

Exercises

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|

Surname, First name

Information Retrieval 1 (52041INR6Y)

Exam 1

|   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|
|   |   |   |   |   |   |   |   |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 3 | 3 | 3 | 3 | 3 | 3 | 3 | 3 |
| 4 | 4 | 4 | 4 | 4 | 4 | 4 | 4 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 |
| 7 | 7 | 7 | 7 | 7 | 7 | 7 | 7 |
| 8 | 8 | 8 | 8 | 8 | 8 | 8 | 8 |
| 9 | 9 | 9 | 9 | 9 | 9 | 9 | 9 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

Pay attention to the following instructions:

- Write your name and student number on the front page. Don't forget to mark the digits!
- Write all your answers on this exam booklet.
- Write your answers inside the boxes provided. It's okay to slightly go outside the margins.
- The answer boxes should be large enough for your answer.
- If you need to clear an answer box in order to start over, ask the invigilator for a blank sticker

1.5p

**1a** Consider the following two rankings with binary relevance labels. Both rankers returned 8 documents

| Relevance of Ranking A | Relevance of Ranking B |
|------------------------|------------------------|
| <u>1</u>               | 0                      |
| 1                      | 0                      |
| 1                      | 1                      |
| 0                      | 0                      |
| 0                      | 1                      |
| 0                      | 1                      |
| 1                      | 1                      |
| 0                      | 0                      |

Considering these rankings, do you think precision@K would be a good evaluation metric to decide which ranker outperforms the other? What about recall@K? (Consider different K to illustrate your answer)

[illegible]

**1b** Given the above ranking, what are two ranking metrics that would be preferable to precision@K? Why would these metrics be preferable?

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |

**1c** In some search scenarios high recall is required, whereas other settings favor precision. Suppose you have an existing ranking system that uses BM25 or QL (you may assume either). What is one way this ranking system could be adapted to increase precision? What is one way to adapt it to increase recall? Your approach may modify any part of the ranking system. Make sure to explain why your approach should have the desired effect (i.e., increasing precision or recall).

[illegible]

- 1p **1d** Name two new scenarios that could correspond to the requirements given in the previous question, and explain why these scenarios are a good fit. Full credit will only be awarded for new scenarios that were not discussed in lectures/Q&A.

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |

### Document representation and matching

- 2p **2a** Given a query containing tokens  $T_1, \dots, T_n$ , the score for a given document  $D$  could be computed as:  $P(T_1, \dots, T_n|D) = \prod_{i=1}^n P(T_i|D)$ . In what scenario would this naive approach return a score of 0 for a relevant document that includes query token  $T_1$ ?

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |

- 2p **2b** Propose a modified QL formula that fixes the above problem in a principled way. Are there any scenarios in which your new formula would still incorrectly return a score of 0? Explain why or why not. If this can still happen, what changes or assumptions would be needed to fix this problem completely?

|  |
|--|
|  |
|  |
|  |
|  |
|  |



Semantic matching methods

1p 3a For two-stage ranking, we often choose term-based methods for first-stage retrieval and semantic models for reranking. Why? Make sure to explain your answer.

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |

1p 3b Consider the reverse approach of using semantic models for initial retrieval and term-based methods for reranking. Can this be a reasonable approach as well? Why or why not?

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |

3p 3c You are developing a "query by document" patent search system in which the user uploads a new, under-development patent to serve as the query and then the system returns existing related documents (patents). Your system's goal is to return all existing patents that are related to the user's query document (the new patent). These patent documents are long and complex.

Your existing system uses BM25. Your boss would like to augment this with a Average Word Embedding (AWE) semantic matching method that reranks the top 25 patents returned, while preserving the BM25 ranking for positions 26-end. Compared to the existing system, what are the advantages and disadvantages of this approach?

|  |
|--|
|  |
|  |
|  |
|  |

[illegible]

## Offline LTR

2p

**4a** Name 3 categories of features that could be used as input representations (feature vectors) for a Learning to Rank (LTR) system. Give one example of a feature for each category.

