DATA MINING TECHNOLOGY FOR BUSINESS AND SOCIETY HW1

Cecilia Martinez Oliva, Giorgio Giannone

Contents

Introduction

This work uses *The Cranfield Collection* to perform a simple Information Retrieval analysis. Collected in the United Kingdom starting in the late 1950s, it contains 1398 abstracts of aerodynamics journal articles, a set of 225 queries and exhaustive relevance judgments of all (query, document) pairs.[1]

1. A simple statistic on the used dataset

1400 documents and *225* queries (3 of these queries return always zero results)

```
more cran_defaultStemmer-title.properties

documents=1400
terms=1800
postings=15742
maxcount=6
indexclass=it.unimi.di.big.mg4j.index.
    ↪ QuasiSuccinctIndex
skipquantum=256
byteorder=LITTLE_ENDIAN
termprocessor=it.unimi.di.big.mg4j.index.
    ↪ DowncaseTermProcessor
batches=1
field=title
size=228551
maxdocsize=43
occurrences=16619
```

2. A list of used stemmers

Three different Stemming methods:

- *Default Stemmer*
- *Stemmer English language*
- *Stemmer English language filter stopwords*

3. A list of used scorer functions

Three different Scoring functions:

- *Count occurrences*
- *Tf-Idf*
- *bm25* (this is a statistical improvement of the normal tf-idf scoring function)

