Your Name Raghav Sharma

Your Andrew ID   rvsharma

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.

   If you answered Yes, provide the name(s) of anyone who provided help, and describe the type of help that you received.

   No

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

   If you answered Yes, provide the name(s) of anyone that you helped, and describe the type of help that you provided.

   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.

   If you answered No:
   a. identify the software that you did not write,
   b. explain where it came from, and
   c. explain why you used it.

Yes

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

   If you answered No:
   a. identify the text that you did not write,
   b. explain where it came from, and
   c. explain why you used it.

Yes

Your Name   Raghav Sharma

Your Andrew ID rvsharma

Homework 3

Instructions

1 Experiment 1: Baselines

	Ranked Boolean AND	**Indri**			
		BOW		**Query Expansion**	
		Your System	**Reference System**	**Your System**	**Reference System**
P@10	0.4000	0.4850	0.4950	0.4800	0.4600
P@20	0.3675	0.4375	0.4425	0.4550	0.4425
P@30	0.3417	0.4267	0.4350	0.4400	0.4267
MAP	0.1071	0.1335	0.1358	0.1359	0.1324
win/loss	N/A	14/5	15/5	14/6	14/6

1.1 Parameters

Ranked Boolean AND

- retrievalAlgorithm: RankedBoolean,
- trecEvalOutputLength=100

BOW-Indri My system

- trecEvalOutputLength=100;
- retrievalAlgorithm=Indri
- Indri:mu=2500
- Indri:lamda=0.4

Query Expansion – My System

- Indri:mu=2500
- Indri-lambda=0.4
- fbDocs=10
- fbTerms=10
- fbMu=0
- fbOrigWeight=0.5

Query Expansion – Reference System

- Indri:mu=2500
- Indri-lambda=0.4
- fbDocs=10
- fbTerms=10

- fbMu=0
- fbOrigWeight=0.5
- fbInitialRankingFile=Experiments/Input/Indri-Bow.teIn

1.2 Discussion

Ranked Boolean AND v/s Indri(Self and Reference):
Indri BOW model as expected, performs better in query results accuracy as compared to the baseline(RankedBoolean AND). The MAP, is higher by 20% and significant improvement is seen in precision at higher number of results(P@20, P@30).
Win – loss ratio shows there were 14 queries which peformed better than Boolean AND. Most significant improvement (~96%) happened in queries where AND performed very low(map-0.0061) which was query:

721:Census data applications

Hypothesis for this performance is that the Indri model uses smoothing over query terms and document length and Boolean AND might retrieve documents which are not relevant but still might be retrieved due to term frequency in the corpus. Indri provides query length smoothing using lambda parameter and document length normalization using the mu parameter. Here mu might be responsible for providing and idf like weight and fetching relevant documents.

For query
784:mersenne primes
Indri(map=0.1625) did 50% worse than Boolean AND(map-0.3214). This is not as bad as expected. This might be due to the query words being highly specific and Boolean AND will be better at retrieving relevant documents that have a low collection term frequency.

These queries have a similar effect on Indri query expansion as well.

Indri-BOW v/s Indri Query expansion:
Using query expansion the MAP for system increases although not considerably(~2%). This might be due to some terms in the query expansion hurting the query results which leads to retrieval of some non-relevant document.

The biggest win (~33%) compared to Indri BOW model was for the query
766:diamond smuggling
The learned query was:
766: #wand (0.0042 diamond 0.0015 sierra 0.0013 leone 0.0008 stone 0.0008 country 0.0007 trade 0.0006 digger 0.0006 have 0.0006 from 0.0006 origin)

This might be due to the maximum terms being associated with the original query terms and it led to increase in the recall for relevant documents.

The biggest loss to query expansion (~90%) when compared to Indri-BOW reference was for query

741:Artificial Intelligence
741: #wand (0.0124 artificial 0.0110 intelligence 0.0053 class 0.0044 acm 0.0039 subj 0.0036 abs 0.0034 fiction 0.0034 author 0.0034 i 0.0034 title)

As we can see from the learned query it includes ambiguous terms from academia(class), a computing society(acm), and science fiction writing(fiction, author, title). This leads to a fall in the map score for this particular query.

In terms of computational complexity, as we include the query expansion, the query for results becomes slower and takes more time to compute each term.

