Information Retrieval -Index, Search

Borna Feldsar 01638974, Kresimir Kasal 00026127

April 2018

1 Index creation

We parsed document in a way to look for a line that matches the pattern *<DOCNO>(.+?)<DOCNO>* which actually is documents ID. After it we looked for a lines which match a start tag of text and the end tag. User has to specify folder which has document files and all files are then parsed recursively. User has an option to use preproccessing, e.i. case folding, stemming, normalization, stop words. For stemming we used *Lucene* library and we did normalization in a way that we check if the word is number and if not we remove all dots,e.i. word U.S.A. matches USA with normalization. There are two tokenizers that user can use, *BreakIterator* from package *java.text.BreakIterator* or split method by the regex pattern. Tokens which are only special character or empty string are skipped and not added in the index. User can also specify size of an index block in a way that size is defined by the number of terms in index block.

For index creation method *Single pass in memory indexing* , SPIMI is used. In index block we stored a HashMap in which key is term and value is list of postings. List of postings consist of Posting class. Its consists of document id and term frequency, i.e. number of word repetition in the document. Beside that we store a map which has a document id as key and number of unique words as value and also a map which has a document id as key and total number of words in it.

2 Search

For the search, the query according to the topic number is searched first. Afterwards, in the class CosineScore the weight according to the scoring function is calculated. Three functions are implemented: Tf-Idf, BM25 and BM25VA. For the Tf-Idf function, weights for the query and the document are calculated, multiplied and afterwards divided by the multiplication of the two lengths (query and document lengths). For the BM25 and BM25VA scoring functions, required parameters are passed via command line. Default parameter values for the BM25 and BM25VA functions were taken from literature (k1=1.2, k3=8, b=0.75). In an example run where the query consisted of two words

401			

Table 1: Caption

("peace" and "freedom"), documents FR940105-0-00048 and FR940105-0-00050 were found as the ones with the highest ranking score.

3 Results