The University of Texas at Dallas

CS 6322

Information Retrieval

Spring 2020

Class Project Report

Project TITLE: Search Engine for Countries

Group: No. 3

Students: Darrel Donald, darrel.donald@utdallas.edu

Ruchi Singh, rxs180057@utdallas.edu

Sasipreetam Morsa, sxm171330@utdallas.edu

Anshul Pardhi, anshul.pardhi@utdallas.edu

Minal Bonde, msb190000@utdallas.edu

1. Introduction

The search engine we created focuses on gathering information about Countries. The work for the different aspects of the search engine was divided as follows:

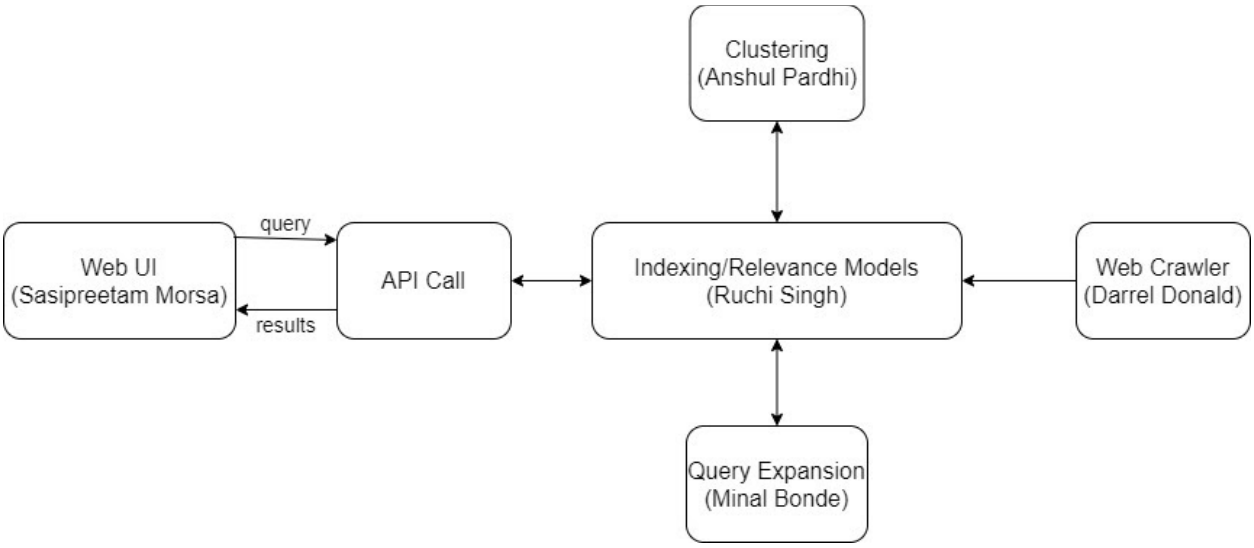
Crawling: Darrel Donald

Indexing and Relevance: Ruchi Singh

User Interface and Comparisons with Google and Bing: Sasipreetam Morsa

Clustering: Anshul Pardhi

Query Expansion and Relevance Feedback: Minal Bonde



The main thing we learned and also our biggest challenge was working with large amounts of data. When compared to the number of webpages on the internet, the amount of data we worked with is relatively small, and yet even working with this amount of data was very difficult and time consuming. As such, we also learned that there are many open-source tools available that are designed to be used with large quantities of data and that make working with this data much easier and smoother. We also created REST endpoints for transferring data between the client and the server, which was a new learning experience for some of the teammates.

Overall, we enjoyed this project because it was different from what we have seen before and gave us a new and unique experience that will help us become better developers.

Darrel Donald
DLD180001

2. Crawling

Initially, Scrapy was being used for crawling. Scrapy was able to obtain 100,000 unique pages, however, most of the pages were twitter links that were not useful to our search engine. We decided to try crawling again with Apache Nutch. Using Apache Nutch yielded more useful webpages.

We gave Nutch the following seed list of 50 URLs:

'https://en.wikipedia.org/wiki/Lists_of_countries_and_territories',

'https://en.wikipedia.org/wiki/Lists_by_country',

'https://en.wikipedia.org/wiki/List_of_sovereign_states',

'https://en.wikipedia.org/wiki/Dependent_territory',

'https://en.wikipedia.org/wiki/Country',

'https://en.wikipedia.org/wiki/Autonomous_republic',

'https://history.state.gov/countries/all',

'https://data.worldbank.org/country',

'https://www.factmonster.com/world/countries',

'https://www.infoplease.com/world/countries',

'https://www.cia.gov/library/publications/the-world-factbook/',

'https://www.traveldocs.com/world-atlas',

'https://www.countryreports.org/',

'https://www.worldometers.info/geography/how-many-countries-are-there-in-the-world/',

'https://history.state.gov/countries/all',

'https://www.countries-ofthe-world.com/all-countries.html',

'https://www.nationsonline.org/oneworld/countries_of_the_world.htm',

'https://www.britannica.com/topic/list-of-countries-1993160',

'https://worldview.stratfor.com/article/how-many-countries-are-there-world-2019',

'https://www.usnews.com/news/best-countries/slideshows/the-25-best-countries-in-the-world',

'https://www.worldatlas.com/articles/how-many-countries-are-in-the-world.html',

'https://worldpopulationreview.com/countries/how-many-countries-are-there/',

'https://theodora.com/wfb/mobile.html',

'https://www.listchallenges.com/the-195-countries-of-the-world',

Darrel Donald
DLD180001

'https://www.cntraveler.com/gallery/are-these-the-best-countries-in-the-world',

'https://www.jetpunk.com/quizzes/how-many-countries-can-you-name',

'https://www.usatoday.com/story/money/2019/07/07/richest-countries-in-the-world/39630693/',

'https://examples.yourdictionary.com/list-of-all-countries-in-the-world.html',

'https://www.thoughtco.com/number-of-countries-in-the-world-1433445',

'https://www.boldtuesday.com/pages/alphabetical-list-of-all-countries-and-capitals-shown-on-list-of-countries-poster',

'https://geology.com/world/world-map.shtml',

'https://www.forbes.com/sites/laurabegleybloom/2020/03/20/ranked-20-happiest-countries-2020/#66d45f9c7850',

'https://www.businessinsider.com/best-countries-in-the-world-ranked-us-news-world-report',

'https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/world-map.html',

'https://www.ethnologue.com/browse/countries',

'https://ourworldindata.org/countries',

'https://www.history.com/news/what-is-the-smallest-country-in-the-world',

'https://freedomhouse.org/countries/freedom-world/scores',

'https://www.heritage.org/index/ranking',

'https://www.who.int/choice/demography/by_country/en/',

'https://www.roughguides.com/gallery/most-beautiful-country-in-the-world/',

'https://www.weforum.org/agenda/2020/02/countries-manufacturing-trade-exports-economics/',

'https://www.sporcle.com/games/g/world',

'https://www.internetworldstats.com/list1.htm',

'https://www.internetworldstats.com/stats8.htm',

'https://www.iqair.com/us/world-most-polluted-countries',

'https://www.visualcapitalist.com/the-most-miserable-countries-in-the-world/',

'https://www.playgeography.com/games/countries-of-the-world/',

'https://www.ducksters.com/geography/',

'https://www.mapsofworld.com/map-of-countries.html'

Darrel Donald
DLD180001

We gathered 126,434 pages useful pages using Nutch. We were aiming for more but had to cut it short due to running out of time. We ensured we did not have duplication in our crawl by using Nutch's built in remove duplicates feature. To pass on the hyperlink information, I gave read permissions to all of the data I was able to collect and explained the output of Nutch to my teammates so that they would be able to find what they needed from me and copy all of the data I collected.

3. Indexing and Relevance

Indexing

Apache Solr is used for indexing the crawled pages. The output of the crawler contains three folders, crawldb, linkdb and segments. While creating the index in Solr, all these databases has been fed as the input. The following command has been used to create the index of crawled pages in Solr with the help of nutch:

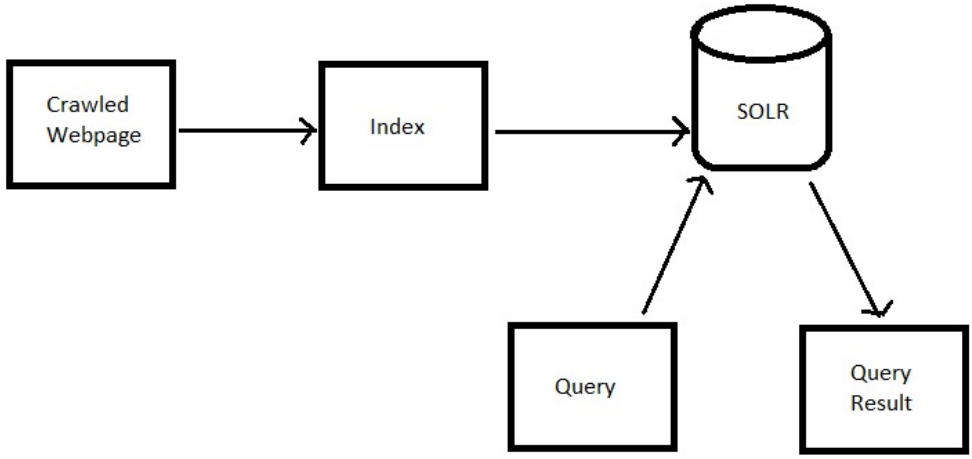
bin/nutch index crawl/crawldb/ -linkdb crawl/linkdb -dir crawl/segments/ -filter -normalize -deleteGone

bin/nutch is the binary of Apache nutch. The index command takes the content from one or more segments and passes it to all enabled IndexWriter plugins which sends the documents to Solr.

Crawl is the directory where nutch puts all the crawled pages and created three directories mentioned above in the command.

1) **CrawlDb** - It contains all the link parsed by the Nutch.
2) **LinkDB** - It contains for each URL the outgoing and the incoming URLs.
3) **Segment** - It contains the list of URLs to be crawled or being crawled.

Index Creation Flow Diagram



Webgraph and Statistics

A web graph has been created from the set of crawled URLs. We have used nutch command to create the webgraph of all the crawled webpages.

bin/nutch webgraph -segmentDir crawl/segments/ -webgraphdb crawl/webgraphdb

The webgraph command of nutch takes multiple segments to process and requires an output directory in which to place the completed web graph components. The webgraph creates three different components: an inlink database, an outlink database and a node database. The inlink database is a listing of url and all of its inlinks. The outlink database is a listing of url and all of its

outlinks. The node database is a listing of url with node meta information including the number of inlinks and outlinks.

Statistics

Total Number of links = 711101
Total Number of nodes = 178112
The largest number of ingoing links = 21437
The largest number of outgoing links = 91

Web graph information connected to index

While creating index in Solr, we have run the LinkRank program of nutch to perform an iterative link analysis with the webgraph. LinkRank is a pagerank-like link analysis program that converges to stable global scores for each url. The LinkRank program starts with a common score for all urls. It then creates a global score for each url based on the number of incoming links and the scores for those links and the number of outgoing links from the page.

Relevance Models

Vector Space Relevance model

Solr provides tf-idf based relevance model to score the webpages and we made use of this model as vector space relevance model. In vector space relevance model, documents and queries are both vectors. Similarity of a document vector to a query vector can be computed by taking the cosine of the angle between them. Cosine similarity measure has been used to get the similarity between document and query. This scoring model depends on number of factors –

Term frequency - The frequency with which a term appears in a document. Given a search query, the higher the term frequency, the higher the document score.

Inverse document frequency - The rarer a term is across all documents in the index, the higher its contribution to the score.

Tf-idf weighting scheme has been used to create the relevance model. In this weighting scheme,

$$W_{d,t} = tf_{d,t} \times idf_t$$

Tf-idf weighting scheme is the most common weigthing scheme in vector space relevance model.

Page Rank

To develop the relevance model based on Page Ranking, we used inbuilt nutch command. The damping factor used is 0.85. Following are the steps we followed in order to apply PageRank scores into crawled database so that query results will display web pages in the descending order of page ranking values –

1. **Create a web graph from the set of URLS fetched by the crawler**
   bin/nutch webgraph -segmentDir crawl/segments/ -webgraphdb crawl/webgraphdb

2. **Run PageRank algorithm iteratively until the score converges**

bin/nutch linkrank -webgraphdb crawl/webgraphdb/

3. **Update the page rank score into the crawled database**
   bin/nutch scoreupdater -crawldb crawl/crawldb - webgraphdb crawl/webgraphdb/

4. **Finally, index the updated crawled database using Solr and nutch command:**
   bin/nutch index crawl/crawldb -linkdb crawl/linkdb -dir crawl/segments/

HITS (Hyperlink Induced Topic Search)

For the HITS score, we created our own python program to generate the authority and the hub score in python. HITS uses hubs and authorities to define a recursive relationship between webpages.

Given a query to a Search Engine, the set of highly relevant web pages are called **Roots**. They are potential **Authorities**. Pages which are not very relevant but point to pages in the Root are called **Hubs**. Thus, an Authority is a page that many hubs link to whereas a Hub is a page that links to many authorities.

We used the **networkx library** to create the graph through the inlinks and outlinks information of all the nodes (urls) and ran the hits algorithm on the graph to generate the hub and the authority scores. To get these inlinks, we used nutch and generated a dump of the all the inlinks of crawled pages. With the inlinks dictionary, we created the outlinks as well. Based on the query provided by the user, root set of web pages are collected based on page rank results. The root set is converted to a base set by adding out link and inlinks of the root set. After that, HIT score calculation is done by our python program and based on maximum authority score, relevant web pages are sorted and returned to the User Interface for display.

Among the webpages we crawled,

1. Highest hub score was assigned to : https://advisor.visualcapitalist.com/ score - 0.0229952
2. Highest authority score was assigned to: https://advisor.visualcapitalist.com/ score - 0.0229952

Topic based page ranking

For the purpose of testing Topic based page ranking, we have chosen 2 topics among countries:

1. Countries by continents
2. Countries by population

The results for **countries by continents** are as follows from our search engine: (included corresponding title and PageRank score for the webpage)

"**url**":"https://www.worldatlas.com/amp/continents.html"
"**title**":"7 Continents of the World - WorldAtlas.com"
"**score**":7.1495075 – Highest Page rank score

"**title**":"World Map With Continents, Map of Continents",
"**url**":https://www.mapsofworld.com/continents/
"**score**":7.141836

"**title**":"Continents by Population - WorldAtlas.com",
"**url**":"https://www.worldatlas.com/articles/continents-by-population.html"
"**score**":7.1285152

"**title**":"Is North America And South America One Continent? - WorldAtlas.com",
"**url**":https://www.worldatlas.com/articles/is-north-america-and-south-america-one-continent.html
"**score**":7.1129417

The results for **countries by population** are as follows from our search engine: (included corresponding title and PageRank score for the webpage)

"**title**":"Population (disambiguation) - Wikipedia",
"**url**":https://en.wikipedia.org/wiki/Population_(disambiguation)
"**score**":4.105301 - Highest Page rank score

"**title**":"Countries With the Lowest Population Density - WorldAtlas.com",
"**url**":https://www.worldatlas.com/amp/articles/countries-with-the-lowest-population-density.html
"**score**":4.096262

"**title**":"List of European countries by population - Wikipedia",
"**url**":"https://en.wikipedia.org/wiki/List_of_European_countries_by_population"
"**score**":4.090098

"**title**":"List of Asian countries by population - Wikipedia",
"**url**":"https://en.wikipedia.org/wiki/List_of_Asian_countries_by_population"
"**score**":4.085929

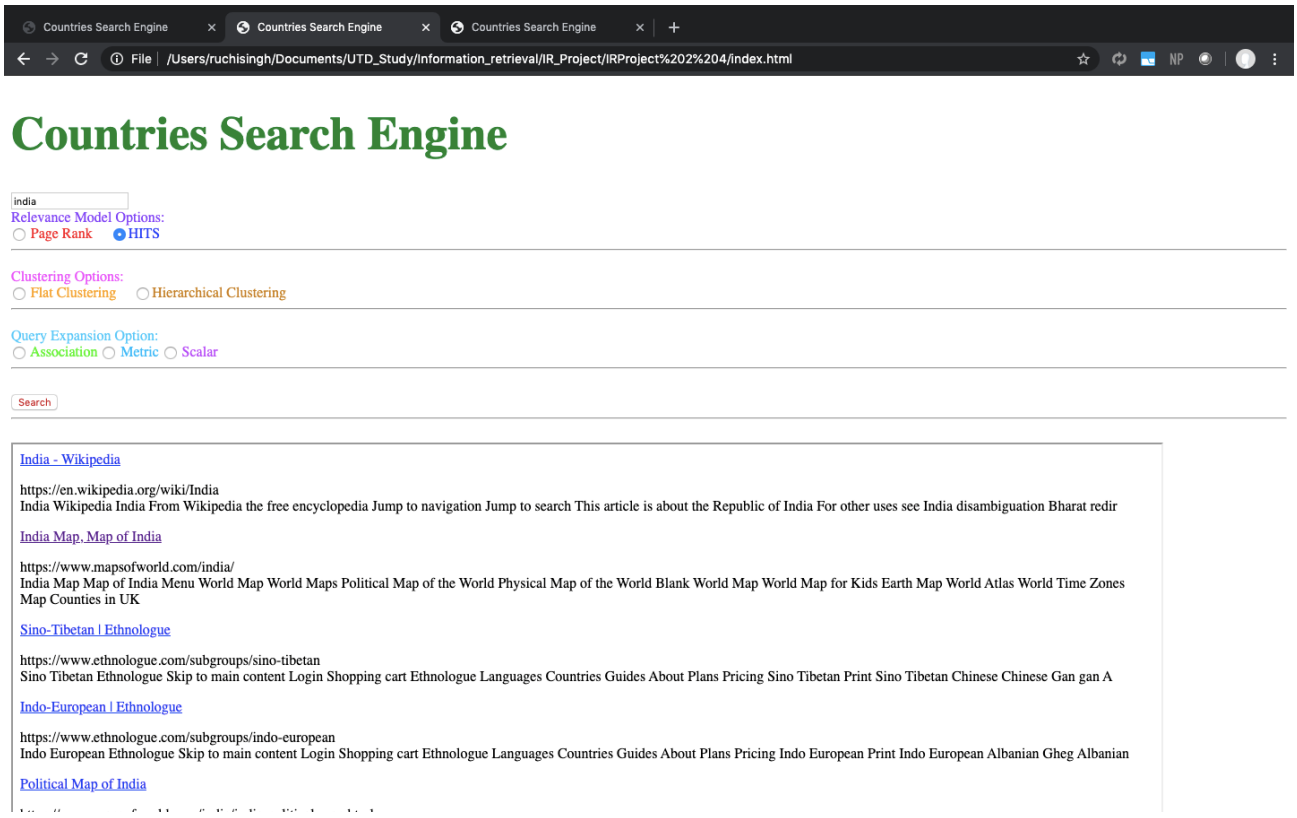
Collaboration with UI and test relevance models results

I have generated several queries to test the various relevance models I have implemented in this search engine. I have generated 5 queries and tested on the UI by differentiating the results of pagerank and HITS relevance model results. I have judged the results in some ways such as, HITS relevance model was giving most relevant results in comparison of page rank. For example,

