UCI Information Retrieval

Project 3 – Search Engine

Due dates: 2/21, 2/28 and 3/13

This assignment is to be done in groups of 1, 2 or 3, preferably the same groups that were in place for Project 2. Although this is presented as one single project that will take until the end of the quarter to complete, internally it is organized in 3 separate milestones, each with a specific deadline, deliverables and score. In doing milestones #1 and #2, make sure to look at the evaluation criteria not just of those milestones but also of milestone #3 –part of the milestones' evaluation will be delayed until the final meeting with the Reader.

You can use code that you or any classmate wrote for the *previous* projects. You cannot use code written for *this* project by non-group-member classmates. Use code found over the Internet at your own peril -- it may not do exactly what the assignment requests. If you do end up using code you find on the Internet, you must disclose the origin of the code. **As stated in the course policy document, concealing the origin of a piece of code is plagiarism**.

Use the Message Board for general questions whose answers can benefit you and everyone.

Goal: Implement a complete search engine for the ICS domain. At the end of this project, you should have a [web] interface that provides the user with a text box to enter queries and returns relevant results.

Milestones Overview

	Deadline	**Goal**	**Deliverables**	**Contribution for score**
#1	2/21	Produce an index for the ICS web pages	Short report + Demonstration	25%
#2	2/28	Add search interface and page retrieval	Short report (no demo)	15%
#3	3/13	Search performance improvement	Code + report+ Demonstration	60%

Milestone #1

Goal: Build an index

Using the pages that you stored by crawling the ics.uci.edu domain in the previous project, construct an index that maps words to documents (pages). As pay load you should add at least the TF-IDF and the position of the words in each document.

Note that this will be the first draft of your index. You may need to redesign it as you improve your search engine.

NOTE: you are being asked to build your own index. While the use of the powerful Lucene library would be recommended for a real search engine, in using it you will miss most of the valuable lessons that this class can teach you. So: no Lucene!

Deliverables:

- Submit a report (pdf) to EEE with the following content: a table with assorted numbers pertaining to your index. It should have, at least the number of documents, the number of [unique] words, and the total size (in KB) of your index on disk.
- A live demonstration of your indexer in action for a small (100) subset of your documents.

Evaluation criteria:

- Did your report show up on time?
- Are the reported numbers plausible?
- Did the indexer work and produce an index file?
- Did you answer questions about your indexer satisfactorily?

Milestone #2

Goal: Complete your search engine

Develop an interface to search your index that retrieves documents according to a relevance score. You are the one in charge of designing your relevance score, although for this milestone this is not very important.

Your search engine should work like what's expected of search engines: the user types a query and a list of relevant hits is shown, showing at least the URL. Bonus points will be given for also showing relevant text snippets in the hits.

The UI doesn't need to be a Web UI at this point, although you may want to do that at some point, because it's nicer (and you can show it off to others on the Internet).

At least the following queries should be used to test your search engine:

1 - mondego
2- machine learning
3- software engineering
4 - security
5 - student affairs
6 - graduate courses
7- Crista Lopes
8 - REST
9 - computer games
10 - information retrieval
(Feel free to use more)

Deliverables:

- Submit a report (pdf) to EEE with the following content:
    - the top 5 URLs for each of the 10 queries above
    - a picture of your search interface

Evaluation criteria:

- Did your report show up on time?
- Are the reported numbers plausible?
- Do you have a user interface?

Milestone #3

Goal: Improve the ranking performance of your search engine

The Oracle (i.e. ground truth) for the ranking of the results is Google. Write a program that sends the 10 queries above to Google appended with "site:ics.uci.edu" and returns Google's search results. If your ICS crawler of project 2 stored only HTML pages, you should eliminate non-HTML documents from the Google results. For each query, the ordering of the Google results should be considered the "right order" (i.e. the Oracle)

1) Compute NDCG@5 for your search engine before any performance improvements.
2) Improve your search engine's NDCG@5.

Deliverables:

- Submit a zip file to EEE containing all the tools/scripts you wrote for your search engine as well as a report (pdf) with the following content:
  1. Your NDCG@5 for 1) and 2)
  2. Explain what you did in order to improve NDGC@5.
  3. Include any other information that may be relevant for this milestone
- A live demonstration of your search engine

Evaluation criteria:

- Is your second NDCG@5 good enough and/or were you able to improve it wrt the first?
- How general are the heuristics that you employed to improve the performance?
- Does your search engine work as expected of search engines?
- How complete is the UI? (e.g. links to the actual pages, snippets, etc.)
- Do you demonstrate in-depth knowledge of how your search engine works? Are you able to answer detailed questions pertaining to any aspect of its implementation?