Assignment 5

Information Retrieval and Web Search

Ahmed Rekik - 790063 & Tim Henning - 789242

December 17, 2017

1. Showing Results

1.a. Think about result visualization for a comment search engine. What information could be shown in a snippet?

In general, the result is returned what the search engine considers as relevant documents in terms of a given query by the user. Certainly, the result could differ in its ranking or even in its relevance consideration in terms of the employed informational retrieval model. In all cases, a search engine aims to return an optimal result for the user in many formats (text, images, videos, link to documents), but it could also include sponsored advertisements that have a relationship to the query or most viewed/rated documents (important news, shop offer...) in sophisticated and lucrative search engine as "Google" or "Bing". In the "Comment search engine" frame those type of result could apply.

On the result snippet, it is reasonable that the result will include a ranked vertical list of matched comments at the front of the page in a textual form (a normalized snippet if the comment is very long) since they represent the main product and the potential information need to the user. Besides that, other information/suggestion could occur such as the article related to the comment, the comment author, the comment-date or something irrelevant to the user informational need such that the most rated comments of the week or month of all the newspaper-website to show interactivity in the search engine.

1.b. What elements are similar, different, or the same compared to result pages of Google, Bing, or any other web search engine?

Although all search engines have an identique aim (returning relevant information to the user), may there result visualization and their ranking differ. As a point of similarity that most of famous web search engine share, is that the query result is returned in a vertical list of snippets from the document headed with a link for the page titled by the page title and that have the same length size. The sponsored advertisments (that could have a relationship with the query or not) has been also more and more introduced by all prominent web search engine in the result visualisation since the last decade.

Otherwise, the result visualization differs from an engine to another due to their different offered feature. This difference can occur even in the matched query result. The first reason for this fact is that some search engine favorize the document shared by websites related to their company than others. For example: by "*Yahoo*" web search engine documents provided by "Yahoo *news/sports/answers*" may be ranked better than all other documents (usually on the top of the result) or "*Wikipedia*" pages are always taking the top-5 result in "*Google*". Other information could be shown also in terms of the query context due to the web searcher feature. For example: If a user types a football team name, a snippet of latest match results, current or former players and club rivals could be listed by "*Google*", otherwise by "*Bing*" they will scroll other information and may suggest tickets sellers or bets website and win probability for the next games.

2. Evaluation Measures

For a given query and a collection of 100 Web pages (which contains 40 relevant pages), a search engine produces the following ranking: R, N, R, R, N, N, R, N, R, R. R denotes a relevant document and N denotes a non-relevant document. Calculate the following evaluation measures:

2.a. Precision and recall

Precision: P $= 7 \div 11 = 0,63$
Recall: R $= 7 \div 40 = 0,175$

2.b. Precision at 7 and recall at 7

Precision at 7: P $= 4 \div 7 = 0,571$
Recall at: R $= 4 \div 40 = 0,1$

2.c. MAP

Table 1: MAP

i	1	2	3	4	5	6	7	8	9	10	11
Result	R	N	R	R	N	N	R	N	R	R	R
i-Precision	1	0.5	0.67	0.75	0.6	0.5	0.571	0.5	0.55	0.6	0.63

$\Rightarrow 1 + 0.67 + 0.75 + 0.57 + 0.55 + 0.6 + 0.63 = 4,77$
$MAP = 4.77 \div 7 = \mathbf{0.681}$

Table 2: Normalized DCG

i	1	2	3	4	5	6	7	8	9	10	11
Result	R	N	R	R	N	N	R	N	R	R	R
DCG	1	1	1.63	2.13	2.13	2.13	2.48	2.48	2.79	3.09	3.37
Perfect result	R	R	R	R	R	R	R	N	N	N	N
P-DCG	1	2	2.63	3.13	3.56	3.94	4.29	4.29	4.29	4.29	4.29
N-DCG	1	0.5	0.619	6.8	0.598	0.54	0.578	0.578	0.65	0.72	0.78

2.d. NDCG (assume binary gain value (relevant/non-relevant))

3. (Programming) Index compression

3.a. Report the sizes of the compressed and uncompressed index files in your implementation.

Table 3: Index Sizes (bytes)

Part	uncompressed	compressed	compression rate	bytes / element
Seek List	2,449,689	857,665	64.9%	7.71
Postings	124,015,956	86,309,936	30.4%	9.67

3.b. Write down which kind of compression techniques you are using.

To compress the seek list, we are using delta compression for both the token strings and the byte offsets of their postings lists. To implement the string delta compression, the first byte of each token is the number of characters that are equal to the token before, followed by the characters that changed. This way, a token where only the last character changed ('negotiate' -> 'negotiated') consumes only two bytes (instead of 10 in this example).

In addition, we removed all unnecessary overhead in the seek file, while still only using ASCII chars. In the end, this means that we need only 7.7 bytes per token on average (~4 bytes for the token and ~3 bytes for the byte offset).

The posting lists are compressed by storing the comment IDs and word positions with base 64 instead of base 10, represented by ASCII characters. This means we do not need two bytes to store the number '10' anymore, but only one. Furthermore we removed some overhead from the file structure. One pair of comment ID (which is a 64bit memory offset) and the word position in the comment takes 9.6 bytes on average, with only ASCII characters used.