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	0.4000	0.4950	0.4600	0.4700	0.4750	0.4750	0.4750	0.4700
P@20	0.3675	0.4425	0.4425	0.4475	0.4500	0.4575	0.4825	0.4650
P@30	0.3417	0.4350	0.4267	0.4300	0.4350	0.4583	0.4850	0.4633
MAP	0.1071	0.1358	0.1324	0.1345	0.1345	0.1371	0.1391	0.1380
win/loss	N/A	15/5	14/20	14/20	14/20	14/20	14/20	14/20

2.1 Parameters

```
queryFilePath=Experiments/HW3/Indri-Bow.qry.txt
trecEvalOutputLength=100
retrievalAlgorithm=Indri
Indri:mu=2500
Indri:lambda=0.4
fb=true
fbTerms=10
fbMu=0
fbOrigWeight=0.5
fbInitialRankingFile=Experiments/HW3/HW3.1-ReferenceIndri-Bow.fbRank
fbExpansionQueryFile=Experiments/HW3/OUTPUT_DIR/HW3.5-QryExpRefFBT30L1.0.qryOut
```

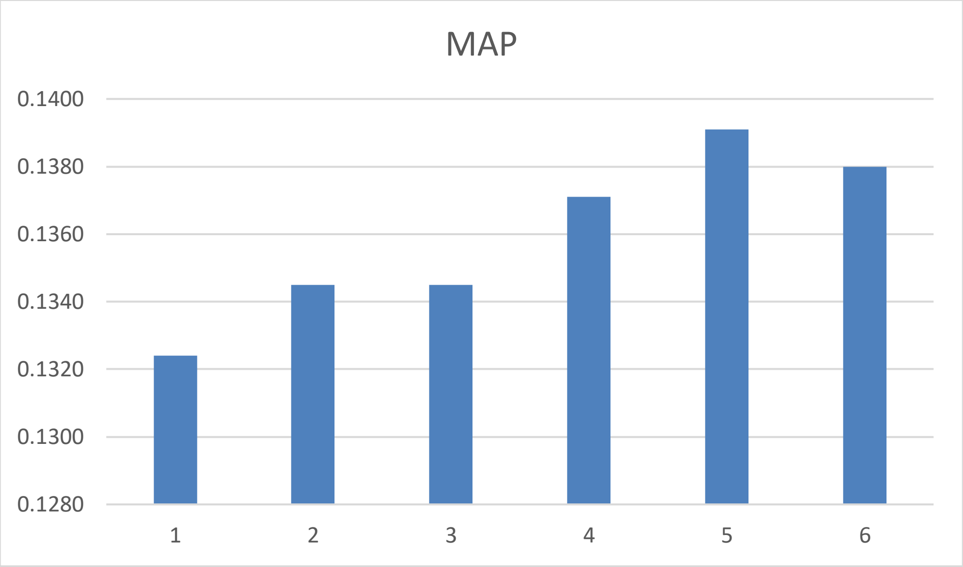
These parameters were held constant while performing the experiments

2.2 Discussion

In this experiment we increase the number of feedback documents on which we base our expansion terms. As we can say intuitively, higher the number of feedback documents to base the expansion terms on higher the possibility of an increase in MAP. This is evident from the experiment performed.

The most significant increase(~100%) in MAP is for fbDocs = 50. We can also see that P@n is the highest for the value for fbDocs = 50.

The variation of the MAP value is depicted in the graph below. As we can see more the number of feedback documents the MAP increases till fbDocs = 50 and is at highest. For fbDocs = 100, the value again decreases.



The winning query compared to Ranked Boolean AND, for most MAP increase(~100%) is for query
735:Afghan women condition
Learned Query = 735: #wand (0.0074 afghanistan 0.0066 women 0.0017 children 0.0016 taliban 0.0013 rights 0.0010 said 0.0010 we 0.0010 s 0.0010 from 0.0009 assistance)

As we notice that the query contains subjects and terms mostly related to ‘afghan women condition’ and closely related to the information need.

The biggest looser is query #741 which decreases by 94%(map=0.0011) as compared to the Ranked Boolean AND(map=0.0198).

741:Artificial Intelligence
741: #wand (0.0206 artificial 0.0183 intelligence 0.0088 class 0.0072 acm 0.0065 subj 0.0060 abs 0.0057 fiction 0.0057 author 0.0056 i 0.0056 title)
This is as similar to the experiment above and might have decreased due to the ambiguity of the terms in the learned queries.

