Your Name: Leonardo Neves

Your Andrew ID: lneves

Homework 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)? 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 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.

Your Name: Leonardo Neves

Your Andrew ID: lneves

Homework 1

1 Structured query set

1.1 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.

To create the queries, I have tried to use the NEAR operator on terms that are part of a named entity or that have a different meaning when put together. In addition, for terms that would be alone on the query or that are acronyms, the documents where those terms appear on the title, url or as keywords should be more relevant. A third approach, for cases where some names would be interesting but the order might not be well known, an AND was used to search for terms in any order, differently from the NEAR where the order is strict. A fourth strategy is to look for popular terms on keywords and titles instead of only in the body, since they can be misleading.

1.2 Structured queries

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

1. which strategy (from Question 1.1) was used for that query,
2. any important deviations from your default strategies, and
3. your intent, i.e., why you thought that particular structure was a good choice.

69:#AND(sewing.keywords instructions) – This merged the third and forth strategies. We know that we only look for the terms alone, they might be misleading, so we combine them using and, since we don't really need them to occur in any particular order or to be near each other. Also, a lot of instructional documents might mention sewing without it being the subject of the document, making it better for look for it as a keyword, like proposed by strategy #4.

79:#OR(voyager.url voyager.keywords) – The query uses the second strategy. As it is a term alone, looking for it on the url and keywords makes it more likely to find documents with relevant content.

84:#NEAR/4(continental plates) – First strategy. Like query #69, the terms have a different meaning when put together, making the near operator to increase the relevance of found documents.

89:#OR(ocd.url ocd.keywords) – Second strategy. This is even better since acronyms are normally used on urls. Looking for ocd on the url should make the results better.

108:#AND(#OR(ralph owen) brewster)– This uses the third strategy. Since Ralph Owen Brewster is a famous person, it is usually referred as both Ralph Brewster or Owen Brewster. Brewster is not a common name and is his last name, so it should increase the relevance even if found alone. Ralph and Owen are more common and would only be interesting if found together with Brewster. So we use the **AND** here with **OR** inside.

141:#OR(#AND(dmv.url registration.url) #AND(dmv va registration))– Again, dmv is an acronym and most important documents that talk about it are under the dmv url. Since we are looking for registration, looking for a url that have both dmv and registration might be a good idea. This uses the second strategy for dmv and the fourth strategy for registration.

Va and registration are too broad and would only make sense if put together in no particular order, using strategy three for the second **AND**.

146: #NEAR/4(sherwood regional library) – This uses the first strategy. All words alone or on in a different order might be misleading, so we look for them in this particular order and appearing together, since we are looking into a named entity.

153:#OR(pocono pocono.url) – This query uses the second strategy, looking for the noun on the url to make the text more relevant. Since the term is not common, we also look in the body.

171:#OR(#NEAR/3(ron howard) #AND(ron howard)) – This query uses the first and third approaches, putting together terms that form a named entity. In order to avoid having a low recall in situations where only one of the name's is used, we also use the last **AND**.

197:#OR(#AND(idaho state) #AND(idaho flower.title)) – This query uses the second and third approaches. We put Idaho and state together but in no particular order, as strategy 3 does, and we look for the flower in the title, as strategy 2, so that we find documents where flower is an important term.

2 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.

2.1 Unranked Boolean

	BOW #OR	**BOW #AND**	**Structured**
P@10	0.0000	0.1100	0.1200
P@20	0.0150	0.1350	0.2250
P@30	0.0200	0.1533	0.2233
MAP	0.0020	0.0665	0.0727
Running Time	00:06:096	00:01:139	00:01:244

2.2 Ranked Boolean

	BOW #OR	**BOW #AND**	**Structured**
P@10	0.1700	0.3700	0.4800
P@20	0.2800	0.4450	0.5250
P@30	0.3367	0.4633	0.4867
MAP	0.1071	0.1882	0.1964
Running Time	00:06:632	00:01:355	00:01:334

3 Analysis of results: Queries and ranking algorithms

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

Hint: Do not just summarize the results from the previous sections; we can see those results above. You are expected to provide 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

Hint: Probably this section doesn't need to be longer than ¾ of a page (not counting these instructions).

It is expected for the precision of an **OR** operator to be lower than the **AND** operator and to have a higher recall. My goal here was to have a good P@10, a higher P@20 and a stable P@30. This would mean that I would be having a good precision but also a reasonable recall, since my precision would be maintained when increasing the number of documents. If it were a web search engine, I would have made increasing my P@10 my priority instead of balancing both metrics. We can see that the precision for **OR** operators is much smaller, even getting a 0.0 P@10 for the unranked Boolean. This happens because we are not narrowing our search and retrieving any document that shows the term, prioritizing recall. And, on the other hand, is doing the opposite. We look for all documents that have all terms, what means that if we forgot a term or introduced a term that is not important, we will change the result a lot. Because of that, it might favor precision, but would not be the highest in terms of MAP and P@n since it is too sensitive to the query terms. The custom queries try to narrow a little bit of **OR** results and, at the same time, expand the results for **AND**. A good example of this is query #108, where we realize the name will not always be written as a full name but having each part of the name to be searched alone is also not a good idea. We put OR and **AND** together to maximize our precision without losing important documents.

Unranked Boolean has lower accuracy than ranked, as expected, while showing a little increase in performance for not having to find MIN and MAX of each term. This might be more relevant for larger queries and larger corpus, but for this small experiment the ranked Boolean seems to be a better alternative.

4 Analysis of results: Query operators and fields

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?

Hint: Same hints as above.

I have tried to improve my performance query by query. I found that, for named entities, it is not always the case that using a **NEAR** operator might be a good choice. It might get a high precision for lower ranks but the recall is bad and the precision drops a lot when we increase the rank. On the other hand, using a combination of **AND** and **OR** was more effective. In addition, to improve **NEAR** recall, it could be used together with an **AND** inside an **OR**.

My strategy for acronyms was not as successful as expected. I thought that, since acronyms like dmv are normally used for urls, it should increase precision a lot. Also, I have tried the query **#AND(dmv.inlink va registration)** in order to find websites that would talk about the registration and have pointers to dmv website as a source or as a registration link. This was completely wrong, making made all my P@n to be 0 until P@30.

Another experiment that didn't work as well as expected was using the near operator to match sewing and instructions. As my first strategy was to put together names that would have a different meaning when put together, I tried the query **#NEAR/3(sewing instructions)** but I just got a high P@5 value but a really low precision for the other rankings. Looking for instructional documents with the sewing keyword had better results, probably because sewing instructions are not as used as "how to sew" or "sewing projects".

Using the fields other than the body are good for enhancing precision, but we will normally have a really bad recall. Also, the url and keywords narrow the results too much, making it too sensitive to bad terms.