Information Retrieval – Final Exam (Spring 2021)

1. [*3+6+1 points*] Consider the following document collection:

Doc 1 New peace drug, new hope.
Doc 2 New process for peace in our time.
Doc 3 New hope for new peace process.
Doc 4 Breakthrough drug - for peace.
Doc 5 Peace brings hope.

(a) Draw the term-document incidence matrix for the collection after normalizing by applying case folding and removing punctuation.

(b) Draw the inverted index for the collection of documents after normalizing by applying case folding and removing punctuation. Include document frequencies and term frequencies in your picture.

(c) What are the returned results for these queries:

i. peace **AND** drug
ii. new **AND NOT** (drug **OR** hope)

a)

	doc 1	doc 2	doc 3	doc 4	doc 5
breakthrough	0	0	0	1	0
bring	0	0	0	0	1
drug	1	0	0	1	0
for	0	1	1	1	0
hope	1	0	1	0	1
in	0	1	0	0	0
new	1	1	1	0	0
our	0	1	0	0	0
peace	1	1	1	1	1
process	0	1	1	0	0
time	0	1	0	0	0

b)

terms		postings					doc freq.	doc ID (colour-code)					within doc frequency (position counts)					term position				
breakthrough	->	4					1	4					1					1				
bring	->	5					1	5					1					1				
drug	->	1	4				2	1	4				1	1				3	2			
for	->	2	3	4			3	2	3	4			1	1	1			3	3	3		
hope	->	1	3	5			3	1	3	5			1	1	1			5	2	3		
in	->	2					1	2					1					5				
new	->	1	2	3			5	1	2	3			2	1	2			1	4	1	1	4
our	->	2					1	2					1					6				
peace	->	1	2	3	4	5	5	1	2	3	4	5	1	1	1	1	1	2	4	5	4	1
process	->	2	3				2	2	3				1	1				2	6			
time	->	2					1	2					1					7				

c) Query results for:

i) peace **AND** drug = **1 , 4**

ii) new **AND NOT (**drug **OR** hope**) = 2**

2. [*3 + 3 points*] Encode the sequence of integers 6, 26, 226 using

(a) Elias γ codes;

(b) Vbyte codes.

a) Elias-gamma codes represent an integer n>0 as a pair of selector and offset. Procedure:

- *Offset: the integer in binary, with the first digit cut off*
- *Selector: length of the offset (nr of digits in the offset) written in Unary = that same amount in zeros, followed by a 1.*
- *Write a new number (elias-gamma code) which is the selector followed by the offset*

6:

- Offset Bin(6) = 110 ; cutting off 1st digit = 10
- Selector nr of digits in offset = 2 ; Unary(2) = 001
- Elias-gamma(6) = 00110

26:

- Offset Bin(26) = 11010 ; cutting off 1st digit = 1010
- Selector nr of digits in offset = 4 ; Unary(4) = 00001
- Elias-gamma(26) = 000011010

226:

- Offset Bin(226) = 11100010 ; cutting off 1st digit = 1100010
- Selector nr of digits in offset = 7 ; Unary(7) = 00000001
- Elias-gamma(226) = 000000011100010

b) For computing VB code:

- *Convert given number to binary*
- *From the least significant bit (from the right) take 7 bits; it will be your last Byte and a "1" will be added to the left of that Byte*
- *Repeat from the remaining of the number: take 7 bits (if available) and add a "1" to the left. If less than 7 digits remaining, then take them and put zeros to complete the Byte (total 8 digits).*
- *Write the numbers in the order that you processed them.*

6:

- Bin(6) = 110
- VB code (6) = 00000110

26:

- Bin(26) = 11010
- VB code (6) = 00011010

226:

- Bin(226) = 11100010
- VB code (6) = **1**11000100000000**1**

3. [*2 + 2 points*] Consider the table below of term frquencies for three documents from a collection of 906799 documents.

Table 1: Term frquencies and document frequencies.

term	Doc 1	Doc 2	Doc 3	df_t
tears	4	16	18	11816
from	28	33	67	606881
compound	0	8	5	1924
eye	14	33	24	12523

(a) First compute the idf values for each term and then compute the tf.idf weights for the terms for each document using the document frequency values (df_t column) in the table.