However, as the fbDocs increases from 10 to 20, the MAP for this query increases from 0.0011 to 0.0026, which is about 55% increase in MAP from fbDocs = 10 to fbDocs = 20. However, it is still low compared to the Ranked Boolean AND. The new learned query is
741: #wand (0.0436 artificial 0.0355 intelligence 0.0138 fiction 0.0125 class 0.0101 acm 0.0100 non 0.0099 title 0.0094 subj 0.0093 author 0.0087 abs)
As we see, new terms like non, and increase in the weight of some terms. This leads to an increase in the precision in fbDocs = 20.

As we keep increasing the fbDocs parameter from 20, to 30 till 50. The MAP for query 704 keeps increasing by the highest level among all queries, and sees a good improvement in MAP from baseline MAP of 0.0025 to MAP of 0.1120 for fbDocs = 50.
The original query is
704:Green party political views
Learned form:
704: #wand (0.0388 party 0.0141 candidate 0.0115 politics 0.0102 view 0.0094 total 0.0090 map 0.0079 0.0055 san 0.0052 election 0.0048 john)

As value of fbDocs increases we see significant slowing down of the search.

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	0.4000	0.4950	0.4500	0.4750	0.5150	0.5150	0.5200	0.5300
P@20	0.3675	0.4425	0.4450	0.4825	0.5050	0.5150	0.5175	0.5200
P@30	0.3417	0.4350	0.4367	0.4850	0.5000	0.5000	0.4983	0.5017
MAP	0.1071	0.1358	0.1350	0.1391	0.1441	0.1447	0.1465	0.1462
Win/loss	N/A	15/5	14/6	14/6	14/6	14/6	14/6	14/6

3.1 Parameters

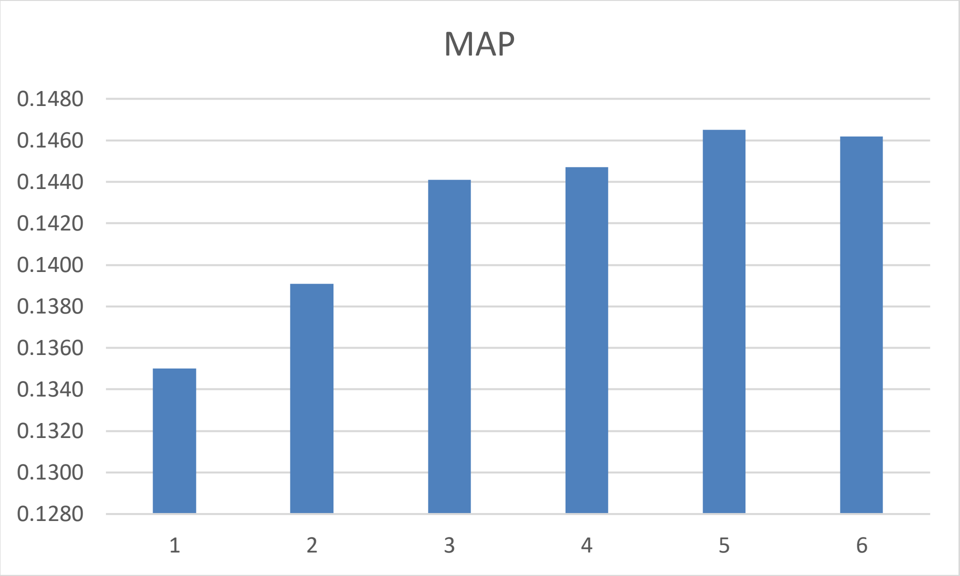
```
trecEvalOutputLength=100
retrievalAlgorithm=Indri
Indri:mu=2500
Indri:lambda=0.4
fb=true
fbDocs=50
fbMu=0
fbOrigWeight=0.5
fbInitialRankingFile=Experiments/HW3/HW3.1-ReferenceIndri-Bow.fbRank
fbExpansionQueryFile=Experiments/HW3/OUTPUT_DIR/HW3.5-QryExpRefFBT30L1.0.qryOut
```

These parameters are held constant during the experiments. fbDocs is chosen as 50 due to it giving the maximum MAP.

3.2 Discussion

fbTerms is the number of top expansion terms that are taken from the ranking of terms. Intuitively we feel that more the number of terms higher the accuracy of the results. So as the fbTerms parameter increases the MAP also increases.