4. Collection, 5. Inverted indexes, 6. Results search engine

Script to create the collections, the inverted indexes (for all stemmers) and to obtain the results from the search engine (for all scorer function)

```
#!/bin/bash

printf "Start!\n"
sleep 1s

mkdir output
cd output
mkdir defaultStemmer
mkdir englishStemmer
mkdir englishStopwords
cd ..


printf "\n create collections \n"
sleep 1s
#1 Create a collection on the set of html documents with MG4J.

find ./Cranfield_DATASET/Cranfield_DATASET/cran -iname \*.html
    ↪ | java it.unimi.di.big.mg4j.document.
    ↪ FileSetDocumentCollection -f HtmlDocumentFactory -p
    ↪ encoding=UTF-8 cran_defaultStemmer.collection

find ./Cranfield_DATASET/Cranfield_DATASET/cran -iname \*.html
    ↪ | java it.unimi.di.big.mg4j.document.
    ↪ FileSetDocumentCollection -f HtmlDocumentFactory -p
    ↪ encoding=UTF-8 cran_englishStemmer.collection

find ./Cranfield_DATASET/Cranfield_DATASET/cran -iname \*.html
    ↪ | java it.unimi.di.big.mg4j.document.
    ↪ FileSetDocumentCollection -f HtmlDocumentFactory -p
    ↪ encoding=UTF-8 cran_englishStopwords.collection
```

```
#2 Create an inverted index (with MG4J) on the collection
   ↪ trying different stemming methods:

#a. default stemmer,
#b. English stemmer  and
#c. English stemmer able to filter stopwords.


printf "\n create inverted indices \n"
sleep 1s

java it.unimi.di.big.mg4j.tool.IndexBuilder --downcase -S
   ↪ cran_defaultStemmer.collection cran_defaultStemmer

java it.unimi.di.big.mg4j.tool.IndexBuilder -t it.unimi.di.big.
   ↪ mg4j.index.snowball.EnglishStemmer -S cran_englishStemmer
   ↪ .collection cran_englishStemmer

java it.unimi.di.big.mg4j.tool.IndexBuilder -t homework.
   ↪ EnglishStemmerStopwords -S cran_englishStopwords.
   ↪ collection cran_englishStopwords

#3 Obtain results for each query using the software " homework.
   ↪ RunAllQueries_HW " trying different scorer functions:
   ↪ CountScorer, TfIdfScorer and BM25Scorer.


printf "\n run queries on cran_defaultStemmer \n"
sleep 1s

#cran1
java homework.RunAllQueries_HW "cran_defaultStemmer" ./
   ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
   ↪ "CountScorer" "text_and_title" ./output/defaultStemmer/
   ↪ output_cran_defaultStemmer_count.tsv

java homework.RunAllQueries_HW "cran_defaultStemmer" ./
   ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
   ↪ "TfIdfScorer" "text_and_title" ./output/defaultStemmer/
   ↪ output_cran_defaultStemmer_tfidf.tsv

java homework.RunAllQueries_HW "cran_defaultStemmer" ./
```

```
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "BM25Scorer" "text_and_title" ./output/defaultStemmer/
    ↪ output_cran_defaultStemmer_bm25.tsv

printf "\n run queries on cran_englishStemmer \n"
sleep 1s

#cran2
java homework.RunAllQueries_HW "cran_englishStemmer" ./
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "CountScorer" "text_and_title" ./output/englishStemmer/
    ↪ output_cran_englishStemmer_count.tsv

java homework.RunAllQueries_HW "cran_englishStemmer" ./
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "TfIdfScorer" "text_and_title" ./output/englishStemmer/
    ↪ output_cran_englishStemmer_tfidf.tsv

java homework.RunAllQueries_HW "cran_englishStemmer" ./
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "BM25Scorer" "text_and_title" ./output/englishStemmer/
    ↪ output_cran_englishStemmer_bm25.tsv

printf "\n run queries on cran_englishStopwords \n"
sleep 1s

#cran3
java homework.RunAllQueries_HW "cran_englishStopwords" ./
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "CountScorer" "text_and_title" ./output/englishStopwords/
    ↪ output_cran_englishStopwords_count.tsv

java homework.RunAllQueries_HW "cran_englishStopwords" ./
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "TfIdfScorer" "text_and_title" ./output/englishStopwords/
    ↪ output_cran_englishStopwords_tfidf.tsv

java homework.RunAllQueries_HW "cran_englishStopwords" ./
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "BM25Scorer" "text_and_title" ./output/englishStopwords/
    ↪ output_cran_englishStopwords_bm25.tsv
```

```
printf "\n run queries on cran_englishStopwords BM25Scorer for
    ↪ title \n"
sleep 1s

java homework.RunAllQueries_HW "cran_englishStopwords" ./
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "BM25Scorer" "title" ./output/
    ↪ output_cran_englishStopwords_bm25_title.tsv


printf "\n run queries on cran_englishStopwords BM25Scorer for
    ↪ text \n"
sleep 1s

java homework.RunAllQueries_HW "cran_englishStopwords" ./
    ↪ Cranfield_DATASET/Cranfield_DATASET/cran_all_queries.tsv
    ↪ "BM25Scorer" "text" ./output/
    ↪ output_cran_englishStopwords_bm25_text.tsv


printf "All Done :)"
```

7. The Average R-Precision

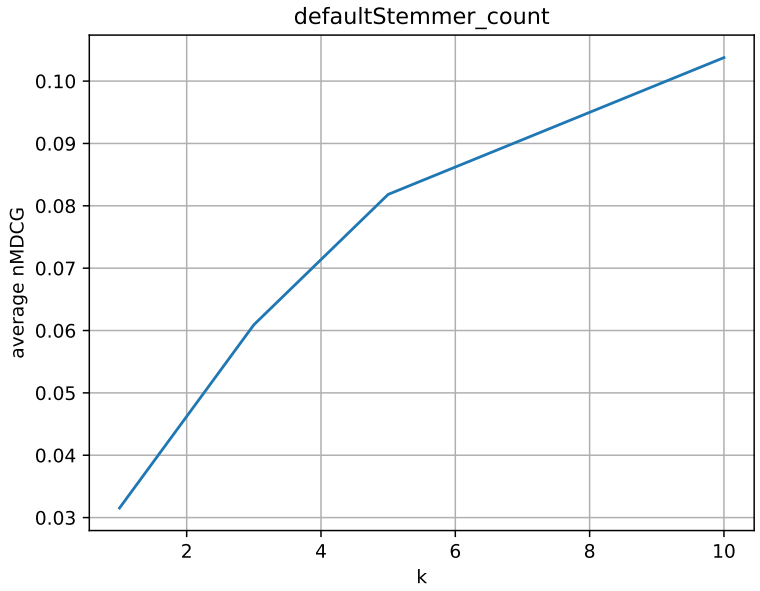
The Average R-Precision for each stemmer scorer function configuration(9) and for the aggregation algorithm(3)

