Homework 5

1 Introduction

1.1 Collaboration and Originality

1. Did you receive help of any kind from anyone in developing your software for this assignment (Yes or No)? It is not necessary to describe discussions with the instructor or TAs.

   No

2. Did you give help of any kind to anyone in developing their software for this assignment (Yes or No)?

   No

3. Did you examine anyone else's software for this assignment (Yes or No)? It is not necessary to mention software provided by the instructor.

   No

4. Are you the author of every line of source code submitted for this assignment (Yes or No)? It is not necessary to mention software provided by the instructor.

   Yes.

5. Are you the author of every word of your report (Yes or No)?

   Yes.

2 Experiment: Diversity and relevance baselines

2.1 Experimental Results

	Indri (Exp-2.1a)	Indri + PM2 (Exp-2.1b)	Indri + xQuAD (Exp-2.1c)	BM25 (Exp-2.1d)	BM25+ PM2 (Exp-2.1e)	BM25+ xQuAD (Exp-2.1f)
P-IA@10	0.1192	0.1783	0.1192	0.1417	0.1775	0.1858
P-IA@20	0.1487	0.1638	0.1504	0.1671	0.1542	0.1692
αNDCG@20	0.3177	0.3693	0.3009	0.3719	0.4146	0.4197
P@10	0.26	0.41	0.26	0.28	0.46	0.45
P@20	0.29	0.39	0.3	0.295	0.47	0.47
MAP	0.2152	0.1572	0.1216	0.1946	0.1844	0.1838

2.2 Parameters

trecEvalOutputLength=100

retrievalAlgorithm=Indri

Indri:mu=1500

Indri:lambda=0.1

BM25:b=0.75

BM25:k_1=1.2

BM25:k_3=0.0

2.3 Discussion

Overall:

Using diversity algorithm can improve the diversity metrics like P-IA@10, P-IA@20 and aNDCG@20, but could decrease non-diversity metrics like P@10, P@20 and MAP. In our experiments, after using PM2 and xQuAD algorithm, diversity metric improves but MAP gets worse. More interestingly, P@10 and P@20 gets higher. This means that the diversity algorithm could help with ranking the top matching documents. The reason could be that in normal ranking, all top rankings documents are related to one intent and the intent is not what the query is actually looking for. Therefore, the Precision at top range could be low comparing to diversity algorithm. On the other hand, since diversity algorithm will keep balancing documents that relates to different intent, it could make some diverse but not so related documents ranked ahead those highly related but not so diverse documents. Therefore, the overall ranking score could be low comparing to non-diversity algorithm. This tradeoff shows that we need to tune the balance between non-diversity algorithm and diversity algorithm carefully.

Examples:

For query 23, “Yahoo”, diversity algorithm has a better performance than the non-diversity algorithm. The non-diversity algorithm has zero score for P@10 and P@20, which means it is returning documents that has high ranking score but totally not relevant documents to the user. However, after applying diversity algorithm, it is able to have 0.3 and 0.15 for P@10 and P@20.

For query 21, “Volvo”, non- diversity algorithm has a better performance than the diversity algorithm. For P@10, both algorithms have 0.4. But for P@20 and P@30, non-diversity algorithm has higher scores. It might rank some irrelevant but more diverse documents ahead, which causes the ranking score to be lower.

3 Experiment: Effect of λ

3.1 Experimental results

Indri + PM2				
	λ=0.0 (Exp-3.1a)	**λ=0.33 (Exp-3.1b)**	**λ=0.67 (Exp-3.1c)**	**λ=1.0 (Exp-3.1d)**
P-IA@10	0.1783	0.1833	0.1733	0.1308
P-IA@20	0.1613	0.1638	0.1588	0.1546
αNDCG@20	0.3799	0.3927	0.3517	0.423
P@10	0.36	0.41	0.42	0.4
P@20	0.365	0.38	0.395	0.445
MAP	0.1517	0.1541	0.1555	0.1644
Indri + xQuAD				
	λ=0.0 (Exp-3.2a)	**λ=0.33 (Exp-3.2b)**	**λ=0.67 (Exp-3.2c)**	**λ=1.0 (Exp-3.2d)**
P-IA@10	0.1192	0.1192	0.1225	0.1792
P-IA@20	0.1487	0.1504	0.1537	0.1692
αNDCG@20	0.3177	0.3023	0.3212	0.3468
P@10	0.26	0.26	0.26	0.37
P@20	0.29	0.295	0.32	0.415
MAP	0.1193	0.1208	0.1259	0.1615