As we can see from the table above the highest MAP is for fbterms 40 and 50. It increases about 25% from baseline comparison of Ranked Boolean AND. It increases about ~2% every time we increase fbTerms by 10. The graphical representation of this increase is depicted below



The highest increase for MAP is for the query 729
729:Whistle blower department of defense
Learned Query for fbTerms = 10
729: #wand (0.0024 whistle 0.0022 blower 0.0017 department 0.0013 employee 0.0012 act 0.0012 state 0.0011 investigate 0.0011 office 0.0010 claim 0.0010 fraud)
The MAP for the AND query was 0.0000 indicating the query was not able to produce relevant results. This value increased to MAP = 0.0049 and the system fetched relevant results from the learned query given above.

The other highest increase in MAP is for query 704
704:Green party political views
Learned query:
704: #wand (0.0388 party 0.0141 candidate 0.0115 politics 0.0102 view 0.0094 total)

The MAP improves by ~97% for this query as compared to the baseline Ranked Boolean AND. We see many terms which are relevant in the query above for fbTerms = 5.

The percentage change from increase in each 10 terms is also highest for this query and is maximum when the fbTerms changes from 5 to 10. This %change is ~35% increase in this case and the learned query for this is given below:

704: #wand (0.0388 party 0.0141 candidate 0.0115 politics 0.0102 view 0.0094 total 0.0090 map 0.0079 0 0.0055 san 0.0052 election 0.0048 john)

However, the win/loss ratio remains the same for each increase in fbTerms implying that the queries which perform poorly than Ranked Boolean AND do not perform better with more number of terms.

Here also as similar to above experiments the biggest loss is for the query 741:

741:Artificial Intelligence
741: #wand (0.0680 artificial 0.0546 intelligence 0.0206 class 0.0167 acm 0.0158 title)

This may be again due to the ambiguity of the expansion terms in the expansion query. The MAP(original = 0.0198) decreases by 16%.

For any loss queries, the MAP does not increase as we increase the fbTerms from 5 to 50. For eg the query 738: 738:Anthrax hoaxes, for fbTerms =5 is 0.2576 to fbTerms = 50 is 0.2541. This implies that the MAP is not affected by the number of terms in the query if that query perform poorly to the Ranked Boolean AND.

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	0.4000	0.4950	0.5200	0.5300	0.5250	0.5200	0.5200	0.4850
P@20	0.3675	0.4425	0.5100	0.5250	0.5200	0.5075	0.4850	0.4375
P@30	0.3417	0.4350	0.4883	0.5000	0.5050	0.4917	0.4700	0.4267
MAP	0.1071	0.1358	0.1353	0.1466	0.1473	0.1452	0.1410	0.1335
Win/loss	N/A	15/5	13/6	14/6	14/6	14/6	14/6	14/6

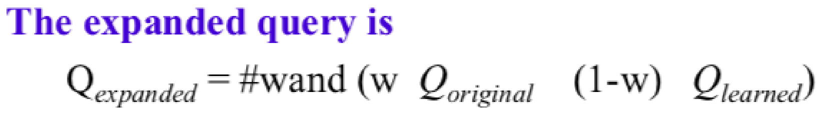
4.1 Parameters

```
trecEvalOutputLength=100
retrievalAlgorithm=Indri
Indri:mu=2500
Indri:lambda=0.4
fb=true
fbDocs=50
fbTerms=50
fbMu=0
fbInitialRankingFile=Experiments/HW3/HW3.1-ReferenceIndri-Bow.fbRank
fbExpansionQueryFile=Experiments/HW3/OUTPUT_DIR/HW3.5-
QryExpRefFBT30L1.0.qryOut
```

These parameters are held constant during the experiments. fbDocs and fbTerms are chosen as 50 due to them giving the maximum MAP at these values

