

Information Retrieval in Historical Domain

Monsoon 2020

Contributors

Aditi Verma (S20180010006 / aditi.v18@iiits.in)

Nethra Gunti (S20180010061 / nethra.g18@iiits.in)

Overview

The aim of this project is to consolidate a dataset using a web crawler and to make an information retrieval system for articles in the History domain, along with auxiliary features such as categorical filtering, key figures involved, and recommendations for related articles.

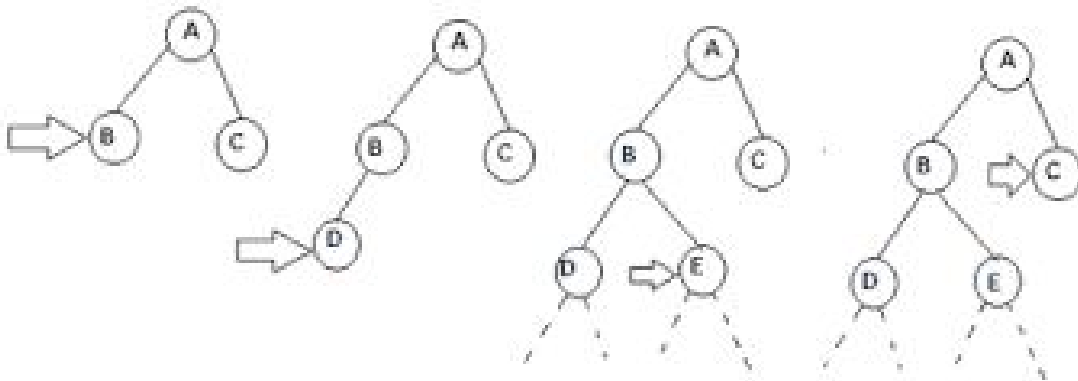
APIs & Libraries

This task has been done in **Python 3.8** and following are the libraries used:

1. Wikipedia API
2. NLTK
3. Spacy
4. Numpy
5. Sqlite
6. JSON (python package)

Data Collection

The first step to this project was to collect a dataset relevant to our needs. Given the domain of the problem, we decided to crawl Wikipedia for the same. Given that crawling using HTML pages for wikipedia would be very counterproductive, we decided to look for APIs. We came across a Python wrapper for the **official Wikipedia API**, which was a perfect fit for our purpose. Then, we created a web crawler which took a few seeded URLs (majority being from the Most Popular pages of the History WikiProject) and found out the categories it was part of. This crawler uses **Depth Limit Search (DLS)** algorithm to crawl all files in a particular category.



This process was executed sequentially, in part not to make requests too often. The other motivation behind it was Python's lack of true parallel multithreading. The information was stored in JSON format, in order to facilitate quick retrieval as well as to enable other features of the system.

The Dataset

After being done with the crawling, we decided to explore the dataset to get a cursory understanding of it. We found that we had crawled about **263000 files**. These files have been stored in JSON format. Each of these JSON files follows the given structure:

```

{
    "title": "title_of_article",
    "url": "url_to_article",
    "tags": "categories_of_article",
    "text": "full_text_of_article"
}

```

Despite having created a reasonably large dataset, we decided to create our IR system on a smaller subset of it, in order to make up for the lack of computational resources.

Approach

I. Process

Our goal was to create a basic IR system for the dataset, as well as to implement a few extra features. For this, we needed to build an inverted index for all of our files. Since the dataset was quite large, we decided to create indices for files in batches. We chose **batch_size=1000** and used multithreading for it.


Each of these indices was created by cleaning the data in each file, tokenizing it, removing stop words, and making a posting list for each unique token. We also implemented lemmatization which reduced the number of tokens significantly. However, seeing that most of the queries for our particular system would be based on nouns and other parts of speech would not have much impact on it, we decided against applying it to the final set of indices, in the interest of saving time and computational resources. The code snippet can still be found in the indexing file.

Then, we merged the index files created iteratively, into common indices, until we were left with a single file. The merging was also done in batches where the batch size was taken as $\log_2(N)$ where N is the number of index files remaining after each iteration.