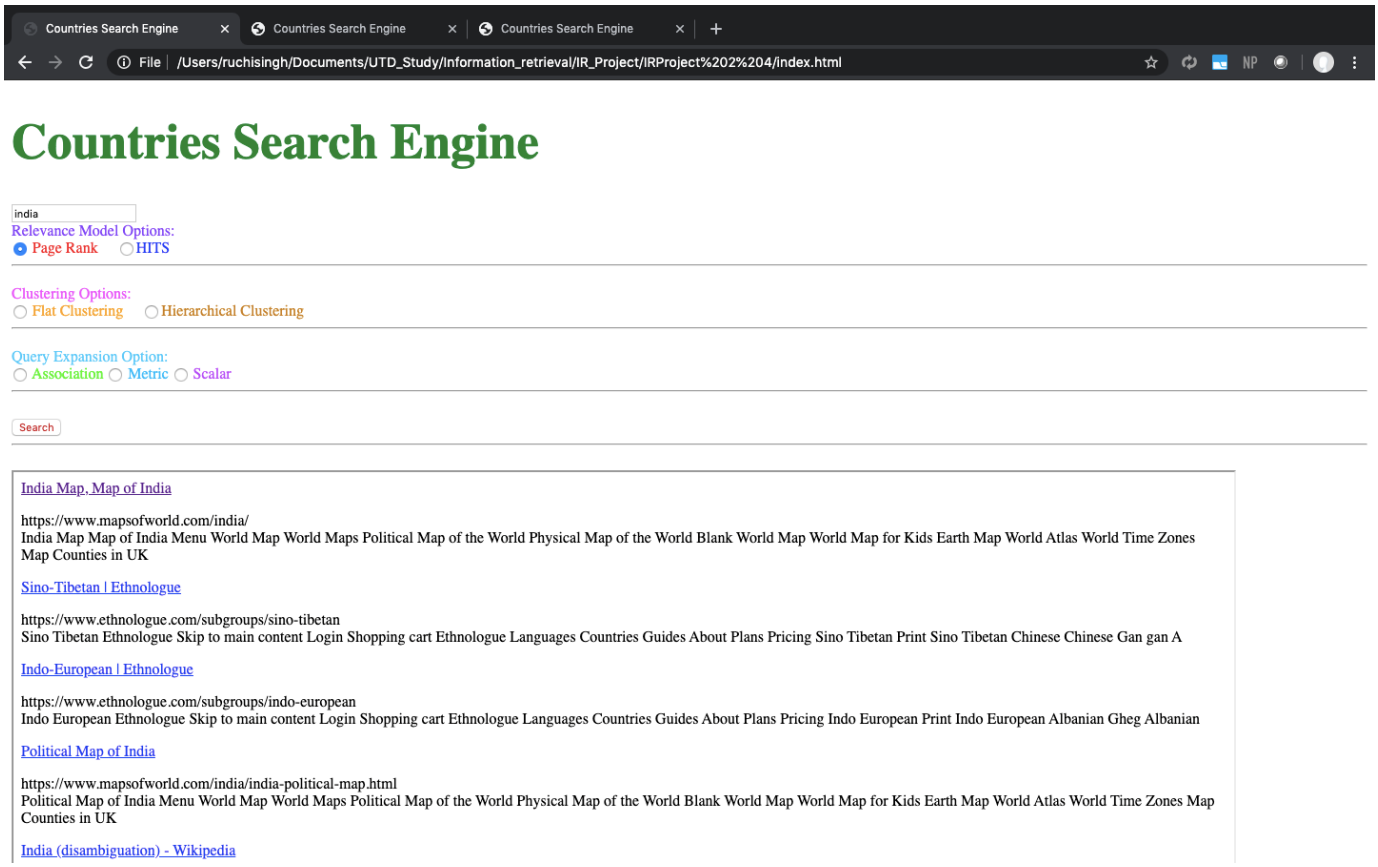
Ruchi Singh
RXS180057

For query – india

HITS gives the results in which the pages should come first as Wikipedia but page rank ranked the other pages more than Wikipedia. In the 1st screenshot, HITS option is selected.

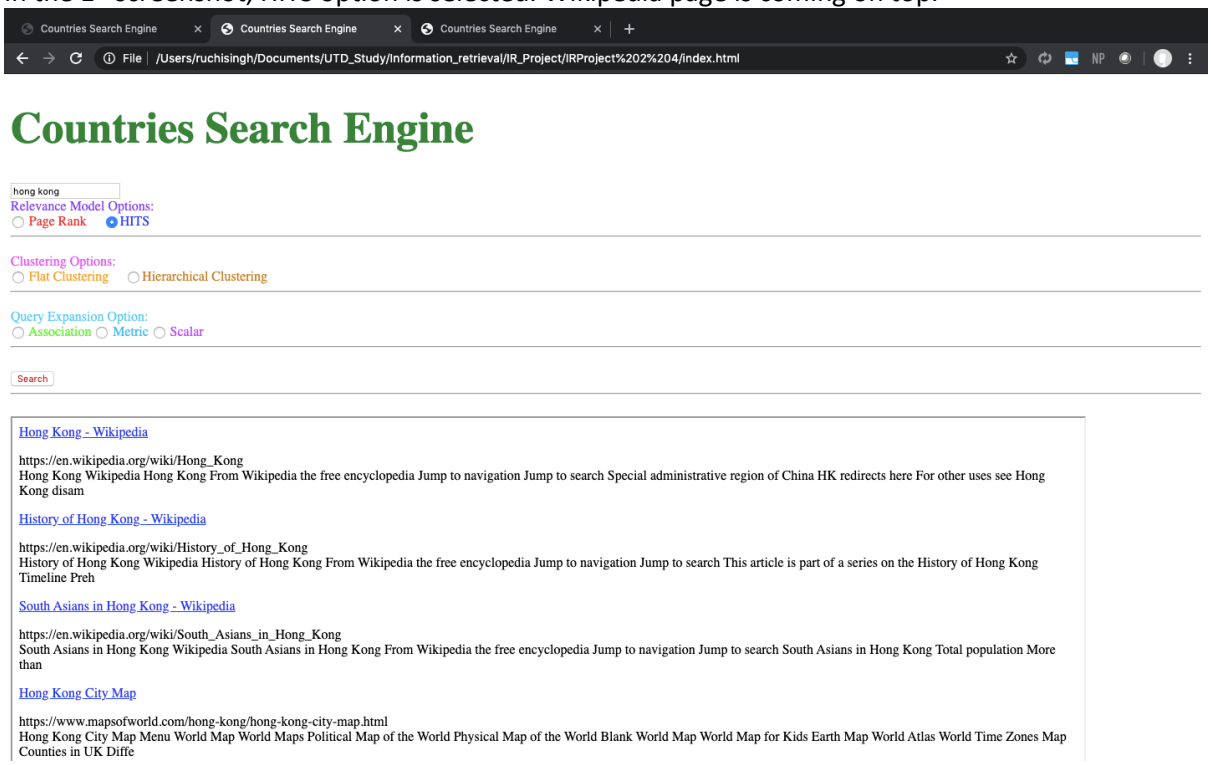


In the 2nd screenshot, page rank option is selected.

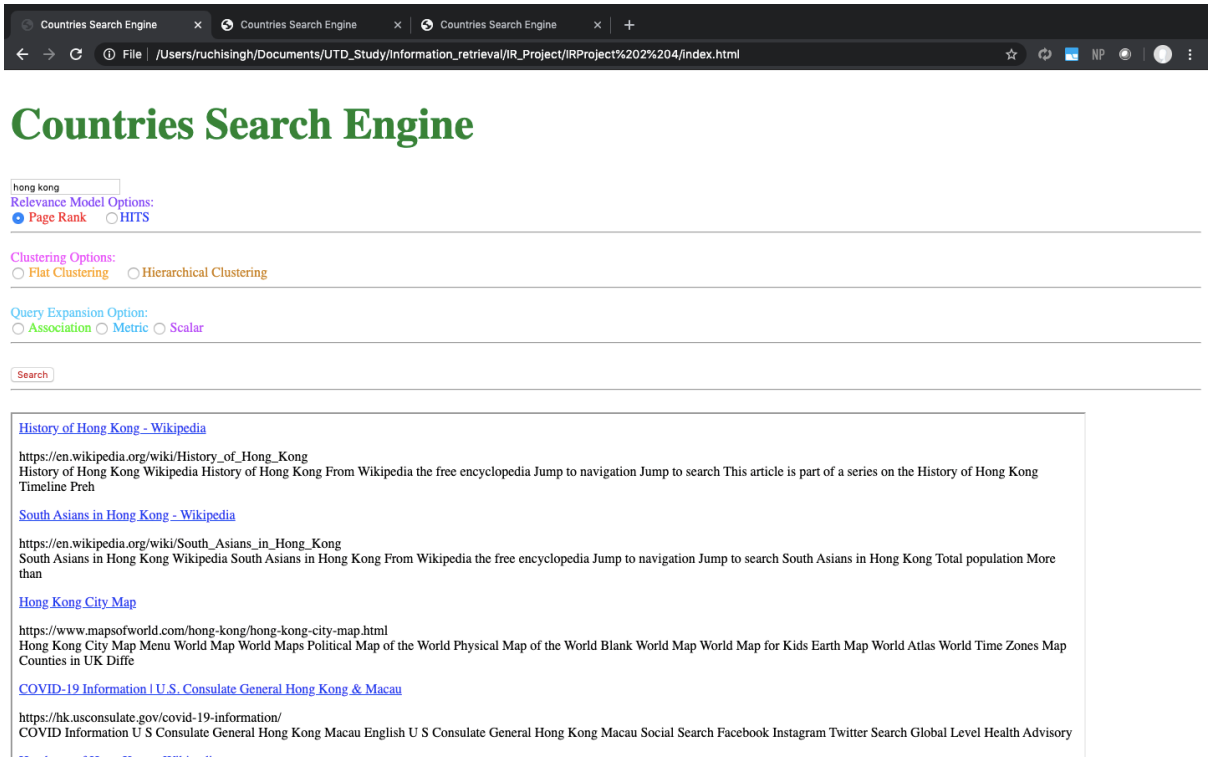


For query – hong kong

In the 1st screenshot, HITS option is selected. Wikipedia page is coming on top.

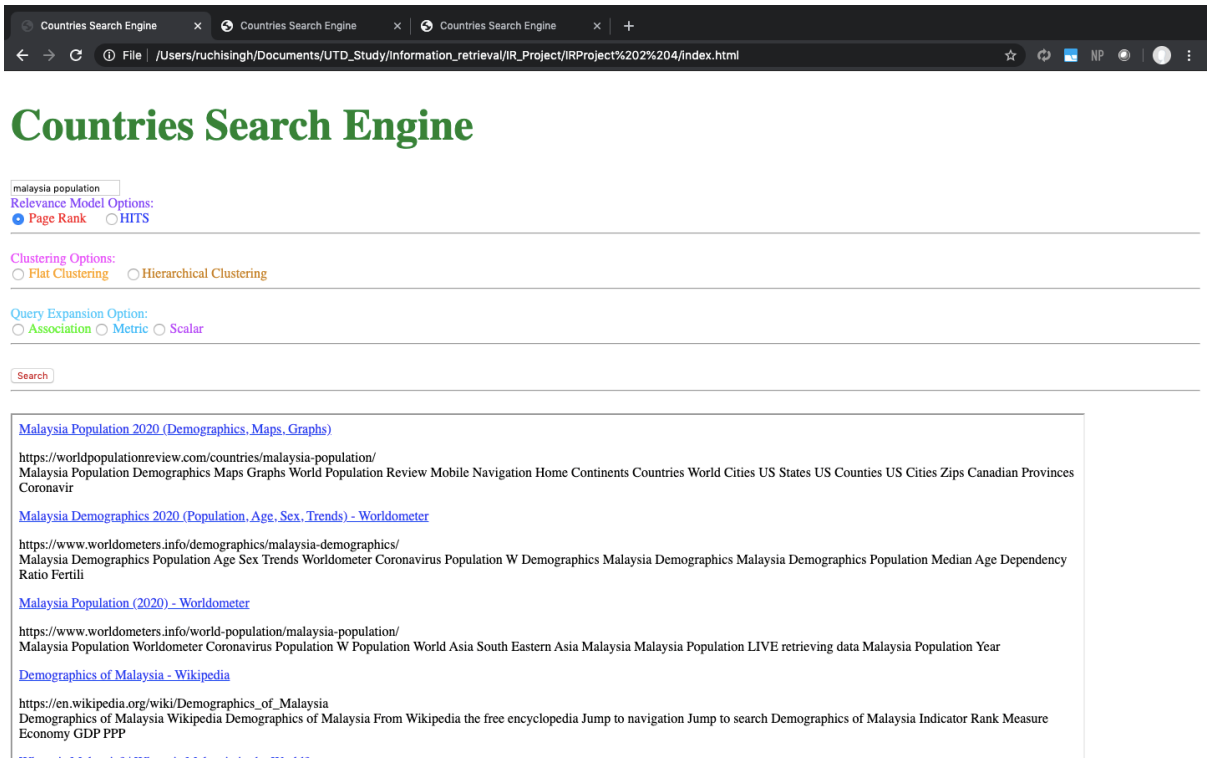


In the 2nd screenshot, page rank option is selected. History of hong kong page is coming on top.

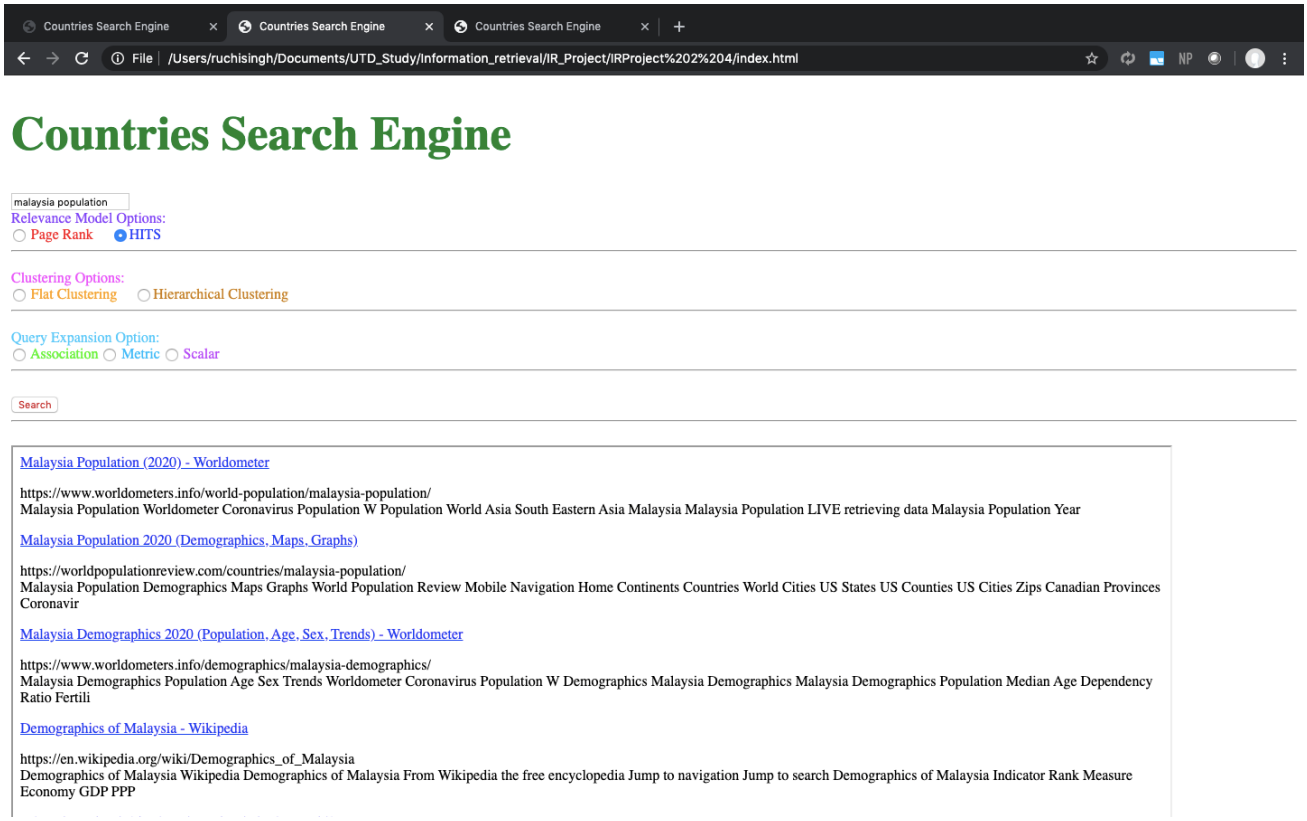


For query – Malaysia population

In the 1st screenshot, page rank option is selected. Demographics page is coming on top.

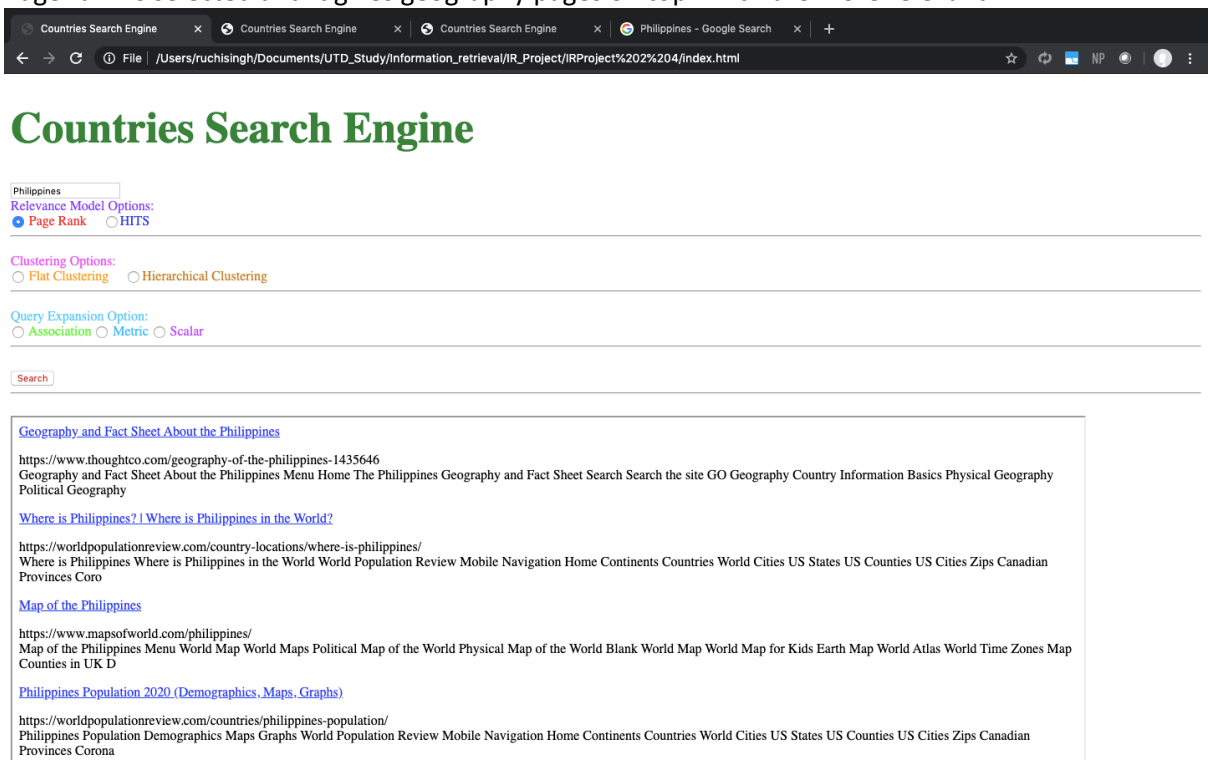


In the 2st screenshot, HITS option is selected. Worldometer page is coming on top about population.