4.2 Discussion

The expanded query is given as



The expanded query is

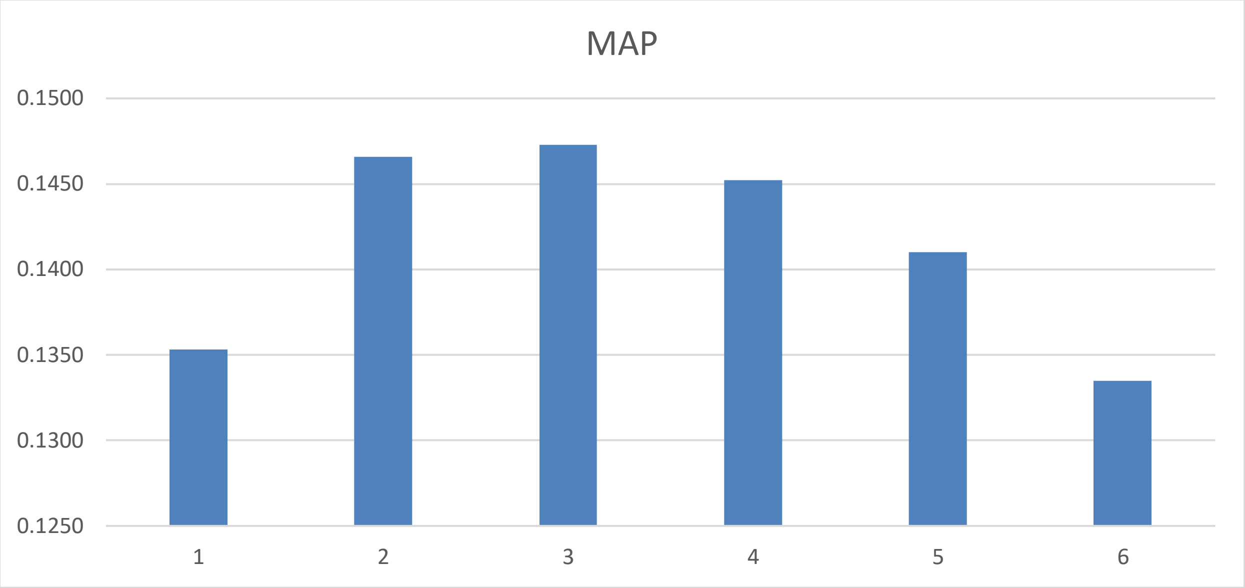
$$Q_{expanded} = \#wand\ (w \quad Q_{original} \quad (1\text{-}w) \quad Q_{learned})$$

Reference: Slide 36 at https://boston.lti.cs.cmu.edu/classes/11-642/Lectures/11-RfAndPrf.pdf

W = fbOrigWeight; so higher the fbOrigweight the original query is given more importance when compared to the learned query. And similarly lower the fbOrigWeight more importance is given to the learned query terms.

In the experiments the fbDocs and fbTerms are kept constant at 50 each since these were the values at which we received the highest MAP in earlier experiments. For fbOrigWeight = 0.0, only learned terms are considered and it gives the highest weight to them so the results all depend on these terms and weights of these terms.

With fbOrigWeight = 0, only learned terms are considered. When comparing learned terms to the original query we might expect the MAP of learned terms to be higher compare to the the original query. But as we can see from the graph below and table above, this is not the case. The MAP is only ~1.5% better when compared to the original query. As we go from weight of 0.0 to 0.4, the MAP increases linearly with the increase in fbOrigWeight. But after reaching 0.4, it reduces drastically as more weightage is given to the original query.



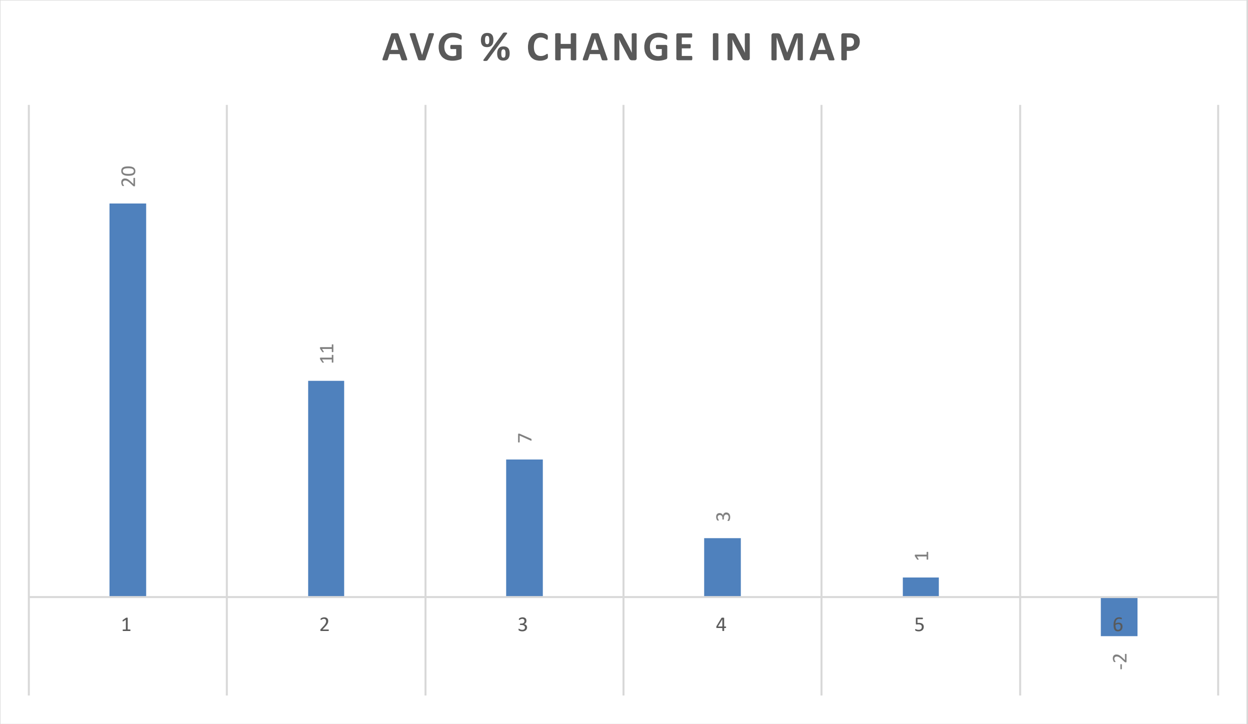
This may be due to weightage given to the original query more and as it is increased then the small weightage that is given to the learned terms is harmed and they become increasingly non relevant. When the weightage of the learned terms is higher (fbOrigWeight < 0.5), MAP is increased due to the positive effect of adding Pseudo relevance feedback to the queries. But as you take that away from the queries (fbOrigWeight > 0.5) the decline is much more rapid.

