Comments on 1st Web Intelligence Mini Project

Eric Ruder, Mathew Yapp, Simon Støvring

October 7, 2014

The following are suggestions on how we think, that you can improve your solution.

1 Crawler

- The near duplicate detection algorithm is called Jaccard similarity.
- You have not implemented a heap to keep track of which hosts, the crawler is allowed to visit. Might end up with a small amount of backqueues and a lot of consecutive hits on the same domain/host.
- A static website might fake being a dynamic website, by making regular and unnoticeable changes. This could be done to exploit the search engine and give the static website a better ranking.
- Have you really implemented sketches? To hash each shingle and create a set of shingle hashes for each document, in order to compare with other sets of shingle hashes is not sketches.
- Why not exclude files that is not HTML and plain text then, when other formats give “text encoded in a strange manner”?

2 Indexer

- You only stem on the English language, but you exclude Danish and English stop words. Does this make sense? How do you detect what words are English and which are Danish?
- You also mention you only remove English stop words, but your choice of seeds includes a Danish website.
- In figure 3, it is implied that the results of the indexer is stored in a database/saved on disk, but you write that your postings list are stored in memory, therefore the figure does not entirely match your implementation.

3 Ranker

- How is the search query handled? Is it tokenized, stemmed and filtered for stop words? Is it normalized? What is the smart notation of your ranker? What formulas did you use?
- What SMART notation does your ranker use?
- How did you implement CosineScore?
- How did you implement the Champions list?