

Lab Class IR**Exercise 1 : Abstract Ranking Model**

We introduced an abstract model of ranking, where documents and queries are represented by features.

- (a) What are advantages of representing documents and queries by features?
- (b) What are disadvantages?

Exercise 2 : Abstract Ranking Model

Documents can easily contain thousands of non-zero features. Why is it important that queries have only a few non-zero features?

Exercise 3 : Inverted Index

Indexes are not necessary to search documents. Your web browser, for instance, has a “Find” function in it that searches text without using an index. Also the UNIX tool `grep` does not use an index.

- (a) When should you use an inverted index for text search?
- (b) What are some advantages of using an inverted index? What are some disadvantages?

Exercise 4 : Inverted Index

We have seen many different ways to store document information in inverted lists of different kinds.

- (a) What kind of inverted lists might you build if you needed a very small index?
- (b) What kind would you build if you needed to find mentions of cities, like Los Angeles or São Paulo?

Exercise 5 : Wildcard indexing

How may a search engine that uses an n -gram inverted index be modified to support these wildcards:

- Token-Wildcard ? that can match any token (e.g., *to* ? or *not to be*)
- Character-Wildcard * that can match any character in a token (e.g., *in*m*ion* should match among others *information*)

Which components need to be changed and how?

Exercise 6 : Term-document matrices & inverted indices

The term-document matrix in [Table 1](#) contains documents *Antony* and *Cleopatra*, *Julius Caesar*, ... and terms *Antony*, *Brutus*, ...

[Table 1](#): Term-Document Matrix of Shakespearean plays. Cell entries denote term weights $w_{i,j} = tf(t_i, d_j)$ (i.e., term-frequency)

	Antony and Cleopatra	Julius Caesar	The Tempest	Hamlet	Othello	...
Antony	382	128	0	0	0	
Brutus	4	379	0	1	0	
Caesar	289	272	0	2	1	
Calpurnia	0	16	0	0	0	
Cleopatra	271	0	0	0	0	
:						..

(a) Using set retrieval, which documents are returned for the following queries?

- $q_1 : \text{Antony}$
- $q_2 : \text{Antony} \wedge \text{Caesar}$ (i.e., conjunctive multi-term query)
- $q_3 : (\text{Antony} \vee \text{Caesar}) \wedge (\neg \text{Calpurnia})$ (i.e., disjunctive multi-term query)

(b) Do you see any shortcomings of this representation?

(c) How would the corresponding space-efficient inverted index look like?

Exercise 7 : Index configurations

Match query types to optimal index configurations for ranked retrieval.

Query types	Index configurations
• Single-term queries (A)	• Postlists ordered by document ID (D)
• Disjunctive multi-term queries (B)	• Postlists ordered by document quality (E)
• Conjunctive multi-term queries:	• Postlists ordered by term weight (F)
– Boolean AND queries (C1)	• Positional indexing (G)
– Proximity queries (C2)	
– Phrase queries (C3)	

Exercise 8 : Inverted indices

- (a) Describe all components of the inverted index shown in [Table 2](#).
- (b) In what order are the postings arranged, and which query types are better or worse suited to this ordering?
- (c) Compute the collection of documents relevant to $q = t_1 \wedge t_2$ (i.e., perform the list intersection operation for terms t_1 and t_2).

[Table 2: Inverted index of Shakespearean plays.](#)

T	Postings
t_1	$(d_2, w_{1,2}, \text{len, skip}) \quad (d_4, w_{1,4}) \quad (d_8, w_{1,8}) \quad (d_{16}, w_{1,16}, \text{skip}) \quad (d_{19}, w_{1,19}) \quad (d_{23}, w_{1,23}) \quad (d_{28}, w_{1,28}, \text{skip}) \quad \text{NIL}$
t_2	$(d_1, w_{2,1}, \text{len, skip}) \quad (d_2, w_{2,2}) \quad (d_3, w_{2,3}) \quad (d_5, w_{2,5}, \text{skip}) \quad (d_8, w_{2,8}) \quad (d_{41}, w_{2,41}) \quad (d_{51}, w_{2,51}, \text{skip}) \quad \text{NIL}$
:	

Exercise 9 : Merge indices

Merge indices from [Table 3](#) and [4](#).

[Table 3: Inverted index 1.](#)

T	Postings
t_1	$(d_4, w_{1,4}, \text{len, skip}) \quad (d_{19}, w_{1,29}) \quad (d_{23}, w_{1,23}) \quad (d_{28}, w_{1,28}, \text{skip}) \quad (d_{50}, w_{1,50}) \quad \dots$
:	

[Table 4: Inverted index 2.](#)

T	Postings
t_1	$(d_2, w_{1,2}, \text{len, skip}) \quad (d_8, w_{1,8}) \quad (d_{16}, w_{1,16}) \quad (d_{41}, w_{1,41}, \text{skip}) \quad (d_{77}, w_{1,77}) \quad \dots$
:	