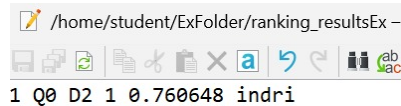


# Information Retrieval - HW2

## Part A

1. (a) One document was retrieved — **D2**.

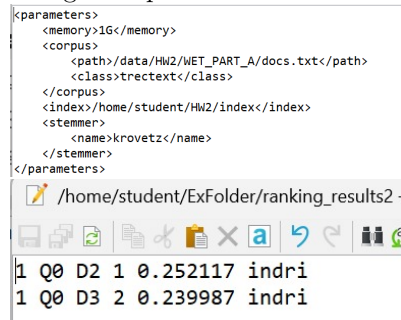


```
/home/student/ExFolder/ranking_resultsEx -
1 Q0 D2 1 0.760648 indri
```

- (b) There might be other documents relevant to the query, but only one document contained the word *corporation* (D2), while another document (D3) contained the word *corporations*, which was not retrieved even though it likely refers to the same concept.

To overcome this issue — as suggested in section 5 of the theoretical part — we need to apply stemming, which reduces word variants to their base forms. In this case, *corporations* would be stemmed to *corporation*, and D3 would be retrieved as expected.

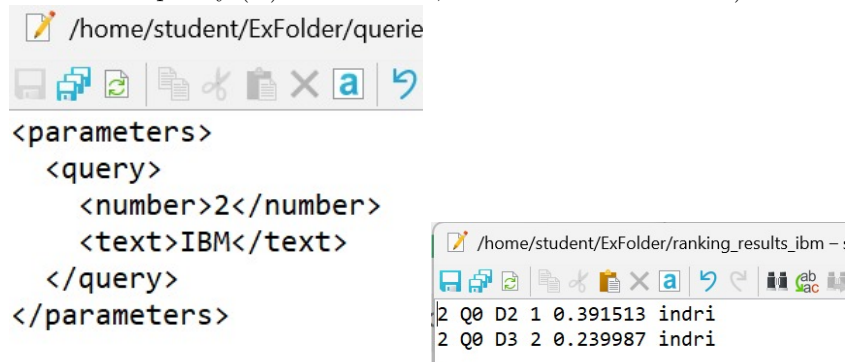
Using the updated index with stemming, we retrieved both documents as expected.



```
<parameters>
<memory>1G</memory>
<corpus>
  <path>/data/HW2/HW2_PART_A/docs.txt</path>
  <class>trectext</class>
</corpus>
<index>/home/student/HW2/index</index>
<stemmer>
  <name>krovetz</name>
</stemmer>
</parameters>
/home/student/ExFolder/ranking_results2 -
1 Q0 D2 1 0.252117 indri
1 Q0 D3 2 0.239987 indri
```

2. We noticed that the word *IBM* appears in only two documents — D2 and D3.

Given that the tf-idf scoring is based on term frequency, and assuming that the idf is the same in both cases, the document that contains the word more frequently (D2) will receive a higher score. (Only the term frequency (tf) matters here, since the idf is identical.)



```
/home/student/ExFolder/querie
<parameters>
  <query>
    <number>2</number>
    <text>IBM</text>
  </query>
</parameters>
/home/student/ExFolder/ranking_results_ibm -
2 Q0 D2 1 0.391513 indri
2 Q0 D3 2 0.239987 indri
```

3. For the same reasons we provided in the previous question, and considering that the documents were indexed without stopword removal, and that stopwords are the only words shared between document

D1 and the other documents in the corpus, we ran the query "in" and obtained the following results:

```
<parameters>
```

```
<query>
```

```
<number>3</number>
```

```
<text>in</text>
```

```
</query>
```

```
</parameters>
```

```
3 Q0 D1 1 0.102134 indri
3 Q0 D4 2 0.0814975 indri
3 Q0 D3 3 0.063545 indri
```

4. (a) Document D4 is **not relevant**, even though it contains the phrase *Michael Jackson*. A user searching for "Michael Jackson" is likely interested in his life and career. However, in D4, Michael Jackson is only mentioned in passing, in a sentence about Lady Gaga. So D4 is not truly relevant.
- (b) Given that D4 is about Lady Gaga, it makes sense to run the query "*Lady Gaga*".

The results were as expected:

- Score for query *Michael Jackson*: 1.36686
- Score for query *Lady Gaga*: 1.85722

```
<parameters>
```

```
<query>
```

```
<number>4</number>
```

```
<text>Lady Gaga</text>
```

```
</query>
```

```
</parameters>
```

```
/home/student/ExFolder/ranking_results
```

```
/home/student/ExFolder/ranking_resu
```

```
4 Q0 D4 1 1.85722 indri
```

```
4 Q0 D4 1 1.36686 indri
```

## Part B

Stopword Removal	Stemmer	P@5	P@10	MAP
with	with	0.3919	0.3725	0.2113
with	without	0.3933	0.3658	0.1860

Table 1: Evaluation results using trec\_eval

### Analysis:

The retrieval method **with stemming** produced **better MAP scores**.

This is expected: stemming reduces word variation by returning words to their base form. This increases the number of documents that might be matched for a given query.

The increase in **retrieved relevant documents** leads to a higher **recall**, and therefore, to a **higher MAP**.

Thus, the better performance of the method using stemming is logical and expected.