

Week 10: Evaluation

- This assignment is due on **6th January, 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 ($13 \cdot 20 \cdot 0.8 = 208$).
- Submission at
<https://www.dcl.hpi.uni-potsdam.de/submit>
 - only zip files
 - one zip file per group per week (week10.zip)
 - put your names on *each* sheet in your pdf file

Assignment 1: Test Collections

- a) Find three test collections used for information retrieval evaluation in the literature not presented in the lecture and describe them (number of documents, number of queries, how relevance was assessed) **4 P**

Assignment 2: Measures

For a given query and a collection of 100 Web pages (which contains 40 relevant pages), a search engine produces the following ranking:

- 1 relevant
- 2 irrelevant
- 3 relevant
- 4 relevant
- 5 irrelevant
- 6 irrelevant
- 7 relevant
- 8 irrelevant
- 9 relevant
- 10 relevant

Compute the following evaluation measures:

- a) Precision and recall **3 P**
- b) Precision at 7 and recall at 7 **3 P**
- c) MAP **3 P**
- d) NDCG (assume binary gain value (relevant/non-relevant)) **3 P**

Assignment 3: (Programming) Evaluation

This week we will implement an evaluation algorithm based on the ranking of the Google Patent Search Engine (google.com/patents).

- Download the latest version of the Java template code from the lecture's resources page. There is a new file called *WebFile* to get the Google ranking results for a given query. The method to get the Google ranking is called `getGoldRanking(String query)`. You will use this ranking to compute the NDCG metric.
 - Assume the best result in the gold ranking has a gain value of 10, the second and third best 9, ..., the 50th best a gain value of 1. We assume an exponential decay of the gain value: $gain(rank) = 1 + floor(10 * 0.5^{0.1 * rank})$.
 - Override abstract `Double computeNdcg(ArrayList<String> goldRanking, ArrayList<String> ranking, int p)` to compute NDCG for a given goldRanking from Google and your ranking at a position p in the rank.
- a) Execute the below queries using as *topK* the value of 5. Print the titles, snippets and the NDCG values for the patents that match the queries. (Note: if the query doesn't contain the symbol #, then your pseudo-relevance feedback method should be disabled)
- "add-on module" 1 P
 - "digital signature" 1 P
 - "data processing" 1 P
 - ""a scanning"" 1 P