Your Name: Wenhan Lu

Your Andrew ID: wenhanl

Homework 3

Collaboration and Originality

1. Did you receive help <u>of any kind</u> 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
   If you answered Yes, provide the name(s) of anyone who provided help, and describe the type of help that you received.

2. Did you give help <u>of any kind</u> to anyone in developing their software for this assignment (Yes or No)?
   No
   If you answered Yes, provide the name(s) of anyone that you helped, and describe the type of help that you provided.

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

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

Your Name: Wenhan Lu

Your Andrew ID: wenhanl

Homework 3

1 Experiment 1: Baselines

	Ranked Boolean	**BM25 BOW**	**Indri BOW**
P@10	0.1500	0.3000	0.2300
P@20	0.1800	0.2950	0.2800
P@30	0.1667	0.2967	0.2900
MAP	0.0566	0.1304	0.1277

Parameters:

BM25:

k_1=1.2
b=0.75
k_3=0

Indri:

mu=2500
lambda=0.4

2 Experiment 2: Different representations

	Indri BOW (body)	**0.20 url** **0.20 keywords** **0.20 title** **0.20 body** **0.20 inlink**	**0.50 url** **0.10 keywords** **0.10 title** **0.20 body** **0.10 inlink**	**0.10 url** **0.50 keywords** **0.10 title** **0.20 body** **0.10 inlink**	**0.10 url** **0.10 keywords** **0.50 title** **0.20 body** **0.10 inlink**	**0.30 url** **0.10 keywords** **0.10 title** **0.40 body** **0.10 inlink**
P@10	0.2300	0.1600	0.1700	0.1700	0.1500	0.2200
P@20	0.2800	0.2000	0.2250	0.2000	0.2100	0.2500
P@30	0.2900	0.2167	0.2300	0.2133	0.2300	0.2600
MAP	0.1277	0.1087	0.1110	0.1075	0.1070	0.1149

Set 1: all weights of the different fields are chosen evenly. I choose this set because I want to see what improvement I can get from bow case when I consider 5 fields instead of one. It's also a base case for comparison with all other weight combinations.

Set 2 to Set 4: One of the fields will be given a high weight (0.5), and the weight of body remains 0.2, others will be given 0.1. The most weighted fields I choose are "url", "keywords" and "title" accordingly. I set these three set in order to find out which field is most important and which is second. Intuitively I think "url", "keyword" and "title" should be more important than "body". Here I have these experiments to verify it and find out the most important field in order to construct my last weight set.

Set 5: body 0.4, url 0.3, others 0.1. This set of weight is set from knowledge I got from previous experiment. From the observation that all sets I chosen before are all worse than bow case. I set the weight of "body" higher to 0.4. And in previous experiment only "url" to 0.5 have better performance than evenly distributing weight. So I set "url" to second highest weight 0.3. All others just set to 0.1.

Trends I observe: Higher weight of 'url' tends to lead to high precision if to keep 'body' a reasonable number as well. 'body' is a underestimated field. A balanced weight of 'url' and 'body' tends to give a better result. Other fields tend to only give bias into result and should be set lower.

These representations actually behave out of my expectation. I expected 'body' to be a less important field than at least 'url' and 'title'. It turns out it is more important than 'title' at least. And I didn't expect none of my constructed queries get a worse score than bow case. I thought at least the last one will get a better precision but I still failed. I think that's may caused by bias introduced by giving no important fields too many weight.

To construct in #and(#wsum(different fields)) way obviously improve recall because all fields will sum up to contributed to final score. The more weights are evenly distributed, the high recall we can get. And the better distributed according to importance of fields, the high precision we can get.

There seems no clear relationship between accuracy and computational cost. But in this case as we have better accuracy in bow case, it's totally a waste to have so complex query to increase computation.

3 Experiment 3: Sequential dependency models

Example Query: Provide your structured query for query "fickle creek farm".

102:#wand(0.3 #and(fickle creek farm) 0.4 #and(#near/1(creek farm) #near/1(fickle creek)) 0.3 #and(#window/8(creek farm) #window/8(fickle creek)))

	Indri BOW (body)	**0.34 AND 0.33 NEAR 0.33 WINDOW**	**0.80 AND 0.10 NEAR 0.10 WINDOW**	**0.10 AND 0.80 NEAR 0.10 WINDOW**	**0.10 AND 0.80 NEAR 0.10 WINDOW**	**0.30 AND 0.40 NEAR 0.30 WINDOW**
P@10	0.2300	0.3500	0.2700	0.3400	0.3400	0.3500
P@20	0.2800	0.3650	0.3400	0.3750	0.3650	0.3750
P@30	0.2900	0.3600	0.3500	0.3733	0.3633	0.3600
MAP	0.1277	0.1891	0.1522	0.1833	0.1667	0.1889

Set 1 (0.34/0.33/0.33): This is a even distribution of weights of three parts. I set it like this in order to provide a base case of considering all three parts.

Set 2 (0.8/0.1/0.1) Set 3 (0.1/0.8/0/1) Set 4 (0.1/0.1/0.8): They are set to only have a super weighted part. They are used to determine which of the three parts is the most important one.

Set 5 (0.3/0.4/0.3): This set is chosen based on observation of previous experiment. Because 'near' is tested as the most important part, and first set (balanced) get the best result, I change weight from set 1 a little bit towards 'near'.

Trends observed: The more balanced they are, the better precision we can get. Even I adjust a little to separately most important part, result is still worse than most balanced case. This may because of importance of bag-of-words model, n-gram and window constraint are tight to each other. They are more powerful when used together and meaningless to compare them separately.

Match Expectation: This experiment matches my expectation that more complex query well constructed like SDM can improve precision.

Relationship between accuracy and computation cost: Not linearly related. But balanced distribution means complex queries like NEAR should take some reasonable parts.

In this case precision improvement is profound. So I think it's worth the cost on computation.

4 Experiment 4: Multiple representations + SDMs

Example Query: Provide your structured query for query "fickle creek farm".

102:#wand(0.2 #AND(#WSUM (0.1 fickle.url 0.1 fickle.title 0.1 fickle.keywords 0.3 fickle.inlink 0.4 fickle.body)#WSUM (0.1 creek.url 0.1 creek.title 0.1 creek.keywords 0.3 creek.inlink 0.4 creek.body)#WSUM (0.1 farm.url 0.1 farm.title 0.1 farm.keywords 0.3 farm.inlink 0.4 farm.body)) 0.8 #wand(0.3 #and(fickle creek farm) 0.4 #and(#near/1(creek farm) #near/1(fickle creek)) 0.3 #and(#window/8(creek farm) #window/8(fickle creek))))

	Indri BOW (body)	**w=1.0 (Exp 2)**	**w_1=0.83**	**w_2=0.67**	**w_3=0.5**	**w_4=0.33**	**w_5=0.16**	**w=0.0 (Exp 3)**
P@10	0.2300	0.2200	0.1889	0.2000	0.2300	0.2400	0.2600	0.3500
P@20	0.2800	0.2500	0.2100	0.2200	0.2500	0.2500	0.2850	0.3650
P@30	0.2900	0.2600	0.2400	0.2667	0.2833	0.2966	0.2767	0.3600
MAP	0.1277	0.1149	0.1167	0.1247	0.1322	0.1467	0.1552	0.1891

Trends: A linear trend of increasing precision as weight decreases. This pretty much match my expectation because using SDM is clearly a better construction of query. Having more weight on SDM will give a better precision as expected.

It worth the cost on computation for accuracy improvement.