Your Name: Leonardo Neves

Your Andrew ID: lneves

Homework 2

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 2

1 Experiment 1: Baselines

	Ranked Boolean	BM25 BOW	Indri BOW
P@10	0.1700	0.4200	0.4000
P@20	0.2800	0.3500	0.4700
P@30	0.3367	0.3667	0.4233
MAP	0.1071	0.1985	0.2057

2 Experiment 2: Queries with Synonyms and Phrases

2.1 Queries

69:#NEAR/5(sewing #SYN(lesson instructions teaching))
79:voyager #SYN(voyager.title mission.title)
84:#SYN(continental tectonic) plates
89:#SYN(ocd obsessive–compulsive)
108:#SYN(ralph owen) brewster
141:#NEAR/5(#SYN(virginia va) dmv) registration
146:#NEAR/3(sherwood regional library)
153:pocono #SYN(pocono.title mountains.title)
171:#NEAR/2(#SYN(ronald ron) howard)
197:idaho state #SYN(flower syringa)

2.2 Query descriptions

For each query, provide a brief (1-2 sentences) description that identifies which strategy was used for that query, any important deviations from your default strategies, and your intent, i.e., why you thought that particular structure was a good choice.

69:#NEAR/5(sewing #SYN(lesson instructions teaching)) – The information need seems to be related to learning how to sew. Having synonyms of instructions to other words that have a similar meaning might be very helpful. The near operator would improve by making sure the instructions were related to sewing.

79:voyager #SYN(voyager.title mission.title) – Assuming the query was about NASA's Voyager missions and not about the word voyager, this would allow us to make the search more precise by looking for the word on the body and on the title. Also, for the title, we would be looking fro mission too, finding situations like "NASA mission" that were actually talking about voyager.

84:#SYN(continental tectonic) plates – Continental plates are often written as tectonic plates, so this is why we have the syn operator, to allow finding situations where the other name is used.

89:#SYN(ocd obsessive–compulsive) – The most meaningful word from the acronym was used as a synonym to find documents that use the full name.

108:#SYN(ralph owen) Brewster – The person is known by the last uncommon name. Using synonym would allow us to find documents that use any of first names.

141:#NEAR/5(#SYN(virginia va) dmv) registration – Va is acronym for Virginia. I have used near because I wanted to make sure we were dealing with Virginia's dmv.

146:#NEAR/3(sherwood regional library) – There are libraries that are not regional and regional libraries that are not from Sherwood. Using the near operator allows us to search for the topic we are really looking for, not a different library.

153:pocono #SYN(pocono.title mountains.title) – Pocono is known for the mountains looking for both names on the title and for Pocono on the body might increase accuracy.

171:#NEAR/2(#SYN(ronald ron) howard) – Ron is not the full name, so using Ronald allow us to find documents that have the full name too. Having the near operator would only find situations where we are looking for this specific person, since Ronald and horward are common names.

197:idaho state #SYN(flower syringa) – The Idaho state flower is a kind of flower from Idaho, also known as Syringa. Looking for the actual name might increase performance.

2.3 Experimental Results

	Ranked Boolean	BM25 BOW	Indri BOW	Ranked Boolean Syn/Phr	BM25 Syn/Phr	Indri Syn/Phr
P@10	0.1700	0.4200	0.4000	0.2900	0.4800	0.5100
P@20	0.2800	0.3500	0.4700	0.3950	0.4800	0.5250
P@30	0.3367	0.3667	0.4233	0.3900	0.4433	0.5367
MAP	0.1071	0.1985	0.2057	0.1568	0.2102	0.2451

2.4 Discussion

Discuss any trends that you observe; whether the use of synonyms and phrases behaved as you expected; and any other observations that you may have.

As expected, we have observed an increase in performance on all the ranking models. The idea of looking for synonyms on the title, as done in queries #79 and #153, made it possible to distinguish documents that were related to the subject and not only other meanings of the words. Having the synonym on query #197 made it possible to make 3 general terms to point into a more specific direction, better representing the information need. For question #141, we saw a great improvement by using the synonym operator to find the full name of an acronym, the same as we did on #89. The near operator on queries #171 and #141 was important for us to specify the exact name of the person and the exact dmv we were looking for. Using

the near operator on both #69 and #146 helped we preserve the meaning of what we were looking for. Losing the order would change this meaning. Query #84 was also interesting because the two synonym names are often used and having the operator allowed for us to identify documents disregarding the vocabulary used.