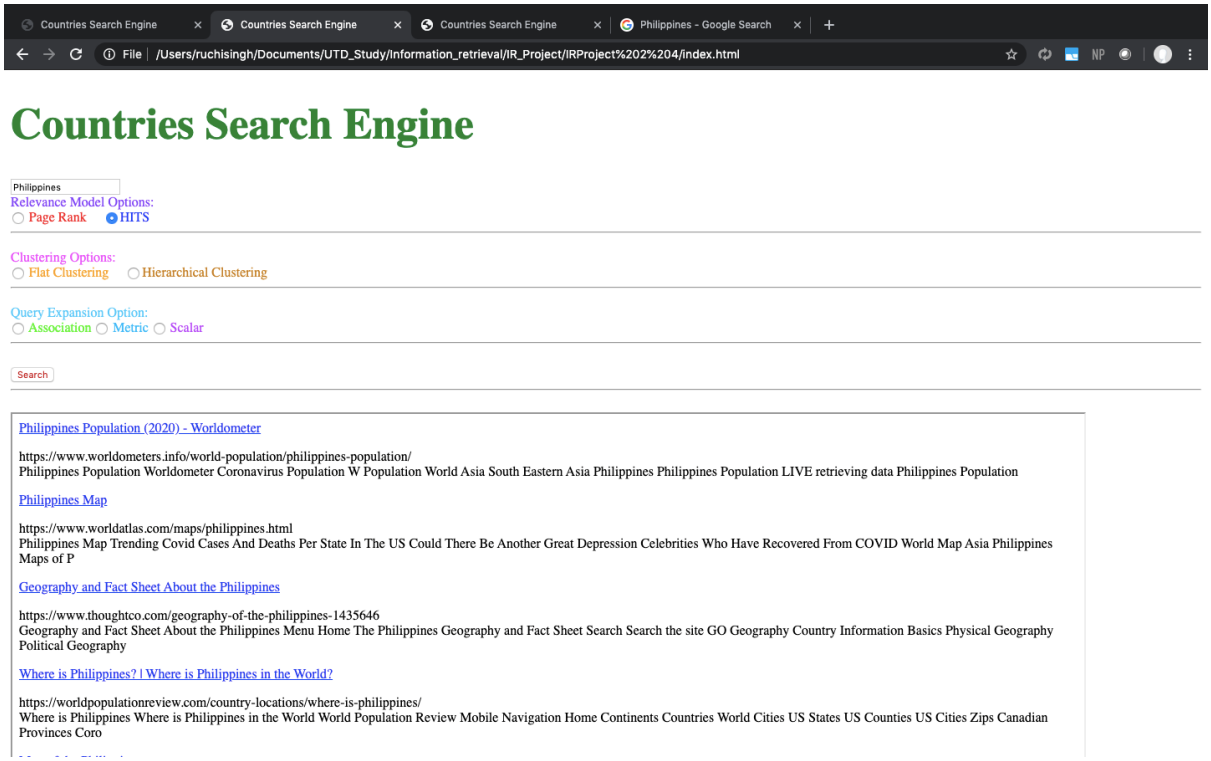


For Query – Philippines

Page rank is selected and it gives geography pages on top which are more relevant.

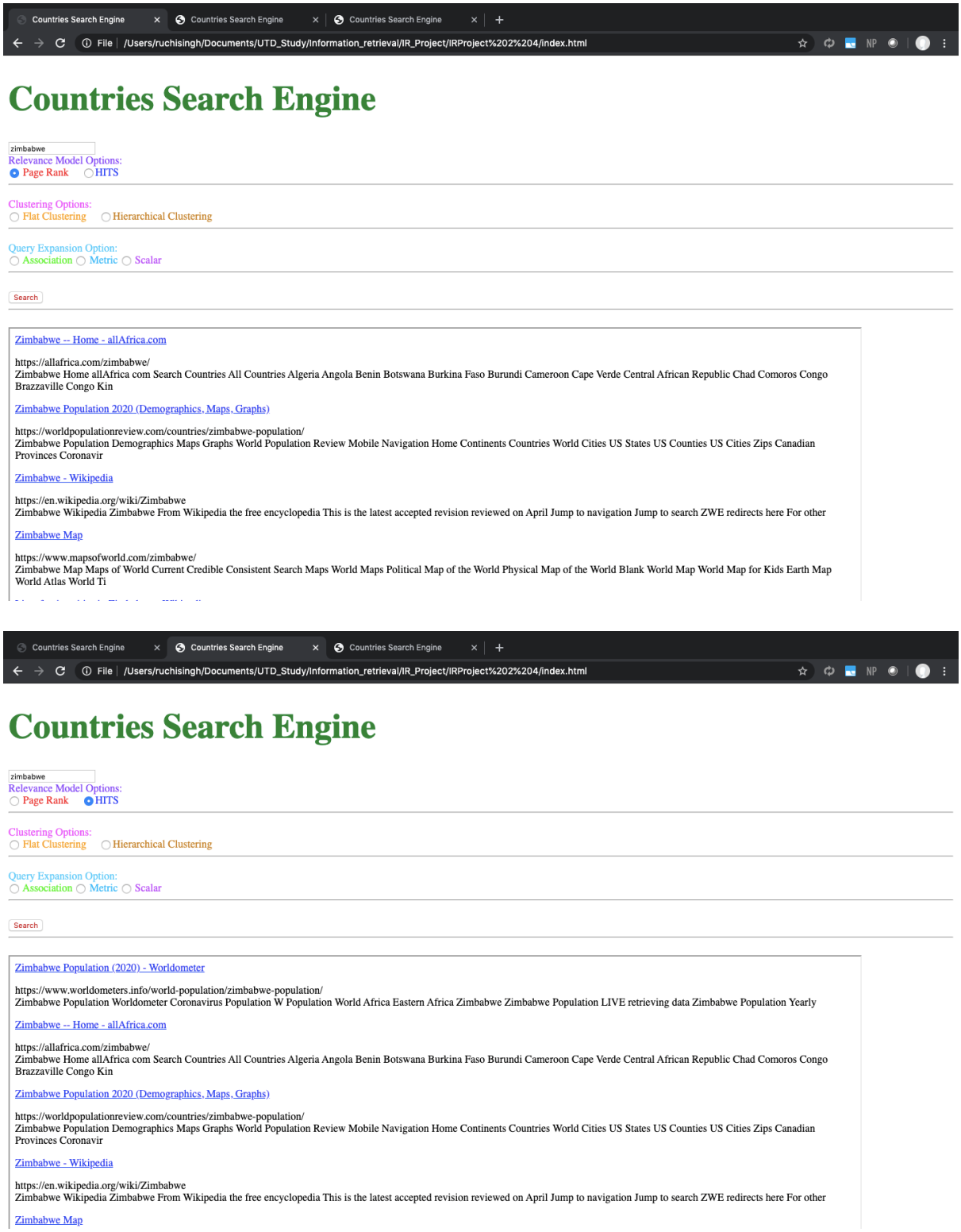


HITS is selected and it gives less relevant results for this query in comparison of page rank.



For Query – Zimbabwe

Page rank is selected and it gives the same results as HITS.



Collaboration with Clustering

I have collaborated with clustering to improve the results of the relevance models I have used. For all above queries mentioned, clustering (Hierarchical clustering) gives better results in comparison of page rank and HITS relevance models.

User Interface and Comparisons with Google and Bing

4. User Interface and Comparisons with Google and Bing (Sasipreetam Morsa)

The visual part of the User Interface is made of a form and 3 iframes.

The form includes a search bar to enter queries, radio buttons to select the method used to rank the results, and a submit button that, when clicked, runs JavaScript functions to display the results in the different iframes.

The 3 iframes contain our custom search engine, Google, and Bing, which are displayed in that order. The Google and Bing iframes start with their respective home pages, while our custom iframe starts out being empty.

To get the results for the user interface, I make a get request, with the query and type as parameters, to an API that connected to the relevance models, provided by Ruchi. The API uses the query and type parameters to build a JSON file with the results ordered by ranking. The different type parameters are as follow:

- Page Rank – API returns JSON with page rank based relevance model
- HITS – API returns JSON with HITS based relevance model
- Flat Clustering – API returns JSON with Flat Clustering based relevance model
- Hierarchical Clustering – API returns JSON with Hierarchical Clustering based relevance model
- Metric Query Expansion – API returns JSON with relevance model based on metric cluster query expansion
- Association Query Expansion – API returns JSON with relevance model based on metric association query expansion
- Scalar Query Expansion – API returns JSON with relevance model based on metric scalar query expansion

Ruchi and I together only used a few queries to test the communication between the User Interface and Ruchi's API, but separately I have tested over 20 queries to both ensure that the API requests could handle many requests back to back and to check that each of the different parameters were being sent to the API properly.

The work for clustering was done by Anshul. For the given query, Ruchi's relevance model would provide the top 50 results to Anshul's clustering program. The clustering program would then rearrange the results such that the results would start with the same top results but would then have the results in the

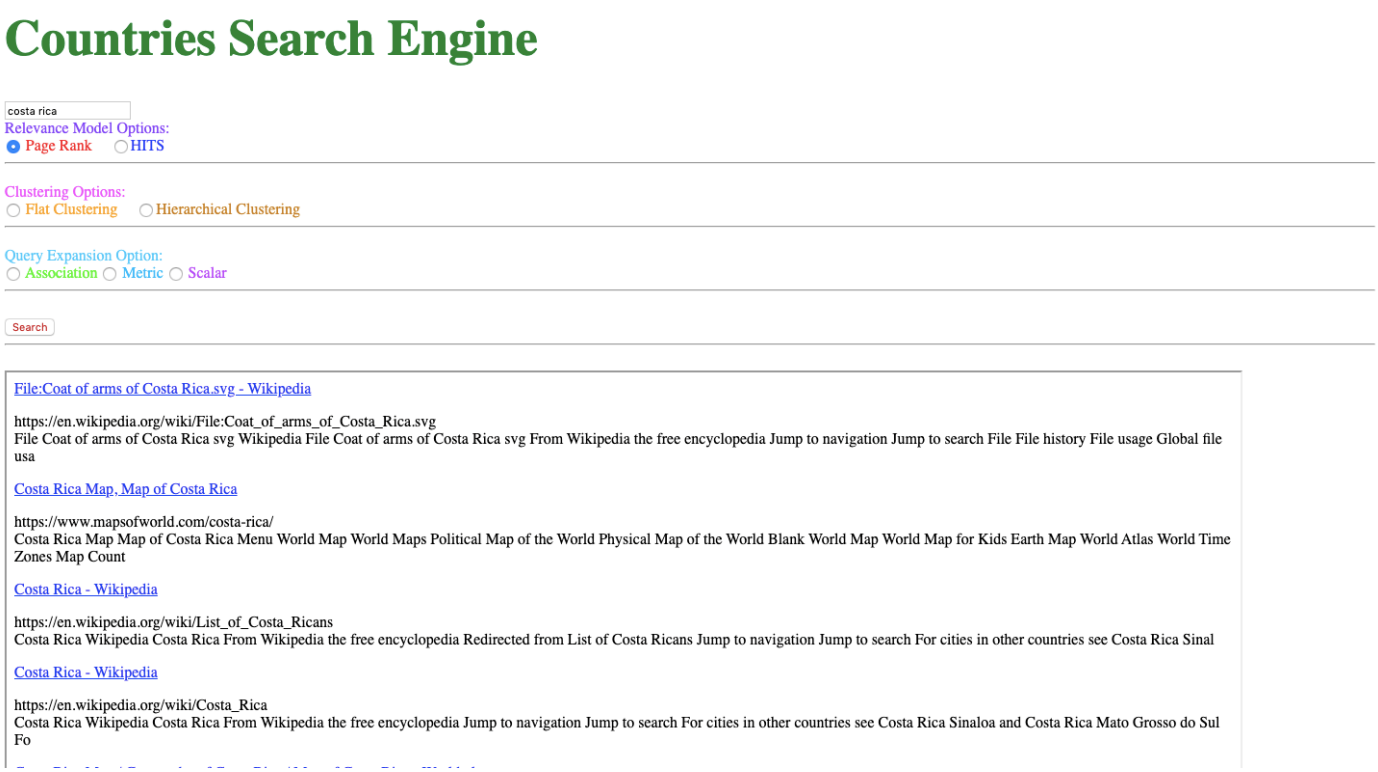
same cluster as the top result be the next results. So basically, the results were rearrange based on clusters containing the top result and so on.

Overall, when compared to large search engines like Google or Bing, our search engine is not as reliable. While there are some queries on which our search engine can keep up with or even beat the large search engines, it is not on the same level as Google or Bing because we don't have access to as many webpages as those search engines and we also don't have a way to keep updating the results in real time based on which websites more users are gravitating towards given a certain type of query.

For the demonstration, we have chosen queries that we believe have been showing both accurate results and show improvement as we include clustering or query expansion.

Below are the results of some of the queries we have tried:

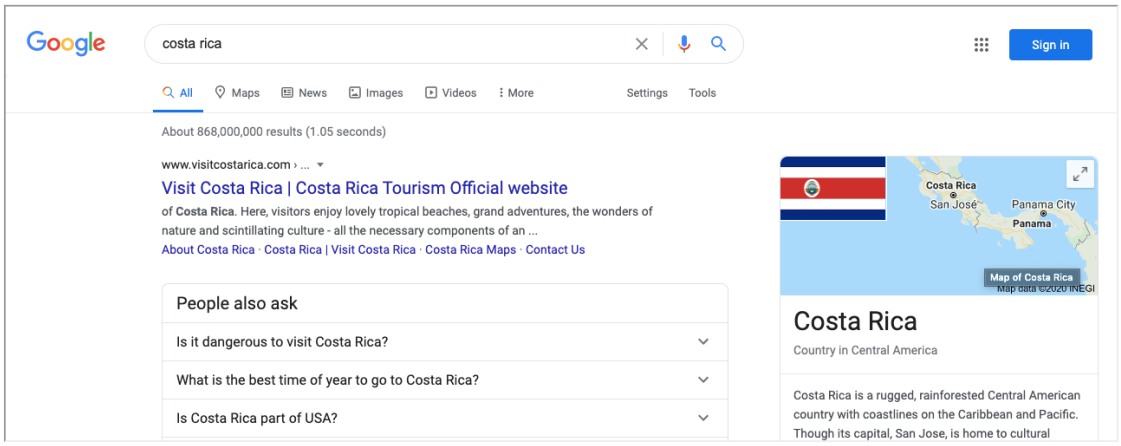
Results for query "costa rica":

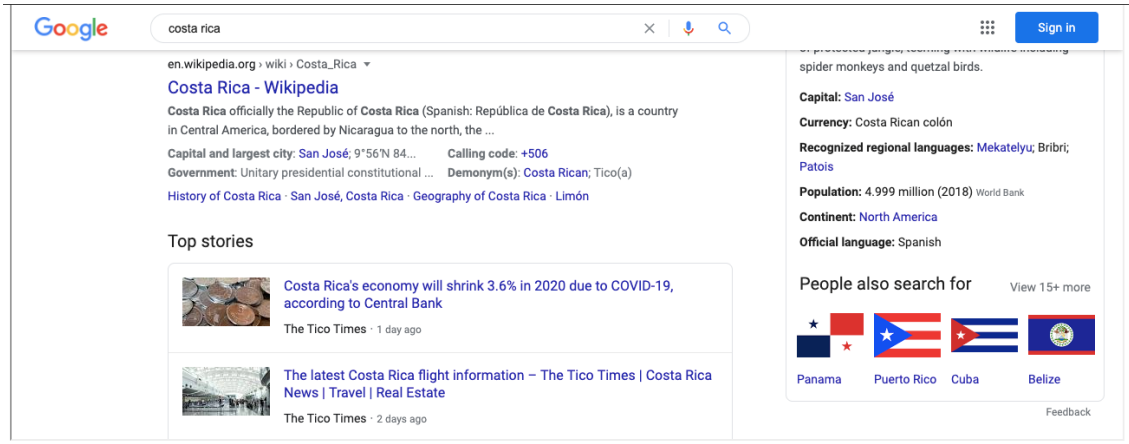


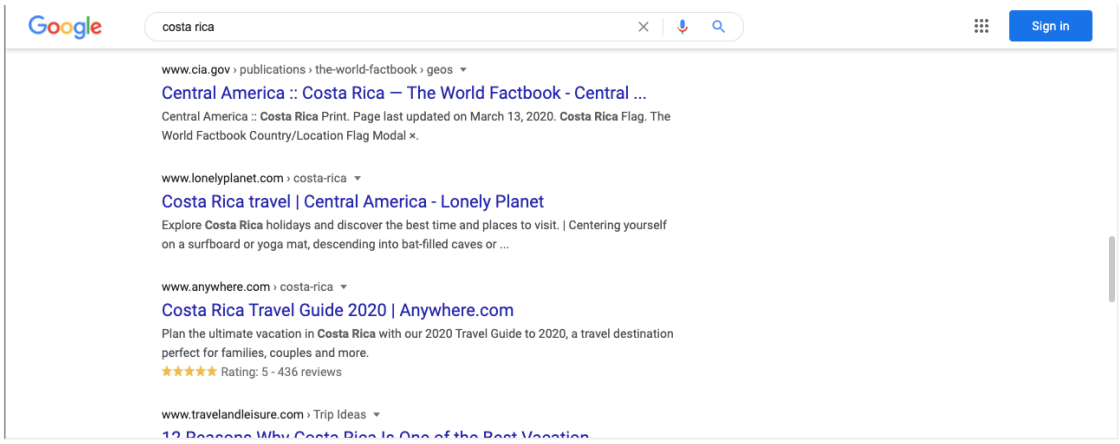
Sasipreetam Morsa/sxm171330

User Interface and Comparisons with Google and Bing

Google Search



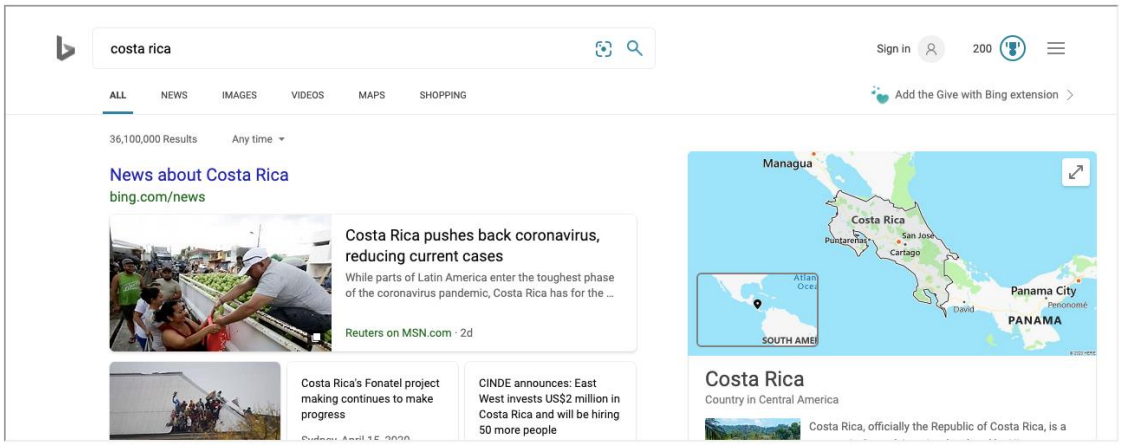




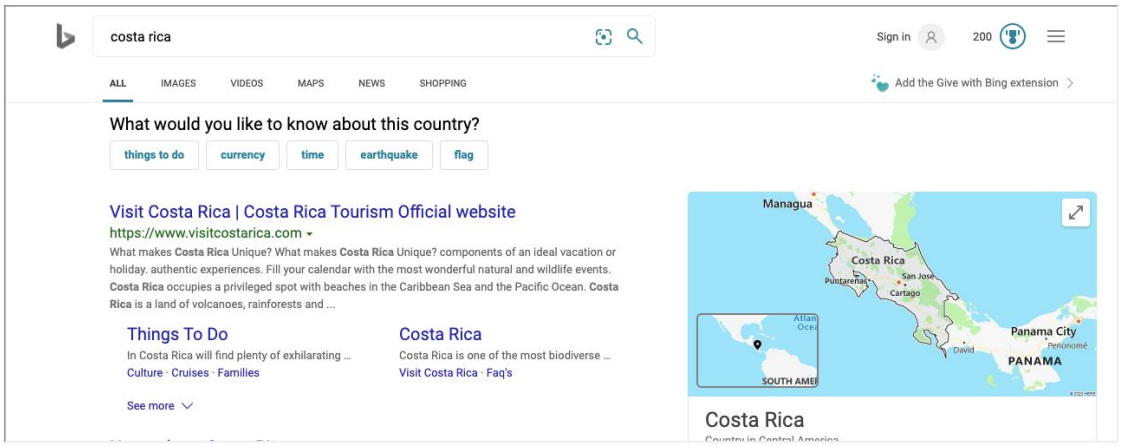
Sasipreetam Morsa/sxm171330

User Interface and Comparisons with Google and Bing

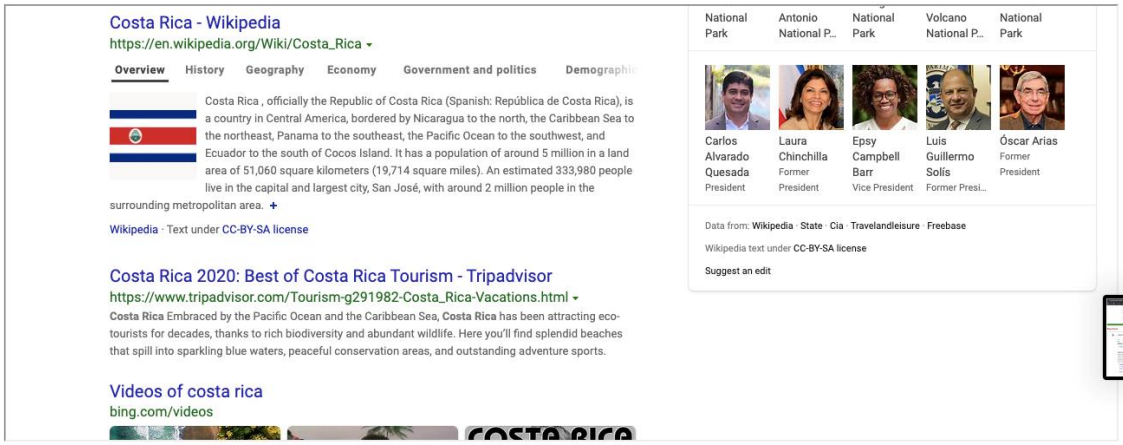
Bing Search



Bing Search



Bing Search



Sasipreetam Morsa/sxm171330

User Interface and Comparisons with Google and Bing

Results for query “israel”:

Countries Search Engine

israel

Relevance Model Options:
(•) Page Rank ( ) HITS

Clustering Options:
( ) Flat Clustering ( ) Hierarchical Clustering

Query Expansion Option:
( ) Association ( ) Metric ( ) Scalar

Search

Bene Israel - Wikipedia

https://en.wikipedia.org/wiki/Bene_Israel
Bene Israel Wikipedia Bene Israel From Wikipedia the free encyclopedia Jump to navigation Jump to search Jewish ethnic group Not to be confused with Bani Isra il or Beta Israel Bene Israel

Israel Population 2020 (Demographics, Maps, Graphs)

https://worldpopulationreview.com/countries/israel-population/
Israel Population Demographics Maps Graphs World Population Review Mobile Navigation Home Continents Countries World Cities US States US Counties US Cities Zips Canadian Provinces Coronavirus

Demographics of Israel - Wikipedia

https://en.wikipedia.org/wiki/Demographics_of_Israel
Demographics of Israel Wikipedia Demographics of Israel From Wikipedia the free encyclopedia Jump to navigation Jump to search Demographics of Israel Population of Israel since Population

Israel - Wikipedia

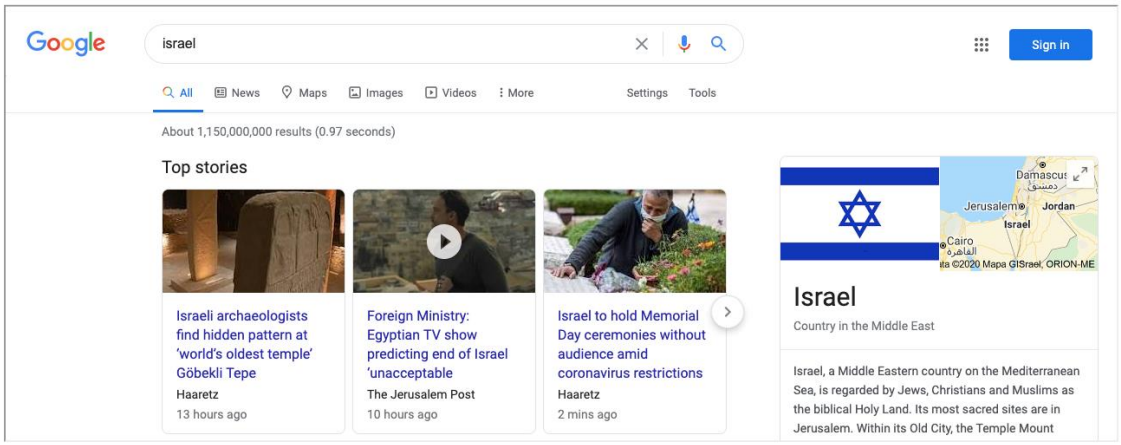
https://en.wikipedia.org/wiki/Israel
Israel Wikipedia Israel From Wikipedia the free encyclopedia Jump to navigation Jump to search This article is about the modern country For other uses see Israel disambiguation Country in Wes

Israelis - Wikipedia

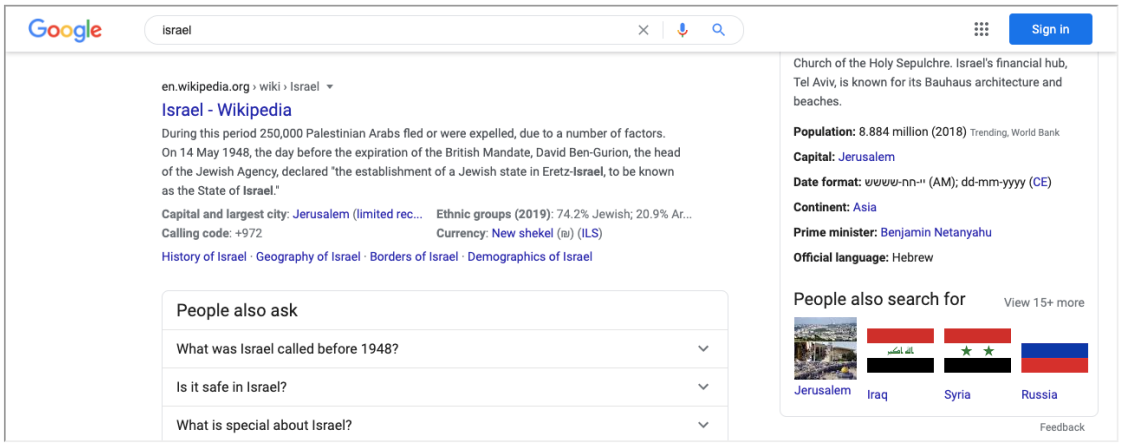
Sasipreetam Morsa/sxm171330

User Interface and Comparisons with Google and Bing

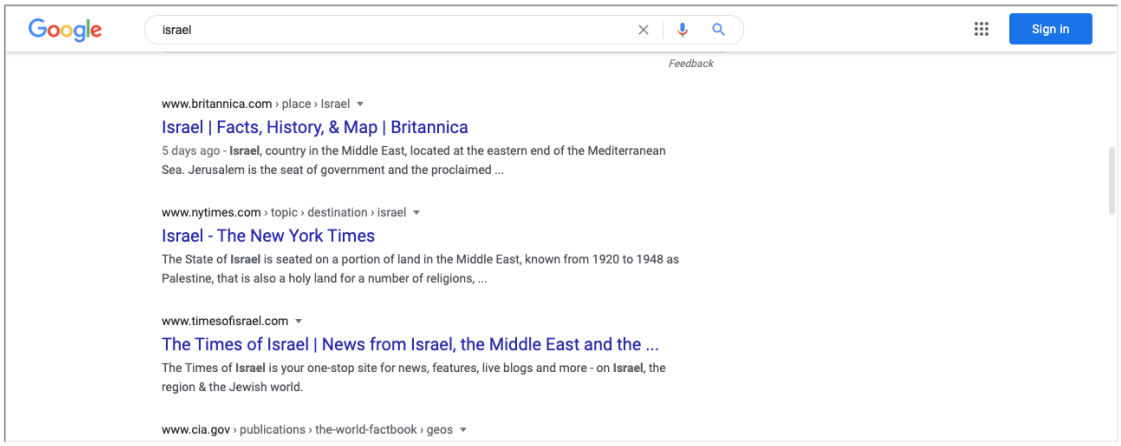
Google Search



Google Search



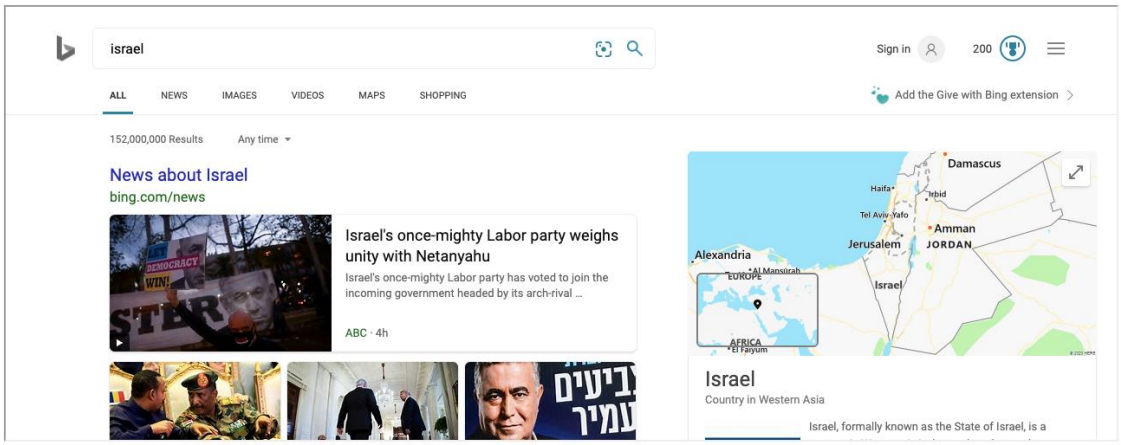
Google Search



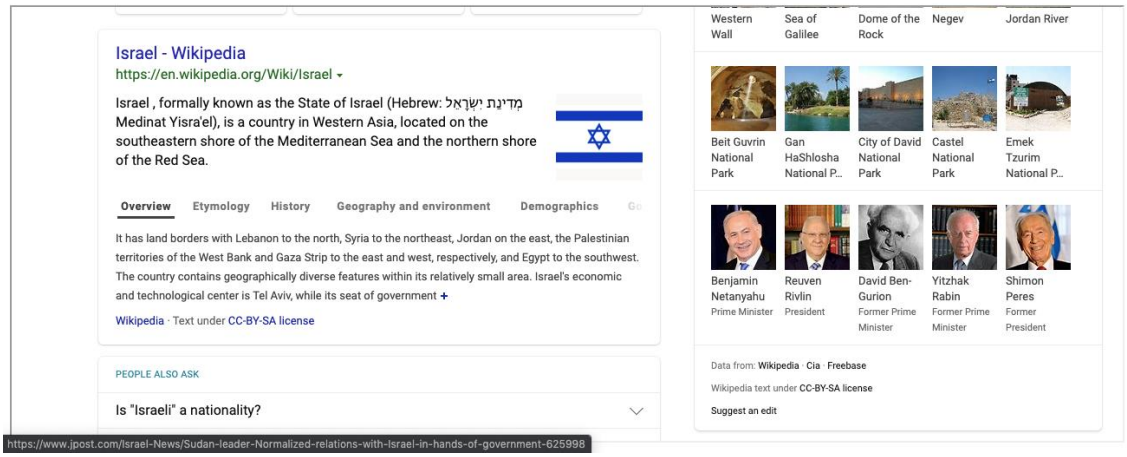
Sasipreetam Morsa/sxm171330

User Interface and Comparisons with Google and Bing

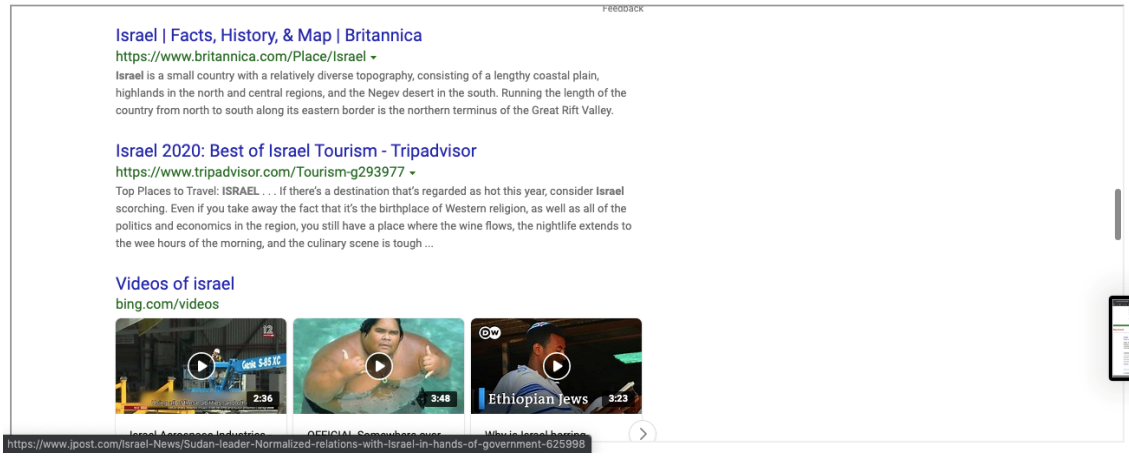
Bing Search



Bing Search



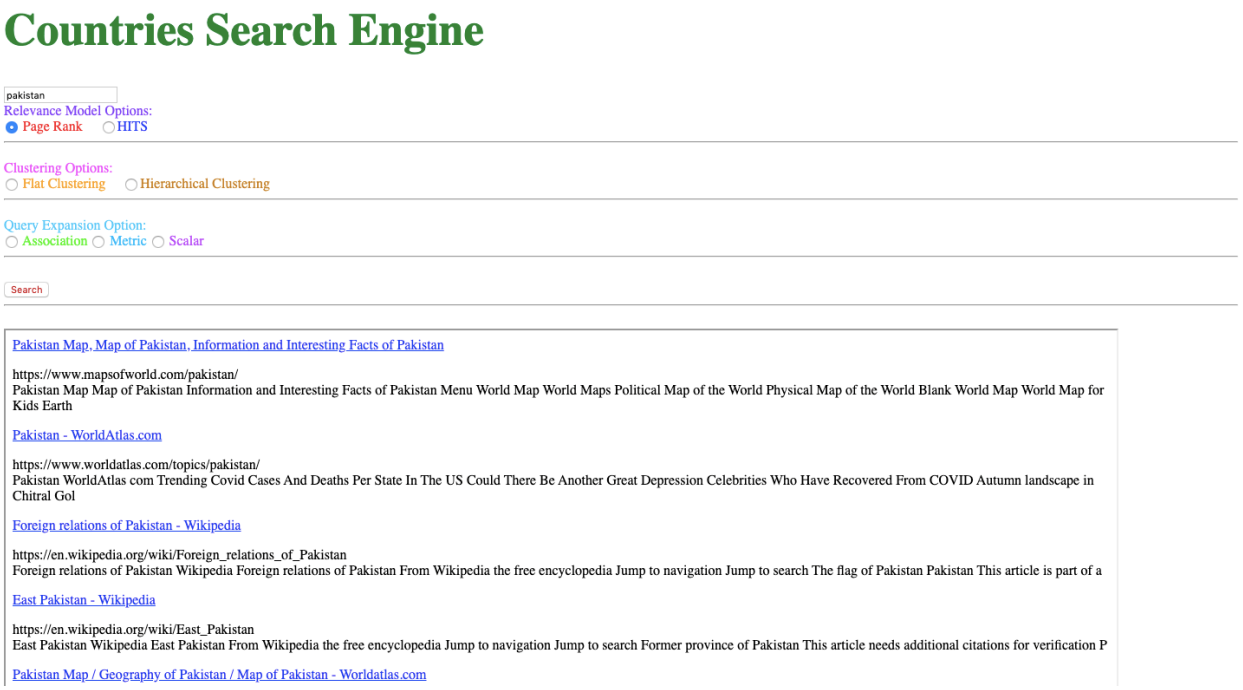
Bing Search



Sasipreetam Morsa/sxm171330

User Interface and Comparisons with Google and Bing

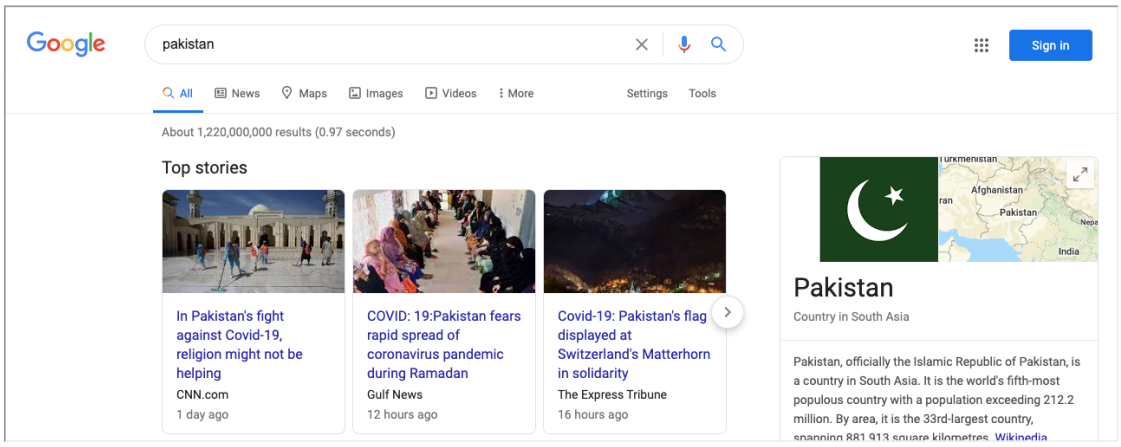
Results for query “pakistan”:



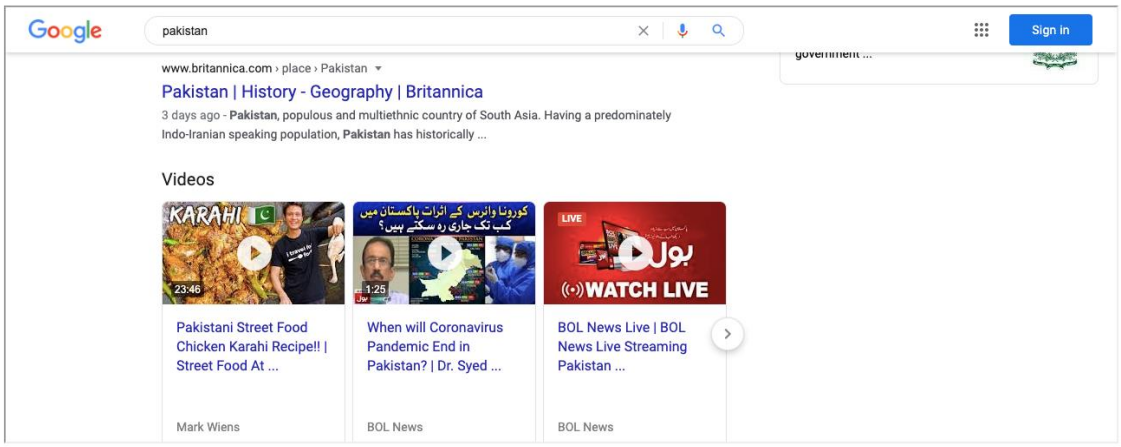
Sasipreetam Morsa/sxm171330

User Interface and Comparisons with Google and Bing

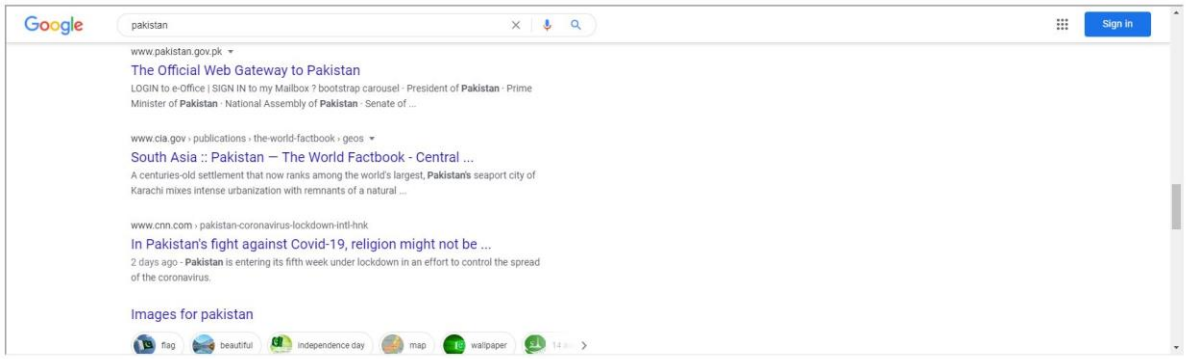
Google Search



Google Search



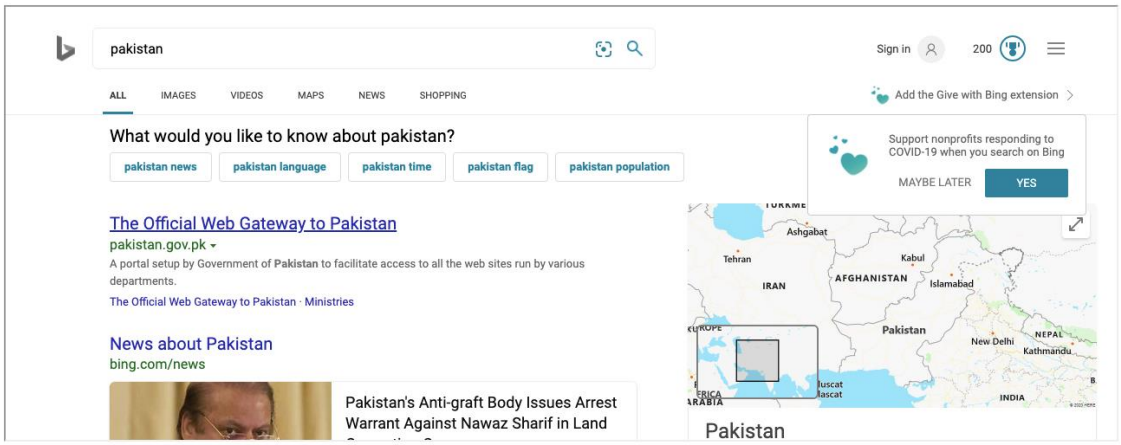
Google Search



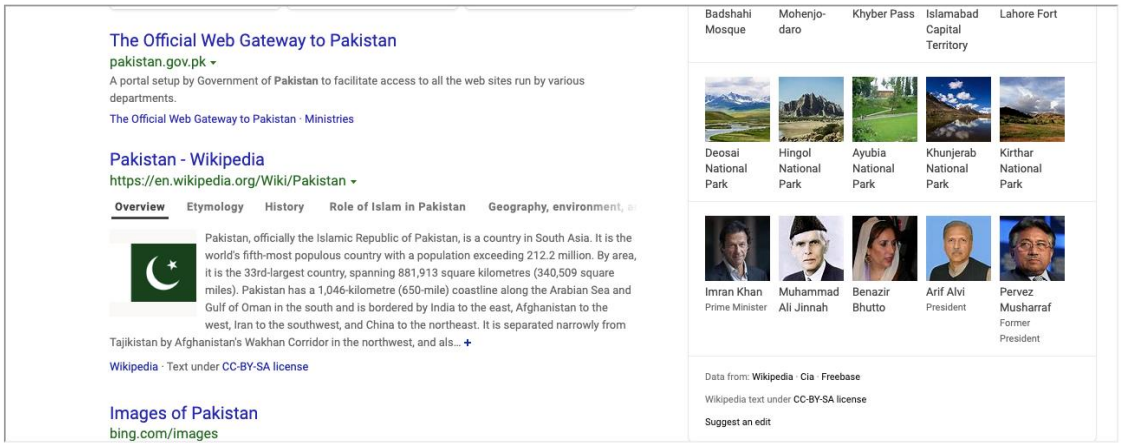
Sasipreetam Morsa/sxm171330

User Interface and Comparisons with Google and Bing

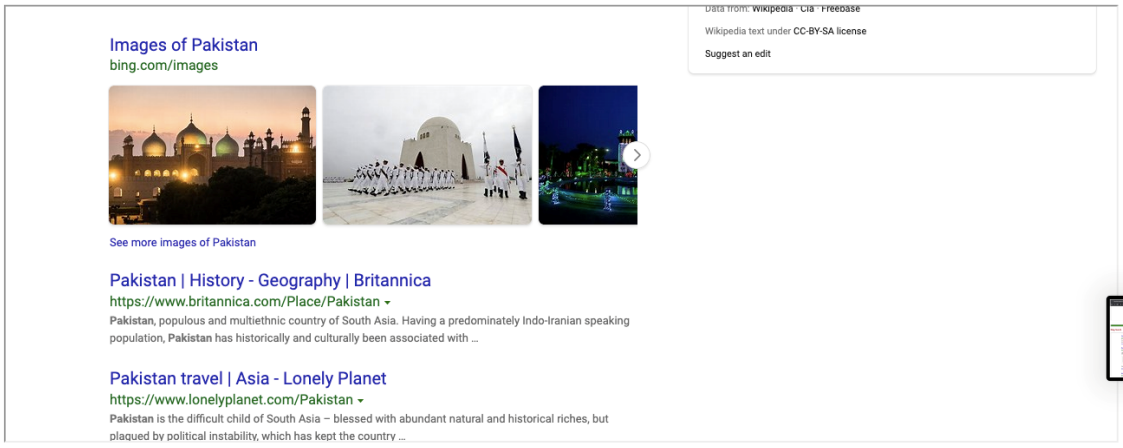
Bing Search



Bing Search



Bing Search



5. Clustering (Anshul Pardhi)

Flat Clustering

K-means clustering algorithm is implemented for flat clustering of documents.

K-means clustering algorithm

Given a set of observations ($\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_n$), where each observation is a *d*-dimensional real vector, *k*-means clustering aims to partition the *n* observations into *k* (≤ *n*) sets $\mathbf{S} = \{S_1, S_2, ..., S_k\}$ so as to minimize the within-cluster sum of squares (WCSS). This WCSS is the sum of the squares of distances of all the observations belonging to a cluster from the centroid of the cluster. Formally, the objective is to find:

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \|\mathbf{x} - \boldsymbol{\mu}_i\|^2$$

where $\boldsymbol{\mu}_i$ is the mean of points in S_i.

Selection of predefined clusters (value of k)

Values of k, ranging from 3 to 20 was tested on a sample of the dataset and the observed clusters were compared to the relevancy of the documents. Elbow method was also used to determine an appropriate value of k. After observing the results testing different values of k, a value of k = 11 gave most relevant results. As a result, all the fetched documents are stored in one of the 11 clusters.

Flat Clustering Algorithm Design & Implementation

All the documents crawled by Darrel (the student responsible for crawling) were indexed by Ruchi (the student responsible for indexing). Ruchi gave me the response generated by Apache Solr when a query "*" was given. This query retrieved all the crawled documents in a JSON format, and the results were dumped to a file. The size of this file was 1 GB.

- I extracted the "content" key's value from the JSON, which contained the content of the document in plain text format, removing the DOM and HTML tags.
- URL corresponding to the content was also extracted.
- The "content" value from all the crawled documents was stored in a list.
- The content was later vectorized using scikit-learn's TfidfVectorizer.

- Scikit-learn's KMeans was then used to apply the K-means clustering algorithm, with a value of k chosen as 11.
- The clusters obtained for a document after applying K-means algorithm as well as the URLs corresponding to that document were converted to Pandas series.
- The results were stored in a text file in the following format
    - URL, ClusterNumber

A sample of the generated clusters from the text file is shown below.

https://en.wikipedia.org/wiki/Foreign_relations_of_Afghanistan,1.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Algeria,6.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Antigua_and_Barbuda,4.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Argentina,4.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Armenia,7.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Australia,8.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Azerbaijan,1.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Brazil,4.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Denmark,7.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Finland,7.0

https://en.wikipedia.org/wiki/Foreign_relations_of_France,7.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Germany,7.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Morocco,6.0

https://en.wikipedia.org/wiki/Foreign_relations_of_New_Zealand,8.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Niue,8.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Northern_Cyprus,1.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Norway,7.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Pakistan,1.0

https://en.wikipedia.org/wiki/Foreign_relations_of_Palestine,1.0

https://en.wikipedia.org/wiki/Foreign_relations_of_South_Korea,1.0

https://en.wikipedia.org/wiki/Foreign_relations_of_South_Sudan,6.0

https://en.wikipedia.org/wiki/Foreign_relations_of_the_Cook_Islands,8.0

As observed from the sample above, the clusters are created based on continents where a country belongs to (all Asian countries are grouped in cluster 1.0, all European countries are grouped in cluster 7.0 and so on).

The generated file contains cluster information of all the crawled pages.

Clustering results incorporation and integration with UI

- On the UI, there are 2 clustering modes: flat and hierarchical.
- If any of the clustering modes is selected, the GET request contains that clustering mode in the type parameter.
- Solr fetches the top 50 relevant results based on page ranking and HITS score and sends it to clustering computation handler.
- Inside the clustering computation handler, the results are ranked in such a way that cluster to which the top-ranked result belongs to, all the other results belonging to the same cluster in the order of decreasing page rank and HITS scores are ranked next.
- The process repeats for the next highest ranked result and all the pages belonging to its cluster are ranked next.
- This process repeats till we reach our threshold of 50.
- The updated ranks are then sent as a response to the UI.

Hierarchical Clustering

Agglomerative clustering algorithm with single link is implemented for hierarchical clustering of documents.

Agglomerative clustering algorithm

Hierarchical clustering is a method of cluster analysis which seeks to build a hierarchy of clusters. Agglomerative clustering is a bottom up approach of hierarchical clustering. Here, each observation, document here, starts in its own cluster, and pairs of clusters are merged as one moves up the hierarchy. The results of hierarchical clustering are usually represented by a dendrogram.

For computing distance, there are various distance metrics to choose from: Euclidean, Manhattan, Mahalanobis, to name a few. For the project, I have used Euclidean distance metric.

For computing the linkage criteria, to compute the distance between different clusters, there are a lot of metrics to choose from: maximum or complete link, minimum or single link, unweighted average link, weighted average link, centroid link, ward etc. For the project, I have used single link metric.

ANSHUL PARDHI
ARP180012

Hierarchical Clustering Algorithm Design & Implementation

All the documents crawled by Darrel (the student responsible for crawling) were indexed by Ruchi (the student responsible for indexing). Ruchi gave me the response generated by Apache Solr when a query "*" was given. This query retrieved all the crawled documents in a JSON format, and the results were dumped to a file. The size of this file was 1 GB.

- I extracted the "content" key's value from the JSON, which contained the content of the document in plain text format, removing the DOM and HTML tags.
- URL corresponding to the content was also extracted.
- The "content" value from all the crawled documents was stored in a list.
- The content was later vectorized using scikit-learn's TfidfVectorizer.
- Vectorizer's output was converted to match agglomerative clustering algorithm's input.
- Python's fastcluster library is used to efficiently compute the results of hierarchical clustering faster.
- Firstly, linkage is calculated, and then the single method of fastcluster library is called to compute the clusters to which a URL belongs to.
- A dendrogram is plotted as well to show the results of clustering visually.
- The clusters obtained for a document after applying agglomerative clustering algorithm as well as the URLs corresponding to that document were converted to Pandas series.
- The results were stored in a text file in the following format
    - URL, ClusterNumber

A sample of the generated clusters from the text file is shown below.

https://www.worldatlas.com/webimage/countrys/namerica/usstates/sc.htm,6.0

https://www.worldatlas.com/webimage/countrys/namerica/usstates/ustravel.htm,6.0

https://www.worldatlas.com/webimage/countrys/oceania/mp.htm,1.0

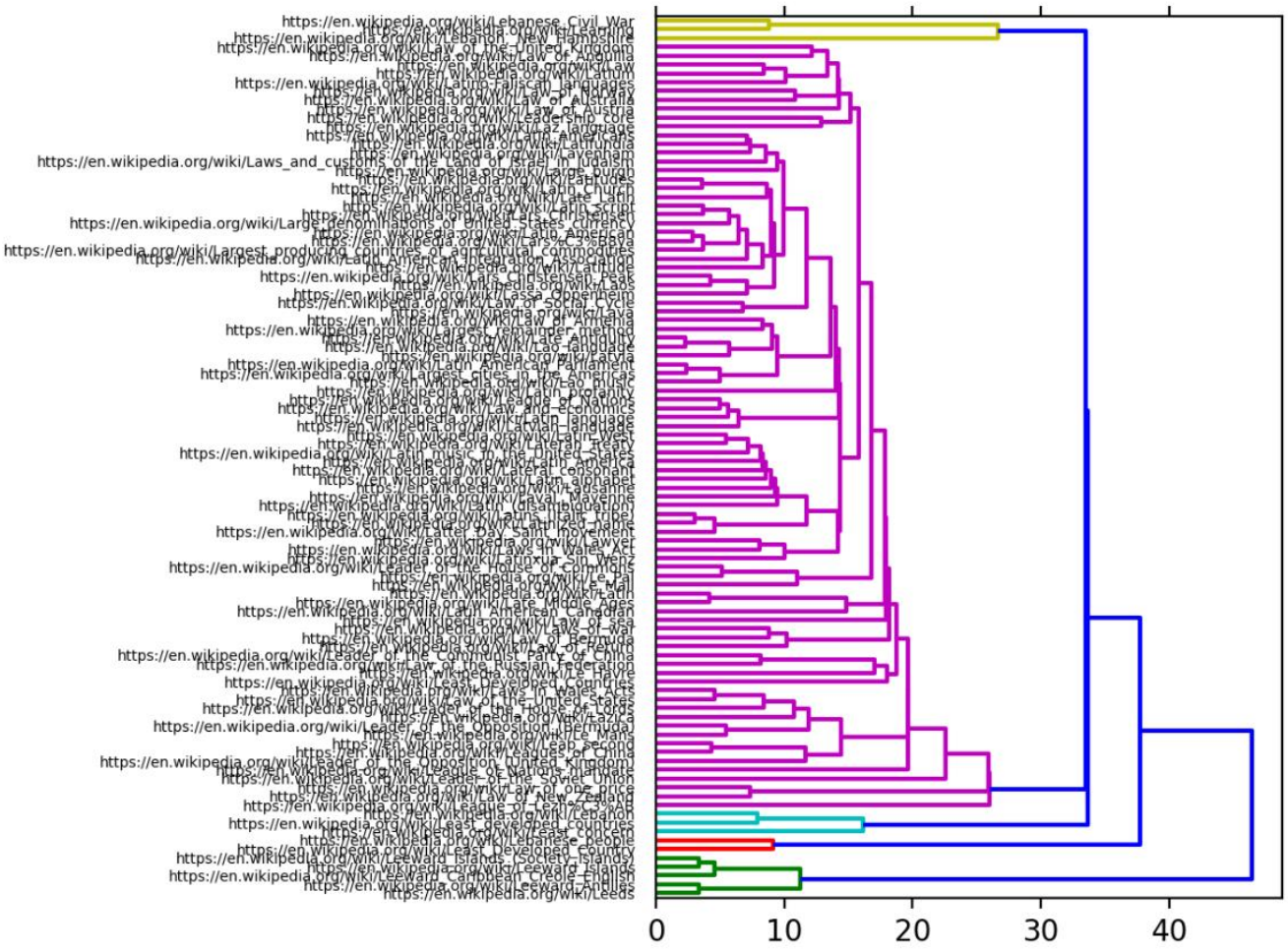
https://www.worldatlas.com/webimage/countrys/oceania/fm.htm,1.0

https://www.worldatlas.com/webimage/countrys/oceania/tk.htm,1.0

Since a dendrogram containing every links can get a lot cluttered and difficult to visualize, I have generated a couple of dendrograms, taking samples from the data. A dendrogram containing all the webpages is shown as well in the end.

