Universiteit Leiden

LIACS
Leiden Institute for Advanced Computer Science

Information Retrieval and Text Analytics

W. Kraaij and C. Veenman

Final Exam // ANSWERS

8 June 2017
B02
14.00-17.00

Student name:

Student number:

This exam consists of 5 pages and 10 questions for a total of 500 points. The grade will be computed as follows: ((points/50)*0.9 + 1)

Instructions:

- Carefully read the instructions and all exercises at the start of the exam.
- Write your name and student number on this hand-out and the answer sheets.
- Verify that your copy of this hand-out is complete and legible.

- Write your answers in a readable form.
- Always provide an explanation for your answer.

- This is a closed-book, closed-notes, individual exam.
- You are not allowed to use your laptop, smartphone or any photographing or telecommunication device. Only a simple calculator is allowed.
- You can work on this exam only within the allocated time-slot.
- Do not unstaple or tear off pages of this hand-out.
- Do not write your answers on this hand-out.
- You must return all pages of this hand-out to the proctor at the end of the exam, regardless of whether or not you have written anything on it.

1. Boolean retrieval and posting lists

a. Time is O(N) (where N is the total number of documents in the collection) assuming we need to return a complete list of all documents satisfying the query. This is because the length of the results list is only bounded by N, not by the length of the postings lists. (EXERCISE WEEK1) (20p)

b. $(L \vee A) \wedge \neg R$
$= (L \wedge \neg R) \vee (L \wedge \neg R)$
(15p) (EXERCISE WEEK1)

c. Standard posting list: 6 (5p) skip pointers: 5 (10p)

2. Stemming is an example of a term normalization function *Fn* generating equivalence classes.

a. dictionary size is reduced (15p)

b. improving recall (15p)

c. terms with same meaning in one class, terms with dfferent meaning in different classes (20p)

3. Query processing

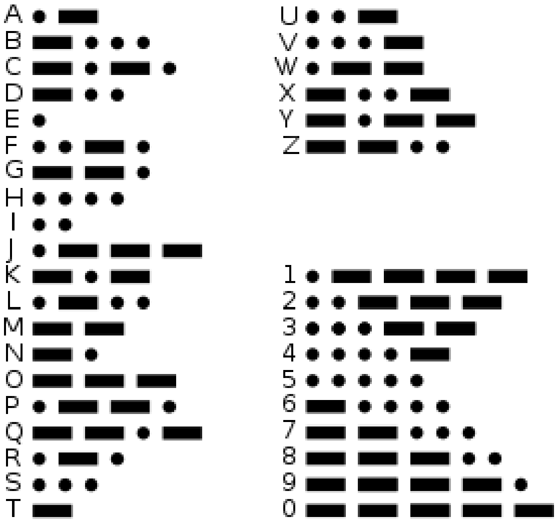
a. Hash tables: pro: O(1) lookup time; con: no support for pre_x queries, hash function may lead to collisions. BST: pro: support pre_x search; con: rebalancing necessary, slower search O(log(N)).
0% - completely wrong; 50% - half good; 100% - good (15p)

b. Jaccard coefficient equals 1/7 or 3/9 when a prefix and suffix character are added. (10p)

c. Levenshtein distance equals 4. Matrix can be derived using procedure on textbook page 54. Only correct answer without matrix yields 10 points. (25p) (NOTE 4 equals 2 inserts and 2 deletes, each with a cost of 1)

4. Consider the Morse code alphabet in the figure below

International Morse Code



a. Explain why the Morse code table has different code lengths for different characters. (10p)
Solution:
More frequent characters get shorter codes. This decreases the total length of the encoded
message.

b. Now consider that we replace all but the 5 most frequent letters by the symbol '\$'. The new alphabet consists thus of 6 symbols. Relative symbol frequencies are tabulated in the table below. What is the theoretically minimum average number of bits per symbol for the language modeled in the table? Please motivate. (20p)

