Norwegian University of Science and Technology

Assignment Title

Assignment 4

-

PageRank and HITS, Structured Indexing and Retrieval in Lucene

Course

TDT4117 Information Retrieval

Semester

FALL 2018

Date Submitted

09/11/2018

Submitted by

Dusan Jakovic

Task 1: PageRank and HITS

1. Compare PageRank and HITS and briefly describe the main ideas of both approaches and point out their differences.

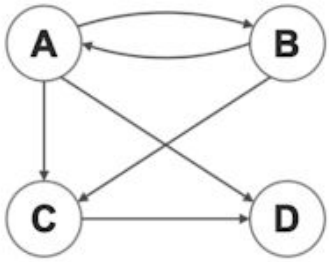
Hits
Hyperlink-Induced Topic Search (HITS) assigns two scores for each page; Authorities and Hubs. Authority estimates the value of the content of the page. Hub value estimates the value of its links to other pages.

PageRank
The PageRank score for a page is determined by summing the PageRanks of all pages that point to it. The “PageRanks “weight” for a page is higher if the page is considered important, which helps to make other pages more important.

Difference
PageRank is robust against spam, query independent, is difficult for new pages and it is weak against inlink-farms. HITS is quick on small neighborhood graphs, ranking according to user relevance, weak against advertisement and spam.

What makes the biggest difference between these two is that PageRank produces ranking independent of a user’s query.

2. Given the graph below, compute hub and authority scores for webpages labeled as A, B, C and D using HITS algorithm. Perform at least 3 iterations of the algorithm and illustrate your computations by providing formulas filled with values for at least one iteration.



ure 1: Graph of websites connected by links.

Adjacency matrix with nodes: 0 1 1 1, 1 0 1 0, 0 0 0 1, 0 0 0 0

HITS Algorithm with three iterations:

Start with each node having a hun score and authority score of 1:

	A	B	C	D
Hub (Out-degree)	1	1	1	1
Authority (In-degree)	1	1	1	1

Iteration 1:
Run the Authority and Hub update rule:

	A	B	C	D
Hub	3	1	1	1
Authority	0	3	2	1

Normalize the values:
Normalization value Hub: **3.464**
Normalization value Authority: **3.464**

Normalized scores:

	A	B	C	D
Hub	0.866	0.289	0.289	0.289
Authority	0.000	0.802	0.535	0.267

Next iteration = True

Iteration 2:
Run the Authority and Hub update rule:

	A	B	C	D
Hub	1.604	0.535	0.802	0.802
Authority	0.000	1.443	1.155	0.866

Normalize the values:
Normalization value Hub: **2.035**
Normalization value Authority: **2.041**

Normalized scores:

	A	B	C	D
Hub	0.788	0.263	0.394	0.394
Authority	0.000	0.707	0.566	0.424

Next iteration = True

Iteration 3:
Run the Authority and Hub update rule:

	A	B	C	D
Hub	1.697	1.576	1.050	0.788
Authority	0.000	1.576	1.050	0.788

Normalize the values:
Normalization value Hub: **2.049**
Normalization value Authority: **2.051**

Normalized scores:

	A	B	C	D
Hub	0.826	0.276	0.345	0.345
Authority	0.000	0.768	0.512	0.384

Next iteration = False

Task 2: Structured Indexing and Retrieval in Lucene

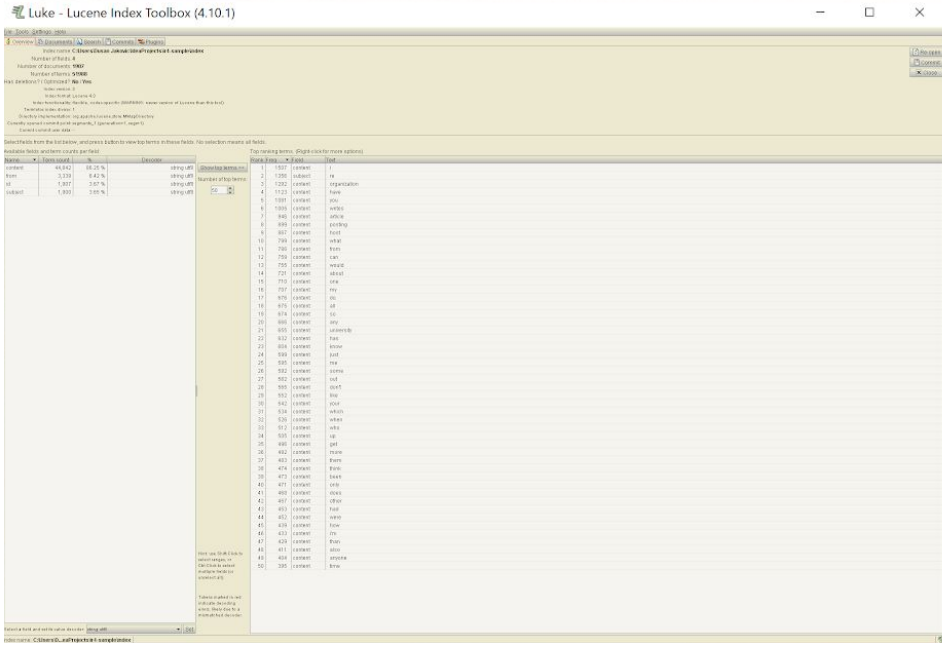
Subtask A. The task is to update the given ‘MyDocument’ class and implement the ‘Document(File f)’ method to index the following fields per document:

