm takes the seed URL and gathers all links in the seed URL and adds them to the queue. The above process is repeated with the remaining items of the queue. Finally, list all URLs in Crawled Urls.txt in the form of "Depth: 1, Rank: 1, URL: https://en.wikipedia.org/wiki/Sport"

So even if a crawled link appears again at another depth, we already have gathered all links on the page. After gathering links in Crawled Urls.txt we scraped information from URLs to the Downloaded Files folder.

2.Created Inverted Index:

Firstly download the dataset(Downloaded Files) using glob and then read the text in all files. Modify the text by removing punctuations and stop words and created to list of tokens(tokenizer using split()) and then we build the inverted index.



Term	Document:Locations
voided	{'doc1': [221]}
coach	{'doc3': [12]}
house	{'doc2': [248]}
singletary	{'doc1': [23, 206, 342]}
mandate	{'doc3': [143]}
innovations	{'doc2': [78]}
edition	{'doc2': [10]}
niners	{'doc1': [0]}
week	{'doc2': [148],'doc1': [178, 186]}
buildings	{'doc3': [384]}
energy	{'doc2': [410]}
football	{'doc1': [326]}
coast	{'doc2': [26]}
job	{'doc1': [256]}
one	{'doc3': [234], 'doc1': [366]}
green	{'doc2': [32, 65]}
team	{'doc1': [335]}
singletary	{'doc1': [23, 206, 342]}
topics	{'doc2': [307]}
smith	{'doc3': [48, 153, 401]}

For the words, we used the Trie data structure as it takes the least time O(length of a word) to search for a word and stored document frequency (df) at the leaf.

For the documents, we used a list data structure and we have stored term frequency (tf) for every document.

We used AVL tree to manage this posting list and store this posting lists in postlinglist.txt

3.Searching:

The query is given for the search

1. Collect unique terms from the query

2. Remove stop words from the query

3. Take the union of the documents in which the terms are present.

   (we also implemented spelling correction)

ie. when a user types a word that is not present in the dictionary we suggest spelling the corrected words by finding the least edit distance among all the words in the dictionary.

4.Ranking:

We get hundreds of documents after a search, we have to choose the documents that are most relevant to the query.

The ranking is the process of finding the documents most relevant to the query.

Ranking procedure:

For QUERY:

For each term in the query, we find tf-raw, tf-weighted, IDF.

Tf-raw: number of documents in which term is present.

Tf-weighted: 1+log(tf-raw)

IDF: log(N/df)

Where,

N: number of documents in the collection

Df: document frequency

For DOCUMENTS:

For each document, we find weights.

For a document

Tf-raw – frequency of the term in the document

Tf-weighted- (1+log(tf-raw))

Normalized weight - document weights after cosine normalization

PRODUCT:

The product of the final query weight and final document weight.

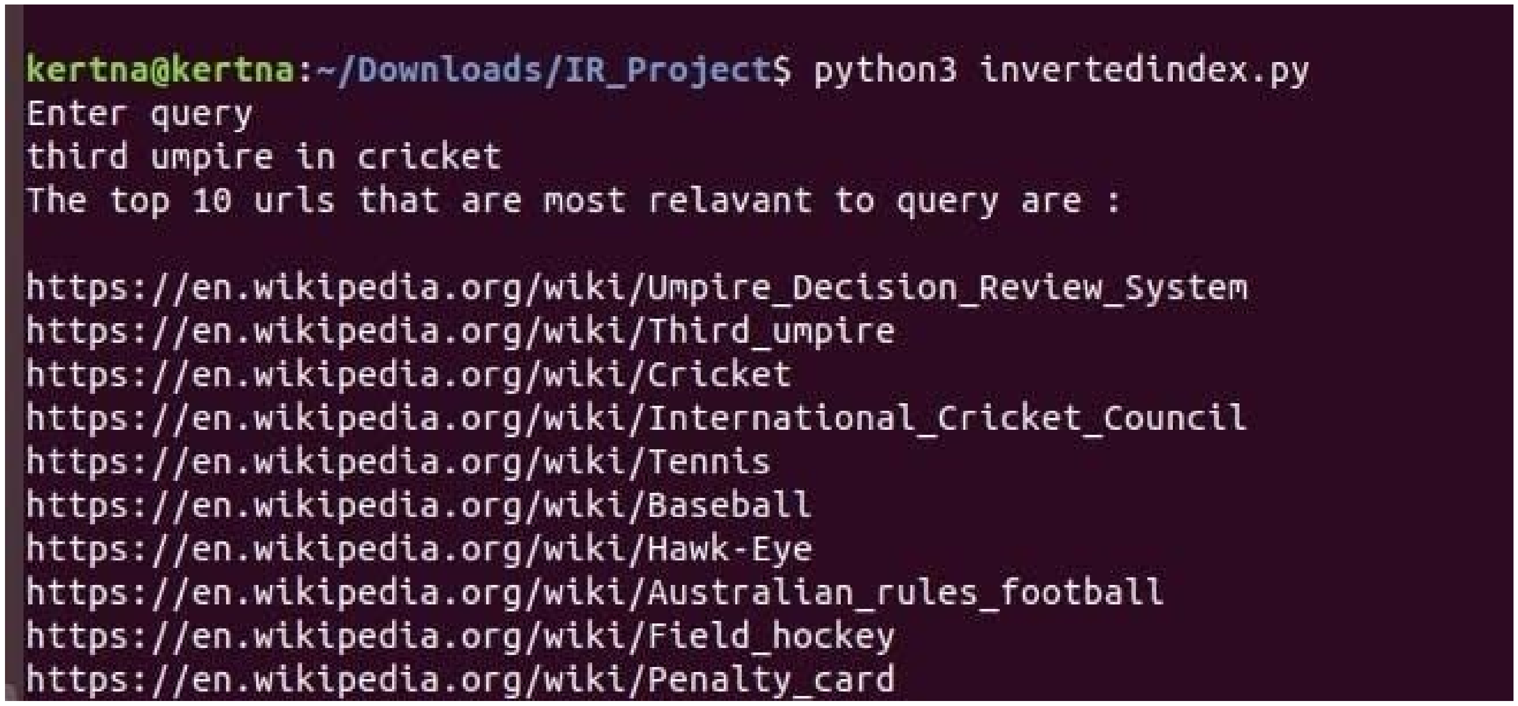
SIMILARITY SCORE:

Σ wqi*wdi

The document with the highest similarity score is the most relevant document.

Summary:

Finally, retrieve the top 10 URLs relevant to the query.



```
kertna@kertna:~/Downloads/IR_Project$ python3 invertedindex.py
Enter query
third umpire in cricket
The top 10 urls that are most relavant to query are :

https://en.wikipedia.org/wiki/Umpire_Decision_Review_System
https://en.wikipedia.org/wiki/Third_umpire
https://en.wikipedia.org/wiki/Cricket
https://en.wikipedia.org/wiki/International_Cricket_Council
https://en.wikipedia.org/wiki/Tennis
https://en.wikipedia.org/wiki/Baseball
https://en.wikipedia.org/wiki/Hawk-Eye
https://en.wikipedia.org/wiki/Australian_rules_football
https://en.wikipedia.org/wiki/Field_hockey
https://en.wikipedia.org/wiki/Penalty_card
```

Contribution:

Sai Divya:50%

Keerthana:50%