Next, we created a relational database using **SQLite** with our dataset. We created 4 tables here:

DOCTABLE	DocID	DocTitle	WordCount	-
TAGSTABLE	TagID	TagTitle	DocID	-
TERMSTABLE	TermID	TermTitle	DocCount	-
POSITIONALINDEX	TermID	DocID	Positions	TermCount

This is the database that we will finally use for the IR system.



Next was the actual retrieval. For this, we used Tf-Idf and cosine similarity as the method of ranked retrieval. We initially reduced the set of documents to be considered for the retrieval using the concept of Boolean Retrieval. We then calculated the Tf-IDF scores and vectors for each document pair from the retrieved set. From these vectors, we calculated the Cosine Similarity and sorted the results accordingly.

II. Non-Trivial Features

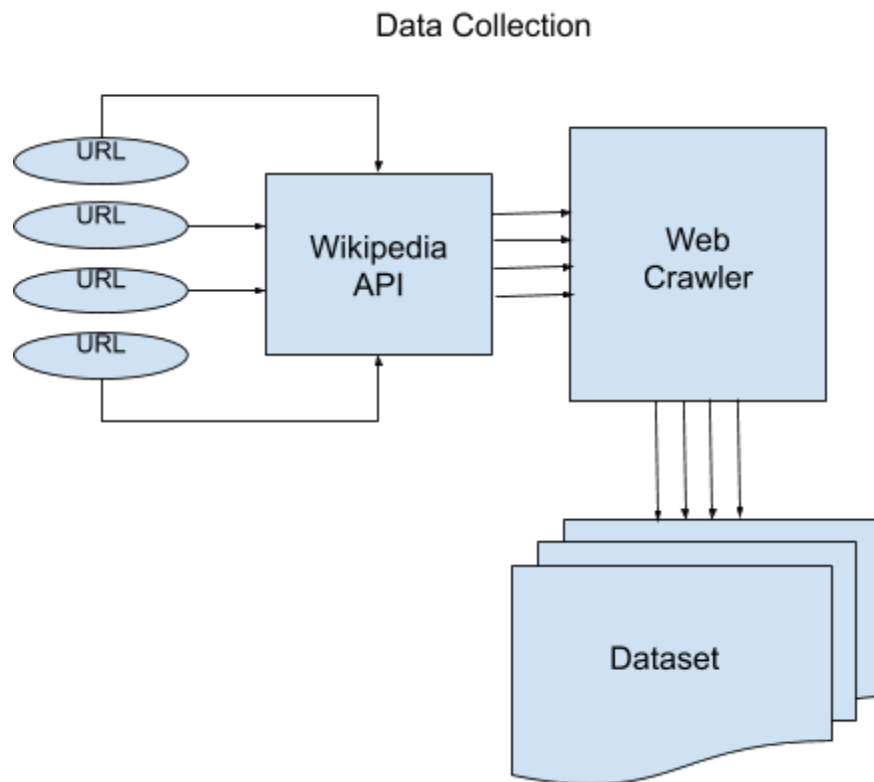
We implemented **spell checker** on the input query. We generated all the possible candidates at edit distances of 1 and 2 for every word in the query and disregarded those not in our dictionary.