Similar to experiments above, the biggest winner among the queries is 704
704:Green party political views
Which increases ~98% when compared to ranked Boolean AND and the learned terms with fbDocs = 50 and fbTerms = 50, fbOrigWeight = 0.4 (for which MAP is the highest) is
704: #wand (0.0388 party 0.0141 candidate 0.0115 politics 0.0102 view 0.0094 total 0.0090 map 0.0079 0 0.0055 san 0.0052 election 0.0048 john 0.0044 democratic 0.0044 vote 0.0042 green 0.0041 republican 0.0039 voter 0.0035 george 0.0032 libertarian 0.0031 primary 0.0029 hollist 0.0029 state 0.0029 kovel 0.0028 hagelin 0.0027 larouche 0.0027 returns 0.0027 santa 0.0026 orrin 0.0026 president 0.0026 nader 0.0026 government 0.0026 trump 0.0025 kip 0.0025 browne 0.0024 mccain 0.0024 america 0.0024 lyndon 0.0024 hine 0.0024 forbes 0.0023 county 0.0023 bauer 0.0023 bowman 0.0023 law 0.0022 reform 0.0022 2000 0.0022 neil 0.0022 gore 0.0022 bradley 0.0022 independent 0.0021 ralph 0.0021 lynn 0.0021 hatch)

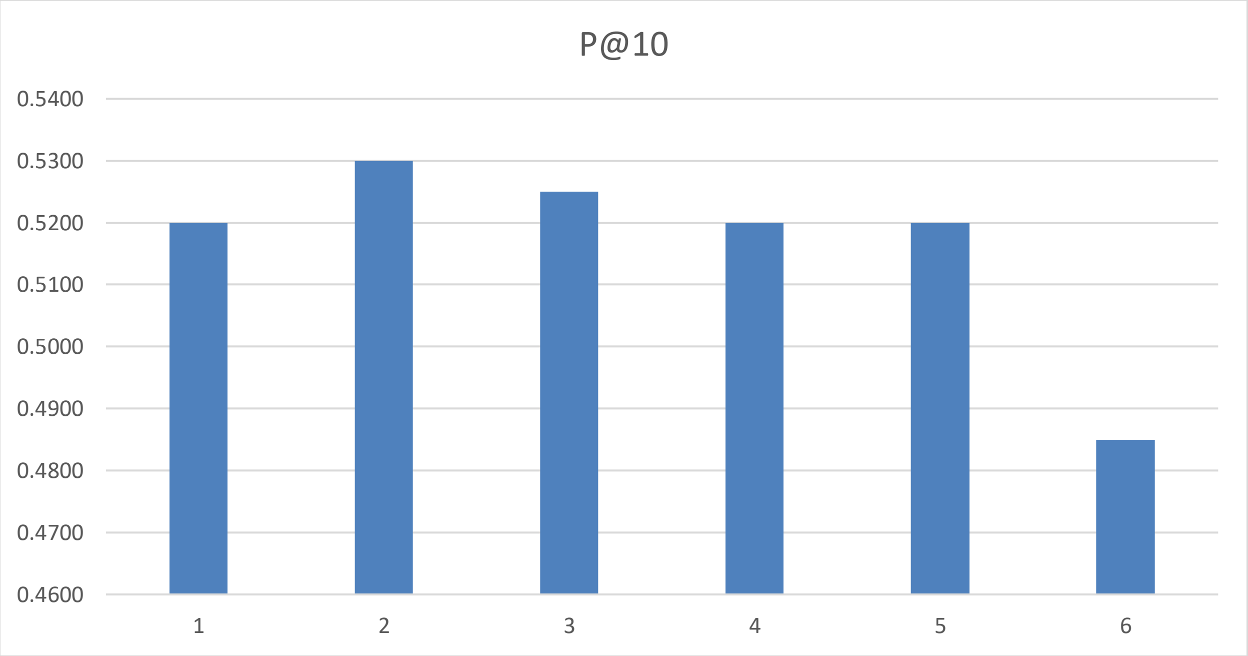
As we can see the learned query has lot of terms with higher weight that satisfies the information need and this results in higher MAP for the query when learned terms are given more weightage.