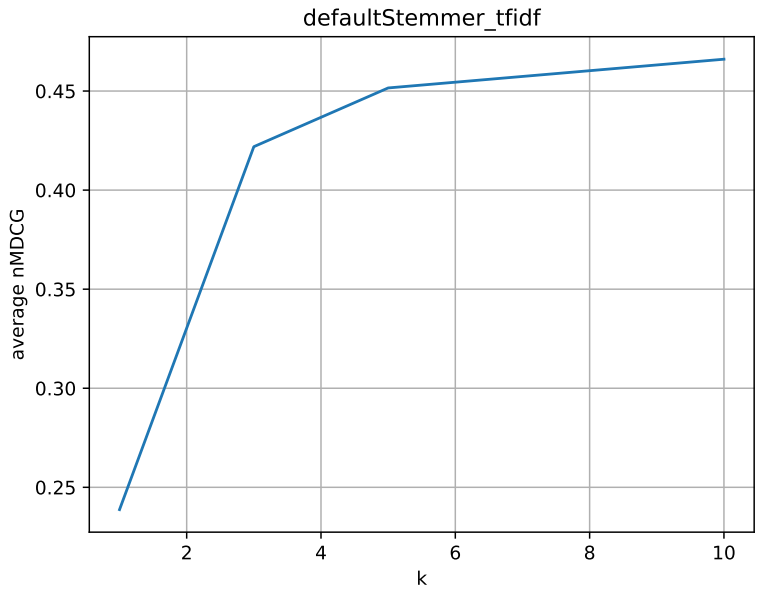
```
 R
defaultStemmer_count
averaged R-precision:   0.0234


defaultStemmer_tfidf
averaged R-precision:   0.1793


defaultStemmer_bm25
averaged R-precision:   0.2549
```

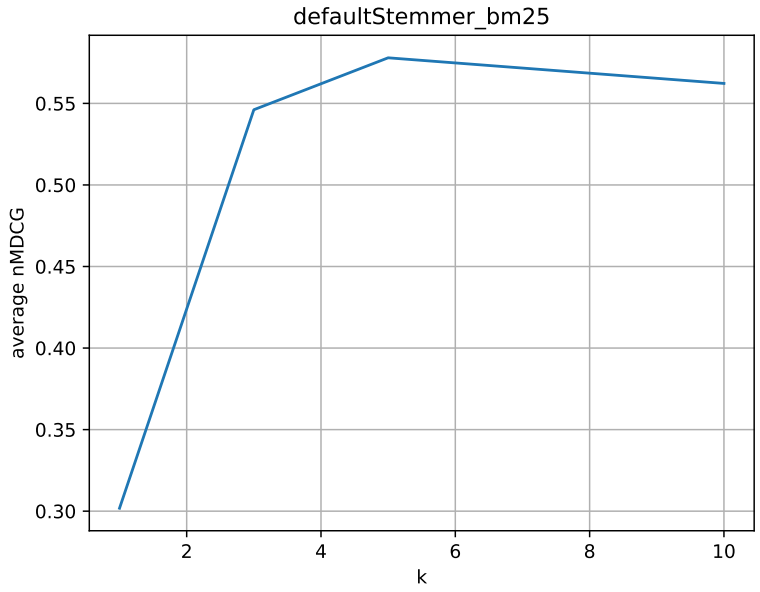
```
englishStemmer_count
averaged R-precision:   0.0266


englishStemmer_tfidf
averaged R-precision:   0.1894


englishStemmer_bm25
averaged R-precision:   0.2624


englishStopwords_count
averaged R-precision:   0.1577


englishStopwords_tfidf
averaged R-precision:   0.1909


englishStopwords_bm25
averaged R-precision:   0.2665



englishStopwords_bm25_title
averaged R-precision:   0.2211


englishStopwords_bm25_text
averaged R-precision:   0.2515


englishStopwords_bm25_title_text
averaged R-precision:   0.248
```