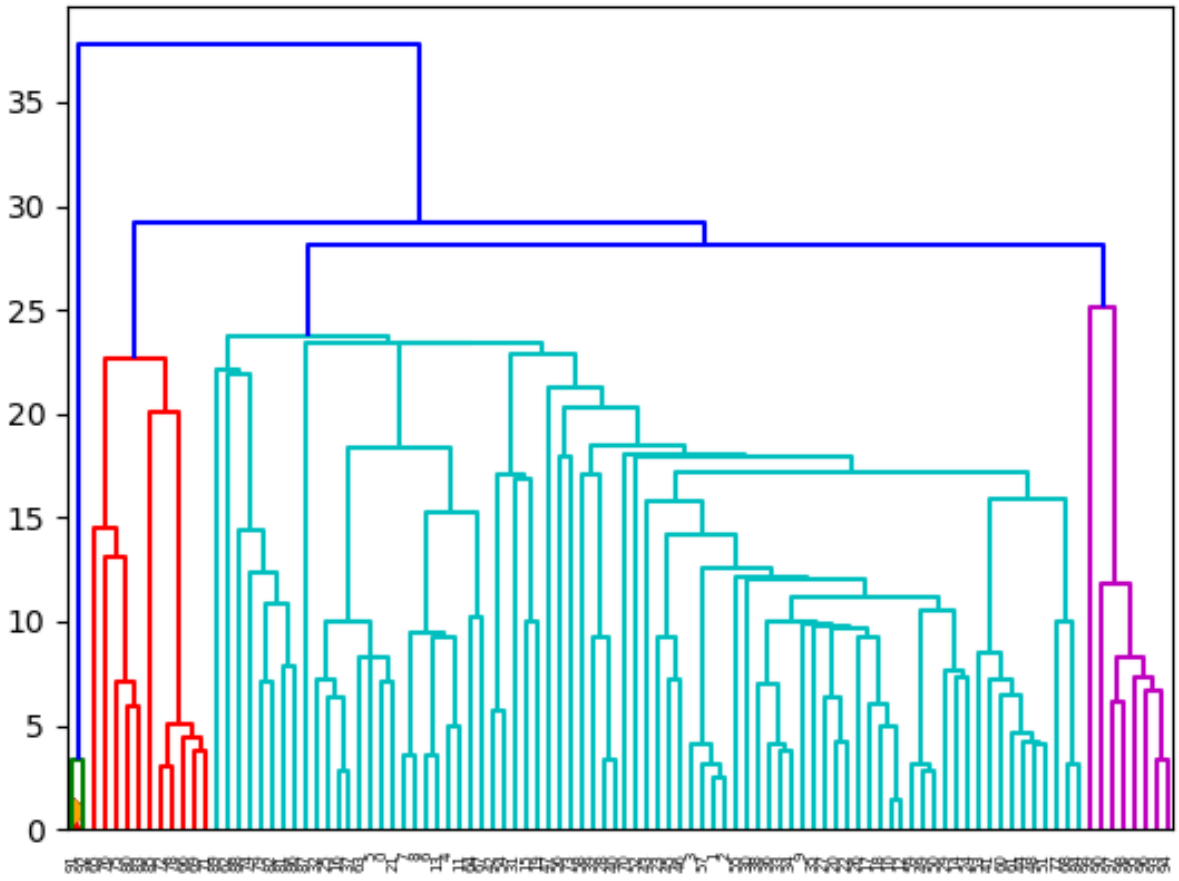
ANSHUL PARDHI
ARP180012

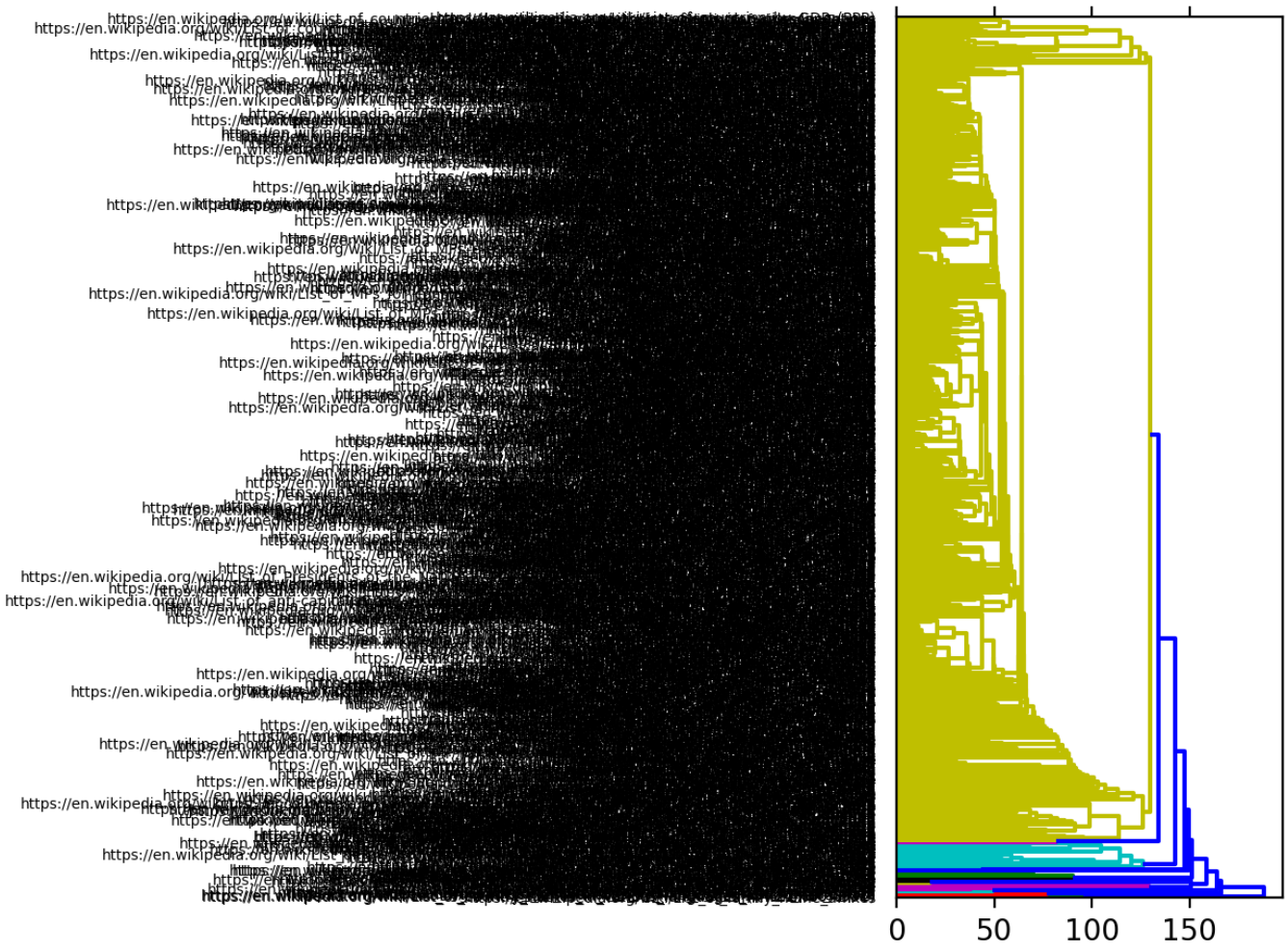
CLUSTERING



ANSHUL PARDHI
ARP180012

CLUSTERING





Number of clusters obtained

It was observed that fastcluster's agglomerative clustering selects the best number of clusters depending on the type of input data provided, and the various distance and linkage metrics used. It was observed that for our dataset, 7 clusters were created and all the links fall into one of these 7 clusters.

Clustering results incorporation and integration with UI

The steps are like those for flat clustering. There are two types of values for the "type" parameter in the request query from the UI, "flat_clustering" and "hierarchical_clustering". Depending on the clustering mode selected, that particular mode's clusters are selected for the re-computation of ranks of retrieved pages.

ANSHUL PARDHI
ARP180012

Number of queries experimented with to improve results

Around 50 queries were used per clustering method to improve the results obtained. Some of the results obtained after running these queries will be listed below in a further subsection. Observations are also provided below the screenshots which shows the impact of clustering on improving the relevancy of the retrieved results.

Query testing generation and impact on results and relevancy

I have used 50 queries for each of the two clustering methods: flat and hierarchical, to test the impact of the results of each clustering method. The queries were the generated manually, depending on the initial results obtained from the original relevance model, using page rank and HITS. The results obtained after applying clustering seem relevant, as similar results are grouped together to be shown earlier than the non-relevant results.

Selection of queries

The criteria for selection of queries is based on the observation obtained from the type of clusters formed for both the clustering methods used. As shown in the sample subset of the results obtained above, one of the query selection criteria is continents, as it was observed that countries belonging to same continents were grouped together. Another selection criteria is the topic asked, such as map or constitution etc. For example, maps of different countries are grouped together while constitutions of different countries are grouped together.

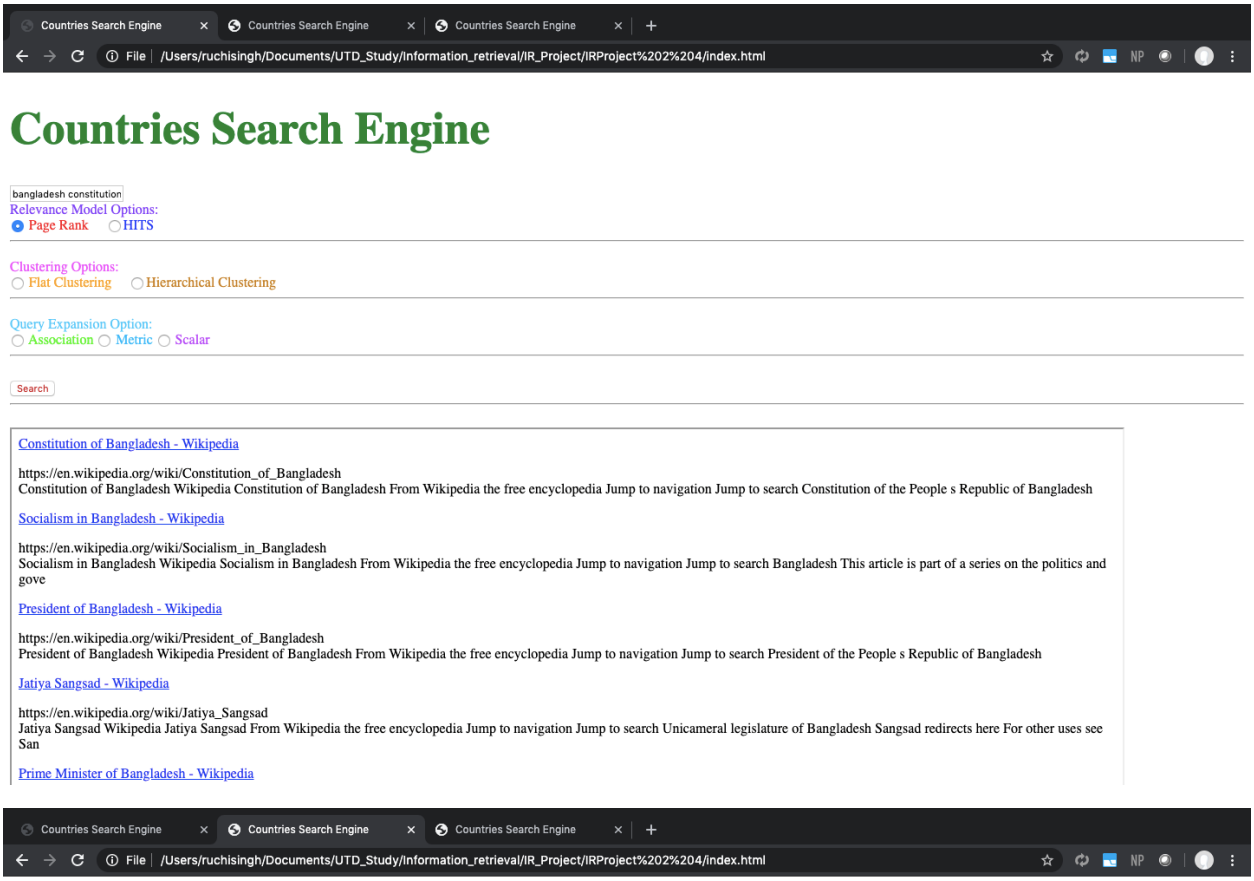
Query examples with clustering

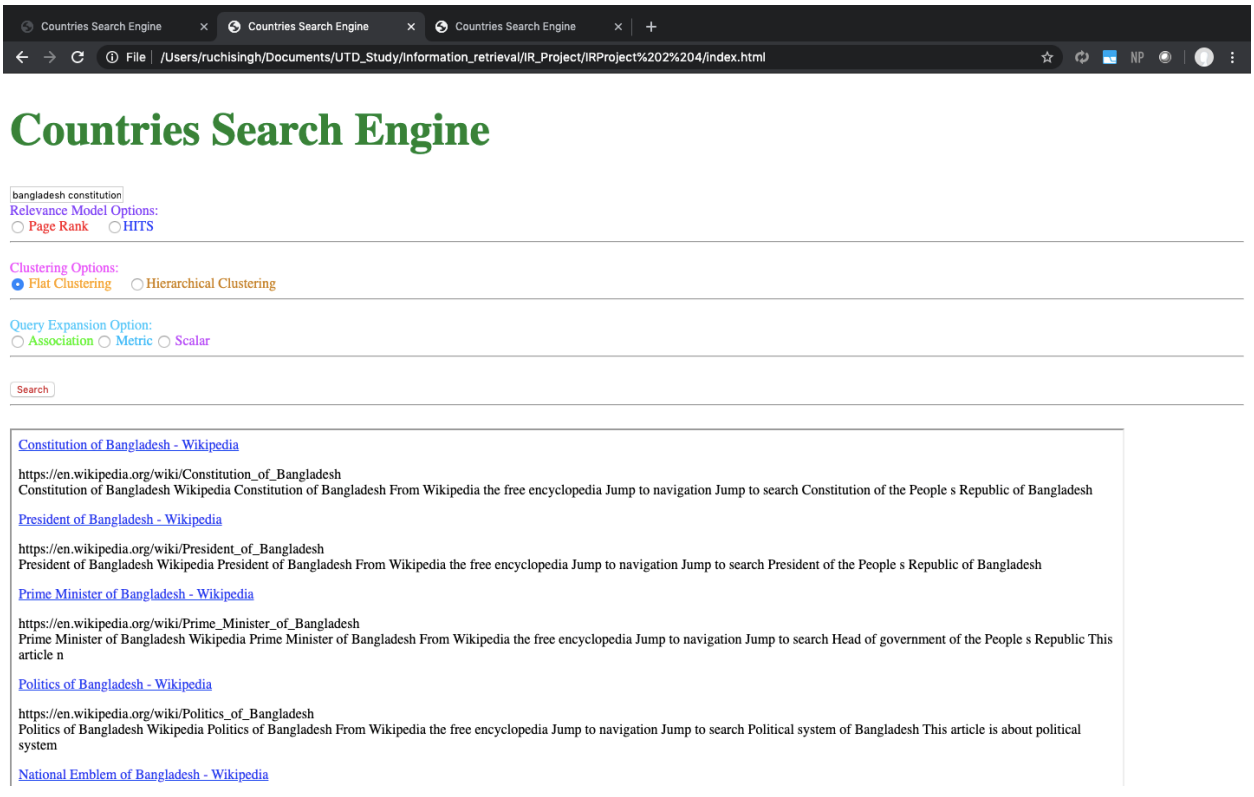
About 50 queries for both flat as well as hierarchical were tested. I have attached the results for 5 such queries below. The first image shows the results obtained by applying page rank. The second image shows the results obtained by applying flat clustering. The third image shows the results obtained after applying hierarchical clustering.

Query 1: Bangladesh constitution

ANSHUL PARDHI
ARP180012

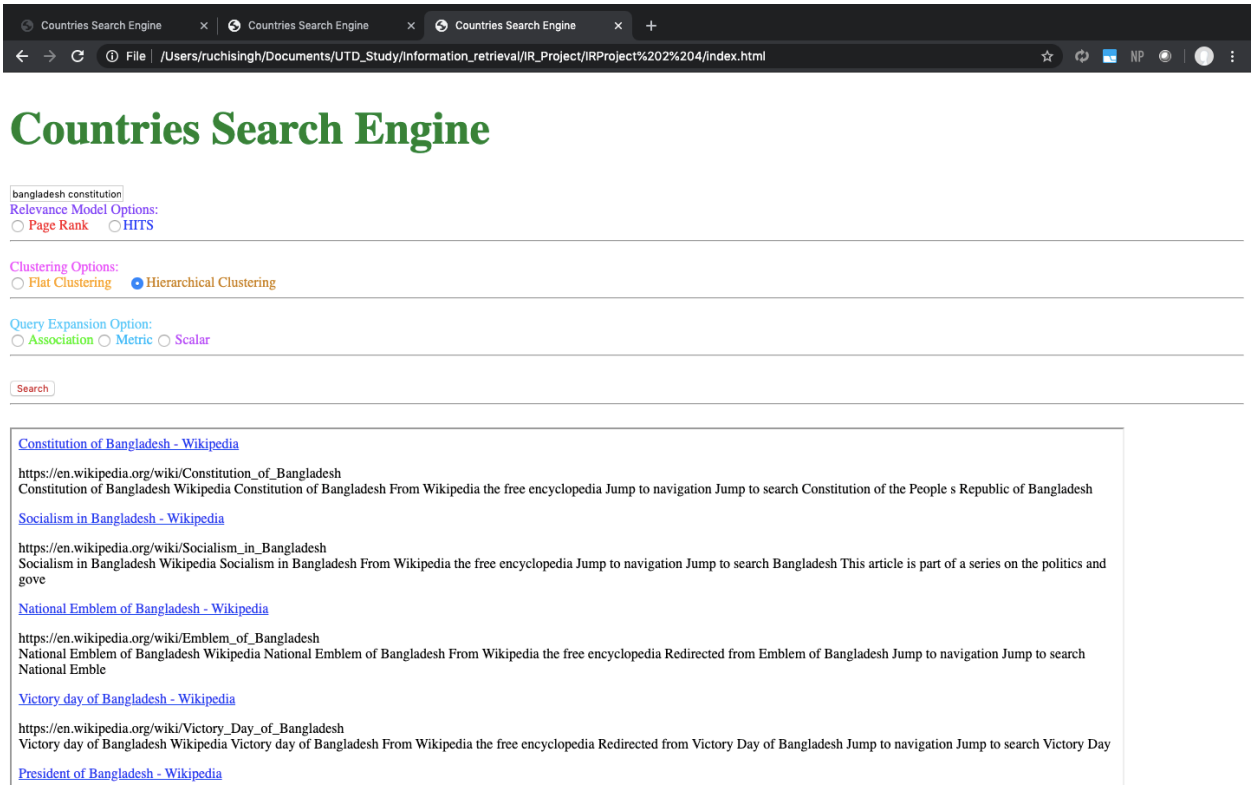
CLUSTERING





ANSHUL PARDHI
ARP180012

CLUSTERING

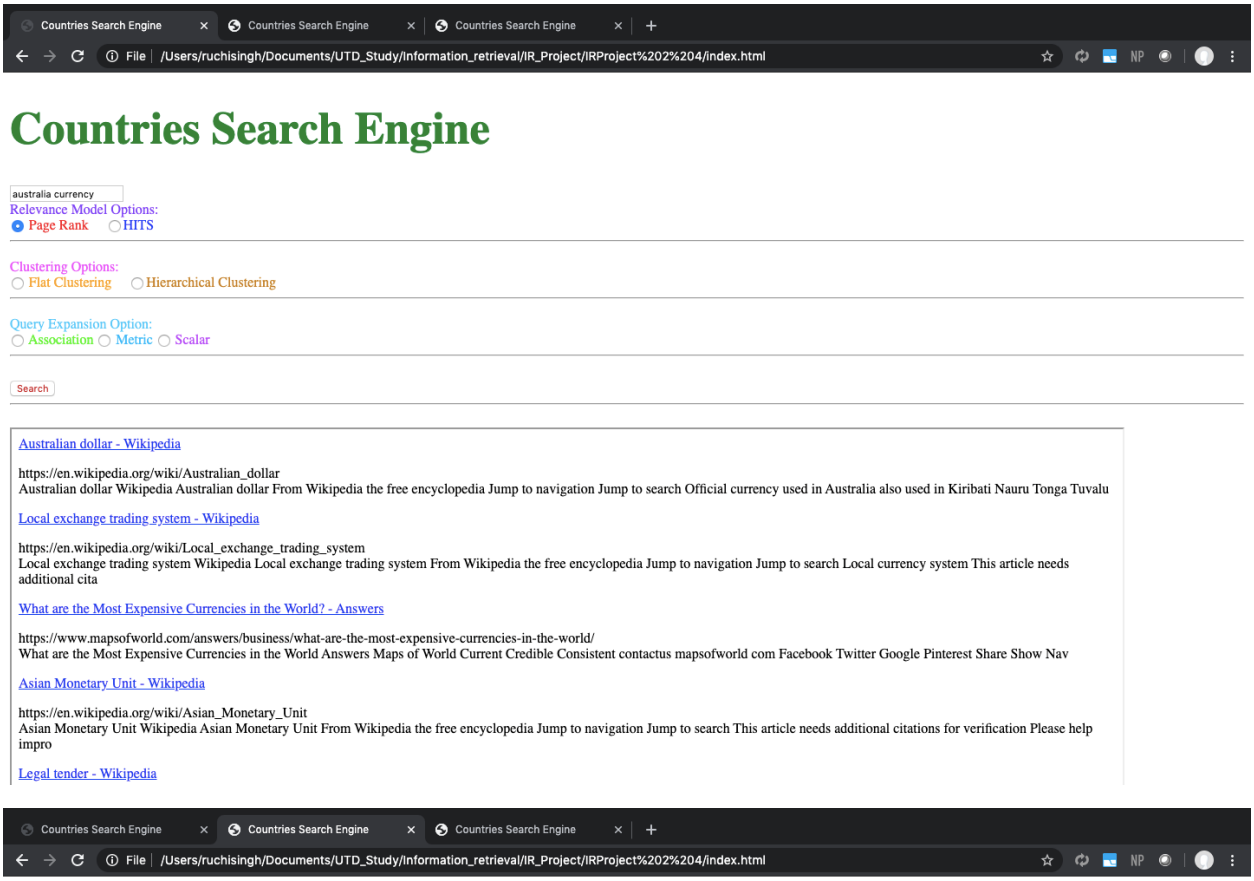


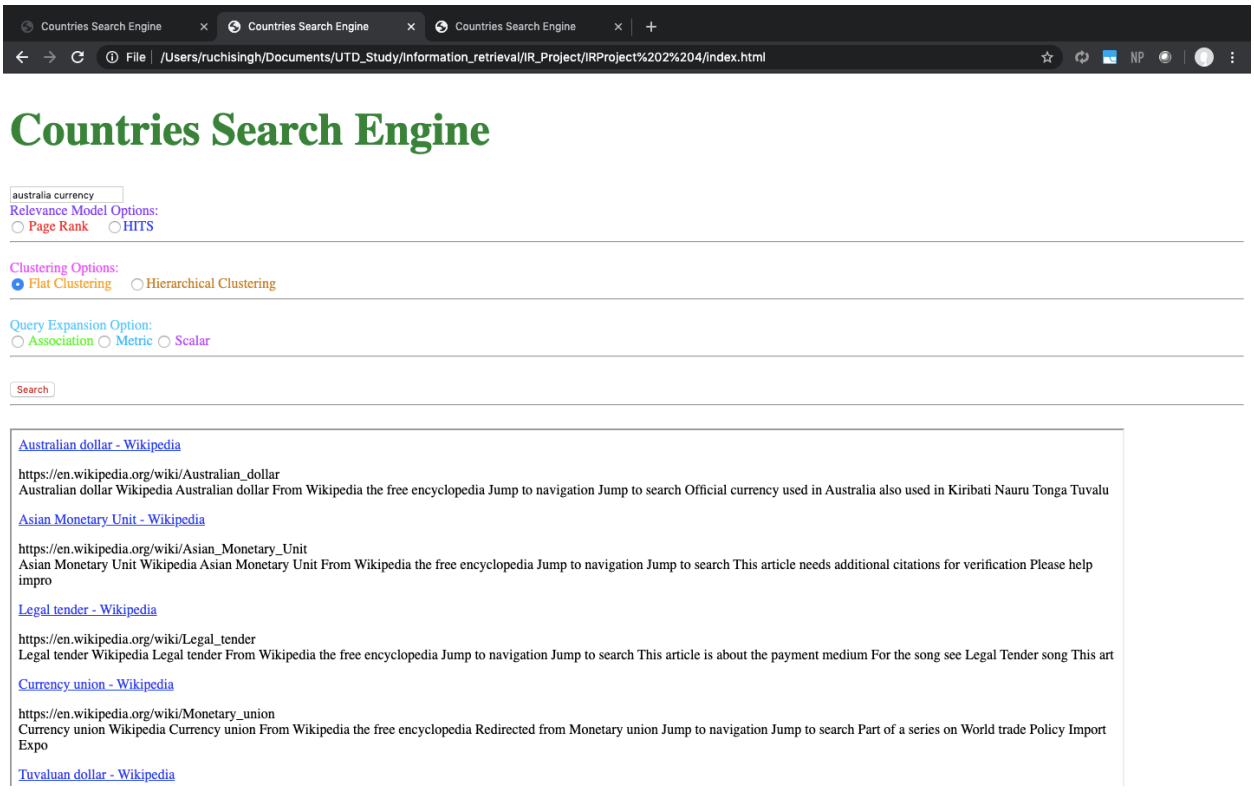
Observations: Clustering results show information relevant to constitution such as information about Prime Minister, President, Emblem, Politics of Bangladesh etc. Such results aren't shown by simply applying page rank.

