Northeastern University

College of Computer and Information Science
Information Retrieval CS6200 – Spring 2017
Prof. Nada Naji najin@ccs.neu.edu

Overview: You have been introduced to the core information retrieval concepts and processes throughout the course of this semester. In this project, you will get to put these into practice by building and using your very own search engines!

Goals: Design and build your information retrieval systems, evaluate and compare their performance levels in terms of retrieval effectiveness

Dataset: CACM test-collection which is comprised of the following:

1- Textual corpus: cacm.tar.gz (3204 raw documents – except for Task 3 part B: use stemmed version cacm_stem.txt)
2- Queries (64 unprocessed cacm.query – except for Task 3 part B: use cacm_stem.query)
3- Relevance judgments (cacm.rel)
4- Stoplist: common_words.txt

Team: Teams of 3 members are to be formed. Send an email to Prof. Naji, and the TAs Maryam, and Abhinav by Thursday April 6th, 2017 (11:59pm) declaring your project team in the subject field ("we're a team!" or "IR project team" or something like that). In the message body, list ALL member names (including yourself!). Once formed, teams cannot be altered.

Milestones:
April 4: Release of the online description for the project
April 6: Team declaration due date
April 18 by 11:59pm: Project & report (implementation & documentation) submission due date

Assessment: The project is to be graded out of 100 points and then scaled to 20% of your overall grade (see syllabus for course grade details)

Implementation: 75 points (detailed point breakdown in project task descriptions)
Documentation: 25 points. Project submissions lacking documentation (report) will NOT be accepted and hence will NOT be graded at all.
Extra credit: 20 points: All or nothing. Awarded credit applies to project & homeworks.

Academic honesty: If you get help from others, you must write their names down on your submission and explain how they helped you. If you use external resources you must mention them explicitly. You may use third party libraries but you need to cite them, too.

Project Description:

Implementation – Phase 1 :: Indexing and Retrieval

Task 1 (5 points): Build your own search engines:

A- From scratch! (You may re-use your indexer and searchers from HWs)

B- Using Lucene: an open source library that provides indexing and searching functionalities (you may re-use your code from HW)

Task 1 Output: *Three baseline runs* (from the three search engines described above). Namely: Your search engine with *BM25* as a retrieval model, your search engine with *tf-idf* as a retrieval model, and Lucene's default retrieval model. The top 100 (at most) retrieved ranked lists (one list per run/search engine) are to be reported.

Task 2 (25 points): Pick *two*[1] of the three runs above and perform query expansion using *one* of the following approaches:
You may choose any of the suggested approaches below, feel free to adopt one that isn't listed but make sure to cite related literature and resources. Justify your design decisions, technical choices, and parameter setting and back them up with demonstrated evidence from literature and/or experiments whenever applicable.

A- Inflectional and/or derivational variants

B- Pseudo relevance feedback

C- Thesauri, ontologies, etc.

Task 2 Output: *Two* runs using *two* of the base search engines incorporating your query expansion technique of choice.

Task 3 (25 points): Use the *two* base search engine setups (retrieval model) that you picked for Task 2 and perform the following:

A- Stopping (using common_words.txt) with no stemming.

B- Index the stemmed version of the corpus (cacm_stem.txt). Retrieve results for the queries in cacm_stem.query. Perform a query-by-query analysis (see documentation) for three queries that you find interesting.

Task 3 Output: *Four runs:* using *two* base search engines (same as chosen for Task 2) X (run with stopping, run with the stemmed corpus and stemmed query subset).

[1] In practice, it is advised to perform Tasks 2 and 3 using all three base search engines from Task 1 to

Implementation – Phase 2 :: Evaluation

(20 points)

By now, you should have *seven* distinct runs with results for all 64 queries. Namely, 3 baseline runs, 2 query expansion runs, and 2 stopping runs (we're not counting the stemming runs here). It is now time to assess the performance of your search engines (runs) in terms of retrieval effectiveness.
Implement and perform the following (do NOT use TREC-Eval):

1- MAP
2- MRR
3- P@K, K = 5 and 20
4- Precision & Recall (provide full tables for all queries and all runs)

Note: Queries that don't have any entries in the relevance judgment should be excluded from evaluation.

Documentation:

A- ReadMe.txt: which explains in detail how to setup, compile, and run your project.
B- Report NOT to exceed 2200 words[2] in PDF format, named as follows:

firstNameInitialLastName1_firstNameInitialLastName2[_firstNameInitialLastName3].pdf

Please follow this structure:

i. First page: Project members' names, course name and semester, instructor name.
ii. Introduction: Short description of your project, detailed description of each member's contribution to the project and its documentation
iii. Literature and resources: overview of the techniques used (chosen query expansion approach) scholarly work and research articles to back up your technique and algorithm choices, resources, third party tools that you used and referred to in your project.
iv. Implementation and discussion: More thorough description of your project and design decisions. Include query-by-query analysis in this section.
v. Results: tables reporting all results obtained for all runs and queries for all required metrics. For query level results, please provide spreadsheets, too.
vi. Conclusions and outlook: state your findings, observations and analyses of the results. Which system do you think works best? Why? For "outlook": write a few sentences stating what you would envision doing to improve your project, what other features would choose to incorporate.
vii. Bibliography: citations and links to resources

[2] This document's word count is about 1000 words

Extra credit (20 points):

This part is optional, and is all or nothing (all 20 points or none). Awarded extra credit points apply to project and homeworks.

1- Perform statistical tests (e.g. *t-test*) to report whether the differences in performance are statistically significant.

2- Implement a *snippet generation* technique and *query term highlighting* within results in one of the non-Lucene runs. It is up to you to figure out which techniques to use, however, you are required to back up your choices with the algorithm(s)/technique(s) details and cite the respective literature.