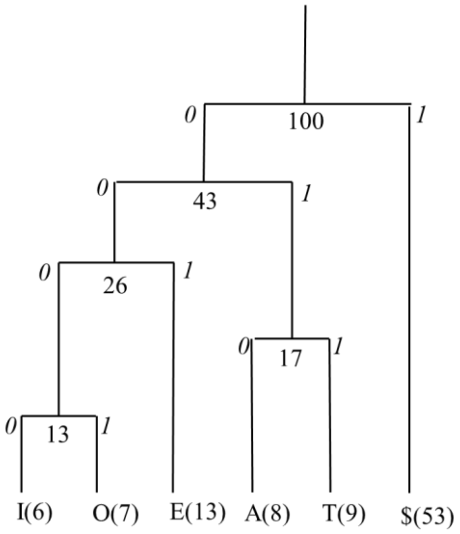
E	13%
T	9%
A	8%
O	7%
I	6%
$	57%

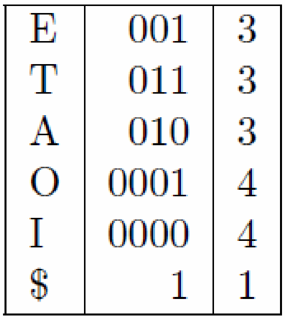
Solution:
Compute the entropy of the discrete probability distribution in table 4. Just mentioning entropy yields 8 points, showing knowledge of the definition of entropy given another 8, showing the full computation: $-(0.13\log_2 0.13 + \ldots)$ *or exact correct answer will qualify for the full 20. = 1.96 (1.36 is correct answer for natural logarithm)*

c. Construct a Huffman coding scheme for this reduced alphabet. What is the average number of bits per symbol used for a typical fragment of the language modeled in the above table, using your code Huffman code scheme? (20p)

Solution:

Huffman scheme 10p. Compute the weighted normalized sum of the product of the relative frequency and code length for each symbol = 1.99, 10p





E	001	3
T	011	3
A	010	3
O	0001	4
I	0000	4
$	1	1

5. Let *D* be a set of documents and *T* a set of terms.

   a. The tf-idf score of a term *t* within a document *d* is given by:

   $$\textit{tf-idf}_{t,d} = (1 + \log(tf_{t,d})) \times \log\left(\frac{N}{df_t}\right)$$

   Explain the name of this formula and explain the components, structure and intuition behind the formula. (25p)

   Solution:

   Measure increases with the number of occurrences (term frequency) within document and with the rarity of the term in the collection (inverse document frequency).

   b. The cosine-measure for document *d* and query *q* is given by:

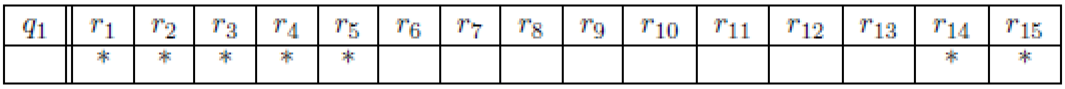
   $$sim(q,d) = \sum_{t \in T} q(t) \cdot d(t)$$

   Explain the name of this formula and explain the components, structure and intuition behind the formula. (25p)

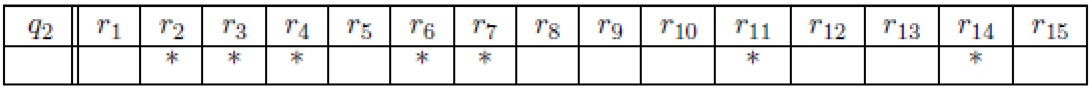
   Solution:

   The cosine between the query and document vector of t can be computed with their dot product. It measures the projected length of one vector on the other.