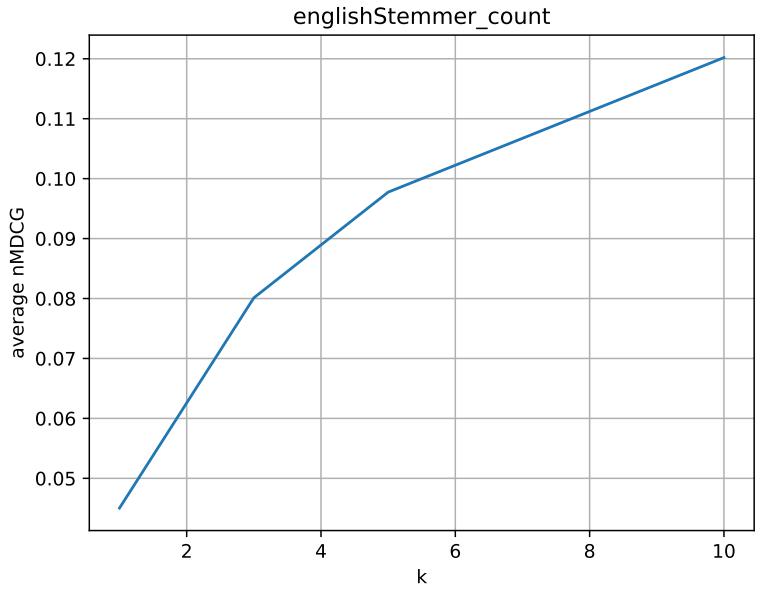
8. The plot of the average nMDCG

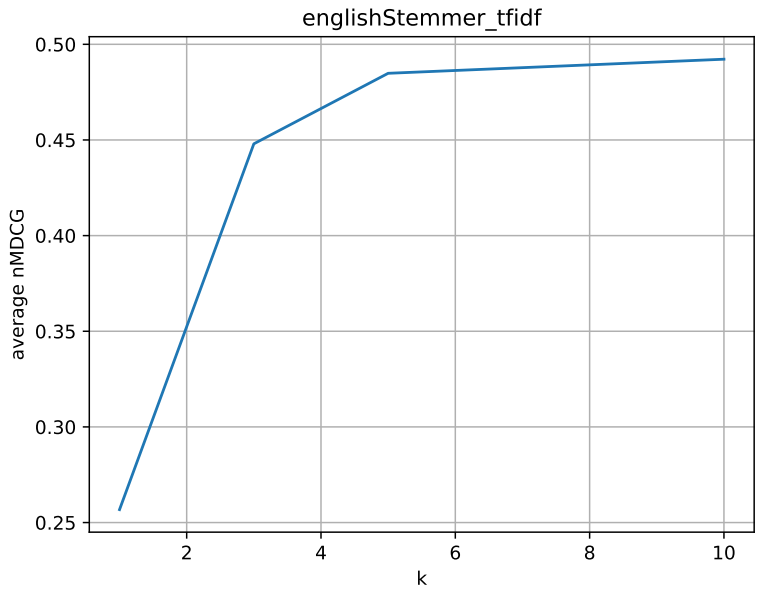
Plot of the average nMDCG for each stemmer scorer function configuration (next page)