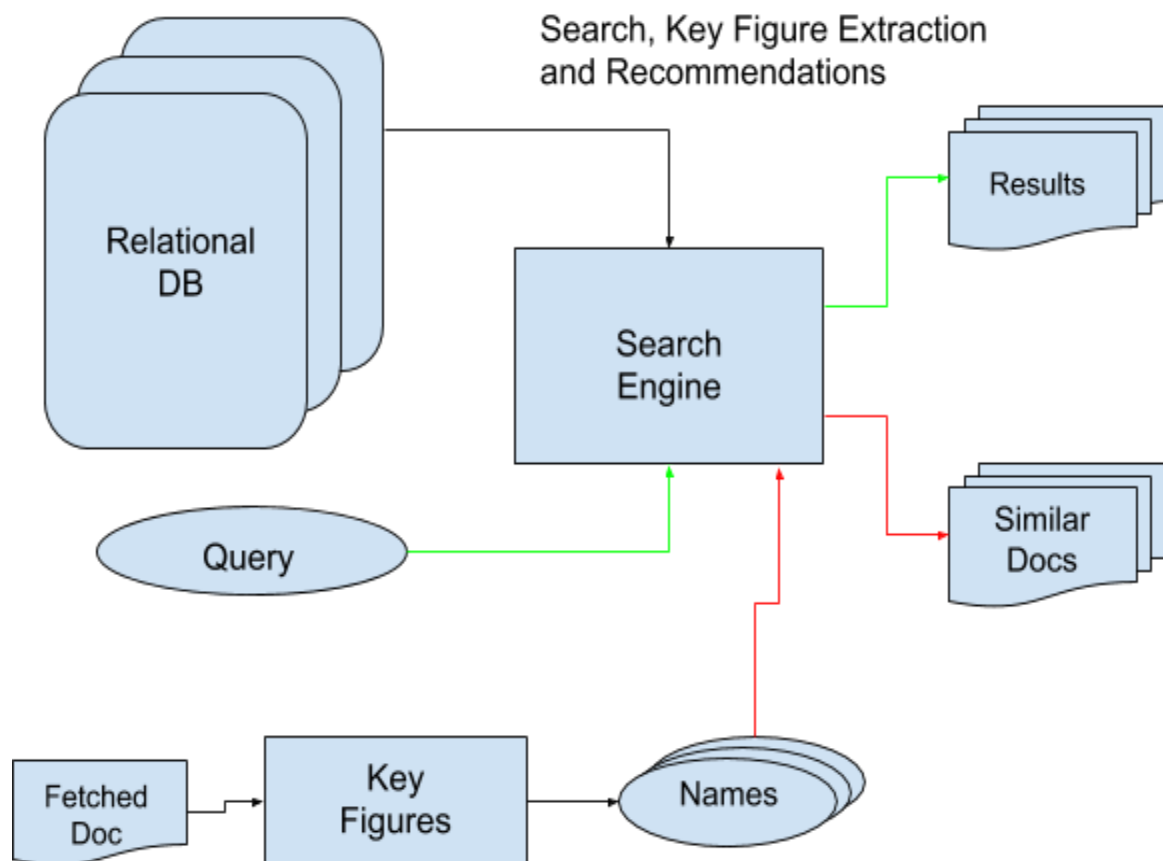
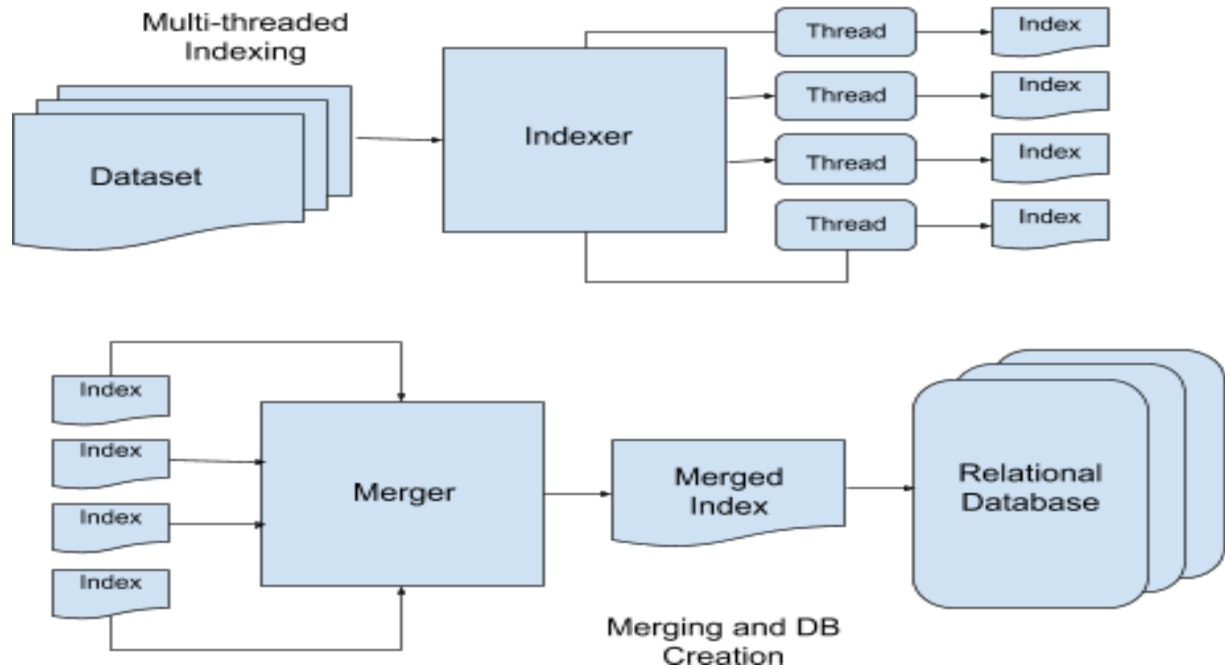
We also implemented **categorical filtering**. For this, the query is treated as a category, and all the files from that category are retrieved. The user can then choose any document to fetch more information. We used Wikipedia's tags as our categories.

