Mini Project

Building a Search Engine

Tutorial - 1

Information Retrieval and Extraction

Project Task

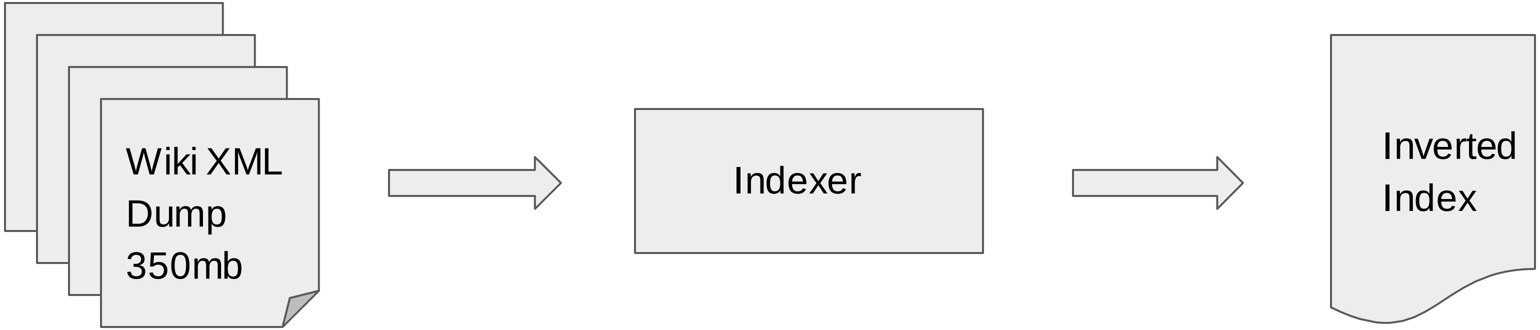
- Data: Wikipedia English Dump
    - Ire-wiki-search-sample.tar.gz (~350 mb for Phase I)
    - enwiki-latest-pages-articles-multistream.xml.bz2 (for Phase II)

- Index size (less than ¼ of data size)
- Support for field queries
- External tools and libraries like Lucene, WikiXMLj, elasticsearch, redis, etc not allowed.

Mini project

- Phase I
    - Inverted index creation on 350 mb wiki dump
    - Basic query implementation.
    - Evaluation for phase-1 will involve only indexing.
- Phase II
    - Inverted index creation on whole wiki dump
    - Implement Ranking mechanism
    - End to End search system

Phase I



Steps involved in Indexing

1. Parsing
2. Tokenization
3. Case Folding
4. Stop Words Removal
5. Stemming
6. Inverted Index Creation

Parsing

- Whole corpus (~ 100 GB) in single XML file
- Phase I
  - XML dump: ~350 MB
  - index size: ~90 MB
  - Index time: within 150 seconds

- Tool - SAX parser / DOM parser (ElementTree)
- WikiXMLj not allowed

Tokenization & Case folding

- Break sentences into individual words called tokens
- Change case to lower case

- Food for thought
  - State-of-the-art V/s state of the art
  - 12-04-1998
  - O’Neill - neill, oneill, o’neill, o’ neill, o neill

Stop Words Removal

- Highly frequent(common) words are of little value
- a, an, the, and, be, by, for, from, …

- Issues (Food for thought)
  - Let it be, To be or not to be
  - Flights from Mumbai

Stop Words Removal

The time of the Elves… is over. Do we leave Middle-Earth to its fate? Do we let them stand alone?	time Elves over leave Middle Earth fate stand alone

Stemming

- Identify root or base word

  is, am, are - be

  operate, operation, operates, operative - oper

  man, men, manliness - man

- Use from the following libraries : pystemmer, nltk (PorterStemmer, SnowballStemmer, WordNetLemmatizer), spacy.

- **Choice of library can heavily impact the index creation time.**

Inverted Index / Posting List

But I am the real Strider, fortunately.I am Aragorn son of Arathorn; and if by life or death I can save you, I will, I am real.

remove stop words

real strider fortunately aragorn son arathorn life death save real

do stemming

Real strider fortun aragorn son arathorn life death save real

Posting List creation

Document 1

real	2
strider	1
fortun	1
aragorn	1
son	1
arathorn	1
like	1
death	1
save	1

Inverted Index / Posting List

Many that live deserve death. And some that die deserve life. Do not be too eager to deal out death in judgement.

