Information Retrieval and Web Search

Practical session n°4: Evaluation

Create a directory named *practice4*. In this directory, create a new file named *practice4_report.txt*.
For each exercise, copy-paste in this file some outputs of your program, showing that you complete the exercise and it works correctly. Add some explanations.

You will build several runs (maximum = 50 runs / submission) for INEX Ad-Hoc Relevant in Context task:

- Each run is stored in a textual file, containing the results returned by your system for the 7 queries bellow (cf. exercise 1). Thus, the size of each file is limited to 10,500 lines.
- the filename of your runs should be named using the following template:
  TeamName_Run-Id_WeigthingFunction_Granularity_Stop_Stem_Parameters.txt

With:

- Run-Id = unique identifier
- WeightingFunction = ltn, ltc, bm25, etc.
- Granularity ϵ { articles, elements, passages }, i.e. the document unit. If "elements", you can add the list of XML tags ϵ { article, header, title, bdy, sec, p, etc.} you consider as document units.
- Stop ϵ { nostop, stopN } with N = size of the stop-list.
- Stem ϵ { nostem, porter, lovins, paice, etc. }
- Parameters: list all the other interesting parameters used, together with their value.

Example: VictorAlbertJulesIsaac_12_bm25_elements_sec_p_stop344_nostem_k1.2_b0.75.txt

At the end of your work:

- copy-paste the source code of your program(s) in the directory *practice4*.
- copy all your runs (maximum = 50 runs) in the directory *practice4* (without any sub-directory).
- compress the directory *practice4* in a file named *practice4_YourTeamName.zip* (or *.tar*, *.gz*, *.rar*, etc.) (e.g.: *practice4_VictorAlbertJulesIsaac.zip*).
- upload this compressed file (one file / team) on the website of the course before 11.59pm November 14th,.

Exercise 1: SMART *ltn* run

Using your IR System, retrieve a ranked list of 1,500 articles from the collection of the Practical session n°3, for each of these 7 queries:

Query id	Query
2009011	olive oil health benefit
2009036	notting hill film actors
2009067	probabilistic models in information retrieval
2009073	web link network analysis
2009074	web ranking scoring algorithm
2009078	supervised machine learning algorithm
2009085	operating system +mutual +exclusion

With these results, build a run for the RIC task of INEX. Check carefully the syntax (cf. lecture n°4, cf. example file *ExempleRun_LTN_articles.txt*).

Exercise 2: SMART *ltc* run

Same question, but using SMART *ltc* weighting function instead of *ltn*.

Exercise 3: *BM25* run

Same question, but using *BM25* weighting function instead of *ltc*. Set the k_1 and *b* parameters as you want. You can use these usual values: $k_1 = 1.2$ and $b = 0.75$.

Exercise 4: Stemmer, stop-words

Build several variants of your first runs, using stop-words or not, using a stemmer or not. With 1 stemmer and 1 stop-list, you can generate at least 3 (weighting) * 2 (stop-list) * 2 (stemmer) = 12 runs.

Exercise 5: BM25 tuning

In order to tune the BM25 weighting function, generate runs exploring the 2-dimensions space of its parameters (k_1 and b). Think about your optimization strategy. A simple one could be to fix k_1 to 1.2 and try 11 values for b (from 0.0 to 1.0, step = 0.1), and then fix b to 0.75 and try 21 values for k_1 (from 0 to 4, step = 0.2). A more efficient one could be to use the gradient descent algorithm.