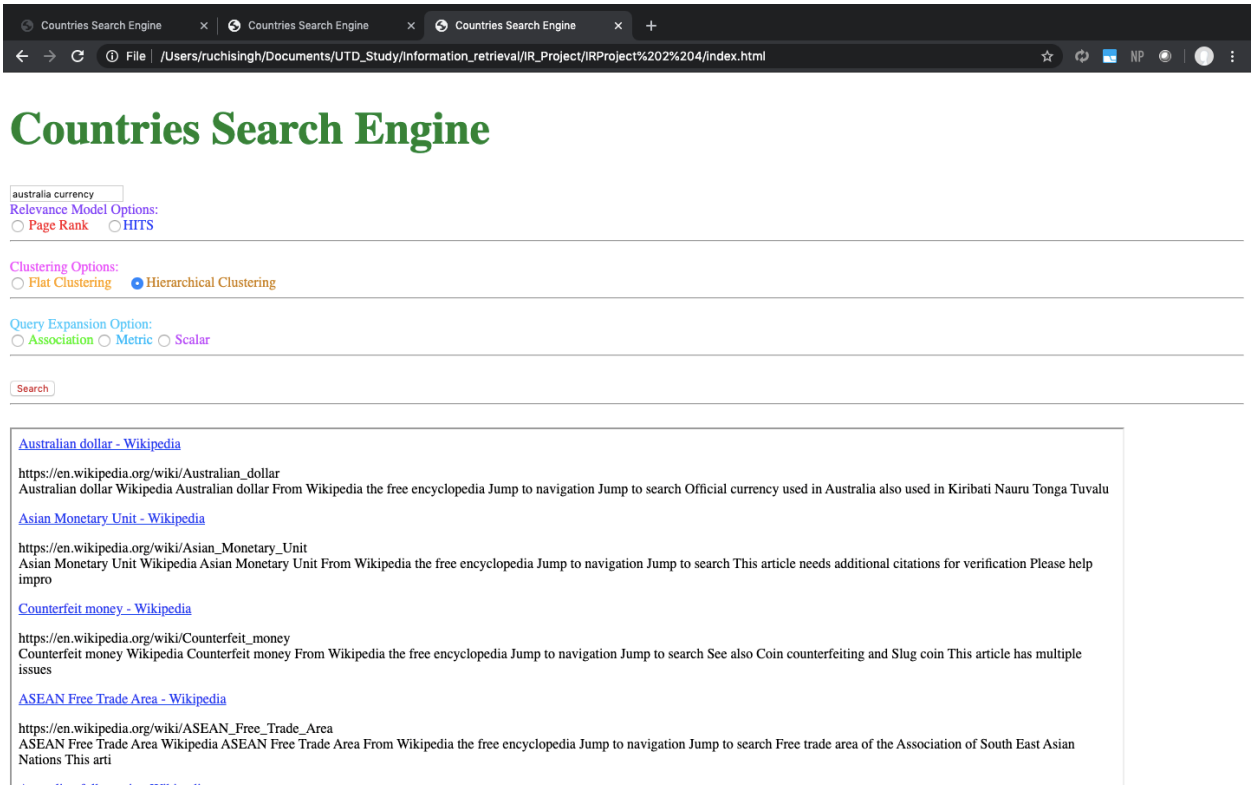
Query 2: Australia currency

ANSHUL PARDHI
ARP180012

CLUSTERING





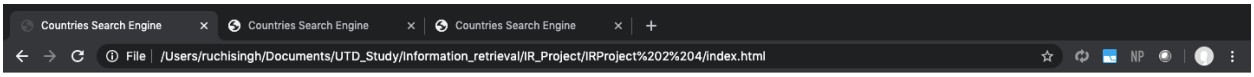


Observations: Clustering results show results like currency such as monetary unit, legal tender, currency union, counterfeit money etc. Furthermore, these results show similar results of countries near Australia such as Tuvalu.

Query 3: European Union

ANSHUL PARDHI
ARP180012

CLUSTERING



Countries Search Engine

european union
Relevance Model Options:
Page Rank HITS

Clustering Options:
Flat Clustering Hierarchical Clustering

Query Expansion Option:
Association Metric Scalar

Search

List of countries by leading trade partners - Wikipedia

https://en.wikipedia.org/wiki/List_of_countries_by_leading_trade_partners
List of countries by leading trade partners Wikipedia List of countries by leading trade partners From Wikipedia the free encyclopedia Jump to navigation Jump to search Wikipedia list article Part

Portal:European Union - Wikipedia

https://en.wikipedia.org/wiki/Portal:European_Union
Portal European Union Wikipedia Portal European Union From Wikipedia the free encyclopedia Jump to navigation Jump to search Wikimedia portal Portal maintenance status February This portal

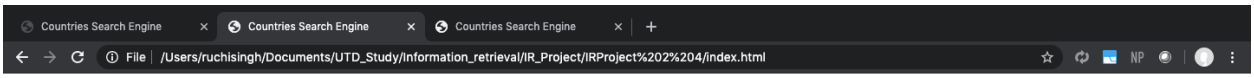
European Union Customs Union - Wikipedia

https://en.wikipedia.org/wiki/European_Union_Customs_Union
European Union Customs Union Wikipedia European Union Customs Union From Wikipedia the free encyclopedia Jump to navigation Jump to search This article is about European Union Customs Union It is

Treaties of the European Union - Wikipedia

https://en.wikipedia.org/wiki/Treaties_of_the_European_Union
Treaties of the European Union Wikipedia Treaties of the European Union From Wikipedia the free encyclopedia Jump to navigation Jump to search Treaties of the European Union Front page of an EU doc

Institutions of the European Union - Wikipedia



Countries Search Engine

european union
Relevance Model Options:
Page Rank HITS

Clustering Options:
Flat Clustering Hierarchical Clustering

Query Expansion Option:
Association Metric Scalar

Search

List of countries by leading trade partners - Wikipedia

https://en.wikipedia.org/wiki/List_of_countries_by_leading_trade_partners
List of countries by leading trade partners Wikipedia List of countries by leading trade partners From Wikipedia the free encyclopedia Jump to navigation Jump to search Wikipedia list article Part

Withdrawal from the European Union - Wikipedia

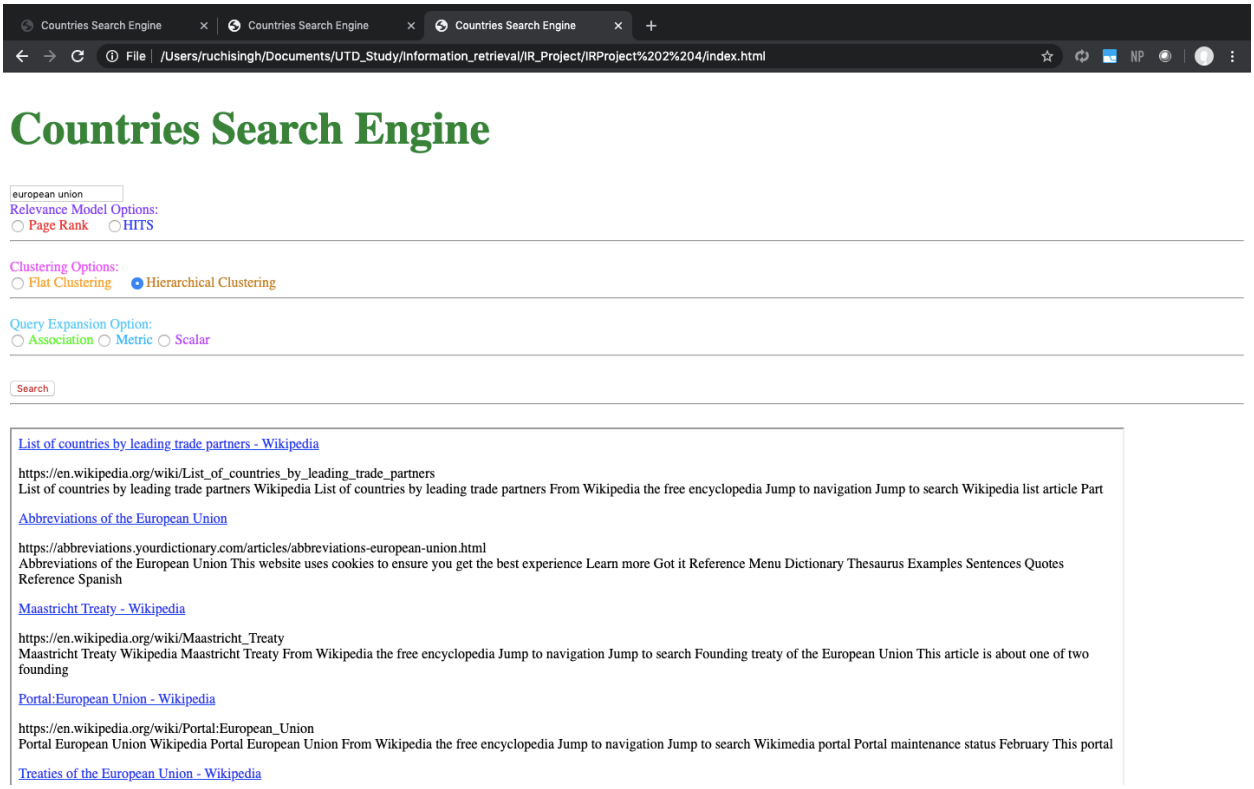
https://en.wikipedia.org/wiki/Withdrawal_from_the_European_Union
Withdrawal from the European Union Wikipedia Withdrawal from the European Union From Wikipedia the free encyclopedia Jump to navigation Jump to search Article redirects here For the former Eu

Abbreviations of the European Union

https://abbreviations.yourdictionary.com/articles/abbreviations-european-union.html
Abbreviations of the European Union This website uses cookies to ensure you get the best experience Learn more Got it Reference Menu Dictionary Thesaurus Examples Sentences Quotes Reference Spanish

Police and Judicial Co-operation in Criminal Matters - Wikipedia

https://en.wikipedia.org/wiki/Police_and_Judicial_Co-operation_in_Criminal_Matters
Police and Judicial Co operation in Criminal Matters Wikipedia Police and Judicial Co operation in Criminal Matters From Wikipedia the free encyclopedia Jump to navigation Jump to search PJCC red

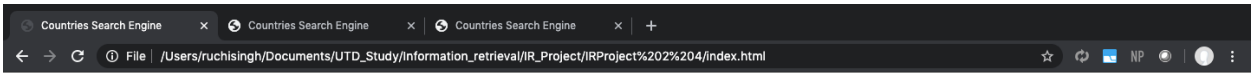


Observations: Hierarchical clustering shows a result Maastricht Treaty. It was the treaty that led to the creation of the European Union. Such results are missing from the top results of page rank.

Query 4: Mumbai

ANSHUL PARDHI
ARP180012

CLUSTERING



Countries Search Engine

mumbai
Relevance Model Options:
Page Rank HITS

Clustering Options:
Flat Clustering Hierarchical Clustering

Query Expansion Option:
Association Metric Scalar

Search

Mumbai - Wikipedia

https://en.wikipedia.org/wiki/Mumbai
Mumbai Wikipedia Mumbai From Wikipedia the free encyclopedia Jump to navigation Jump to search Bombay redirects here For other uses see Bombay disambiguation and Mumbai disambiguation Meg

Mumbai Population 2020 (Demographics, Maps, Graphs)

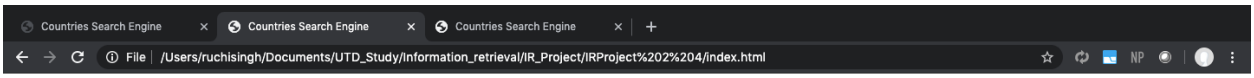
https://worldpopulationreview.com/world-cities/mumbai-population/
Mumbai Population Demographics Maps Graphs World Population Review Mobile Navigation Home Continents Countries World Cities US States US Counties US Cities Zips Canadian Provinces Coronavirus

Administrative divisions of Mumbai - Wikipedia

https://en.wikipedia.org/wiki/Administrative_divisions_of_Mumbai
Administrative divisions of Mumbai Wikipedia Administrative divisions of Mumbai From Wikipedia the free encyclopedia Jump to navigation Jump to search This article needs additional citations for ve

History of Mumbai from an Arabian seaport to Bollywood | Britannica

https://www.britannica.com/video/141204/Video-overview-Mumbai-history
History of Mumbai from an Arabian seaport to Bollywood Britannica Search Britannica Encyclop dia Britannica Login Categories Science Technology Health Medicine Sports Recreation Geography Trav



Countries Search Engine

mumbai
Relevance Model Options:
Page Rank HITS

Clustering Options:
Flat Clustering Hierarchical Clustering

Query Expansion Option:
Association Metric Scalar

Search

Mumbai - Wikipedia

https://en.wikipedia.org/wiki/Mumbai
Mumbai Wikipedia Mumbai From Wikipedia the free encyclopedia Jump to navigation Jump to search Bombay redirects here For other uses see Bombay disambiguation and Mumbai disambiguation Meg

Mumbai Population 2020 (Demographics, Maps, Graphs)

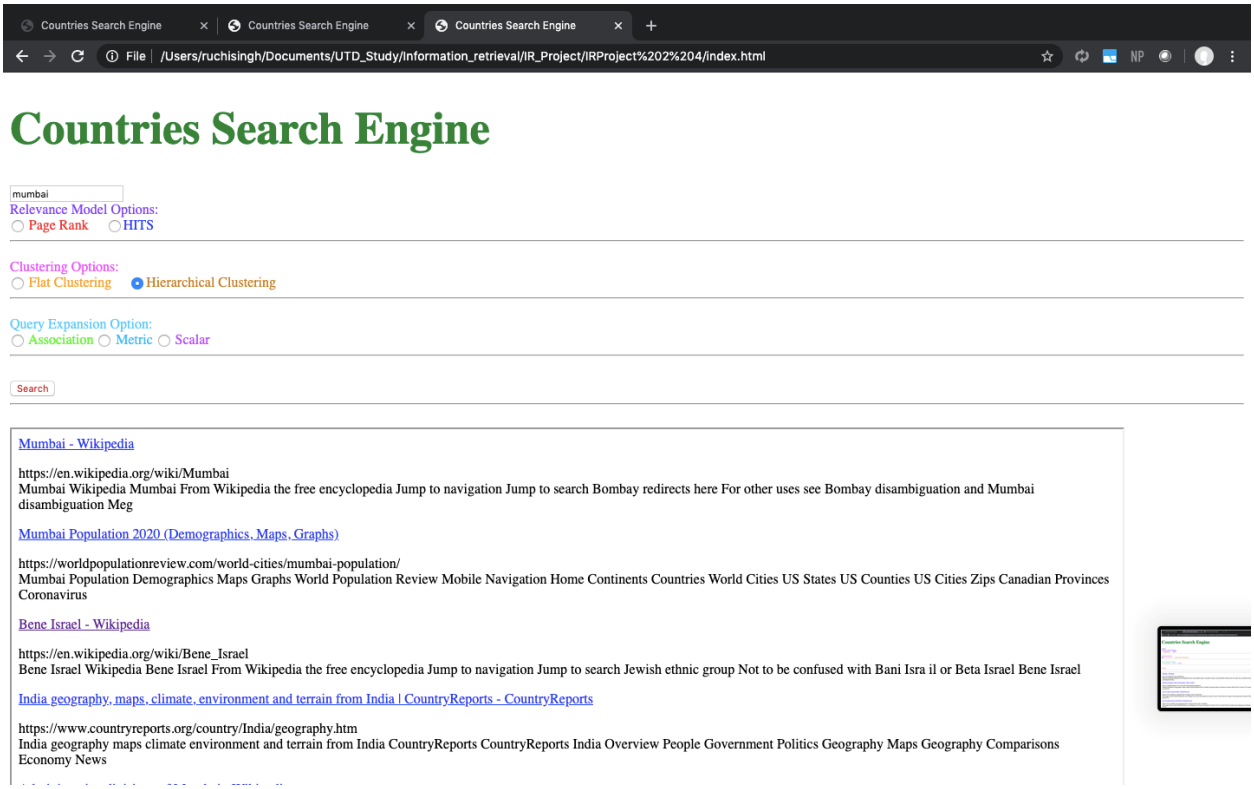
https://worldpopulationreview.com/world-cities/mumbai-population/
Mumbai Population Demographics Maps Graphs World Population Review Mobile Navigation Home Continents Countries World Cities US States US Counties US Cities Zips Canadian Provinces Coronavirus

The 10 Largest Cities in India - WorldAtlas.com

https://www.worldatlas.com/articles/the-10-largest-cities-in-india.html
The Largest Cities in India WorldAtlas com Trending Covid Cases And Deaths Per State In The US Could There Be Another Great Depression Celebrities Who Have Recovered From COVID Society Th

The 10 Largest Cities in the World - WorldAtlas.com

https://www.worldatlas.com/amp/articles/the-10-largest-cities-in-the-world.html
The Largest Cities in the World WorldAtlas com Trending Covid Cases And Deaths Per State In The US Could There Be Another Great Depression Celebrities Who Have Recovered From COVID World

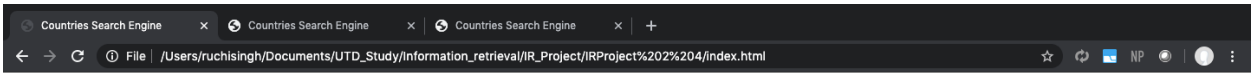


Observations: Clustering not only improves results for countries, but it also improves results for cities. Here flat clustering groups relevant information like the top 10 largest cities in India as well as the world. Furthermore, hierarchical clustering shows a result, Bene Israel. These people are the Jewish community settled predominantly in Mumbai. Such results aren't ranked that high when applying page rank.