(b) Compute the Euclidean normalized document vectors for each of the three documents above, where each vector has four components, one for each of the four terms.

a) Formulas used: idf = log (N / dft) ; tf.idf = (1 + log $tf_{t,d}$) * log(N / dft)

N = 906799

term	doc1	doc2	doc3	dtf	idf
tears	4	16	18	11816	1.885041
from	28	33	67	606881	0.174407
compound	0	8	5	1924	2.673306
eye	14	33	24	12523	1.859803

	tf.idf		
term	**doc1**	**doc2**	**doc3**
tears	3.019948	4.154856	4.25128
from	0.426803	0.439248	0
compound	0	5.087542	4.541867
eye	3.991375	0	4.426723

b) For normalizing the components of the document vectors we need to calculate the L_2 norm, then divide each of the vector components (the tf.idf) by that L_2 normalization factor:

	length-normalized		
term	**doc1**	**doc2**	**doc3**
tears	0.601191	0.631128	0.556793
from	0.084965	0.066722	0
compound	0	0.772804	0.594851
eye	0.794576	0	0.579771
L2 norm	5.023278	6.583225	7.6353

So the normalized vectors for each document become:
Doc1 = [0.601 , 0.085 , 0.000, 0.795]
Doc2 = [0.631, 0.067, 0.773, 0.000]
Doc3 = [0.557, 0.000, 0.595, 0.580]

4. [*6 + 6 + 8 points*] An engineering company with 10,000 employees worldwide has recently developed an in-house prototype search system to replace an old Unix system. The search system is used mostly by engineers who need to find precise information in technical manuals quickly. The engineers also use the system to keep up to date with the latest developments in their specific fields and so occasionally browse through industry specific magazines and journals that the company has online access to. The information systems department of the company has asked you to evaluate the prototype search systems to ensure that it supports the needs of the engineers using it. One of the information systems specialists working for the company also asks you a number of questions about the evaluation of the prototype.

(a) They say that they have read in one of the information systems journals that commercial search engines often use A/B testing. They want to know if you are planning to use this method as well when evaluating their prototype system. Explain to them what A/B testing is and why it is appropriate/inappropriate for the problem at hand.

a) A/B testing (also known as split testing or bucket testing) is a process in which one shows two variations of the same thing (web page, search system, etc) to different groups of users at the same time (groups A and B, for variations A and B) and compare which version is more effective (= better satisfies the users, gives them better results, or whatever the relevant metric at stake is for what is being tested).

The users are assigned one of the two systems randomly and the results are analysed with "two-sample hypothesis testing" which determines whether the results (and the difference in the results) is statistically significant or not.

A/B testing would be appropriate for testing the new search system because we could compare it to the old search system and determine whether the new system is better than the old one. If it's not better (if it's worse or the same), there is no point in going through the expense of implementing it.

(b) They also say that they came across evaluation measures such as MAP and NDCG. They want to know how these two measures differ from more traditional measures of precision and recall and whether there is any advantage of using MAP and NDCG over precision and recall. They want to know if in your evaluation, you will be using any of the discussed methods (i.e. precision, recall, MAP, NDCG) and why/why not.

The basic metrics of any system are **Recall** and **Precision**. Our search system should balance both of them since there they are both important for our needs. Recall is extra important when browsing industry specific literature for keeping up with developments, and Precision is extra important when trying to find precise information in technical manuals.

$$Precision = \frac{\text{number of relevant documents retrieved}}{\text{number of documents retrieved}}$$

$$Recall = \frac{\text{number of relevant documents retrieved}}{\text{number of relevant documents}}$$

MAP stands for **M**ean **A**verage **P**recision. It is calculated over a set of many queries (a test collection) as the arithmetic mean of average precision values. Each query is assumed to have the same weight. MAP assumes that the user is interested in finding many relevant documents for each query and is one of the measures most often used in IR of research papers. It values Recall above precision.

- MAP would be useful to use in studying our new search system since one of the uses that it will be put through is to browse through industry-specific magazines and journals/ research papers.
- MAP should not be the only metric that we study in our new system. Even though we want good Recall when our engineers are keeping up to date with new industry developments, we also want good Precision when they are trying to find precise information in technical manuals very quickly.

NDCG stands for **N**ormalized **D**iscounted **C**umulative **G**ain. It is a measure of ranking quality, which is why NDCG is quite popular in evaluating Web search systems. People have no patience to scroll down long lists of results so the search results should be ordered by rank. Highly relevant documents should have better rank (=closer to the top). Highly relevant documents are more useful than marginally relevant documents, which are more useful than non-relevant documents – and their rank should reflect that.

- This would be also an important metric to study for our new search system. Our engineers have more leisure when browsing through industry literature for the latest developments, but when they need to find precise information in technical manuals they have a need for quick results and can't afford the time to be scrolling down big lists of results looking for what might be the most relevant ones.

(c) The information system specialist thinks it would be a good idea to test the new system versus the old system before deploying it throughout the whole company. The information systems specialist suggests you should conduct a user study using 5 or 6 of the company secretaries working in head office of the company and the results analysed using statistical testing. If the results are significant, then you can move on to deployment. Is the system specialist right? Explain you answer.

The information system specialist is both right and wrong. (continues next page)

He is correct in wanting to test the new system versus the old system. This could be done with A/B testing and the results analysed with "two-sample hypothesis testing" to see whether they are statistically significant or not. We would then determine whether the new system is better than the old one. If it's not better (if it's worse or the same), there is no point in going through the expense of deploying it.

He is wrong in his choice of test users. Our target users are engineers, so using the company secretaries as our test users would not work since they do not possess the specialized knowledge required to make relevance judgements of the results.