- id: the name of the file.
- from: whatever is stored in the from field of the given message.
- subject: the subject of the e-mail.
- contents: the actual e-mail contents

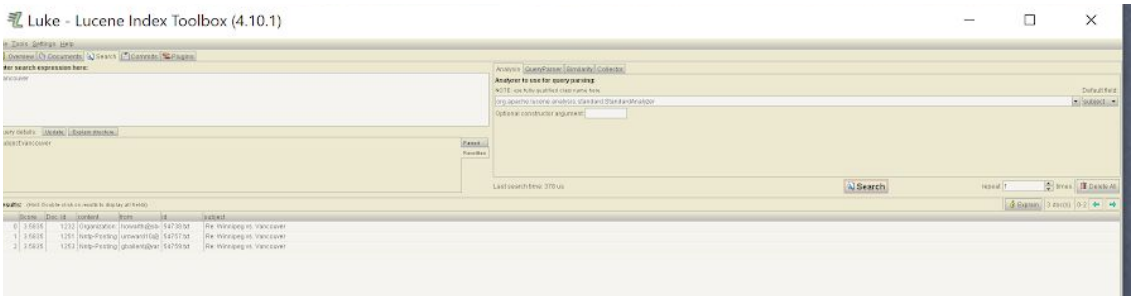
Code:

```
//creating structured lucene document
doc.add(new StringField("id", newsDocument.getId(), Field.Store.YES));
doc.add(new TextField("from", newsDocument.getFrom(), Field.Store.YES));
doc.add(new TextField("subject", newsDocument.getSubject(), Field.Store.YES));
doc.add(new TextField("content", newsDocument.getContent(), Field.Store.YES));
```

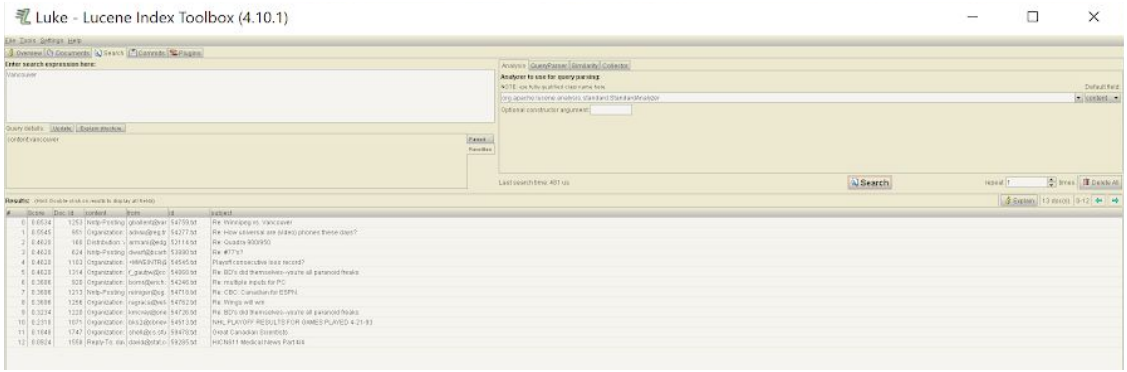
Subtask B. Once you start the application, choose the path to the index directory of subtask A . Use Luke to search for the term ‘Vancouver’ in different fields. Use Luke’s search tab and set the analyzer to ‘org.apache.lucene.analysis.standard.StandardAnalyzer’ and select different fields. Show screenshots of the results and explain the behavior of the system.



Choosing the path to the index directory of subtask A



Searching for term “Vancouver” in **subject**



Searching for term “Vancouver” in **content**

Note: **id** and **from** will not return any results