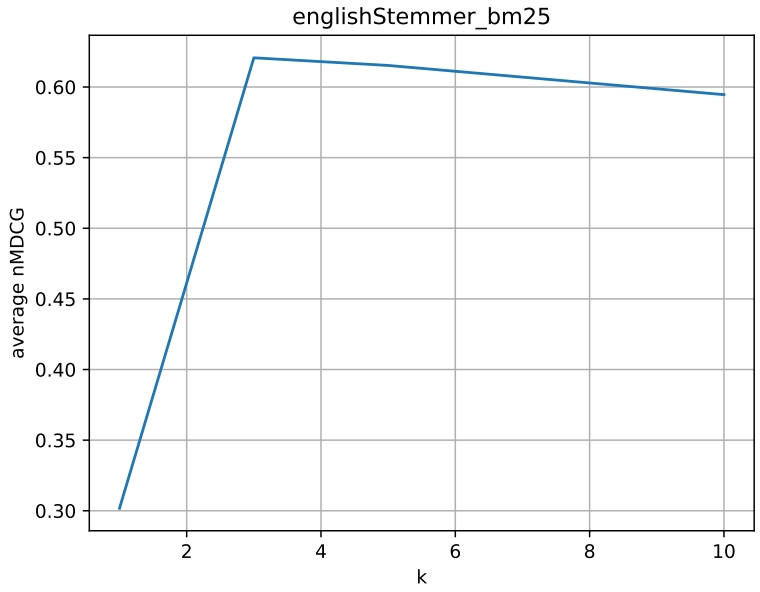

defaultStemmer_count
average nMDCG
0.10
0.09
0.08
0.07
0.06
0.05
0.04
0.03
2
4
6
8
10
k

