PA3 REPORT

Rukmani Ravisundaram
Tayyab Tariq

Task1:

In task1, the query vector is idf weighted. We note that in the given training documents, almost all of the queries do not contain common stop words. So idf weighting here will be effective in those queries that have very common words such as 'the' , 'of', 'to' etc. The idf is calculated as log(N/df). For unseen words, we have made the decision of returning idf of zero, since we have not seen the word in any of the documents, chances are it is a misspelling. So in the absence of spell correction, we choose to ignore it. Normalization of raw term scores in the document is using log scaling. This gave better results than simple L1-norm. Weights for each of the three fields are given in the table. The general rule that was quickly realized was not to give too much importance to the body. So the weights of the body signals are always kept lower than the title and anchor text counterparts. This is because the anchor text and title are much better indicators of the content of a document. A single query term occurring in the title/anchor text is a much stronger signal of relevance than a few occurrences of query terms in disjoint places in the body.

Task2:

The BM25 ranking algorithm is implemented with the constants as shown in the table. The B parameters in the field dependent normalized term frequency, serve to normalize the raw term frequency with respect to the average length of the fields. The B_{title} is given a value closer to 1, since raw term frequencies tend to have a much larger deviation in the range of values and we get better results by normalizing them with respect to average title length. The B_{body} and B_{anchor} are given values that are not too close to 1, since the ratio of term frequencies to the body length doesn't vary too much, so dividing by average length doesn't improve normalization too much. Similarly for the anchor text length as well, due to the assumption that the anchor text is one single document of links, which makes its average size no very different from every anchor document size. Optimizing these parameters are done by keeping 6 values fixed and changing the one weight. The W parameters are given higher values for title and anchor as in Task1, because they are stronger indicators of document relevance than the body.

Task3:

Task1 has been modified to incorporate the smallest window signal. The size of the smallest window in the body, title and anchor text are calculated and the minimum of the three is taken for calculation. The Boost value is set to 1. Since cosine similarity has been tuned to perfection, there is little room for improvement by incorporating the smallest window signal on this particular dataset. With any value of Boost above 1, we are not seeing any improvement over the scores of cosine similarity (And we did not choose to hack to lower the weights of Cosine Similarity in order for smallest window to exceed Task1). An exponential decay function is chosen as it rapidly scales down the boost when window sizes are large. The scores currently obtained are in the single digit range. Therefore intuitively the value of the boost should not be set too high. This is to avoid overfitting to the development set, as a high value of boost say 100 will increase the score of a 1.5 to 150, and another document that has a smallest window of one less than the length of the query will get a much lower boost. To avoid this large fluctuation, we usually would set the boost to a small enough number, while still sufficient to improve scores of relevant documents.

Extra Credit: (to run , pass in task number of 4)

An additional static signal of page importance has been tried out. We used the idea of PageRank, and make an extremely simplified imitation of it. A document is considered important if there are many

links pointing to it. From the given training data, we can glean this information from the anchor text counts given. The counts of all anchor text is added up, which will give the number of links pointing to a page. Of course this method is extremely prone to spam linking. However, for our training set, which is all from the Stanford trusted domain, we will not consider this issue further. Instead of using the raw number of inlinks to a page, we log normalize the score, similar to the way raw term frequencies were normalized. Plus we also divide this number of inlinks to a page by the average number of inlinks over the training set. This kind of normalization gives better performance on the given training set. Task 4 has been added to the task 1 metric. We give a boost equal to the normalized page importance to the already computed score.

Question 2. Other metrics:

Some of the other metrics that could be used are max-tf normalization as described in the text book. This can be used to scale the body field by the maximum term frequency. But since we do not have the entire content of the document in this particular assignment, implementing this is not straightforward.

Raw term frequencies in BM-25 can be further log normalized in addition to normalizing them by average document lengths. This can avoid giving higher scores to documents that have repeated occurrences of a single query term, but doesn't contain all query terms.

Number of links to a page is also a good indicator of the static relevance of a page. In general, a page is considered important, statically, if there are more links to it. This has been implemented as the extra credit task.

Question 3.

B_{title}, B_{body}, B_{anchor} serve to discount the raw term frequencies in each field by the average field length. However we do not want to completely discount smaller fields, like the title field since the average length of the title length does not exhibit too much deviation from the mean, whereas the body fields tends to vary a lot across documents. In order to account for the difference in the nature of the various fields, we have three parameters that control how much we want to normalize with respect to the length of the field.

Question 4.

Varying B, the boost factor makes a significant change to ranking. If we have a large values of boost, we're effectively using only the smallest window as our strongest signal. In this case the weights on the three fields bear little, if not any impact on the score. Having a boost above 1 but below 10, gives approximately equal importance to all signals, namely the smallest window and term frequency weights of the each field.

Tables:

Table 1

Task 1 - Cosine Similarity	Task 2 - BM-25	Task 3 - Smallest Window	Task 4 - Static Page Importance
C1 = 0.1	$B_{title} = 0.84$, $B_{body} = 0.14$, $B_{anchor} = 0.02$	C1 = 0.1	C1 = 0.1
C2 = 0.1	$W_{title} = 21$, $W_{body} = 0.5$, $W_{anchor} = 16.5$	C2 = 0.1	C2 = 0.1

C3 = 0.8	K1 = 24	C3 = 0.8 Boost B = 1	C3 = 0.8

Table 2

Task	**NDCG Score on dev set**
Cosine Similarity	0.9048
BM25	0.9086
Smallest Window	0.9048
Page Rank	0.9019