Cerca i Anàlisi de la Informació (CAI)
Grau en Ciència i Enginyeria de Dades, UPC

Session 2: Models

Exercise List, Fall 2021

Basic comprehension questions.
Check that you can answer them before proceeding.

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.
5. Write down the definitions of recall, precision, coverage, and novelty. Explain them in words in a way that you think your classmates would understand.
6. Explain to yourself how to compute a precision/recall graph.
7. True or false or criticize: To maximize user satisfaction, aim at a balance between recall and precision
8. Write down Rochio's formula for user relevance feedback.

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:

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

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

```
(fire OR gold) AND (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. 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.

Term	% docs
computer	10%
software	10%
bugs	5%
code	2%
developer	2%
programmers	2%

Given the query Q="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:

- D1 = "programmers write computer software code"
- D2 = "most software has bugs, but good software has less bugs than bad software"
- D3 = "some bugs can be found only by executing the software, not by examining the source code"

(Answer: I get similarities 0.766, 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 and B) or (C and D)`. How large can the answer to this query be, in the worst case?
2. And for the query `(A and B) or (A 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:

Term	# docs
computing	300,000
networks	200,000
computer	100,000
files	100,000
system	100,000
client	80,000
programs	80,000
transfer	50,000
agents	40,000
p2p	20,000
applications	10,000

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 BUTNOT Components"?
3. Compute the idf value of the terms "Computer" and "Components".
4. Compute the vector model representation of Doc4 using `tf-idf` weights.
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)

Exercise 7

A user tells us that, after asking a query to our search system, she found 10 relevant documents in positions 2, 6, 12, 18, 20, 22, 30, 36, 40, and 50. Assuming there are no more relevant documents in the collection, draw a precision-recall graph of the answer at 10 recall levels. Make sure you give the table of numbers that you used to plot the graph.

Exercise 8

We have a document collection with 100 documents, identified by numbers $1 \ldots 100$. Suppose that the relevant ones for a given query are those numbered $1 \ldots 20$.

Two information retrieval systems give as a result to the query the following answers:

S1= { 1,2,21,22,3,23,25,4,28,5,29,30,6,7,31,32,33,40,41,42,8,43,44,
9,45,10,50,51,11,52,53,54,12,60,62,13,63,64,14,15,16,70,78,80,17,
81,82,83,85,18,90,19,91,92,20,93,94,95,96,98 },

S2= { 25,26,1,27,28,2,3,29,30,4,35,36,5,37,6,7,8,38,9,40,10,42,11,45,46,
12,48,50,51,13,60,61,64,14,70,72,15,78,79,90 }.

For this query and each of the two systems,

a) Compute the recall, precision, and F-measure (with $\alpha = 1/2$, $\alpha = 1/4$, and $\alpha = 3/4$).

b) Compute the novelty and coverage measures, assuming that the user already knew the documents with odd index and did not knew about those with even index.