We also implemented the feature to **extract important figures** (both people and organizations) involved/mentioned in a particular article. We used SpaCy's Named Entity Recognition (NER) module for this purpose. Then, we extracted the required information. This information is available when the user wants to fetch a particular document.

Finally, we implemented the feature of **recommending similar docs** for reading, based on the figures obtained in the previous step. Like the key figures feature, this is available when viewing a particular document.

III. Pipeline





Results

A few snapshots of the system:

1. Query

```
-----
Enter Query: world war
results retrieved in 10.472901105880737 secs
{3: 0.022039213549457214, 4: 0.019426958634929974, 7: 0.008174102814752145, 1: 0.007890950174421214, 10: 0.007495361169128025, 6: 0.005962612361000328}
3
DocID: 000003
Title: Baltic Sea campaigns (1939-45)
URL: https://en.wikipedia.org/wiki/Baltic_Sea_campaigns_(1939%E2%80%9345)
=====
4
DocID: 000004
Title: Battle of Barking Creek
URL: https://en.wikipedia.org/wiki/Battle_of_Barking_Creek
=====
7
DocID: 000007
Title: Battle of Czajánek's barracks
URL: https://en.wikipedia.org/wiki/Battle_of_Czaj%C3%A1nek%27s_barracks
=====
1
DocID: 000001
Title: 1936-1939 Arab revolt in Palestine
URL: https://en.wikipedia.org/wiki/1936%E2%80%931939_Arab_revolt_in_Palestine
=====
10
DocID: 000010
Title: Battle of Changsha (1939)
URL: https://en.wikipedia.org/wiki/Battle_of_Changsha_(1939)
=====
6
DocID: 000006
Title: Bloody Sunday (1939)
URL: https://en.wikipedia.org/wiki/Bloody_Sunday_(1939)
=====
-----
```

2. Tag


```
-----
Enter Tag: Conflicts in 1939
1
DocID: 000001
Title: 1936-1939 Arab revolt in Palestine
URL: https://en.wikipedia.org/wiki/1936%E2%80%931939_Arab_revolt_in_Palestine
=====
2
DocID: 000002
Title: Ariostazo
URL: https://en.wikipedia.org/wiki/Ariostazo
=====
3
DocID: 000003
Title: Baltic Sea campaigns (1939-45)
URL: https://en.wikipedia.org/wiki/Baltic_Sea_campaigns_(1939%E2%80%9345)
=====
4
DocID: 000004
Title: Battle of Barking Creek
URL: https://en.wikipedia.org/wiki/Battle_of_Barking_Creek
=====
5
DocID: 000005
Title: Battle of Petsamo (1939)
URL: https://en.wikipedia.org/wiki/Battle_of_Petsamo_(1939)
=====
6
DocID: 000006
Title: Bloody Sunday (1939)
URL: https://en.wikipedia.org/wiki/Bloody_Sunday_(1939)
=====
7
DocID: 000007
Title: Battle of Czajánek's barracks
```

