Week 7:
Visualizing the Search Results

- This assignment is due on **2nd December, 2015 (13:30)**
- You can discuss the problems with other groups of this course or browse the Internet to get help. However, copy and paste is cheating.
- There are 13 weekly exercises in total. In each one of them, all assignments sum up to 20 points. You need to achieve at least 80% of all assignments during the course in order to participate in the final exam. Hence, you need to achieve at least 208 points in total (*13*20*0.8=208*).
- Submission at
  `https://www.dcl.hpi.uni-potsdam.de/submit`
  - only zip files
  - one zip file per group per week (week7.zip)
  - put your names on *each* sheet in your pdf file

Assignment 1: Clustering Results

a) Have you ever used any other search engine except for Google? Try using Yippy and Carrot2 with patent-related terms. These search engines provide a list of relevant topics to the query terms along with the corresponding pages of each topic. What do you think about the clustering results? Are the topics of the clusters sufficiently relevant to your query? **3 P**

b) Consider the characteristics of the patent data that you either discovered so far in the U.S. patent xml files or in patent-related material on the Web. Can you think of a user-friendly way to present your search engine results clustered? For example, based on the names of the authors, or the classification of each patent or the content of the titles? (hint: there are many xml elements apart from the title and the abstract that could be useful for the patent clustering) **5 P**

Assignment 2: Showing Results

Consider some patent-related queries. Compare the result pages of Google and Bing for the above queries.

a) What elements are similar, different, the same? **2 P**

b) Find two different result snippets for the same page and say which one you prefer and why. **4 P**

Assignment 3: (Programming) Snippet Generation

This week we will implement snippet generation and adjust the blind feedback method so that it only uses the terms in the snippets.

- In order to generate the snippets you should use the testData.xml file as index. To access each patent in the xml file, you should build a seek list that maps each patent document ID to its offset in the file. You could also map the elements of the patents that you are interested in, for instance the offset of the abstract or the title of each patent.
- A simple way to print snippets could be to print a few words before and after each query term that is included in the document. On the other hand, it is useful to the user to provide the sentences/segments that contain as many query terms as possible.

- You should also ignore the xml tags and only print the text content. Highlighting or coloring the output is also a user-friendly way of showing your results.
- As in the previous assignment, you should use the prf variable as the number of the documents that will be used in the pseudo-relevance feedback method. The difference in this assignment is that for each of the prf documents, you will consider its snippet instead of its abstract for the pseudo-relevance mechanism. You should again use the value of 2 for the prf variable and 10 for the k variable.

a) Explain in a few words how your snippet generation method works in your pdf file.

b) Execute the below queries using pseudo-relevance feedback based on snippets. Print the improved visualization of the ranking of the patents that match the below queries in a seperate txt file.

- *access control* **1 P**
- *computers* **1 P**
- *data processing* **1 P**
- *web servers* **1 P**
- *vulnerability information* **1 P**
- *computer-readable media* **1 P**

Query language

a) These are the operators that your search engine should generally support at the end of the lecture.

- no operators (simple keyword search)
- " " (phrase queries wrapped in ' ')
- AND, OR, NOT (three different boolean operators)
- * (prefix queries like dat*, inf*)
- #5 (# enables pseudo-relevance feedback, prf is 5)

b) Having the above query types and different operators in mind, the following queries are the possible combinations that you may be asked to answer. Let's consider the terms "data", "information", "mobile" and "processing".

- *data* AND *information*
- *mobile* OR "*data processing*"
- *data* NOT *info**
- "*data proces**" NOT *processing*
- "*data processing*" #2
- "*data proces**" #4
- "*data processing*"*mobile* #2
- *mobile data* #3
- *mobile dat** #2

c) The queries issued in your patent engine in the evaluation phase will be of the above form. It is not necessary for your system to support all of them, but the more you support the more query types you will be able to answer at the end. However, it is manadatory to support the simple keyword search, since at least 50% of the queries will be of this kind. You should also let the user decide whether to use feedback or not, by taking the # symbol into account. Finally, although it is not compulsory to implement all of the above combinations, it is important to use the same operators, so that there are no incompatability issues during the evaluation time.