Z534 Search Assignment 2 Report

Yan Song

Task 1

For task 1, the code is in file easySearch.java. The code prints out all the non-zero documents for a given query sequentially.

Task 2

For task 2, the code is in file searchTRECtopic.java. The code generates two files: TREC_short.txt and TREC_long.txt containing the search results for the TREC topics.

Task 3

For task 3, the code is in file compareAlgorithms.java. The code implements four different searching algorithms and generates the results files accordingly.

Task 4

The comparison is in the following table.

For short queries

Evaluation metric	Your algorithm	Vector Space Model	BM25	Language Model with Dirichlet Smoothing	Language Model with Jelinek Mercer Smoothing
P@5	0.1480	0.2920	0.3040	0.3480	0.2840
P@10	0.1480	0.3020	0.3000	0.3300	0.2820
P@20	0.1340	0.2640	0.2700	0.2890	0.2470
P@100	0.0978	0.1642	0.2487	0.1690	0.1600
Recall@5	0.0234	0.0529	0.0478	0.0621	0.0524
Recall@10	0.0551	0.0961	0.0879	0.1018	0.0913
Recall@20	0.0829	0.1451	0.1377	0.1460	0.1154
Recall@100	0.2202	0.3556	0.3557	0.3452	0.1337
MAP	0.1054	0.1990	0.2011	0.2072	0.1942
MRR	0.2856	0.4798	0.4778	0.4786	0.4542
NDCG@5	0.1627	0.3107	0.3212	0.3520	0.3015
NDCG@10	0.1633	0.3194	0.3196	0.3448	0.3010
NDCG@20	0.1613	0.3081	0.3115	0.3329	0.2910
NDCG@100	0.1921	0.3120	0.3251	0.3309	0.3108

For long queries

Evaluation metric	Your algorithm	Vector Space Model	BM25	Language Model with Dirichlet Smoothing	Language Model with Jelinek Mercer Smoothing
P@5	0.1280	0.2560	0.2840	0.2560	0.2320
P@10	0.1240	0.2440	0.2440	0.2420	0.2140
P@20	0.1140	0.2210	0.2340	0.2340	0.2120
P@100	0.0742	0.1406	0.1488	0.1460	0.1378
Recall@5	0.0187	0.0349	0.0402	0.0403	0.0406
Recall@10	0.0368	0.0621	0.0703	0.0708	0.0658
Recall@20	0.0595	0.1064	0.1159	0.1270	0.1136
Recall@100	0.1743	0.2929	0.3167	0.3340	0.2901
MAP	0.0664	0.1529	0.1676	0.1586	0.1514
MRR	0.2563	0.4528	0.4597	0.3475	0.3640
NDCG@5	0.1388	0.2819	0.3029	0.2499	0.2348
NDCG@10	0.1341	0.2682	0.2729	0.2473	0.2294
NDCG@20	0.1310	0.2586	0.2718	0.2590	0.2410
NDCG@100	0.1466	0.2694	0.2872	0.2753	0.2609

Summary:

1. As more documents included in the evaluation stage, precision score tends to decrease while recall tends to increase, which is because of the nature of these two measurements.
2. NDCG seems to be quite stable on different degrees.
3. Our simple algorithm performs much worse than the build-in algorithms.
4. Storing index offline has huge improvements on the performance of the online search algorithms. Index helps to narrow down the targeting files and some pre-stored information can be very helpful like the document length.
5. The search result for the long queries are not necessarily better than the ones got from the short queries. This might be caused by the noise in the long queries.
6. Fixed search algorithms always produce the same search results, but the results might be far away from information need of the users. This is why we need to consider the feedback from the user to adjust the ranking results dynamically.