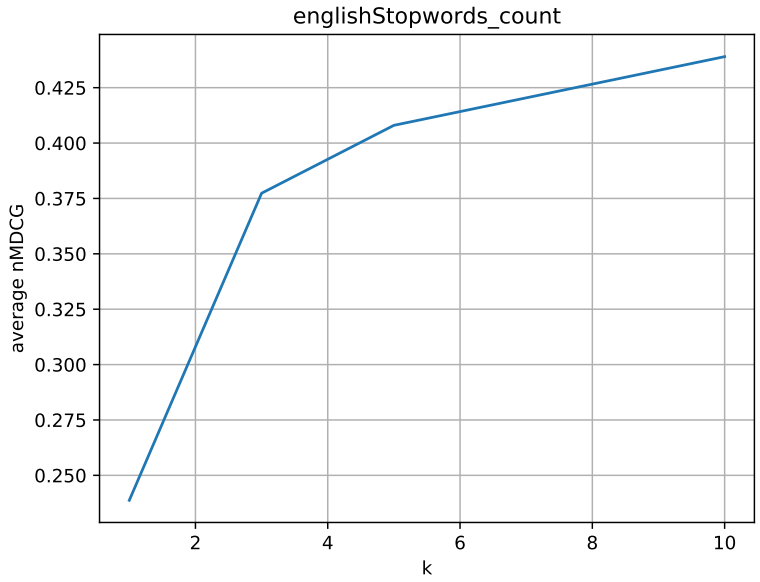

defaultStemmer_tfidf
average nMDCG
0.45
0.40
0.35
0.30
0.25
2
4
6
8
10
k

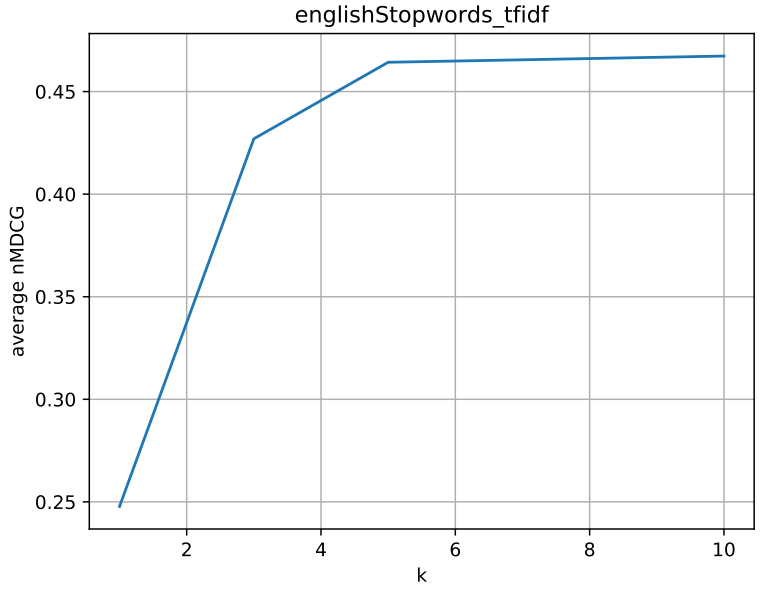

defaultStemmer_bm25
average nMDCG
0.55
0.50
0.45
0.40
0.35
0.30
2
4
6
8
10
k

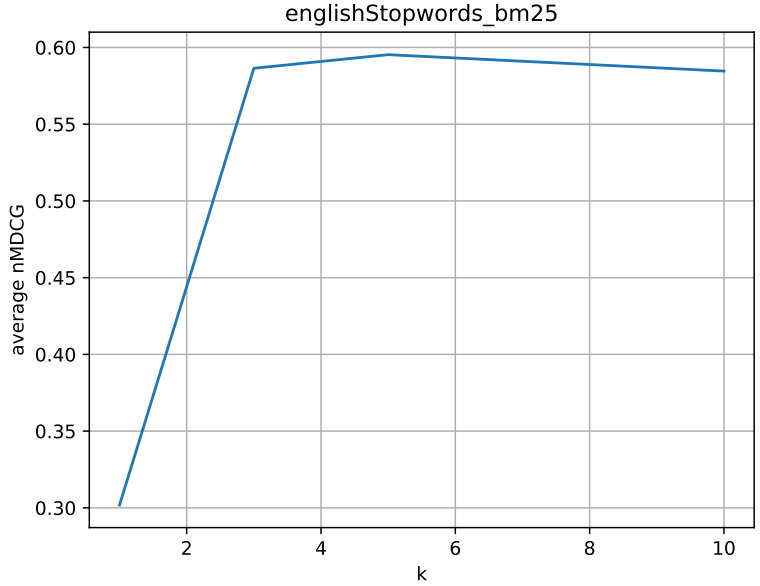

englishStemmer_count
average nMDCG
0.12
0.11
0.10
0.09
0.08
0.07
0.06
0.05
2
4
6
8
10
k


englishStemmer_tfidf
average nMDCG
0.50
0.45
0.40
0.35
0.30
0.25
2
4
6
8
10
k


englishStemmer_bm25
average nMDCG
0.60
0.55
0.50
0.45
0.40
0.35
0.30
2
4
6
8
10
k


englishStopwords_count
average nMDCG
0.425
0.400
0.375
0.350
0.325
0.300
0.275
0.250
2
4
6
8
10
k


englishStopwords_tfidf
average nMDCG
0.45
0.40
0.35
0.30
0.25
2
4
6
8
10
k


englishStopwords_bm25
average nMDCG
0.60
0.55
0.50
0.45
0.40
0.35
0.30
2
4
6
8
10
k

9. Q/A

1. Which is the best combination stemmer scorer function?
    - Using the average R-precision the best combination is englishStopwords as Stemmer and bm25 as scoring function (0.2665)
2. Which is the best stemmer?
    - Using the average R-precision the best stemmer is englishStopwords
3. Which is the best scorer function?
    - Using the average R-precision the best scoring function is bm25