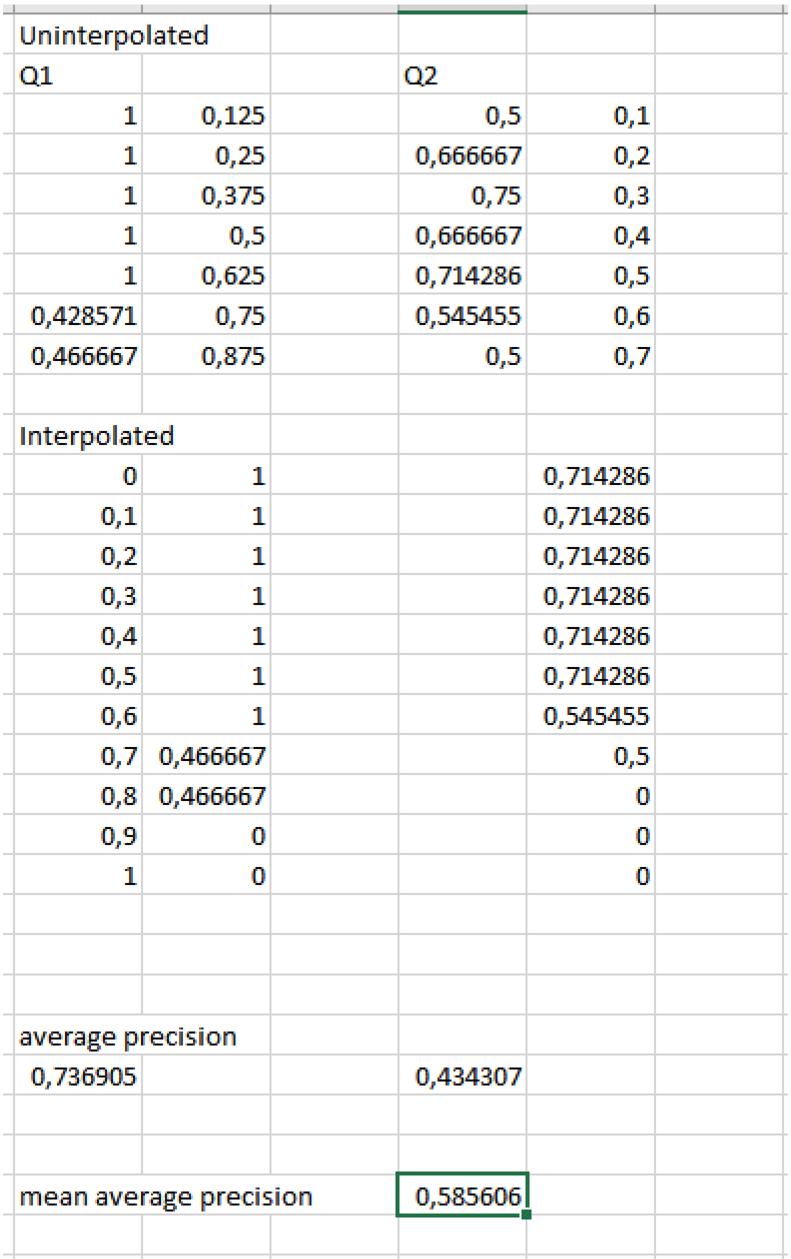
6. We assume a test collection consisting of 20 documents, two queries q_1 and q_2 and a set of relevance judgements. The following table shows the relevance judgements for the top 15 results for each query using system *S*. The '*' symbol indicates a document being relevant. There are 8 relevant documents in the result set for query q_1 and 10 for q_2.



q_1	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}	r_{11}	r_{12}	r_{13}	r_{14}	r_{15}
	*	*	*	*	*									*	*



q_2	r_1	r_2	r_3	r_4	r_5	r_6	r_7	r_8	r_9	r_{10}	r_{11}	r_{12}	r_{13}	r_{14}	r_{15}
		*	*	*		*	*				*			*	



