LBSC 796/INFM 718R: Information Retrieval Systems
Spring, 2006

Midterm Exam

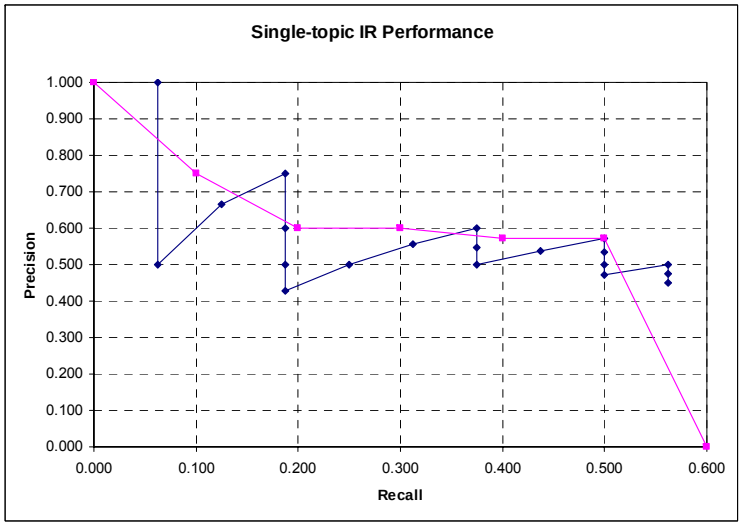
Name: _____

- Show sufficient work to demonstrate your understanding of the material. This is mostly for your benefit, as it will allow partial credit to be awarded.
- It might be a good idea to read over the entire exam first to get a sense of how to manage your time.
- You will have until 3:45pm to complete the exam.
- Good luck!

Question	Points	Total
1	16	16
2	8	8
3	28	28
4	12	12
5	18	18
6	18	18
Total	100	100

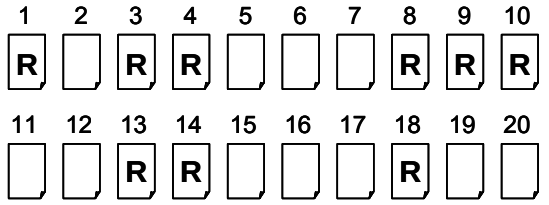
Question 1. Evaluation (16 points)

The following is the uninterpolated precision-recall graph of an IR system on a particular topic. You know that 20 hits were retrieved, and that there are 16 relevant documents for this topic (not all of which are retrieved).



A. (4 points) What does the interpolated graph look like? Draw neatly on the graph above.

B. (6 points) In the diagram below, each box represents a hit. Based on the above precision-recall graph, which hits are relevant? Write an "R" on the relevant hits. Leave the non-relevant hits alone.



C. (2 points) What is the Mean Average Precision (MAP)?

(1 + 2/3 + 3/4 + 4/8 + 5/9 + 6/10 + 7/13 + 8/14 + 9/18) / 16 ≈ 0.355

D. (2 points) What is the R-precision?

8/16 = 0.5

E. (2 points) What is Precision at 10?

6/10 = 0.6

Question 2. TREC Collections (8 points)

TREC-style evaluations with reusable test collections assume a model of the user and information-seeking behavior that is oversimplified and unrealistic. *Briefly* list two of these oversimplifications.

1. Focus on the IR black box; disregarding the interactive IR environment

2. Binary relevance judgments

3. Relevance judgments made independently on each document

Question 3. Boolean and Vector Space Retrieval (28 points)

Assume the following fragments comprise your document collection:

Doc 1: banking on banks to raise the interest rate
Doc 2: jogging along the river bank to look at the sailboats
Doc 3: jogging to the bank to look at the interest rate
Doc 4: buzzer-beating shot banked in!
Doc 5: scenic outlooks on the banks of the Potomac River

Assume that you drop stopwords.
Assume that you stem.