BM25 + PM2				
	λ=0.0 (Exp-3.3a)	**λ=0.33 (Exp-3.3b)**	**λ=0.67 (Exp-3.3c)**	**λ=1.0 (Exp-3.3d)**
P-IA@10	0.1758	0.1875	0.1725	0.1092
P-IA@20	0.1562	0.1567	0.1517	0.1354
αNDCG@20	0.3547	0.4073	0.4379	0.3897
P@10	0.51	0.48	0.44	0.31
P@20	0.525	0.49	0.445	0.4
MAP	0.2001	0.1897	0.1772	0.1397
BM25 + xQuAD				
	λ=0.0 (Exp-3.4a)	**λ=0.33 (Exp-3.4b)**	**λ=0.67 (Exp-3.4c)**	**λ=1.0 (Exp-3.4d)**
P-IA@10	0.1417	0.1858	0.1858	0.1858
P-IA@20	0.1671	0.1717	0.1617	0.1617
αNDCG@20	0.3719	0.4214	0.4053	0.4051
P@10	0.28	0.46	0.45	0.46
P@20	0.295	0.47	0.47	0.47
MAP	0.1029	0.1885	0.1848	0.1857

3.2 Parameters

Indri:mu=1500

Indri:lambda=0.1

BM25:b=0.75

BM25:k_1=1.2

BM25:k_3=0.0

3.3 Discussion

Discussion on the effect of lambda to the diversity metrics

For xQuAD algorithm, if lambda is higher, the algorithm will focus more on selecting diverse document. For PM2, if lambda is higher, the algorithm will tend to choose the document that fits a certain intent better. Since the target intent is changing over time, the final ranking result will be diverse. If the lambda is higher, the ranking will fit different query at each moment more intensively, thus resulting in a more diverse ranking. However, in my experiments the relation between diversity metric and the lambda is not so linear. Only in the Indri + xQuAD experiment, the diversity metric grows largely when lambda is larger.

Discussion on the effect of lambda to the precision metrics

The P@10, P@20 and MAP score are very correlated to the diversity metrics like P-IA@10, P-IA@20. If the diversity metric is high, the precision metric is high too. This is interesting because it probably means the ground-truth relevant documents are diverse too, instead of relating to only a few intents. Therefore, when we take diversity into consideration, the matching precision actually becomes better.

4 Experiment: The effect of the re-ranking depth

4.1 Experimental results

Indri + PM2				
	25 / 25 (Exp-4.1a)	**50 / 25 (Exp-4.1b)**	**100 / 50 (Exp-4.1c)**	**200 / 100 (Exp-4.1d)**
P-IA@10	0.1542	0.1983	0.1733	0.1792
P-IA@20	0.1533	0.1946	0.1588	0.1613
αNDCG@20	0.3896	0.462	0.3517	0.4116
P@10	0.41	0.46	0.42	0.43
P@20	0.35	0.47	0.395	0.425
MAP	0.0792	0.1187	0.1555	0.257

BM25 + PM2				
	25 / 25 (Exp-4.2a)	**50 / 25 (Exp-4.2b)**	**100 / 50 (Exp-4.2c)**	**200 / 100 (Exp-4.2d)**
P-IA@10	0.1383	0.19	0.1875	0.1733
P-IA@20	0.1429	0.1792	0.1567	0.1683
αNDCG@20	0.3351	0.467	0.4073	0.3773
P@10	0.3	0.5	0.48	0.38
P@20	0.33	0.435	0.49	0.455
MAP	0.0441	0.1117	0.1897	0.2852
BM25 + xQuAD				
	25 / 25 (Exp-4.3a)	**50 / 25 (Exp-4.3b)**	**100 / 50 (Exp-4.3c)**	**200 / 100 (Exp-4.3d)**
P-IA@10	0.1467	0.1875	0.1858	0.1783
P-IA@20	0.1496	0.1821	0.1717	0.1808
αNDCG@20	0.3496	0.4836	0.4214	0.4075
P@10	0.34	0.49	0.46	0.4
P@20	0.305	0.435	0.47	0.465
MAP	0.0442	0.115	0.1885	0.289

4.2 Parameters

Indri:mu=1500

Indri:lambda=0.1

BM25:b=0.75

BM25:k_1=1.2

BM25:k_3=0.0

diversity:lambda=0.67

4.3 Discussion

If maxInputRankingsLength is larger, more candidates can be considered to be the matching document. This can increase the chance of more related documents to be found. But at the same time, if we include too many documents with low-ranking score, these documents could happen to be picked into the ranking in the re-ranking phase and harm the performance. Because the ways we calculate the score, and the re-ranking score are different. We want to achieve a balance to include enough good documents to re-rank but avoid adding too many noisy documents that could possibly be picked due to the re-rank algorithm. If we look at P@10 and P@20 metrics across different ranking lengths, we can see this phenomenon. While adding more ranking candidates, the precision could go up first because more documents are taken into consideration for re-rank, and then go down later because noisy documents start to influence the reranking.

The parameter MaxResultRankingLength should be chosen according to the need of user and also the maxInputRankingsLength. If we make the two parameters the same, no documents will be discarded in the original ranking. Otherwise, we are picking some of the documents in the ranking. Comparing two groups of experiment, A and B, they both produce a result of 25 documents, but A only have 25 documents in the initial ranking, B has 50 documents. The performance of B is much better than A (both diversity metrics and non-diversity metrics). Therefore, it is probably better to have more candidate documents than actual needed when doing re-ranking. This allows some low ranked documents in the initial result to be picked out in the reranking. However, this will increase the computational complexity. And depending on the algorithm in use, the complexity could have a O(n^2) relation to the document number.