Your Name: Chih-Wei Chang

Your Andrew ID: cchang3

Homework 3

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.

   Yes. Yu-Heng Lei explained the TermVector API to me and discussed possible pitfalls in collecting terms from top N docs.

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

   No.

3. 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

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

   Yes

Your Name: Chih-Wei Chang

Your Andrew ID: cchang3

Homework 3

Instructions

1 Experiment 1: Baselines

	Ranked Boolean AND	Indri			
		BOW		Query Expansion	
		Your System	Reference System	Your System	Reference System
P@10	0.3400	0.2900	0.3200	0.3200	0.3100
P@20	0.3900	0.3475	0.3675	0.3650	0.3525
P@30	0.3983	0.3333	0.3417	0.3317	0.3417
MAP	0.1501	0.1514	0.1591	0.1546	0.1549
win/loss	N/A	9/11	13/7	13/7	12/8

1.1 Parameters

For base retrieval, I used Indri retrieval algorithm with mu equals to 1000 and lambda equals 0.7. For query expansion, the number of feedback documents fbDocs is set to 10, the number of terms fbTerms is set to 10, and the smooth coefficient fbMu is set to 0.

1.2 Discussion

First thing we note is that the win/loss is boosted significantly by query expansion. This shows the power of expanding query terms. However, we also notice that even though the win/loss is improved, the MAP score is not changed too much. One possible reason behind this difference might be that query expansion can boost some queries but also hurt some queries at the same time. Thus, it is possible that the expansion indeed makes the scores more distinguishable and help the system differentiate between vague documents and then improve the AP for some queries. However, the expansion also hurt the retrieval system in another way by lowering the AP for some queries. As a result, the win/loss might be better, but averaging over all the queries might reduce the overall improvement.

Another thing we found in this experiment is that the quality of the ranking used for finding expansion terms might not be very important. The reason is that even though our system produced a worse ranking comparing to reference system, when they were both used to expand the queries, the one used better ranking only outperform very slightly to another one. Thus, we

might guess that as long as the ranking is reasonable, it has the ability to improve the performance.

2 Experiment 2: The number of feedback documents

	Ranked Boolean AND	Indri BOW, Reference System	Query Expansion, Reference System Initial Results					
			Feedback Documents					
			10	20	30	40	50	100
P@10	0.3400	0.3200	0.3100	0.3100	0.2950	0.2900	0.3000	0.3050
P@20	0.3900	0.3675	0.3525	0.3675	0.3575	0.3625	0.3725	0.3750
P@30	0.3983	0.3417	0.3417	0.3533	0.3500	0.3450	0.3517	0.3500
MAP	0.1501	0.1591	0.1549	0.1584	0.1582	0.1596	0.1627	0.1678
win/loss	N/A	13/7	12/8	13/7	11/9	11/9	12/8	13/7

2.1 Parameters

For base retrieval, I used Indri retrieval algorithm with mu equals to 1000 and lambda equals 0.7. For query expansion, the number of feedback documents fbDocs range from 10 to 100, the number of terms fbTerms is fixed to 10, and the smooth coefficient fbMu is set to 0.

2.2 Discussion

The main goal in this experiment setting is trying to find the optimal fbDocs for the query expansion stage. Intuitively, we might guess that more documents we used the more accurate we can approximate the true terms distribution and thus the more improvement can result from the query expansion. Indeed, our experiment confirmed this conjecture. We can observe that the MAP is increasing in a steady way although the win/loss does not change too much. However, other than the improvement on the MAP, one another significant trend is that the running time tends to become much longer on larger fbDocs than smaller fbDocs. The reason is straightforward: more feedback documents we used implies we need to calculate score for more terms. And if the terms distribution follow the Heaps' Law, every one more fbDocs might increase our computational complexity by a square root speed. Thus, we might need to consider the trade-off between good MAP against reasonable running time carefully when choosing the fbDocs parameter. Personally, I think 50 fbDocs can produce reasonable good MAP while keep the running time in an acceptable range.

3 Experiment 3: The number of feedback terms

	Ranked Boolean AND	Indri BOW, Reference System	Query Expansion, Reference System Initial Results					
			Feedback Terms					
			5	10	20	30	40	50
P@10	0.3400	0.3200	0.3050	0.3050	0.3300	0.3400	0.3300	0.3350
P@20	0.3900	0.3675	0.3500	0.3750	0.3850	0.3800	0.3775	0.3800
P@30	0.3983	0.3417	0.3267	0.3500	0.3450	0.3483	0.3400	0.3417
MAP	0.1501	0.1591	0.1572	0.1678	0.1713	0.1731	0.1756	0.1766
Win/loss	N/A	13/7	11/9	13/7	13/7	14/6	16/4	16/4

3.1 Parameters

For base retrieval, I used Indri retrieval algorithm with mu equals to 1000 and lambda equals 0.7. For query expansion, the number of feedback documents fbDocs is set to the optimum retrained from previous experiment as 100, the number of terms fbTerms is ranging from 5 to 50, and the smooth coefficient fbMu is set to 0.