A. (5 points) Construct the term-document matrix for the above documents that can be used in Boolean retrieval. The index terms have already been arranged for you alphabetically in the following table (assume terms not mentioned to be stopwords):

Term	Doc 1	Doc 2	Doc 3	Doc 4	Doc 5
bank	1	1	1	1	1
buzzer-beating	0	0	0	1	0
interest	1	0	1	0	0
jog	0	1	1	0	0
look	0	1	1	0	0
outlook	0	0	0	0	1
Potomac	0	0	0	0	1
raise	1	0	0	0	0
rate	1	0	1	0	0
river	0	1	0	0	1
sailboat	0	1	0	0	0
scenic	0	0	0	0	1
shot	0	0	0	1	0

B. (2 points each) What documents would be returned in response to the following queries?

bank NOT interest

docs 2, 4, 5

(interest AND rate) NOT jog

doc 1

bank AND (scenic OR jog)

docs 2, 3, 5

Doc 1: banking on banks to raise the interest rate
Doc 2: jogging along the river bank to look at the sailboats
Doc 3: jogging to the bank to look at the interest rate
Doc 4: buzzer-beating shot banked in!
Doc 5: scenic outlooks on the banks of the Potomac River

C. (10 points) Construct the vector space term-document matrix for the above documents (repeated from before) using *tf.idf* term weighting. Normalize your vectors. The following blank tables are provided for your convenience. You can use as many or as few of them as you wish. Clearly indicate your final answer.

Term frequencies:

Term	idf	**Doc 1**	**Doc 2**	**Doc 3**	**Doc 4**	**Doc 5**
bank	0	2	1	1	1	1
buzzer-beating	.699				1	
interest	.398	1		1		
jog	.398		1	1		
look	.398		1	1		
outlook	.699					1
Potomac	.699					1
raise	.699	1				
rate	.398	1		1		
river	.398		1			1
sailboat	.699		1			
scenic	.699					1
shot	.699				1	

tf.idf (before length normalization)

Term	**Doc 1**	**Doc 2**	**Doc 3**	**Doc 4**	**Doc 5**
bank					
buzzer-beating				.699	
interest	.398		.398		
jog		.398	.398		
look		.398	.398		
outlook					.699
Potomac					.699
raise	.699				
rate	.398		.398		
river		.398			.398
sailboat		.699			
scenic					.699
shot				.699	
length	.897	.981	.795	.989	1.27

tf.idf (after length normalization)

Term	**Doc 1**	**Doc 2**	**Doc 3**	**Doc 4**	**Doc 5**
bank					
buzzer-beating				.706	
interest	.444		.500		
jog		.406	.500		
look		.406	.500		
outlook					.550
Potomac					.550
raise	.779				
rate	.444		.500		
river		.406			.313
sailboat		.713			
scenic					.550
shot				.706	

D. (3 points each) Simulate the retrieval of documents in response to the following queries. Indicate the order in which documents will be retrieved, and the similarity score between the query and each document.

bank interest

(note that I didn't bother to normalize the query vectors)

doc 3: 0.500
doc 1: 0.444
doc 2: 0
doc 4: 0
doc 5: 0

bank river

(note that I didn't bother to normalize the query vectors)

doc 2: 0.406
doc 5: 0.313
doc 1: 0
doc 3: 0
doc 4: 0

E. (1 points) In two sentences or less, describe why indexing word senses does not yield higher retrieval performance.

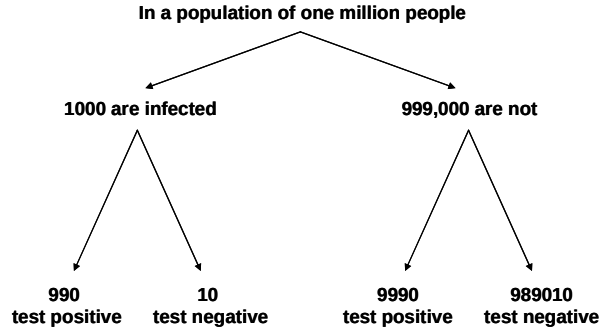
Word-sense disambiguation techniques are far from perfect, and queries often have a built-in disambiguation effect already.