Query 5: Guatemala

ANSHUL PARDHI
ARP180012

CLUSTERING



Countries Search Engine

guatemala
Relevance Model Options:
Page Rank HITS

Clustering Options:
Flat Clustering Hierarchical Clustering

Query Expansion Option:
Association Metric Scalar

Search

Guatemala Map | Map of Guatemala

https://www.mapsofworld.com/guatemala/
Guatemala Map Map of Guatemala Menu World Map World Maps Political Map of the World Physical Map of the World Blank World Map World Map for Kids Earth Map World Atlas World Time Zones Map Coun

Guatemala City - Wikipedia

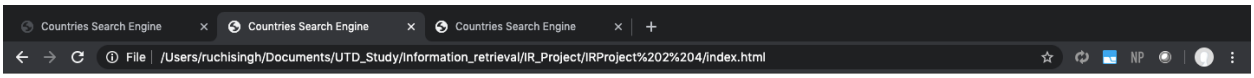
https://en.wikipedia.org/wiki/Guatemala_City
Guatemala City Wikipedia Guatemala City From Wikipedia the free encyclopedia Jump to navigation Jump to search Capital of Guatemala Capital City in Guatemala Guatemala Guatemala City Guatemala Cap

Guatemala Population 2020 (Demographics, Maps, Graphs)

https://worldpopulationreview.com/countries/guatemala-population/
Guatemala Population Demographics Maps Graphs World Population Review Mobile Navigation Home Continents Countries World Cities US States US Counties US Cities Zips Canadian Provinces Coronavi

Guatemala - Wikipedia

https://en.wikipedia.org/wiki/Guatemala
Guatemala Wikipedia Guatemala From Wikipedia the free encyclopedia This is the latest accepted revision reviewed on April Jump to navigation Jump to search This article is about the cou



Countries Search Engine

guatemala
Relevance Model Options:
Page Rank HITS

Clustering Options:
Flat Clustering Hierarchical Clustering

Query Expansion Option:
Association Metric Scalar

Search

Guatemala Map | Map of Guatemala

https://www.mapsofworld.com/guatemala/
Guatemala Map Map of Guatemala Menu World Map World Maps Political Map of the World Physical Map of the World Blank World Map World Map for Kids Earth Map World Atlas World Time Zones Map Coun

Current Local Time in Guatemala City, Guatemala

https://www.timeanddate.com/worldclock/guatemala/guatemala
Current Local Time in Guatemala City Guatemala Menu timeanddate com Search Site Articles City Country Social Share this page Follow us on Facebook Tweet Follow Twitter Visit us on Faceboo

Mayan | Ethnologue

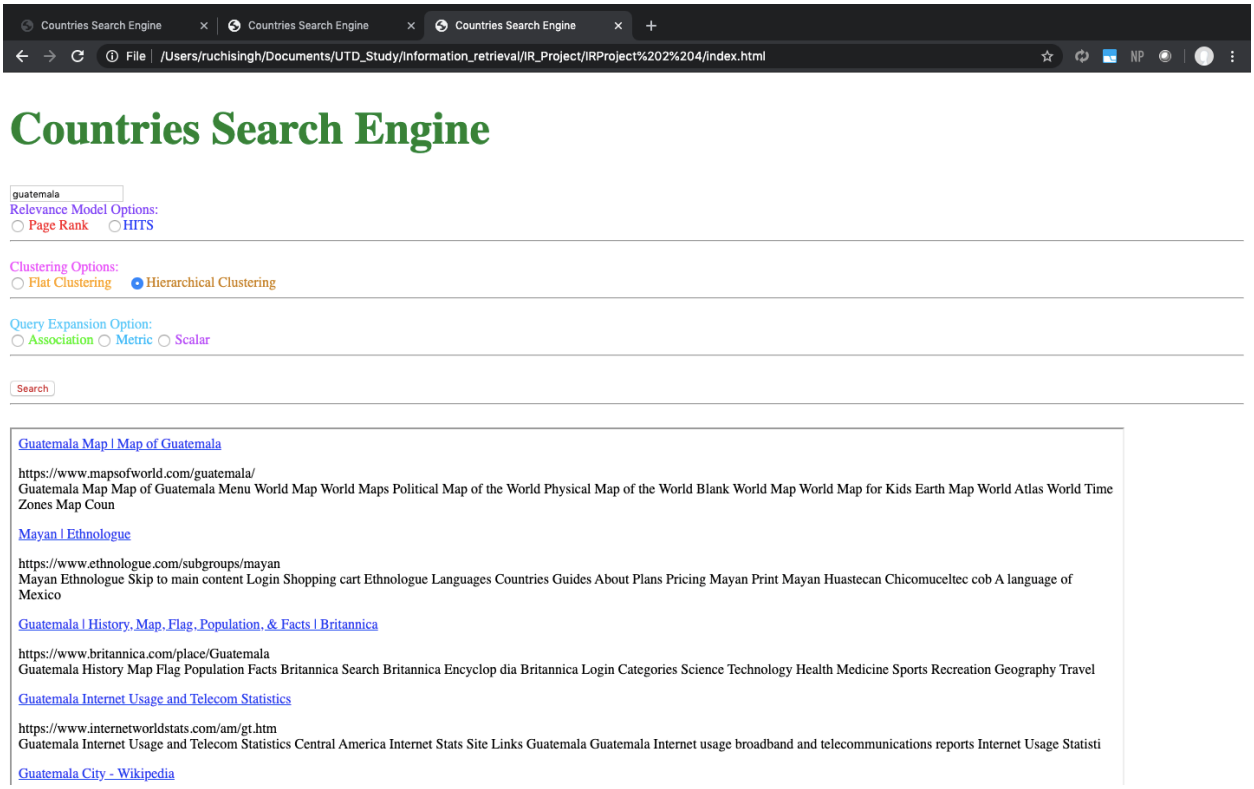
https://www.ethnologue.com/subgroups/mayan
Mayan Ethnologue Skip to main content Login Shopping cart Ethnologue Languages Countries Guides About Plans Pricing Mayan Print Mayan Huastecan Chicomuceltec cob A language of Mexico

Guatemala City Population 2020 (Demographics, Maps, Graphs)

https://worldpopulationreview.com/world-cities/guatemala-city-population/
Guatemala City Population Demographics Maps Graphs World Population Review Mobile Navigation Home Continents Countries World Cities US States US Counties US Cities Zips Canadian Provinces Cor

ANSHUL PARDHI
ARP180012

CLUSTERING



Observations: The majority of indigenous people in Guatemala are of Mayan origin. Clustering handles such details and shows Mayan results on querying Guatemala. Such results don't get a top rank when page rank is used.

6.Query Expansion and Relevance Feedback

Query expansion is used to improve search results by following local methods:

1) Relevance Feedback (Roccio Algorithm)
2) Pseudo Relevance Feedback
   a) Association cluster
   b) Metric cluster
   c) Scalar cluster

The following formula is used to implement Roccio algorithm:

- Used in practice:

$$\vec{q}_m = \alpha\vec{q}_0 + \beta\frac{1}{|D_r|}\sum_{\vec{d}_j \in D_r}\vec{d}_j - \gamma\frac{1}{|D_{nr}|}\sum_{\vec{d}_j \in D_{nr}}\vec{d}_j$$

- D_r = set of known relevant doc vectors
- D_{nr} = set of known irrelevant doc vectors
  - Different from C_r and C_{nr} 
- q_m = modified query vector; q_0 = original query vector; α,β,γ: weights (hand-chosen or set empirically)
- New query moves toward relevant documents and away from irrelevant documents

According to this image, 20 queries were fed as q_0 as input and the algorithm was tested for alpha =1.0, beta = 0.9 and gamma =0.1 values , so that a set of weights could be found that can separate relevant documents from the irrelevant documents.

The 20 queries were selected based on number of query parameters, spelling mistakes, regional name of countries and queries containing words that are states or cities (outside of country domain).

1) 4 queries with one keyword for the data that was crawled
   a) australia
   b) china
   c) ethiopia
   d) pakistan
2) 4 queries with two keywords for the data that was crawled
   a) bangladesh area
   b) pakistan terrorism
   c) china population
   d) india pakistan
3) 4 queries with spelling mistakes in the query words for the data that was crawled
   a) australai
   b) paksitan
   c) chna
   d) germny
4) 4 queries with regional name for countries
   a) hindustan
   b) britaan
   c) franche

d) gana

5) 4 queries with the words that are outside of country domain
   a) texas
   b) europe
   c) mumbai
   d) doha

examples of the web pages that are found relevant:

Query	**Relevant web pages**
australia	https://en.wikipedia.org/wiki/Australia
	https://www.britannica.com/place/Australia
	https://en.wikipedia.org/wiki/Demographics_of_Australia
	https://en.wikipedia.org/wiki/Indigenous_Australian_music
china population	https://worldpopulationreview.com/countries/china-population/
	https://en.wikipedia.org/wiki/Chinese_people
	https://en.wikipedia.org/wiki/One-child_policy
	https://www.statista.com/topics/753/china/
india pakistan	https://en.wikipedia.org/wiki/Pakistan_Movement
	https://en.wikipedia.org/wiki/Kashmir_conflict
	https://worldview.stratfor.com/topic/india-pakistan-rivalry

examples of the web pages that are found irrelevant:

Query	**Irrelevant web pages**
australia	https://en.wikipedia.org/wiki/Cocos_Islands
	https://en.wikipedia.org/wiki/Demographics_of_Cocos_(Keeling)_Islands
	https://en.wikipedia.org/wiki/Applied_linguistics
china population	https://en.wikipedia.org/wiki/Taiwan,_China
	http://www.spacetoday.org/China/ChinaGlance.html
	https://www.thoughtco.com/peoples-republic-of-china-facts-history-195233
india pakistan	https://en.wikipedia.org/wiki/Victory_Day_of_Bangladesh https://en.wikipedia.org/wiki/Biharis_in_Bangladesh
	https://en.wikipedia.org/wiki/Gilgit-Baltistan

modified queries after applying Roccio algorithm to above queries:

Original query	**Query after roccio algorithm**
australia	australia map
china	china republic
ethiopia	ethiopia map
pakistan	pakistan kashmir
bangladesh area	bangladesh area code
pakistan terrorism	pakistan terrorism news
china population	china population policy
india pakistan	india Pakistan bbc
australai	australai coco
paksitan	paksitan east
chna	chna corona

germny	germny map
hindustan	hindustan time
britaan	britaan bhutan
franche	franche fries
gana	gana gana
indo	indo democrat
indo china	indo china wall
image argentina	imagining argentina wiki
iraq for sale	iraq sale people

The approach was used for the 20 queries mentioned above. The results obtained were not very promising and universal set of weights could not be arrived. Following were some of the findings:

1. Some queries fetched no documents – like "hindustan", "britaan" etc. which meant that the data we crawled did not contain neither the pages nor any reference to these words. In such cases, relevance model did not work
2. Some queries fetched all irrelevant documents – like "image argentina" fetched results for "imagining argentina movie". So the modified query was "image argentina satelite"
3. Some queries fetched great relevant results – like "china population" which fetched the china population 2020 and china child policy. In such cases, weights stabilized too early and so the modified query was not much relevant.
4. Performance issues as too much time was taken for some queries to get the stabilized weights, hence could not be performed on the fly
5. Each query needed its own unique set of weights

Due to the above issues, it was decided not to include the code for Rocchio algorithm for expanding the query, rather the results of the observations were used in the pseudo relevance feedback approaches to filter out the stop words, avoid spelling mistakes etc.

Pseudo Relevance Feedback:

1. **Association cluster**: The idea is that stems which co-occur frequently inside documents have a synonymity association
2. Metric cluster: The idea is that the stems which occur far apart in the document are less co-related to the terms that occur closer like in the same sentence.
3. Scalar cluster: The idea is that two stems are more co-related if they have similar neighbourhood.

Following is the list of 54 queries used for pseudo relevance feedback. (Each query run 3 times with 3 different cluster methods as mentioned above):

Query	**Relevant results fetched after expansion(pages)**	**Most relevant expanded query**	**Cluster method used**
china	10	china map republic	association, scalar
china population	10	china population policy child	association,
australia	10	australia coco demographic	association, scalar
bangladesh	9	bangladesh people arm	metric, association
ethiopia	2	ethiopia concur confluence	metric, association, scalar

india pakistan	8	india pakistan kashmir bangladesh war	metric, association
france	5	france wins wine map	metric, association, scalar
germany	5	germany squad language human	metric, association, scalar
france area	8	france area code people	association, scalar
germany flag	10	germany flag color red	association
china coronavirus	9	china coronavirus spread world	metric, association, scalar
pakistan terrorist	4	pakistan terrorist news kashmir	metric, association, scalar
qatar	9	qatar airway oil currency	metric, association,
canada weather	1	canada weather daily toronto canadian	metric, association, scalar
india	9	india map democracy	association
hindustan	0	hindustan article time	metric, association, scalar
gana	0	gana gana	metric, association
franche	0	franche firewatch	association, scalar

Based upon the above results, it was decided that the association cluster method is the best for implementation of the pseudo relevance feedback.

Some challenges:

1. Scalar clustering took too long to run so we can't perform them well
2. Metric clustering also was taking time in some cases.

Query 1: australia
local document set(ids):

491d54c4-885c-11ea-bc55-0242ac130003
491d571c-885c-11ea-bc55-0242ac130003
491d57e4-885c-11ea-bc55-0242ac130003
491d58a2-885c-11ea-bc55-0242ac130003
491d5ab4-885c-11ea-bc55-0242ac130003
491d5b90-885c-11ea-bc55-0242ac130003
491d5c58-885c-11ea-bc55-0242ac130003
491d5d16-885c-11ea-bc55-0242ac130003
491d5f00-885c-11ea-bc55-0242ac130003
491d5fd2-885c-11ea-bc55-0242ac130003
491d6090-885c-11ea-bc55-0242ac130003
491d614e-885c-11ea-bc55-0242ac130003
491d620c-885c-11ea-bc55-0242ac130003
491d62ca-885c-11ea-bc55-0242ac130003
491d6392-885c-11ea-bc55-0242ac130003
491d65cc-885c-11ea-bc55-0242ac130003
491d66a8-885c-11ea-bc55-0242ac130003
491d6766-885c-11ea-bc55-0242ac130003
491d682e-885c-11ea-bc55-0242ac130003
491d68ec-885c-11ea-bc55-0242ac130003
491d69b4-885c-11ea-bc55-0242ac130003

491d6a72-885c-11ea-bc55-0242ac130003
491d6b30-885c-11ea-bc55-0242ac130003
491d6d6a-885c-11ea-bc55-0242ac130003
491d6e32-885c-11ea-bc55-0242ac130003
491d6ef0-885c-11ea-bc55-0242ac130003
491d6fae-885c-11ea-bc55-0242ac130003
491d7076-885c-11ea-bc55-0242ac130003
491d71ac-885c-11ea-bc55-0242ac130003
491d7288-885c-11ea-bc55-0242ac130003

india_pakistan_local_doc.txt

local vocabulary set:

local_vocabulary_set1.txt

local stem set:

local_stem_set1.txt

Query 2: india Pakistan
local document set(ids):
867c15ba-885e-11ea-bc55-0242ac130003
867c193e-885e-11ea-bc55-0242ac130003
867c1b0a-885e-11ea-bc55-0242ac130003
867c1c72-885e-11ea-bc55-0242ac130003
867c1db2-885e-11ea-bc55-0242ac130003
867c1eca-885e-11ea-bc55-0242ac130003
867c205a-885e-11ea-bc55-0242ac130003
867c219a-885e-11ea-bc55-0242ac130003
867c229e-885e-11ea-bc55-0242ac130003
867c23fc-885e-11ea-bc55-0242ac130003
867c253c-885e-11ea-bc55-0242ac130003
867c264a-885e-11ea-bc55-0242ac130003
867c27d0-885e-11ea-bc55-0242ac130003
867c2906-885e-11ea-bc55-0242ac130003
867c2a1e-885e-11ea-bc55-0242ac130003
867c2b7c-885e-11ea-bc55-0242ac130003
867c2cc6-885e-11ea-bc55-0242ac130003
867c2e60-885e-11ea-bc55-0242ac130003
867c2faa-885e-11ea-bc55-0242ac130003
867c30fe-885e-11ea-bc55-0242ac130003
867c3252-885e-11ea-bc55-0242ac130003
867c3392-885e-11ea-bc55-0242ac130003
867c3518-885e-11ea-bc55-0242ac130003
867c36e4-885e-11ea-bc55-0242ac130003

867c38b0-885e-11ea-bc55-0242ac130003
867c39fa-885e-11ea-bc55-0242ac130003
867c3b3a-885e-11ea-bc55-0242ac130003
867c3c98-885e-11ea-bc55-0242ac130003
867c4350-885e-11ea-bc55-0242ac130003
867cac96-885e-11ea-bc55-0242ac130003

india_pakistan_local_doc.txt

local vocabulary set:

local_vocabulary_set2.txt

local stem set:

local_stem_set2.txt

Query 3: china population
local documents (ids) :
e5200812-8861-11ea-bc55-0242ac130003
e5200a10-8861-11ea-bc55-0242ac130003
e5200b0a-8861-11ea-bc55-0242ac130003
e520115e-8861-11ea-bc55-0242ac130003
e52014e2-8861-11ea-bc55-0242ac130003
e520160e-8861-11ea-bc55-0242ac130003
e52017d0-8861-11ea-bc55-0242ac130003
e5202130-8861-11ea-bc55-0242ac130003
e520241e-8861-11ea-bc55-0242ac130003
e520250e-8861-11ea-bc55-0242ac130003
e52025e0-8861-11ea-bc55-0242ac130003
e520269e-8861-11ea-bc55-0242ac130003
e520275c-8861-11ea-bc55-0242ac130003
e520281a-8861-11ea-bc55-0242ac130003
e52029b4-8861-11ea-bc55-0242ac130003
e5202a90-8861-11ea-bc55-0242ac130003
e5202b62-8861-11ea-bc55-0242ac130003
e5202c20-8861-11ea-bc55-0242ac130003
e5202e78-8861-11ea-bc55-0242ac130003
e5202f54-8861-11ea-bc55-0242ac130003
e520301c-8861-11ea-bc55-0242ac130003
e52030e4-8861-11ea-bc55-0242ac130003
e52031ac-8861-11ea-bc55-0242ac130003
e520326a-8861-11ea-bc55-0242ac130003
e5203328-8861-11ea-bc55-0242ac130003
e520362a-8861-11ea-bc55-0242ac130003
e5203710-8861-11ea-bc55-0242ac130003

e52037ce-8861-11ea-bc55-0242ac130003
e5203896-8861-11ea-bc55-0242ac130003
e520395e-8861-11ea-bc55-0242ac130003

china_population_local_doc.txt

local vocabulary set:

local_vocabulary_set3.txt

local stem set:

local_stem_set3.txt

from the table above, we can see that the relevant pages obtained are more when association cluster method is used and the time consumed for association cluster is less. As a result, we can say that association cluster is best suitable when performance is considered. It also takes care of spelling mistakes. The number of relevant pages obtained from table shows the correlations for the queries. It also shows the expanded query for each original query.

Collaboration with UI and relevance model:

1) From the API (URL) call we receive original query and the cluster method type to be used.
2) Based on these parameters, the relevant model provides top 50 documents to the query expansion python code. These results are then used as local documents to form local vocabulary , stem sets.
3) Based on the query parameter the cluster method out of association, metric and scalar is called to return expanded query to relevance model.
4) The relevance model then hits solar in order to received results for expanded query.
5) These results are then passed to front end and displayed.

Query selection for demo:

For the demo, the query “china population” was used since it has two words and can be enhanced with other words. It is tested for association cluster method.

7. Discussion

After completing this project, we have learned the many different aspects required to make a simple and working search engine. We were able to create a search engine that could take crawled webpages and return somewhat relevant results using the different techniques that we learned in class. The project gave us lots of practical knowledge of creating a search engine and also showed just that we have only really scratched the surface of Information Retrieval. When comparing the results that we worked so hard to get with large search engines such as Google and Bing, we realized that the work we did is small compared to what it takes to create and maintain popular and up-to-date search engines like Google or Bing.

8. Conclusion

In conclusion, for this project, we created a search engine — using the knowledge we gained throughout this class and some open source programs — that focuses on information related to Countries. The five of us splits the 5 different tasks required to build the search engine and collaborated to combine our individual parts into a final product that can provide relatively accurate information about Countries. While there were many challenges along the way, we were able to overcome them and create a good search engine. This project provided us with an opportunity to apply the knowledge we gained in this class in a practical way and boosted our understanding of the theoretical concepts we learned.