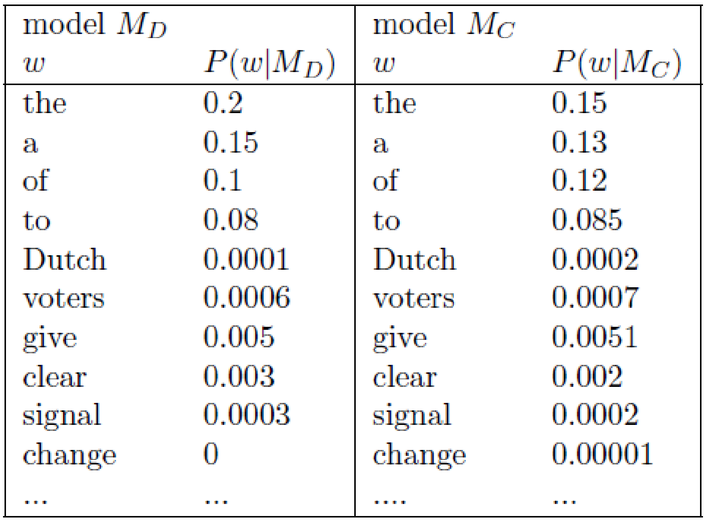
Uninterpolated				
Q1			Q2	
1	0,125		0,5	0,1
1	0,25		0,666667	0,2
1	0,375		0,75	0,3
1	0,5		0,666667	0,4
1	0,625		0,714286	0,5
0,428571	0,75		0,545455	0,6
0,466667	0,875		0,5	0,7
Interpolated				
0	1			0,714286
0,1	1			0,714286
0,2	1			0,714286
0,3	1			0,714286
0,4	1			0,714286
0,5	1			0,714286
0,6	1			0,545455
0,7	0,466667			0,5
0,8	0,466667			0
0,9	0			0
1	0			0
average precision				
0,736905			0,434307	
mean average precision			0,585606	

a. See page 146.

Correct PR graph: 25p or Correct argumenting: 25p

b. apply the definition of p146. Correct reasoning 10pt, correct answer 15pt (total 25)

7. Consider the following table of unigram conditional probabilities for a document *D* and a document collection *C*:



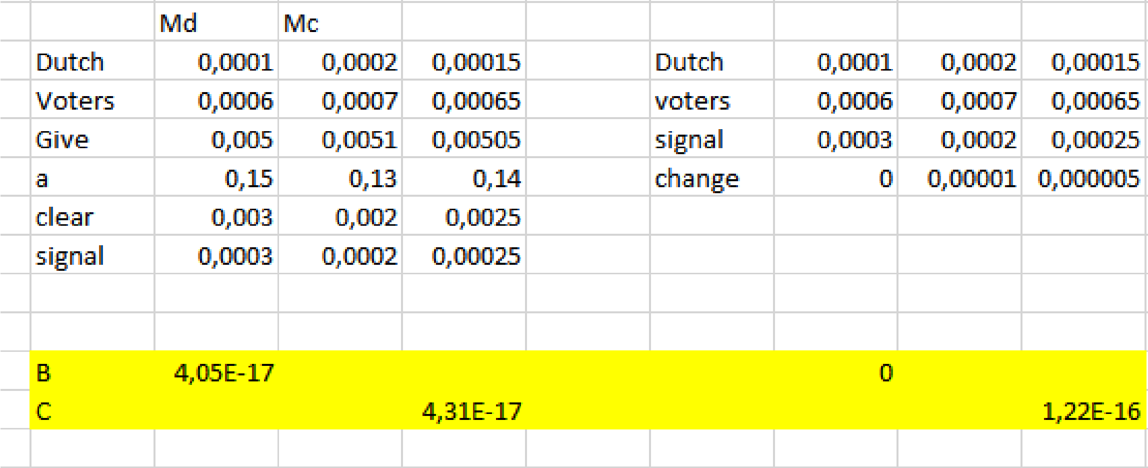
model M_D		model M_C	
w	$P(w\|M_D)$	w	$P(w\|M_C)$
the	0.2	the	0.15
a	0.15	a	0.13
of	0.1	of	0.12
to	0.08	to	0.085
Dutch	0.0001	Dutch	0.0002
voters	0.0006	voters	0.0007
give	0.005	give	0.0051
clear	0.003	clear	0.002
signal	0.0003	signal	0.0002
change	0	change	0.00001
...	...	....	...