3. Fetch (with Key Figures)

```

-----
Enter docid to retrieve: 000009
DocID: 000009 Title: Catalonia Offensive

URL: https://en.wikipedia.org/wiki/Catalonia_Offensive

Key Figures:

Catalan Christian -- Soviet -- Italian -- Republican -- Catalan -- L2 -- Nazis -- Republicans
-- Republic -- Italians -- Spanish -- Nationalists -- Nationalist -- Hernandez -- French --
European -- Garcia Valiño's -- Azaña -- Garcia Valiño -- Juan Negrín -- Muñoz Grandes's -- Th
omas -- Rojo -- Vicente Rojo -- Franco -- Paul -- Helen -- Jackson -- January Garcia Valiño --
-- Harper Perennial -- Muñoz Grandes -- Artesa -- Tarragona -- Ebro -- Negrín -- Sabadell -- P
enguin Books -- Segre -- Fidel Dávila -- Gambara -- Les Borges Blanques -- Harper Collins --
Terrassa -- Gabriel -- Saravia -- Hugh -- Juan Hernandez -- Preston -- Barcelona -- Perea --
Moscardo -- Solchaga -- Cervera -- Cuerpo Legionario Italiano -- Graham -- Juan Modesto -- Of
ensiva de Cataluña -- the Munich Agreement -- Harper & Row -- Badalona -- The Nationalist Arm
y -- Army of the Ebro -- Oxford University Press -- Cortes -- East Army -- Argelès, Gurs, Riv
esaltes and Vernet -- XV Republican Corps -- Solchaga's Army Corps of Navarra -- Eastern Regi
on Army Group -- The Catalan -- Moscardo -- the Ebro Gambara's -- Condor Legion -- Les Borges
Blanques -- Ebro Army -- Army of Urgel -- the 26th Republican Division -- Peñarroya -- the R
epublican army -- Aragon -- Army Group -- Maestrazgo Corps -- the Republican Army -- ISBN --
Moroccan Corps -- Beevor -- Army -- the Moroccan Corps -- Aragon Army Corps -- the Free Frenc
h Forces -- Aragon Army -- the International Brigades -- Princeton -- Navarreses -- GERO -- Y
agüe -- XV Republican corps -- Princeton University Press -- The Battle for Spain --

Doc Excerpt:

The Catalonia Offensive (Catalan: Ofensiva de Catalunya, Spanish: Ofensiva de Cataluña) was
part of the Spanish Civil War. The Nationalist Army started the offensive on 23 December 193
8 and rapidly conquered Republican-held Catalonia with Barcelona (the Republic's capital city
from October 1937). Barcelona was captured on 26 January 1939. The Republican government he
aded for the French border. Thousands of people fleeing the Nationalists also crossed the fro
ntier in the following month, to be placed in internment camps. Franco closed the border with
France by 10 February 1939.

```

4. Similar Docs

Similar Docs:

{3: 0.2754665938334733, 5: 0.21502082708070142, 10: 0.020691911970261986}

3

DocID: 000003

Title: Baltic Sea campaigns (1939-45)

URL: [https://en.wikipedia.org/wiki/Baltic_Sea_campaigns_\(1939%E2%80%9345\)](https://en.wikipedia.org/wiki/Baltic_Sea_campaigns_(1939%E2%80%9345))

=====

5

DocID: 000005

Title: Battle of Petsamo (1939)

URL: [https://en.wikipedia.org/wiki/Battle_of_Petsamo_\(1939\)](https://en.wikipedia.org/wiki/Battle_of_Petsamo_(1939))

=====

10

DocID: 000010

Title: Battle of Changsha (1939)

URL: [https://en.wikipedia.org/wiki/Battle_of_Changsha_\(1939\)](https://en.wikipedia.org/wiki/Battle_of_Changsha_(1939))

=====

What do you want to search today?

1. Query
2. Tag
3. Fetch
4. Exit