A more interesting analysis would be to see how the % change occurs as we increase the fbOrigweight from 0.0 to 1.0.

As we move away from the learned terms to original query we find that the average percentage change in the MAP reduces from 11% to -2% for original query. This agrees with the hypothesis earlier that as we move away from learned terms to original query we expect MAP to reduce.



Another noteworthy evaluation parameter is P@10 which indicates precision of top results. We can observe that as we move away from learned terms to original query, the P@10 parameter reduces which implies lower number of relevant results at the top as we move towards original query with very few relevant results (map = 0.4850) when only original query is considered.



The most loss when compared to Ranked Boolean AND in MAP is
for the query # 741:Artificial Intelligence.
Using the % change in MAP we notice that the MAP increases for this query as we weigh the original query more than the learned terms. Becoming MAP – 0.0117 same as Indri BOW when fbOrigWeight – 1.0. This is due to the ambiguity introduced by the learned terms in the query and as learned terms become less and less important (fbOrigWeight increases), the MAP for this query improves.

5 Experiment 5: Smoothing on longer queries

	Indri BOW, Reference System	**Query Expansion, fbTerms = 10**					
		λ					
		0.4	**Value 1 = 0**	**Value 2 = 0.2**	**Value 3 = 0.6**	**Value 4 = 0.8**	**Value 5 = 1.0**
P@10	0.4950	0.4800	0.5050	0.4850	0.4500	0.4300	0.0000
P@20	0.4425	0.4825	0.4975	0.4900	0.4600	0.4550	0.0000
P@30	0.4350	0.4817	0.4933	0.4900	0.4633	0.4350	0.0000
MAP	0.1358	0.1391	0.1400	0.1409	0.1335	0.1280	0.0000
Win/loss	N/A	11/9	11/9	12/8	10/10	9/11	0/20

	Indri BOW, Reference System	**Query Expansion, fbTerms = 20**					
		λ					
		0.4	**Value 1 = 0**	**Value 2 = 0.2**	**Value 3 = 0.6**	**Value 4 = 0.8**	**Value 5 = 1.0**
P@10	0.4950	0.5150	0.5250	0.5200	0.5050	0.4700	0
P@20	0.4425	0.5100	0.5157	0.5125	0.5050	0.4625	0
P@30	0.4350	0.4967	0.5117	0.5050	0.4850	0.4483	0
MAP	0.1358	0.1452	0.1454	0.1466	0.1408	0.1346	0
Win/loss	N/A	13/7	11/9	12/8	12/8	10/10	0/20

	Indri BOW, Reference System	**Query Expansion, fbTerms = 30**					
		λ					
		0.4	**Value 1 = 0.0**	**Value 2 = 0.2**	**Value 3 = 0.6**	**Value 4 = 0.8**	**Value 5 = 1.0**
P@10	0.4950	0.5200	0.5200	0.5350	0.5050	0.5000	0
P@20	0.4425	0.5125	0.5300	0.5150	0.5125	0.5025	0
P@30	0.4350	0.5050	0.5067	0.5083	0.4950	0.4767	0
MAP	0.1358	0.1481	0.1483	0.1489	0.1447	0.1393	0