[illegible]

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |

[illegible]



## Counterfactual LTR

2p

**5a** Compare and contrast counterfactual learning to rank (LTR) with LTR using manual relevance labels. Your answer should describe their trade-offs.

[illegible]

**5b** Let's assume you want to estimate position bias for your counterfactual LTR model using the RandTop-n algorithm. You randomly shuffled documents into three rankings and collected three user clicks for each of them, which are displayed below. (1 means the user clicked on the document and 0 means they didn't.) Based on the clicks, compute the propensities  $p_i$  for each rank  $i$ . Make sure to explain your approach; answers that contain calculations without any explanation will receive no credit.

| Ranking 1 |            |              | Ranking 2 |            |              | Ranking 3 |            |              |
|-----------|------------|--------------|-----------|------------|--------------|-----------|------------|--------------|
| Rank      | Document   | Click logs 1 | Rank      | Document   | Click logs 2 | Rank      | Document   | Click logs 3 |
| 1         | Document 1 | [1, 0, 1]    | 1         | Document 2 | [1, 0, 1]    | 1         | Document 3 | [1, 1, 1]    |
| 2         | Document 2 | [1, 0, 1]    | 2         | Document 4 | [1, 1, 0]    | 2         | Document 1 | [0, 1, 1]    |
| 4         | Document 4 | [1, 0, 0]    | 4         | Document 1 | [1, 0, 1]    | 4         | Document 4 | [0, 0, 1]    |
| 3         | Document 3 | [0, 0, 0]    | 3         | Document 3 | [0, 1, 0]    | 3         | Document 2 | [1, 0, 0]    |

[illegible]

2p


- 5c** In the previous part, you calculated the propensities  $p_i$  with the help of an intervention (i.e., randomizing the results displayed by a running system). Without the help of an eye tracker, how could these propensities be computed offline? (with no intervention)

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

2p

- 5d** While you were away on vacation, your boss purchased an eye tracker and used it to run a large user study in which participants were recorded issuing queries and navigating to the relevant results. How could this data be used to replace the propensities calculated in the previous question? What are the trade-offs between this approach and the approach you proposed in the previous question?

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |



|  |
|--|
|  |
|  |
|  |
|  |

**Inverse Propensity Scoring**

2p **6a** Different query intents can imply different click behaviors in a SERP.  
For example, a *navigational query* is a search with the intent of finding a specific webpage (e.g., UvA MS AI program homepage), whereas *informational queries* cover a broad topic (e.g., trucks) for which there may be thousands of relevant results.

Compare navigational and informational queries in terms of the expected number of clicks.

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |

3p **6b** If we have two different IPS (inverse propensity scoring) models for unbiased LTR, one for navigational and the other for informational queries, what would be the difference between the propensities in a position-based model (PBM) used by these two IPS models? (That is, assume the propensities are calculated with PBM and explain how the propensities would differ between the two models.)

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |

**6c** A naive LTR refers to the case where the clicks are used as relevance signals, without IPS correction. Compare the ranking performance of naive LTR models, learned on navigational and informational queries.

|  |
|--|
|  |
|  |
|  |
|  |
|  |
|  |
|  |

## Online LTR

**7a** In online LTR, we work with user interactions such as clicks. The first step to running an online LTR experiment is simulating users' behaviors using and producing clicks based on a "dependent click model". ("Dependent" here simply means that this is the click model you are using in your simulation.) It is then possible to use simulated clicks instead of the relevance labels for training the LTR algorithm.

Can you propose an algorithm to simulate users clicking?

**Hint:** define the probability of user click and stop.

[illegible]



|  |  |
|--|--|
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

2p

**8b** Imagine a conversational search system in which the system sometimes has the initiative. What are two different actions the system might take? What are the potential dangers of taking these actions?

[illegible]

This page is left blank intentionally