Basic comprehension questions.
Check that you can answer them before proceeding. Not for credit.

1. True or false: The boolean model does not rank documents in the answer, while the vectorial model allows for ranking.

2. Suppose you are given the frequency of every term in a given document. What other information do you need to compute its representation in tf-idf weights?

3. Hide the course slides. Write down the formula of the cosine measure of document similarity. Now look at the slides. Check your answer. Repeat until correct.

4. Same for the tf-idf weight assignment scheme.

Exercise 1

Consider the following documents:

\( D_1 \): Shipment of gold damaged in a fire

\( D_2 \): Delivery of silver arrived in a silver truck

\( D_3 \): Shipment of gold arrived in a truck

and the following set of terms:

\[ T = \{ \text{fire, gold, silver, truck} \}. \]

Compute, using the boolean model, what documents satisfy the query

\( (\text{fire OR gold}) \ \text{AND} \ (\text{truck OR NOT silver}) \)
and justify your answer. Do the same with the query

(fire OR NOT silver) AND (NOT truck OR NOT fire).

Argue whether it is possible to rewrite these queries using only the operators AND, OR and BUTNOT in a logically equivalent way. This means that it must be equivalent for all possible document collections, not just this one.

**Exercise 2**

Consider the following collection of five documents:

- Doc1: we wish efficiency in the implementation for a particular application
- Doc2: the classification methods are an application of Li’s ideas
- Doc3: the classification has not followed any implementation pattern
- Doc4: we have to take care of the implementation time and implementation efficiency
- Doc5: the efficiency is in terms of implementation methods and application methods

Assuming that every word with 6 or more letters is a term, and that terms are ordered in order of appearance,

1. Give the representation of each document in the boolean model.
2. Give the representation in the vector model using tf-idf weights of documents Doc1 and Doc5. Compute the similarity coefficient, using the cosine measure, among these two documents.

(Answer to 2: I get 0.162.)

**Exercise 3**

We have indexed a collection of documents containing the terms of the following table; the second column indicates the percentage of documents in which each term appears.
Given the query \( Q = \text{“computer software programmers”}, \) compute the similarity between \( Q \) and the following documents, if we use tf-idf weights for the document, binary weights for the query, and the cosine measure. Determine their relative ranking:

- \( D_1 = \text{“programmers write computer software code”} \)
- \( D_2 = \text{“most software has bugs, but good software has less bugs than bad software”} \)
- \( D_3 = \text{“some bugs can be found only by executing the software, not by examining the source code”} \)

\((Answer: I get similarities 0.966, 0.436, 0.244.)\)

**Exercise 4**

Suppose that terms A, B, C, and D appear, respectively, in 10,000, 8,000, 5,000, and 3,000 documents of a collection of 100,000.

1. Consider the boolean query \((A \text{ and } B) \text{ or } (C \text{ and } D)\). How large can the answer to this query be, in the worst case?

2. And for the query \((A \text{ and } B) \text{ or } (A \text{ and } D)\)? Think carefully.

3. Compute the similarity of the documents “A B B A C C” and “D A D B B C C” using tf-idf weighting and the cosine measure.

\((Answers: 1) 11.000 2) 10.000 3) 0.736.)\)

**Exercise 5**

We have an indexed collection of one million documents that includes the following terms:

<table>
<thead>
<tr>
<th>Term</th>
<th>% docs</th>
</tr>
</thead>
<tbody>
<tr>
<td>computer</td>
<td>10%</td>
</tr>
<tr>
<td>software</td>
<td>10%</td>
</tr>
<tr>
<td>bugs</td>
<td>5%</td>
</tr>
<tr>
<td>code</td>
<td>2%</td>
</tr>
<tr>
<td>developer</td>
<td>2%</td>
</tr>
<tr>
<td>programmers</td>
<td>2%</td>
</tr>
</tbody>
</table>
1. Compute the similarity between the following documents D1 and D2 using tf-idf weights and the cosine measure:

D1 = "p2p programs help users sharing files, applications, other programs, etc. in computer networks"

D2 = "p2p networks contain programs, applications, and also files"

2. Assume we are using the cosine measure and tf-idf weights to compute document similarity. Give a document containing two different terms exactly that achieves maximum similarity with the following document

"p2p networks contain programs, applications, and also files"

Compute this similarity and justify that it is indeed maximum among documents with two terms.

(Answer to 1. 0.925.)

Exercise 6

Consider the following collection of four documents:

Doc1: Shared Computer Resources
Doc2: Computer Services
Doc3: Digital Shared Components
Doc4: Computer Resources Shared Components
Assuming each word is a term:

1. Write the boolean model representation of document Doc3.

2. What documents are retrieved, with the boolean model, with the query “Computer BUT NOT Components”?

3. Compute the idf value of the terms “Computer” and “Components”.


5. Compute the similarity between the query “Computer Components” (with binary weights) and Doc4 (with tf-idf weights), with the cosine similarity measure.

(Answer to 4: I get 0.6534)