605.744: Information Retrieval
Problem Set (Module 4)

Sabbir Ahmed

September 26, 2022

Note: for any problem or calculation requiring a logarithm use base-2 logs.

1. (15%) What do Salton and Buckley mean by *tfc* term weighting? What is the difference between *tfc* and *nfc* term weighting?

   Answer: Term weighting is terms that get scaled using their corresponding frequencies, relative frequencies, relevance, etc., or other attributes. *tfc* is considered the original TF-IDF method with cosine normalization. *nfc* uses a normalized TF component in its TF-IDF approach for the term weights.

2. (15%) The Porter stemmer conflates the words *program*, *programs*, and *programming* into the same stem (program). For a given document collection, how would both document term frequency (TF) and inverse document frequency (IDF) weights change in an IR system using the Porter stemmer compared to an IR system that uses plain words as indexing terms? Provide examples with your explanation.

   Answer: Different words getting stemmed to the same root replace the words in the dictionary with the stem word. This increases the term frequencies of the stem words and zeroes the values for the original words. The words *program*, *programs*, and *programming* getting stemmed to *program* increases the TF of *program* and zeros the values for the other words. Since the original words no longer exist in the dictionary, IDF values cannot be computed for them. The document frequency (DF) for the stem word, in this case, *program*, increases. Since the IDF is inversely proportional to this metric, it gets decreased.

3. (20%) *Impact Ordering* and *Index Elimination* are two separate techniques that each reduce computation when calculating document similarity by approximating normal vector cosine. In your own words give a short explanation of each method.

   Answer: Impact ordering refers to the order of the postings list being determined by other metrics besides the doc-IDs. The default method of computing cosine similarities was to traverse through each of the postings lists sorted by their doc-IDs, computing their cosine similarities, and accumulating their values at the end. This method can be inefficient due to computing the metrics for words not in the top-k. Instead of sorting the postings list by doc-IDs, they can instead be sorted by a different metric that can impose a greater impact, such as the term frequency.

   Index elimination is the process of setting an IDF threshold for terms to consider. Terms with low IDF values would not be considered as relevant and may consist of stopwords. This approach would narrow down the documents list to those with many (or all) of the query terms.

4. (50%) Calculate cosine similarities for query Q against just documents *D1* and *D2* from the following term-document matrix using the vector cosine model with TF/IDF weighting. Query Q contains the four words: **bear bear cougar dolphin**. The numbers in the table below are term frequencies. The document collection consists of only these eight documents, and the five terms listed below are the only ones found in any document.

   Recall that for the TF-IDF scheme, the weights for terms in a query or in a document should be the term frequency times the inverse document frequency for that term. Compute accurate cosine scores that do consider the vector length of the query. Show the work in your calculations.

   Table 1: Word Frequencies per Documents

Word	**D1**	**D2**	D3	D4	D5	D6	D7	D8
alligator	0	2	0	0	2	0	0	0
bear	1	1	0	0	0	1	0	1
cougar	2	0	0	0	0	0	0	0
dolphin	2	2	3	2	1	1	2	3
eagle	3	0	3	4	0	0	4	0

   Answer: Adding the query Q vector to the table:

   Table 2: Word Frequencies per Documents and Query

Word	**D1**	**D2**	Q
alligator	0	2	0
bear	1	1	2
cougar	2	0	1
dolphin	2	2	1
eagle	3	0	0

The TF and IDF of the vocabulary were computed using the following values:

$$\begin{aligned} TF(t) &= \text{term frequency in the corpus} \\ IDF(t) &= \text{inverse document frequency} \\ &= log_2\left(\frac{N}{df(t)}\right), \\ &\text{where:} \\ &N = \text{length of corpus} = 40, \\ &df(t) = \text{document frequency} \end{aligned}$$

Table 3: TF(t) and IDF(t) of Vocabulary

term	**TF**	**IDF**
alligator	4	$log_2\left(\frac{40}{2}\right) = 4.92$
bear	1	$log_2\left(\frac{40}{4}\right) = 3.32$
cougar	2	$log_2\left(\frac{40}{1}\right) = 5.32$
dolphin	2	$log_2\left(\frac{40}{8}\right) = 2.32$
eagle	3	$log_2\left(\frac{40}{4}\right) = 3.32$

Finally, computing the cosine similarities.

Table 4: Cosine Similarities of D1 and D2

value	**D1**	**D2**	**Q**
sum of squares	18	9	6
length, $\|\|D\|\| = \sqrt{\sum_{i=1}^{t} w_i^2}$	4.24	3	2.45
dot product	6	4	6
cosine similarity, $\frac{d \cdot q}{\|\|d\|\| * \|\|q\|\|}$	$\frac{6}{4.24*2.45} = 3.47$	$\frac{4}{3*2.45} = 3.27$	1