Lab 10 – Ranked Search over Wikipedia

CC5212-1

May 11, 2016

Today we combine the last two labs: we will use the PageRank scores from the last lab to improve the results of the search engine we built two labs ago. Let's see if the results improve with PageRank!

For this, you will need the code of both Lab 8 and Lab 9. If you did not complete these labs, you can find the code here:

- `http://aidanhogan.com/teaching/cc5212-1-2016/sol/mdp-lab8-sol.zip`
- `http://aidanhogan.com/teaching/cc5212-1-2016/sol/mdp-lab9-sol.zip`

If you haven't built the inverted index or computed the PageRank scores over the Wikipedia data, you will need to follow lab instructions 8 and 9 again.

- `http://aidanhogan.com/teaching/cc5212-1-2016/lab/08/mdp-lab08.pdf`
- `http://aidanhogan.com/teaching/cc5212-1-2016/lab/09/mdp-lab09.pdf`

You need to do lab 8 for sure. However, if you want to skip re-doing lab 9, the final PageRank scores are available here `http://aidanhogan.com/teaching/cc5212-1-2016/data/es-wiki-ranks.tsv.gz`.

- `http://aidanhogan.com/teaching/cc5212-1-2016/lab/08/mdp-lab08.pdf`

The following assumes you have an inverted index at `INV-DIR` and the PageRank scores of the articles at `DIR`/es-wiki-ranks.tsv.gz

- Note that at the end of this lab, you will have to submit your search results using different ranking weights, etc., so please copy and paste them into a text file and make clear what you searched for and what settings you use for which results! The settings may look like

```
== title 5f, abstract 1f, log_10(pagerank) ==
= search "obama" =
Result 1 ...
Result 2 ...
...
= search "neruda" =
...
```

  This says you have set title to have 5 times more weight than abstract and have used the log of the Pagerank Gcore as a boost (it will become more clear later).

- As a baseline, let's collect some search results before adding the PageRank scores to see if the results improve:
  - Run the class `SearchIndex` with the argument `-i` `INV-DIR`.

  - Run some searches. Copy the results into a text file and save them for comparison later (you can just note the settings as `== default ==`). Search for "`obama`", "`boston`" and "`neruda`" and two other searches of your choice. (If you encounter problems with searches involving accents, make sure to set your console to UTF-8: `Run Configurations > Common` and select `Other [UTF-8]` under encoding.)
  - Let us try changing the weighting of title vs. abstract. At the moment, a keyword hit for title is 5 times more than one for abstract. Let's try set them equal. In `SearchWikiIndex`, set `BOOSTS.put(FieldNames.TITLE.name(),1f);`. Try run the same queries again (you can note the settings as `== title 1f, abstract 1f, no pagerank ==`)

- Last but not least, we want to use the PageRanks to increase the score of more important articles in Wikipedia (based on their link structure).
  - First make a copy of the inverted index directory (which we call `INV-DIR-COPY`). We will include the ranks in this copy.
  - Download the code project from `http://aidanhogan.com/teaching/cc5212-1-2016/lab/10/mdp-lab10.zip` and open it in Eclipse.
  - Please review the code to see how it uses the PageRank scores to boost the importance of documents in the inverted index.
  - Please make sure to change `StandardAnalyzer` to `SpanishAnalyzer` in `BoostRanks` (my mistake).
  - Time to run `BoostRanks` over the index:

    `-i` `DIR``/es-wiki-ranks.tsv.gz -igz -o` `INV-DIR-COPY`

    You may need to set more memory on your machine to load the ranks.

- Before you run the searches again, modify `SearchIndex` to print the PageRank of the results. Also, return the field boosts so title is 5 times more important than abstract again.

- Run the same five searches again and copy them to your results. Are the results better than before?

- In BoostRanks, there is a method called `getBoost`. I would like you to make another copy of the raw inverted index and to modify this method to try boost with the log (base 10) of the PageRank score * 100000. Run the queries and copy the results.

- Finally, based on what you've seen, please configure the title/abstract weight and `getBoost` in a way you think would give the best results (hint: there is no correct answer, just go with your gut). Run the queries and copy the results.

- Please submit just your search results to u-cursos.