Your Name: Hailun Zhu

Your Andrew ID: hailunz

Homework 1

1 Collaboration and Originality

Your report must include answers to the following questions:

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
   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 of any kind 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 every line of source code submitted for this assignment (Yes or No)?
   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 every word 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.

2 Structured query set

2.1 Summary of query structuring strategies

Briefly describe your strategies for creating structured queries. These should be general strategies, i.e., not specific to any particular query.

1. Generally, use And and NearN operator could have better precision results, especially for words that appear to be phrases.
2. Also, using field specified queries for words could sometimes yield good results, if the word could be a link or title.

2.2 Structured queries

List your structured queries. For each query, provide a brief (1-2 sentences) description that:

1. identifies which strategy (from Question 2.1) was used for that query,
2. any important deviations from your default strategies, and
   No, generally the results obey my strategies.
3. your intent, i.e., why you thought that particular structure was a good choice.

10:#and(cheap internet.inlink)
Use And operator and inlink field for internet. The internet is likely to be in a url or inlink field.
12:#and(djs djs.keywords djs.url)
Use And operator and keywords, url fields for djs.
26:#Near/4(lower #NEAR/1(heart rate))
Use NearN operator. Heart rate seems to be a phase so use Near/1 to narrow down the range.
29:#OR(#AND(#NEAR/1(ps.title 2.title) games.title) #AND(#NEAR/1(ps.body 2.body) games.body))
Use And operator to each group, ps 2 seems to be a phrase so use Near/1. And use title field and Or operator to combine those two And results.
33:#and(#and(elliptical trainer) #near/15(elliptical.title trainer.title))
Use And, Near operator, and title field.
52:avp.url
Use url field. This word avp has no obvious property, so I just test it with url.
71:#and(#near/1(living in) india)
Use And and Near/1 operator. The ‘living in’ seems to be a phrase so that I use Near/1.
102:#NEAR/1(fickle creek farm)
Use Near/1 operator. These three words compose of a specified phrase.
149:#AND(uplift at #NEAR/10(yellowstone #NEAR/2(national park)))
Use And and NearN operators. The Yellowstone national park seems to be a specified name. Meantime the national park could compose of a phrase, so compare the Yellowstone and national park separately.
190:#and(#AND(#NEAR/1(brooks brothers) clearance) #or(brooks.title brothers.title clearance.title))
Use And and NearN operator, and title field. The ‘brooks brothers’ is a brand, it is likely to be considered together, so use Near/1.

3 Experimental results

Present the complete set of experimental results. Include the precision and running time results described above. Present these in a tabular form (see below) so that it is easy to compare the results for each algorithm.

3.1 Unranked Boolean

	BOW #OR	BOW #AND	Structured
P@10	0.0100	0.0400	0.2400
P@20	0.0050	0.0200	0.2350
P@30	0.0033	0.0433	0.2467
MAP	0.0010	0.0142	0.1107
Running Time	00:14.8	00:03	00:02

3.2 Ranked Boolean

	BOW #OR	BOW #AND	Structured
P@10	0.1500	0.2500	0.3700
P@20	0.1800	0.2600	0.4050
P@30	0.1667	0.2767	0.3600
MAP	0.0566	0.0980	0.1483
Running Time	00:15.5	00:03	00:02

4 Analysis of results

Discuss your observations about the differences between the three different approaches to forming queries, and the two different approaches to retrieving documents in terms of their retrieval performance and running time.

Discuss the effectiveness, strengths, and weaknesses of the query operators and fields, and your success and failure at using them in queries. Did they satisfy your expectations given in Section 3?

Feel free to include other comments about what you observed

In this part of the report, do not just summarize the results from the previous section. We can see your results. You are expected to write your interpretation of the results based on what you learned in the lectures and readings. This is your chance to show what you learned from this homework assignment - take this section very seriously.

Generally, the OR operator will result in lower precision than AND operator. And the NEAR operator is very useful when implementing phrase indexing because this operator can give the information of the relative positions. And the overall performance of RankedBoolean is much better than UnrankedBoolean model, because the RankedBoolean has taken more things into account, that is the term frequency. And as the running time of these two retrieval models are very similar, I conclude that the RankedBoolean is better than UnrankedBoolean model.

For the ten structured queries I have done some separate experiments so I will discuss them individually.