remove stop words

live deserve death die deserve life eager deal death judgement

do stemming

live deserve death die deserve life eager deal death judgement

Posting List creation

Document 2

live	2
deserve	2
death	2
die	1
life	1
eager	1
deal	1
judgement	1

Inverted Index

real	2
strider	1
fortun	1
aragorn	1
son	1
arathorn	1
like	1
death	1
save	1
live	2
deserv	2
death	2
die	1
life	1
eager	1
deal	1
judgement	1

Document 1

Document 2

Sorted Index

aragon:d1(1)
arathorn:d1(1)
deal:d2(1)
death:d2(2), d1(1)
deserv:d2(2)
die:d2(1)
eager:d2(1)
fortun:d1(1)
judgement:d2(1)
life:d1(1), d2(1)
live:d2(2)
real:d1(2)
save:d1(1)
son:d1(1)
strider:d1(1)

Handling Multiple Fields (Field Queries)

Wikipedia Fields:

1. Title
2. Body Text
3. Infobox
4. Categories
5. External Links (outlinks)
6. References

Storing Field types in Index

- Plain query - Sachin Tendulkar Sports
- Field query - t:Sachin b:Tendulkar c:Sports

- Choose a suitable format for storing field type in index file to support field queries.
- Store type along with frequency and docid

Storing field types in Index

Approach 1:

sachin:d1-t1c2b7|d5-t1
tendulkar:d1-t1b1|d6-c1b1

Approach 2:

sachin-t:d1-1|d5-1
sachin-c:d1-2
sachin-b:d1-7
tendulkar-t:d1-1
tendulkar-c:d6-1

Points to note

- Design a scalable index module.
- You can reduce index size by using index compression methods
  - Trade-off between search time efficiency and index size
- Search : Try to implement basic search, reading and parsing the index, parsing the query, producing basic results (ranking won't be evaluated in Phase1)

- Think of secondary index if you need to build (mostly in Phase II)

- Programming Language - C++/Python/Java

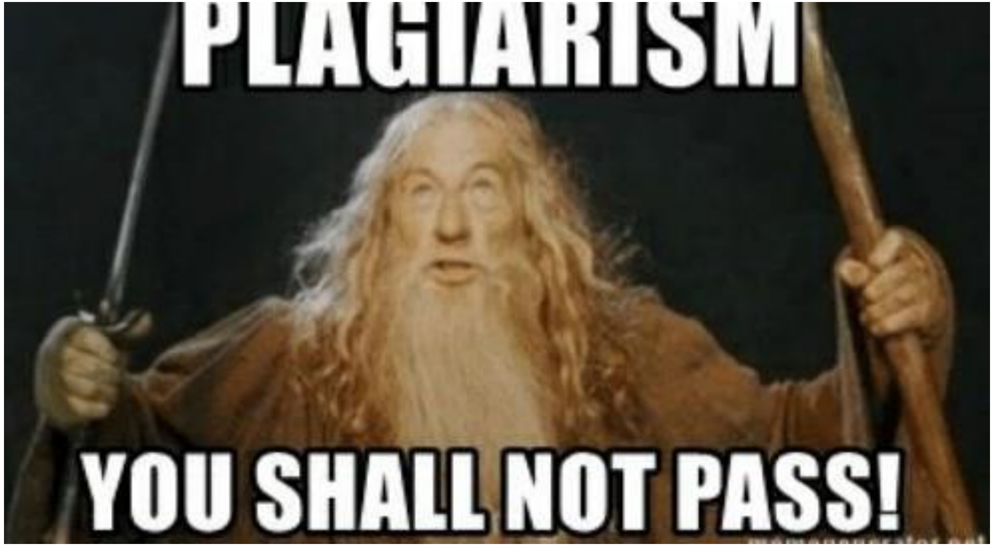
Bonus Task: Handling code mixed queries

- Dump will consist of articles in 2 languages.
- Queries will also be code mixed.
- You will have to return the results where the article should contain the terms from both language.

Example: - Sachin tendulkar करियर

Plagiarism

- **NO plagiarism will be tolerated**. Copying of code, using someone else's index or any sort of malpractice would lead to a **0** in the mini project at the least.



References

Tutorial video video `link

Christopher Manning, Information Retrieval

http://nlp.stanford.edu/IR-book/html/htmledition/irbook.html
Grossman, Frieder- Information Retrieval (Algorithms and Heuristics) -

Chapter 2, Chapter 5

Videos

https://class.coursera.org/nlp/lecture/178

https://class.coursera.org/nlp/lecture/179

https://class.coursera.org/nlp/lecture/180