Conclusion

The R-precision metric gives importance to the relevant results in general; instead the nMDCG gives importance also to the position of these relevant results. Comparing the results obtained we see some differences: for the nMDCG metric the best combination stemmer/scorer is *englishStemmer-bm25*; instead for the R-precision is *englishStopwords-bm25*. Using this information we can suppose that the englishStemmer returns less relevant results but with a better position; instead the englishStopwords returns more relevant results but less ordered.
The best average nMDCG is obtained using a k = 5: with k = 10 we have more relevant results but also a higher possibility to have a not correct position; instead for k = 1 we are constrained to have in the first place a relevant result or the metric will be zero.
A worst score (0.248) is obtained, in terms of R-precision, using the aggregation algorithm than using only the text score (0.2515): the aggregation algorithm tends to create a mean between the two ranking and, because one is not so good in terms of relevance(title: 0.2211), the final result using text and title will be a little worst than the result obtained considering only the best rank(in this case the text rank).

```
nMDCG

defaultStemmer_count
k:  1
averaged nMDCG:  0.0315
k:  3
averaged nMDCG:  0.0609
k:  5
averaged nMDCG:  0.0818
k:  10
averaged nMDCG:  0.1038


defaultStemmer_tfidf
k:  1
averaged nMDCG:  0.2387
k:  3
averaged nMDCG:  0.4219
k:  5
averaged nMDCG:  0.4516
k:  10
averaged nMDCG:  0.4661
```

```
defaultStemmer_bm25
k:  1
averaged nMDCG:  0.3018
k:  3
averaged nMDCG:  0.5461
k:  5
averaged nMDCG:  0.578
k:  10
averaged nMDCG:  0.5622


englishStemmer_count
k:  1
averaged nMDCG:  0.045
k:  3
averaged nMDCG:  0.0801
k:  5
averaged nMDCG:  0.0977
k:  10
averaged nMDCG:  0.1202


englishStemmer_tfidf
k:  1
averaged nMDCG:  0.2568
k:  3
averaged nMDCG:  0.448
k:  5
averaged nMDCG:  0.4848
k:  10
averaged nMDCG:  0.4922


englishStemmer_bm25
k:  1
averaged nMDCG:  0.3018
k:  3
averaged nMDCG:  0.6207
k:  5
averaged nMDCG:  0.6153
k:  10
averaged nMDCG:  0.5946
```

```
englishStopwords_count
k:  1
averaged nMDCG:  0.2387
k:  3
averaged nMDCG:  0.3774
k:  5
averaged nMDCG:  0.408
k:  10
averaged nMDCG:  0.439


englishStopwords_tfidf
k:  1
averaged nMDCG:  0.2477
k:  3
averaged nMDCG:  0.427
k:  5
averaged nMDCG:  0.4643
k:  10
averaged nMDCG:  0.4673


englishStopwords_bm25
k:  1
averaged nMDCG:  0.3018
k:  3
averaged nMDCG:  0.5864
k:  5
averaged nMDCG:  0.5953
k:  10
averaged nMDCG:  0.5846




englishStopwords_bm25_title
k:  1
averaged nMDCG:  0.2703
k:  3
averaged nMDCG:  0.4883
```

```
k:  5
averaged nMDCG:  0.5252
k:  10
averaged nMDCG:  0.5328


englishStopwords_bm25_text
k:  1
averaged nMDCG:  0.2928
k:  3
averaged nMDCG:  0.5429
k:  5
averaged nMDCG:  0.5649
k:  10
averaged nMDCG:  0.5625


englishStopwords_bm25_title_text
k:  1
averaged nMDCG:  0.2883
k:  3
averaged nMDCG:  0.5696
k:  5
averaged nMDCG:  0.5658
k:  10
averaged nMDCG:  0.5607
```

References

[1] Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schtze. *An Introduction to Information Retrieval*. Cambridge university press, Cambridge, 2009.