3 Experiment 3: BM25 Parameter Adjustment

3.1 k_1

	k_1							
	1.2	1.5	3.0	10.0	1.0	0.6	0.3	0.0
P@10	0.5500	0.5400	0.5300	0.4500	0.5500	0.5500	0.5500	0.1200
P@20	0.4550	0.4550	0.4650	0.4000	0.4500	0.4450	0.4500	0.1400
P@30	0.4567	0.4600	0.4500	0.4000	0.4600	0.4667	0.4733	0.1600
MAP	0.2501	0.2487	0.2257	0.1853	0.2512	0.2535	0.2544	0.0686

.

3.2 b

	b							
	0.75	0.8	0.9	1.0	0.5	0.4	0.2	0.0
P@10	0.5500	0.5400	0.5100	0.5100	0.5500	0.5700	0.5300	0.5100
P@20	0.4550	0.4600	0.4600	0.4600	0.5000	0.5100	0.5100	0.5000
P@30	0.4567	0.4567	0.4233	0.4167	0.4833	0.4833	0.4900	0.4967
MAP	0.2501	0.2486	0.2389	0.2247	0.2635	0.2645	0.2591	0.2244

3.3 Discussion

Explain your reasons for choosing the values that you tested, and how those reasons are related to how BM25 works. Discuss any changes in retrieval performance that you observed, and the significance of any trends that you observed.

I have decided first to increase both parameters to see how getting small increases and getting it up one order of magnitude for k1 and up to the max value for b would affect the performance. After that, I have decreased the values first in a small step but then in a larger step until it got to 0, so I could see how this would affect the result.

Increasing k1 is similar to increasing the smoothing. If we get k1 to be too large, the influence of term frequency would be big since the denominator of the middle part of BM25's equation would be huge making the overall values to be small. The term frequency, then, would be representative of the term importance, making it closer to our version of the ranked Boolean when k1 is too large. On the other hand, decreasing k1 to less than 1 will also decrease the effect of the document term frequency to the point, when k1 = 0, it will have zero effect to our system, making terms rare terms to be as important as frequent terms. From what we have learned so far, documents that have more appearances of a non-

stopword term are more likely to be important. This is proven by the really bad results that I got by making k1 to be zero.

The constant b normalizes the term frequency by the document length. When b is equal to 0, there is no normalization, what makes longer documents more likely to be retrieved. On the other hand, when b is 1, it is completely normalized, what might not be as desirable since we might lower the importance of some frequent terms just because a document is long. We can see that, for our collection, the values of b around 0.5 were better and having the value closer to 1 had similar results than having them close to zero, probably because the length our documents are well distributed around the median document length.

4 Indri Parameter Adjustment

4.1 μ

	μ							
	2500	4000	7000	10000	1500	1000	500	0
P@10	0.4800	0.4200	0.4200	0.4300	0.4700	0.5200	0.5700	0.5400
P@20	0.5300	0.5100	0.5350	0.5200	0.5400	0.5300	0.5250	0.4950
P@30	0.4800	0.5200	0.5367	0.5367	0.4867	0.4900	0.5067	0.4533
MAP	0.2397	0.2263	0.2105	0.1997	0.2485	0.2555	0.2659	0.2423

4.2 λ

	λ							
	0.4	0.45	0.6	0.8	1.0	0.35	0.2	0.0
P@10	0.4800	0.4800	0.4800	0.4500	0.0000	0.4700	0.4700	0.4800
P@20	0.5300	0.5300	0.5150	0.4700	0.0050	0.5300	0.5350	0.5350
P@30	0.4800	0.4767	0.4567	0.4533	0.0067	0.4800	0.4800	0.4800
MAP	0.2397	0.2366	0.2274	0.2040	0.0013	0.2413	0.2456	0.2483

4.3 Discussion

Explain your reasons for choosing the values that you tested, and how those reasons are related to how Indri works. Discuss any changes in retrieval performance that you observed, and the significance of any trends that you observed.

Using the same reasoning as section 3, I have tried to first make small changes to the values of the parameters so I could see how small changes would affect the overall performance. Then, I would go to the extremes of both values to see the effect of large and small values.

For μ, increasing the parameter is the same as increasing the smoothing for the term frequency. Having a really large value would make the effect of the term frequency to be less important, which gave us a worse result. Decreasing the value makes the term frequency more important and, when we get to zero, we have zero smoothing on our results. In our case, as expected, we got an increase in precision for the top ranks but clearly a drop in recall for having a smaller precision on the lower ranks.

For λ, the changes on the parameter are similar to controlling the idf effect, or removing importance of a term that occur often but on a lot of documents, what might signal it is a stopword or at least a less representative term. Increasing the value of the parameter increases this importance for more rare words, usually more important for longer queries. Since our experiments were with shorter queries, we can see that getting the value to be 1.0 gave a terrible result as expected, while decreasing the value didn't show many differences, with a slight increase in performance measured by the map value.