BITS - Pilani

S1-21_SSZG537 IR Assignment

Deadline: 13th November, 2021

This assignment is aimed at designing and developing one's own text based information retrieval system.The assignment aims at building the searching module according to Boolean, Vector Space model of Information Retrieval.

Programming languages:
You may implement your assignment in any language you choose. STLs and built-in packages may only be used for text preprocessing and text normalization (C++'s Boost Library, Python's NLTK Package etc.). You are expected to code the core functionality of the model that you choose (TF-IDF in case of Vector Space model etc.)

The following are two tasks descriptions and you can pick any one of them and implement:

1. Domain Specific Information Retrieval System
   The task is to build a search engine which will cater to the needs of a particular domain. You have to feed your IR model with documents containing information about the chosen domain. It will then process the data and build indexes. Once this is done, the user will give a query as an input. You are supposed to return top 10 relevant documents as the output. Your results should be explainable. The design document should clearly explain the working of the model along with detailed explanation of any formulas that you might have used. For Eg:
   - You can build a search engine for searching song lyrics. Dataset: https://labrosa.ee.columbia.edu/millionsong/musixmatch
   - Other domains on which you can find datasets easily are:
     - Product review. https://www.kaggle.com/datafiniti/consumer-reviews-of-amazon-products
     - Healthcare https://www.kaggle.com/xhlulu/medal-emnlp
     - Finance
     - Automobiles

   You may also choose your own domain and crawl data from the web using the crawler described below in the additional resources. You are free to use your own web crawler.

2. Profession Specific Information Retrieval System

   The task is to build a search engine which will cater to the needs of a particular profession. It will then process the data and build indexes. Once this is done, the user will give a query as an input. You are supposed to return top 10 relevant documents as the output. Your results should be explainable. The design document should clearly explain the working of the model along with detailed explanation of any formulas that you might have used. For Eg:

   - Westlaw is a search engine which caters to the needs of lawyers. You may also choose your own domain and crawl data from the web using the crawler described below in the additional resources. You are free to use your own web crawler.
   - Management emails dataset
   - Blogger's corpus
   - Hansard dataset of Canadian parliamentary transaction

Additional Resources:

1. Stemming
   a. Martin Porter's 'Porter Stemmer' can be used for this purpose. Implementation in multiple languages can be found in the above link.
2. Tokenization:
   a. For this step you can use any standard tokenizer or inbuilt package. Following are a few sources:
      i. Python's NLTK package.
      ii. Stanford Tokenizer.
      iii. TM package of R.
3. Datasets:
   - Some data sets can be found at: https://snap.stanford.edu/data/index.html
   - Teams are allowed to use their own corpus
     Just make sure that you are not using a corpus of very small size
4. Incase of any queries kindly mail I/C of the course at ayandas@hyderabad.bits-pilani.ac.in

Deliverables:

The final submission must contain the following documents:

1. **Design Document** – This document should contain the description of the application's architecture along with the major data structures used in the project. Precision and Recall,

if possible, should also be calculated. Running for all the preprocessing should be mentioned. Also mention the running time for search or retrieval.

2. **Code** – The code should be well commented.
3. **Documentation** – All the classes, functions and modules of the code must be documented.
   Software that automatically generate such documents can be used – pydoc for Python, Eclipse for Java etc.
4. **README** – The README file should describe the procedure to compile and run your code for various datasets.

Submission Guidelines:

All the deliverables must be zipped and submitted in Taxilla latest by **deadline 30 th October 2021**.

Evaluation Criteria for Task :

S.No.	Task	Marks
1.	Tokenization and Normalization	2
2.	Efficient usage of Data Structures with justification	2
3.	Index Construction	2
4.	Querying the index and data retrieval	2
5.	Novelty / Out-of-the-box thinking (Anything that is not covered in the lectures.)	2
	Total	**10**

It should be noted that all the assignments would be run through a plagiarism detector and any form of plagiarism will not be tolerated and shall be brought to the notice of WILP division.