Win/los s	N/A	12/8	10/10	12/8	13/7	11/9	0/20

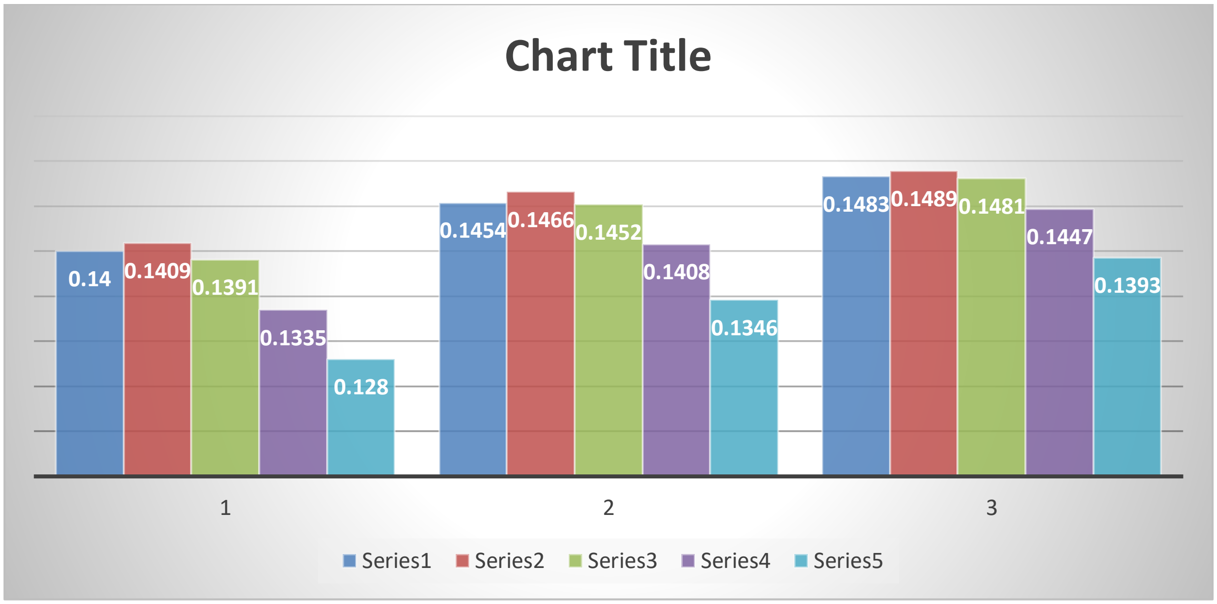
5.1 Parameters

```
indexPath=outlen/index-gov2
queryFilePath=Experiments/HW3/Indri-Bow.qry.txt
trecEvalOutputLength=100
retrievalAlgorithm=Indri
Indri:mu=2500
fb=true
fbDocs=50
fbMu=0
fbOrigWeight=0.4
fbInitialRankingFile=Experiments/HW3/HW3.1-ReferenceIndri-Bow.fbRank
fbExpansionQueryFile=Experiments/HW3/OUTPUT_DIR/HW3.4-QryExpRefFBorig0.0.qryOut
```

These parameters are held constant during the experiments. fbDocs is chosen as 50 due to it giving the maximum MAP at these values. While fbOrigWeight gives maximum MAP at value of 0.4.

5.2 Discussion

The parameter lambda is used for smoothing for larger queries. So the value of lambda is evident when the query is longer. As is evident from the table and the graph below that as the value of terms increases from 10 to 20 to 30, the MAP also increases for longer queries.



The first group represents 10 terms, second shows MAP for 20 terms, and third one shows MAP for 30 terms.
As we see, the MAP for 5 values of lambda 0.0, 0.2, 0.4, 0.6 and 0.8 gradually increases as the fbTerms is increased from 10 to 20 to 30 respectively.

Within each fbTerms group we get the highest value of MAP at lambda = 0.2.

The highest MAP is for lambda = 0.2 at fbTerms = 30 at a value of MAP = 0.1489.

Another inference from the experiment is that as the fbTerms increases, the effect of increase in lambda increases the percentage change in the value of MAP as depicted in the table below

fbTerms	Lambda				
	0	0.2	0.4	0.6	0.8
10	3	3.61958836	2.37239396	-1.7228464	-6.09375
20	3.71389271	3.88813097	4.20110193	5.18465909	4.90341753
30	1.95549562	1.54466085	1.95813639	2.69523151	3.37401292

As we see that when the value of lambda increases in the fbTerms 20 and 30, the percentage increase in MAP for these parameters increases with the value of lambda. This is due to the smoothing effect of lamda for longer queries. For shorter queries (fbterms = 10) the %change in MAP is negative when it is compared to the Indri BOW queries