BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE, PILANI
HYDERABAD CAMPUS
CS F469 - Information Retrieval

Assignment – 2
First Semester 2016-17

TOTAL MARKS: 40
DUE DATE: 20/11/2016

This assignment contains two tasks. You can choose to complete any ONE task.

TASK – 1: CROSS LANGUAGE INFORMATION RETRIEVAL

This task is aimed at designing and developing a Cross Language Search Engine that can process queries in more than one language. This task is divided into two phases. Phase 1 focuses on training the translation models using a parallel corpus to perform the language translation and Phase 2 deals with querying the corpus using the developed search engine. You can either choose to create a search engine from scratch or extend the search engine developed in Assignment 1.

This assignment should be done with the same groups as that of Assignment 1.

Tips and Tricks:

- Phase-1 includes three major models – IBM Model 1, IBM Model 2 and IBM Model 3
- A lot of probabilities need to be computed and bookkeeping is extensive. Ensure that efficient and well-defined data structures are used for easy access while translating text.
- You can choose to either translate the corpus into the foreign language or opt for converting the query from the foreign language to the source language. Study the differences between both of them and choose the one that suits your need the most.

Corpus:

Any parallel corpus can be used for the assignment. Also, a corpus of your choice can be used for developing the search engine. The following is a list of a few Parallel Corpuses. The list of corpuses for the search engine can be obtained from Assignment 1.

1. Europarl Parallel Corpus
2. OPUS – The Open Parallel Corpus
3. IITB – Agriculture Domain Parallel Corpus

Programming Languages:

The assignment can be implemented in any programming language of your choice.
You are expected to code the core functionality of the model that you choose (TF-IDF in case of Vector Space model, EM algorithm etc.)

Phase-1 Information:

Training and obtaining the probabilities of translation and alignments:
The training module should take the parallel corpus as input and should output the probabilities of translation and alignments. This information is then used to translate either the query into the source language or the corpus into the foreign language.
You should be able to output the results of this phase upon request. Failure to do so might lead to negative marks.

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.
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 to **bphc.ir@gmail.com** latest by **November 20, 2016.**
You are expected to demo your application and present your results as per the schedule that will be made available.

Evaluation Criteria:

S.No.	***Task***	***Marks***
1	EM Algorithm	15
2	IBM Model 1 / IBM Model 2 / IBM Model 3	15
3	Viva	5
4	Novelty / Out-of-the-box thinking (Anything that is not covered in the lectures.)	5
	TOTAL	40

TASK – 2: SVD vs CUR DECOMPOSITION

This task is aimed at implementing both the SVD and the CUR Matrix Decomposition Algorithms and comparing the efficiency of both these approaches (In terms of space, time etc.). You are required to show the instances where SVD is better, and instances where CUR is better. The pseudocode for the algorithms can be found in the slides.

This assignment should be done with the same groups as that of Assignment 1.

Tips and Tricks:

- Since CUR Decomposition involves use of random rows and columns for decomposing the matrix (Random Number Generator), ensure that you use a seed so that the output remains consistent over multiple runs.

Corpus:

Any Rating or Review dataset can be used for this assignment.

1. Amazon Book Reviews
2. Book Crossing
3. LibRec

Programming Languages:

The assignment can be implemented in any programming language of your choice.
You are expected to code the core functionality of the algorithm (SVD, EM etc.)

Additional Information:

You are expected to code the core of both the SVD and CUR Decomposition algorithms.
You should be able to output the results of this phase (All the three matrices that the original matrix has been decomposed to) upon request. Failure to do so might lead to negative marks.

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.
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 to **bphc.ir@gmail.com** latest by **November 20, 2016.**

You are expected to demo your application and present your results as per the schedule that will be made available.

Evaluation Criteria:

S.No.	***Task***	***Marks***
1	Efficient Data Structures	5
2	SVD Algorithm	10
3	CUR Decomposition Algorithm	15
4	Comparison between SVD and CUR	5
5	Viva	5
	TOTAL	40