For the #10 query, I have tested the following queries:

1. 10:#and(cheap internet.inlink)
2. 10:#and(cheap internet.url)
3. 10:#and(cheap internet.inlink internet.url)
4. 10:#or (cheap internet.inlink)

The results shown that the first one yield the best precision among the three, and it is much better than not using any field. So when a term appears to have a special property like internet, link etc, using a related field can improve the precision as well as recall. Also as the fourth query gets a nearly 0 precision, the OR operator works not so well when working alone without other operators. And as OR operator computes the union of the result, the term that occurs more often will be dominant, which makes other part of the query useless.

For the #12 and #52 queries, I have tested the following queries:

1. 12:#AND(djs)
2. 12:#and(djs djs.keywords djs.url)
3. 52:#and(avp.body avp.inlink)
4. 52:avp.url
5. 52:#and(avp.body avp.url)

These two queries both have only one term. However the queries that have better performance are different. For #12, the best one is the second one, to use both the body, keywords and url, whereas the best one for #52 is the fourth one to only use url field. Also these experiments shows to get higher precisions the recall may be decreased, that is the case between first and second queries above. However, there are situations that change the structured queries could affect the precision and recall in a same way, that is the case for the No.3, 4, 5 queries above. Only use body and url, or body and inlink could decrease the precision and recall at the same time. That may imply the restriction given by the query are two much so that those measurements are both decreasing.

For #26,#33, #71, #102, #149:

1. 26:#AND(lower #NEAR/5(heart rate))
2. 26:#Near/4(lower #NEAR/1(heart rate))
3. 33:#AND(elliptical trainer)
4. 33:#NEAR/5(elliptical trainer)
5. 33:#and(elliptical trainer) #and(elliptical.keywords trainer.keywords)
6. 71:#and(#near/1(living in) india)
7. 71:#near/5(#near/1(living in) india)
8. 71:#and(#near/2(living in) india.title india)
9. 102:#NEAR/1(fickle creek farm)
10. 149:#OR(uplift at #NEAR/5(yellowstone national park))
11. 149:#AND(uplift at #NEAR/10(yellowstone #NEAR/2(national park)))
12. 149:#and(uplift at #NEAR/3(yellowstone #NEAR/2(national park)))

These queries use the combination of the three operators. The ‘heart rate’ and ‘yellowstonr national park’ are very likely to be considered as phrases, so I choose to use the NEAR operator.

For #26, ‘lower … heart rate’ could also be a complete phrase, thus the result of using two NEAR operator is much better than AND and NEAR operators. By using two NEAR operators the recall and precision both increases. Thus, as for terms that seem to be related, like adjunct words with phrases, using NEAR operator might be a better choice.

However as for #33, I could not decide the distance between ‘elliptical’ and ‘trainer’, so that in this case the AND operator yields a better result. Besides, by using OR operator to combine the results of body field and keywords field, the MAP is a little bit higher.

For #71, by adding title field, the precision goes up while the recall falls. In this case, we need to find the balance between the recall and precision.

For #102, the NEAR/1 and NEAR/10 yield the same result, which may due to our greedy strategy. As the fickle creek farm could usually be considered as one phrase, then our strategy will only match once, though there may be other combination for larger distance between these terms.

For #149, it shows that NEAR/10 is better than NEAR/3 in this case, though we could consider ‘yellowstone national park’ as one phrase. As well AND operator is much better than OR operator. The OR operator is likely to take more irrelevant docs into account and keeps the real relevant docs away, thus decrease both the precision and recall.

For #29, #190:

1. 29:#OR(#AND(#NEAR/1(ps.title 2.title) games.title) #AND(#NEAR/1(ps.body 2.body) games.body))
2. 190:#AND(#NEAR/1(brooks brothers) clearance)
3. 190:#AND(#AND(#NEAR/1(brooks brothers) clearance) #OR(brooks.title brothers.title clearance.title))
4. 190:#OR(#AND(#NEAR/1(brooks brothers) clearance) #AND(brooks.title brothers.title clearance.title))

These two queries use the combination of the three operators and the fields. When using different fields with the same terms set, it is a good choice to use OR operator to combine those two results.

To sum up, appropriately using AND and NEAR operators could yield better result. AND operator could get the docs that contain all the query terms, and it is quick. However, it could not get any information about the relative positions about the required terms. Thus in this case, the NEAR operator is more powerful. The NEAR operator is a more flexible operator than the phrase index or byword index method. But excessive use of the NEAR operator could result in bad results also by setting more restrictions. The use of OR operator will perform well by combination with other operators.