3.2 Discussion

Generally speaking, the more terms we used in the query expansion the better MAP we can get. Also, comparing to experiment 2, we can find that the improvement we can get from increasing the number of feedback terms is somehow more significant than increasing the number of feedback documents. This might inspire us to choose a moderate number of feedback documents while using a larger number of feedback terms. The benefit of doing so is that increasing number of documents will significantly increase the running time while increasing the feedback terms does not really have impact on the running time. The reason is straightforward in that the number of documents will affect the number of candidate terms but the number of feedback terms only affect the final retrieving process. Thus, by tuning these two parameters might help us find a balanced point between running time and MAP.

One interesting point we observed in this experiment is that for different number of feedback terms, although the queries that was improved are quite different, the query that was hurt by the expansion is always Query #91: er tv show. The reason might be that this query contains too vague terms and thus the query expansion might introduce too many irrelevant terms and dilute the precision.

4 Experiment 4: Original query vs. expanded query

	Ranked Boolean AND	Indri BOW, Reference System	Query Expansion, Reference System Initial Results					
			fbOrigWeight					
			0.0	**0.2**	**0.4**	**0.6**	**0.8**	**1.0**
P@10	0.3400	0.3200	0.3700	0.3550	0.3600	0.3100	0.2950	0.2900
P@20	0.3900	0.3675	0.3750	0.3850	0.3775	0.3675	0.3475	0.3475
P@30	0.3983	0.3417	0.3567	0.3500	0.3500	0.3383	0.3400	0.3333
MAP	0.1501	0.1591	0.1895	0.1889	0.1813	0.1691	0.1592	0.1514
Win/loss	N/A	13/7	14/6	15/5	15/5	16/4	15/5	9/11

4.1 Parameters

For base retrieval, I used Indri retrieval algorithm with mu equals to 1000 and lambda equals 0.7. For query expansion, the number of feedback documents fbDocs is set to the optimum retrained from experiment 2 as 100, the number of terms fbTerms is set to the optimum retained from experiment 3 as 50, and the smooth coefficient fbMu is ranging from 0.0 to 1.0.

4.2 Discussion

Before this experiment, I personally thought the best way to combine the original query and the expanded query might be something around 0.5 v.s. 0.5, as too extreme combination might result in unstable retrieval performance. However, from the experiment we can find that 0.0 original weight outperforms other weights. That implies that the original terms might not have too much importance after generating expanded terms. And this might be related to other query expansion strategies such as BM25 that discards the original query immediately after expanding queries. The reason those expansion strategies work might be the same that 0.0 original weight performs the best in our experiment.

We then looked closer at the queries that is affected by the expansion. We found that for Query #79: voyager, the AP actually improves as the original weight decrease from 1.0 to 0.0. This might because that the original query only contains one term, and thus it might be difficult for the retrieval system to distinguish those relevant documents and irrelevant ones. However, after query expansion, we select some informative terms to replace the original single term, which thus gives us some chance to find a better ranking.

5 Experiment 5: Effect of the original query quality

	Ranked Boolean AND	Query Expansion, Reference System Initial Results			
		BOW Original Query		SDM Original Query	
		Original	Expanded	Original	Expanded
P@10	0.3400	0.3200	0.3700	0.3850	0.4400
P@20	0.3900	0.3675	0.3750	0.4375	0.4375
P@30	0.3983	0.3417	0.3567	0.4300	0.4200
MAP	0.1501	0.1591	0.1895	0.1999	0.2153
Win/loss	N/A	13/7	14/6	14/6	17/3

5.1 Parameters

For base retrieval, I used Indri retrieval algorithm with mu equals to 1000 and lambda equals 0.7. For query expansion, the number of feedback documents fbDocs is set to the optimum retrained from experiment 2 as 100, the number of terms fbTerms is set to the optimum retained from experiment 3 as 50, and the smooth coefficient fbMu is set to the optimum retained from experiment 4 as 0.0.

5.2 Discussion

First, the SDM approach has the highest MAP among these four approaches, and this improvement is dramatic when comparing to no-expanded BOW query. However, we might not be that surprised by the huge gap between the MAP of original BOW and the expanded SDM. This is because the no-expanded SDM has already outperformed than origin BOW. In fact, if we compared the gaps between the original BOW and expanded one with original SDM and expanded one, we might find that expanding the BOW query give much more relative improvement. This might inspire us that the expansion has much more impact on the retrieval method that was “originally bad” than those were “original reasonably good”. Combining this observation with the fact that expanded SDM is slightly better than original SDM, we might conclude that there is very likely no harm for us to apply the expansion on our retrieval system, no matter the quality of our original system. That is, the risk that applying expansion will hurt the overall performance is low.