Lab Class IR:II

Exercise 1 : Architecture

In which conceptual "layer" (i.e., indexing, storage, and retrieval) would you locate the following components or processes of a search engine.

(a) Crawler

(b) Language model

(c) Snippet generation

(d) Stemming

(e) Stop word removal

(f) Query analysis

(g) Posting list

Exercise 2 : Document processing

Process this document step-by-step according to a standard document processing pipeline:

> Information retrieval (IR) in computing and information science is the task of identifying and retrieving information system resources that are relevant to an information need. The information need can be specified in the form of a search query. In the case of document retrieval, queries can be based on full-text or other content-based indexing. Information retrieval is the science [1] of searching for information in a document, searching for documents themselves, and also searching for the metadata that describes data, and for databases of texts, images or sounds.[1]

(a) What is the output after tokenization (ignoring uppercase/lowercase)?

(b) What is the output after stop word removal (also remove numbers)?

(c) What is the output after lemmatization?

Exercise 3 : Query processing

Which essential steps happen to a query until it can be used for searching an inverted index?
How do the steps relate to document processing.

Exercise 4 : Posting list

Construct the posting lists of the index terms a, b, and e, given the two documents d_1 and d_2:
$d_1 = [a, a, b, c, b, b, d, e, b, c, a, e, d, c]$
$d_2 = [d, f, c, b, b, d, a, a, b, c]$
Use term frequency (TF) as term weighting scheme.

[1] https://en.wikipedia.org/wiki/Information_retrieval

Exercise 5 : Document statistics

Why could it be useful to store the number of documents that a term occurs in?

Exercise 6 : Relevance scoring

Relevance scoring is the quantification of the (assumed) relevance of an indexed document d to a query q. Consider this simple relevance function ρ:

$$\rho(q,d) = \sum_{t \in T} \omega_Q(t,q)\omega_D(t,d) \tag{1}$$

Here, $\omega_Q(t,q)$ and $\omega_D(t,d)$ are the term weights indicating the importance of term t for the query $q \in Q$ or the document $d \in D$, respectively.
For this example, assume a corpus of just two documents, d_1 and d_2 and the vocabulary $T = \{a, b, c\}$. The pre-computed term weights $\omega_D(t, d_i), t \in T$ are stored in the given postings lists:

$$\begin{aligned} p_a &= [(d_1,3),(d_2,2)] \\ p_b &= [(d_1,4),(d_2,3)] \\ p_c &= [(d_1,1)] \end{aligned}$$

Use the term frequency (TF) to compute $\omega_Q(t,q)$. The query is $q = aac$.

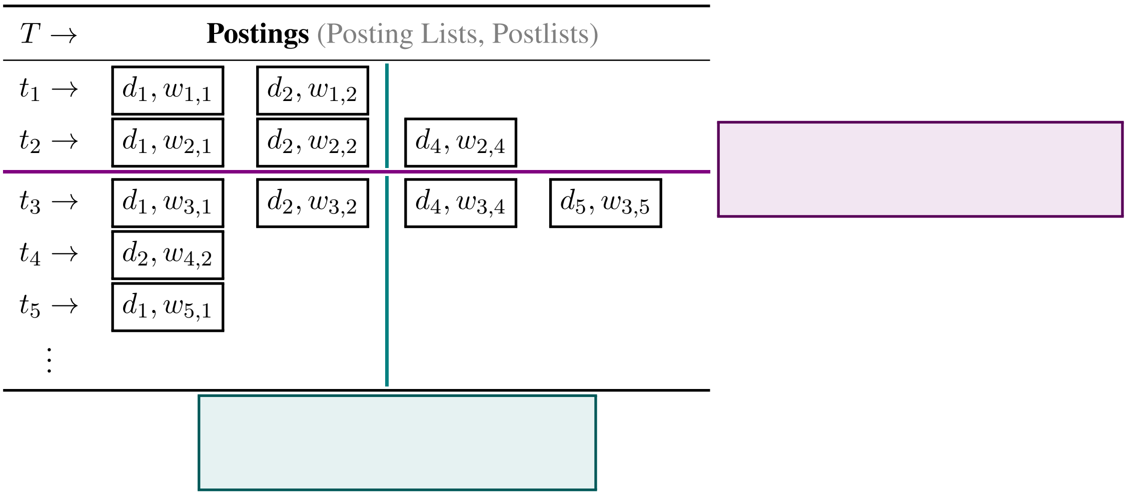
(a) Compute the relevance scores $\rho(q,d_1)$ and $\rho(q,d_2)$ according to Equation 1.

(b) What is the resulting ranking of the documents?

(c) Did you use document-at-a-time scoring or term-at-a-time scoring?

Exercise 7 : Distributed storage

Large indexes are distributed (sharded, partitioned) across machines. The lines represent these partitions. Insert (a) document distribution and (b) term distribution into the corresponding boxes.



$T \rightarrow$	**Postings** (Posting Lists, Postlists)			
$t_1 \rightarrow$	$d_1, w_{1,1}$	$d_2, w_{1,2}$		
$t_2 \rightarrow$	$d_1, w_{2,1}$	$d_2, w_{2,2}$	$d_4, w_{2,4}$	
$t_3 \rightarrow$	$d_1, w_{3,1}$	$d_2, w_{3,2}$	$d_4, w_{3,4}$	$d_5, w_{3,5}$
$t_4 \rightarrow$	$d_2, w_{4,2}$			
$t_5 \rightarrow$	$d_1, w_{5,1}$			
$\vdots$				