Question 4. The dreaded disease… again (12 points)

In lecture 4, we talked about a hypothetical disease that's very rare, but there's a very accurate test for it. By Bayes' Rule, we computed the probability that you actually have the disease, given that you tested positive. This question explores this problem in more detail. Find the probability that you have the disease, given that you tested positive under the following two slightly different scenarios. The goal is for you to gain a better intuition of how probabilities interact.

A. (6 points)
The incidence of the disease: 0.1%
Accuracy of the test: 99%

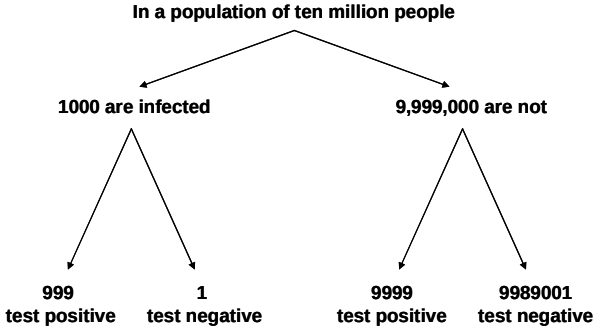
See following analysis:



Out of 10980 (990+9990) people that test positive, only 990 have it. Therefore, the probability that you actually have the disease, given that you tested positive, is 990/10980 ≈ 9.0%.

B. (6 points)
The incidence of the disease: 0.01%
Accuracy of the test: 99.9%

See following analysis:



Out of 10998 (999+9999) people that test positive, only 999 have it. Therefore, the probability that you actually have the disease, given that you tested positive, is 999/10998 ≈ 9.1%.

Question 5. The guessing game (18 points)

Assume we're playing a game in which I think of a number between 1 and 1000, and you have to guess that number in as few attempts as possible. With each guess, you can also ask me a yes/no question about the number and I'll answer truthfully.

To reiterate: each turn consists of you guessing a number, asking me a yes/no question, and receiving the answer. The goal is to guess the number I'm thinking of in as few turns as possible.

A. (6 points) Come up with a linear-time algorithm for guessing this number. By algorithm, I mean a strategy of what number you would guess first, guess next, etc., and what you would ask.

Is it one? Is it two? Is it three? …

B. (6 points) Come up with a logarithmic-time algorithm for guessing this number. By algorithm, I mean a strategy of what number you would guess first, guess next, etc., and what you would ask.

Is it less than or equal to 500?
    If so, ask: is it less than or equal to 250?
    Otherwise, ask: is it less than or equal to 750?

Essentially, perform binary search on the range between 1 and 1000.

C. (3 points) In the worst case, how many guesses would it take to figure out the number using the linear-time algorithm?

In the worst case, you have to go through all numbers, i.e., 1000 tries.

D. (3 points) In the worst case, approximately how many times guesses would it take to figure out the number using the logarithmic-time algorithm?

$\lg_2 1000 \approx 10$

Question 6. Language Processing and IR (18 points)

Three linguistic phenomena are listed below. For each,

- In ten words or less, describe what it is. (1 point)
- Give an original example, not presented in the lecture slides. (1 point)
- In two sentences or less, describe why it poses a problem for IR systems (2 points)
- Does it cause *primarily* precision or recall problems? (2 points)

Morphological Variation
What is it? Different forms of the same word.

Example: computer, computers, computerize

Why is it a problem? Search for one term won't retrieve all variants.

Precision or recall problem? recall

Synonymy
What is it? Different words, same meaning.

Example: lift, elevator

Why is it a problem? Search for one term won't retrieve all synonyms.

Precision or recall problem? recall

Polysemy
What is it? Same word, different meanings.

Example: lift (to lift), lift (elevator)

Why is it a problem? Search for one term will bring back documents in which the term means something different.

Precision or recall problem? precision