BIRLA INSTITUTE OF TECHNOLOGY & SCIENCE, PILANI
HYDERABAD CAMPUS

Information Retrieval(CS F 469)

Assignment 1
First Semester 2015-2016

Total Marks:25
Due Date:28/9/2015

In this assignment you will design and implement your own Text based information retrieval system. The assignment has two phases. In Phase I, you will build the indexing component, which will take a large collection of text and produce a searchable, persistent data structure. In Phase II, you will add the searching component, according to Boolean Retrieval or Vector Space Model or Probabilistic Model.

The project may be done individually, or in a team of four members. All the team members are expected to contribute to all aspects of the assignment: design, implementation, documentation, and testing.

Phase I will have two major components: the tokenization and the normalization. Note that in order to fully implement the vector space model or Probabilistic Model you may need to retain several pieces of information about the documents in the dictionary / document location list / inverted index data structures such as the total number of documents, the maximum term frequency for each document, the length of the weight vector for each document, etc. Try to save values that you anticipate are needed to calculate.

Phase I resources

You may find the following resources useful in Phase I:

Corpus: You may pick up corpus from any online resource. The following are few links to corpus

1. http://trec.nist.gov/data/docs_eng.html
2. https://catalog.ldc.upenn.edu/LDC2015MDP
3. https://snap.stanford.edu/data/wiki-meta.html
4. Stanford Large Network Dataset Collection (SNAP)
5. Social Computing Data Repository
6. Twitter posts
7. E-Commerce
8. online reviews
9. “Twitter Census”

Tokenization: Tokenize the terms in the document.

For this you can use any standard tokenizer.

- Python’s NLTK package provides tools for tokenizing and normalizing the terms.
- http://nlp.stanford.edu/software/tokenizer.shtml
- (tm package of R also has this feature).

Stemmer for normalization: Martin Porter’s web page http://tartarus.org/~martin/PorterStemmer/ contains implementations of the Porter Stemmer in a number of languages.

Phase II

For Phase II, you will write a program which will accept queries from the user and search for documents using the data structures produced in Phase I. Implement the search using Boolean or Vector Space or Probabilistic model to rank the documents, carefully choosing the weighting function.
Your search interface must allow the user to:
1. Input a search query.
2. View a list of search results. Each result should display an internal document ID (sequence number), a title or URL depending on your dataset used.

Deliverables

Design document: Describe your application’s architecture and major data structures used at various stages.
Code: Your code needs to be well documented with comments
Readme file : You also need to include a README.txt file that describes how to compile your program and run it on various datasets and source of your dataset along with the precision and recall for various test cases that were used to test the application.

Submit your solution as a zip file to **bphc.ir@gmail.com**

You are also expected to demo your program including an explanation of your evaluation results to us as per the schedule which will be made available.

Implementation: You may code your project in any programming language of you choice (such as C#, Java or Python).

Scheme of evaluation:

Sl No.	Task	Marks
1.	Tokenizing and normalizing	3
2.	Building dictionary efficiently	5
3.	Building index	5
4.	Retrieving the data accurately	7
5.	Viva	5
	Total	**25**