Consider also two sentences: *s₁*: "Dutch voters give a clear signal." and *s₂*: "Dutch voters signal change."

a.
$$P(s_1) = P(Dutch) \cdot P(voters|Dutch) \cdot P(give|Dutch, voters) \cdot P(a|Dutch, voters, give) \cdot P(clear|Dutch, voters, give, a) \cdot P(signal|Dutch, voters, give, a, clear)$$
(15p)

b. Compute the generative probability (query likelihood) of s_1 and s_2 given the unigram model for *D*: $P(s_2|M_D)$ (see Table above).(20p)

c. Compute the generative probability of s_1 and s_2 where M_D is interpolated with background model M_C using interpolation parameter λ = 0.5 (15p)



	Md	Mc						
Dutch	0,0001	0,0002	0,00015		Dutch	0,0001	0,0002	0,00015
Voters	0,0006	0,0007	0,00065		voters	0,0006	0,0007	0,00065
Give	0,005	0,0051	0,00505		signal	0,0003	0,0002	0,00025
a	0,15	0,13	0,14		change	0	0,00001	0,000005
clear	0,003	0,002	0,0025					
signal	0,0003	0,0002	0,00025					
B	4,05E-17					0		
C			4,31E-17					1,22E-16

8. Text classification

a. Explain the name of the naive Bayes classifier. (15p)

Solution:

The answer should explain which assumptions are made (conditional independence).

Consider the following formula:

$$C_{map} = argmax_{c_i} \frac{P(c_i)P(x_1, \ldots, x_n|c_i)}{P(x_1, \ldots, x_n)}$$

b. How can we simplify it into a Naïve Bayes classifier? (20p)

Solution:

1) The denominator can be dropped, since it is independent of c_i 2) by considering conditional independence the enumerator $P(x_1,x_2..,x_n|c_i)$ can be decomposed into a product of separate probabilities $P(x_1|c_i)P(x_2|c_i)\ldots P(x_n|c_i)$.

c. Text collections have large numbers of features. Give two reasons why the large number of features can be a problem. (15p)
Solution:
Two reasons from 1) increases training and evaluation time 2) creates sensitivity for noise features 3) prone to overfitting

9. Consider the following formula:

$$P(p\,|\,q) = \frac{1}{N}\cdot d + \frac{\delta_{q\to p}}{O(q)}\cdot(1-d)$$

a. Explain the components and the structure of this formula, and the intuition behind this formula. (25p)
ANSWER: This is the probability that a random walk is taking place from node q to node p. The probability is a mixture of a random jump (the first component) and following an outlink (each node *q* has outdegree O(*q*) outlinks).

b. Why is it essential that the Markov chain describing the random walk process is ergodic? (25p) ANSWER: The Markov chain should be ergodic in order that the random walk process coverges to a stable PageRank value (stable probability of being visited).

10. Entities

a. Give examples of three different entity types and their types. (10p)
Solution:
Examples: Person: Mark Rutte, City: Leiden; phone number: 06-12345678

b. Give three fundamentally different ways to recognize entities in texts. (15p)
Solution:
1) lexicons 2) rules, regular expressions 3) machine learning with features vectors

Evaluation of named entity recognition systems

c. What are criteria for correctly identified named entities? (10p)
Solution:
1) Correct entity identification (boundaries)
2) Correct entity classification (types)

d. Give two measures that are typically used for evaluation of entity recognition systems, their formula and meaning. (15p)
Solution:
$Precision = \frac{|F\cap T|}{|F|};\ Recall = \frac{|F\cap T|}{|T|}$

F: entities identified
T: True entities