CS47300 Fall 2020 Assignment 4 Solutions

1 Web Crawling: Duplicate Elimination

For this question, you will use four-term shingles and Jaccard coefficient to compare documents.

1. Compute the Jaccard Coefficient for the following pair of documents. Please show your work.

   D1: i love cats more than dogs

   D2: i love cats less than dogs

   Shingles for D1: {I love cats more, love cats more than, cats more than dogs}

   Shingles for D2: {I love cats less, love cats less than, cats less than dogs}

   $\frac{0}{6} = 0$

2. Now, compute the Jaccard Coefficient for these next two documents. Please show your work.

   D3: i really really do love dogs

   D4: i really really do love cats

   Shingles for D1: {I really really do, really really do love, really do love dogs}

   Shingles for D2: {I really really do, really really do love, really do love dogs}

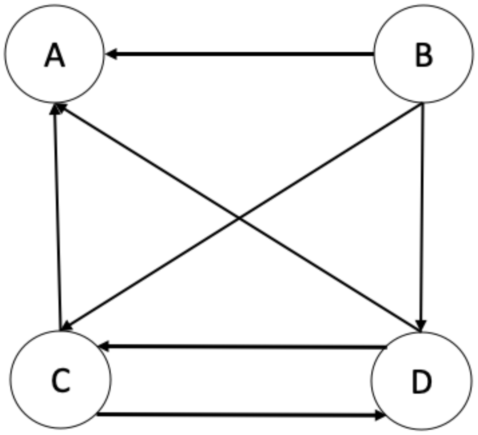
   $\frac{2}{4} = \frac{1}{2}$

   What do you notice about the Jaccard Coefficient scores, in relation to how "similar" each pair of documents is? You should see a a bit of a flaw in this approach - describe what it is and how you mjght deal with it.

You should notice that shingling statistics get skewed towards the end of documents. Words located at the ends of documents often occur in fewer k-shingle for k large. There are many viable ways to attempt to address this issue. A few techniques for solving this issue were: wrap around shingles and using a larger range of k for shingling. While using a larger range of k doesn't necessarily fix the issue, it at least incorporates shingles where the ends of the documents are valuable as well.

2 PageRank

In this graph, A, B, C, and D are four webpages linking to each other.



Assume the probability of following each link is equal.

1. Show the Matrix Notation of these webpages. Order the element in the sequence: A, B, C, D.
   Solution:

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$$

2. Run the HITS algorithm on this graph, simulating the algorithm for three iterations. Show the Authority score and Hubness score for each page in each iteration.
   Iteration 1:
   a(A) = h(B) + h(C) + h(D) = 3
   a(B) = 0
   a(C) = h(B) + h(D) = 2
   a(D) = h(B) + h(C) = 2
   h(A) = 0
   h(B) = a(A) + a(C) + a(D) = 3
   h(C) = a(A) + a(D) = 2
   h(D) = a(A) + a(C) = 2
   Normalized: $a(A) = \frac{3}{7}, a(B) = 0, a(C) = \frac{2}{7}, a(D) = \frac{2}{7}$
   $h(A) = 0, h(B) = \frac{3}{7}, h(C) = \frac{2}{7}, h(D) = \frac{2}{7}$
   Iteration 2:
   a(A) = h(B) + h(C) + h(D) = 7
   a(B) = 0
   a(C) = h(B) + h(D) = 5
   a(D) = h(B) + h(C) = 5
   h(A) = 0
   h(B) = a(A) + a(C) + a(D) = 7
   h(C) = a(A) + a(D) = 5
   h(D) = a(A) + a(C) = 5
   Normalized: $a(A) = \frac{7}{17}, a(B) = 0, a(C) = \frac{5}{17}, a(D) = \frac{5}{17}$
   $h(A) = 0, h(B) = \frac{7}{17}, h(C) = \frac{5}{17}, h(D) = \frac{5}{17}$
   Iteration 3:
   a(A) = h(B) + h(C) + h(D) = 17
   a(B) = 0
   a(C) = h(B) + h(D) = 12
   a(D) = h(B) + h(C) = 12
   h(A) = 0
   h(B) = a(A) + a(C) + a(D) = 17
   h(C) = a(A) + a(D) = 12
   h(D) = a(A) + a(C) = 12
   Normalized: $a(A) = \frac{17}{41}, a(B) = 0, a(C) = \frac{12}{41}, a(D) = \frac{12}{41}$
   $h(A) = 0, h(B) = \frac{17}{41}, h(C) = \frac{12}{41}, h(D) = \frac{12}{41}$

3. Assume the damping factor ($1 - \alpha$ in the formula on slide 40) is 0.85, run the Pagerank algorithm (random walk model) on this graph. Simulate the algorithm for three iteration. Show the pagerank score (normalized) for each page in each iteration. Note: If you did this using the formula on slide 34, with α as the damping factor, that is okay as well. You should be able to easily figure what this will converge to.

   Initially

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\ \frac{1}{2} & 0 & 0 & \frac{1}{2} \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 \end{bmatrix}$$

$$H = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

$$B' = \epsilon H + (1 - \epsilon)B = 0.15H + 0.85B$$

$$B' = \begin{bmatrix} 0.0375 & 0.0375 & 0.0375 & 0.0375 \\ 0.3208 & 0.0375 & 0.3208 & 0.3208 \\ 0.4625 & 0.0375 & 0.0375 & 0.4625 \\ 0.4625 & 0.0375 & 0.4625 & 0.0375 \end{bmatrix}$$

$$R = [\frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}]$$

$$r = B^{'T} r$$

Iteration 1:

$$R = [0.3208, 0.0375, 0.2146, 0.2146]$$

Normalize:

$$R = [0.4037, 0.0472, 0.2700, 0.2700]$$

Iteration 2:

$$R = [0.28, 0.0372, 0.1653, 0.1653]$$

Normalize:

$$R = [0.4324, 0.0575, 0.2553, 0.2553]$$

Iteration 3:

$$R = [0.2708, 0.0375, 0.1623, 0.1623]$$

Normalize:

$$R = [0.4279, 0.0593, 0.2564, 0.2564]$$

4. Please compare HITS and Pagerank. Briefly, elaborate the pros and cons of HITS and Pagerank.

   Both HITS and PageRank are retrieval algorithm based on graph structure. The pros od PageRank is that once the crawler server crawled data, the calculation of PageRank can be done offline. So it does not need to update frequently and response to users rapidly. However, the cons is that PageRank cannot rank a new webpage correctly, because there are not many webpages link to a new page even if it's important.

   The pros of HITS is that HITS requires less iterations and converges quickly. The cons is that it can be affacted by "spam" pages.

5. How close do you think each is converging, and do you think the values you have are close to the values they would converge to?
   HITS is nearly converged, it will converge to aboout $[0.4, 0, 0.3, 0.3]$**. Pagerank is also nearly converged, it will converge to aboout** $[0.4, 0.05, 0.25, 0.25]$

3 Query Expansion

1. Suppose the users of your information retrieval system feel that that none of returned results meet their information needs. Explain the problem in terms of precision and recall. Include an example describing your explanation(s).

   Since the users see no relevant documents, the numerator in both recall and precision (relevant retrieved) is 0. Thus precision is clearly 0. We can't be sure about recall, as it is possible there are no relevant results in the corpus - but if there are, then recall is also 0.

2. Given your example, how would you expect query expansion to affect:

   Query expansion adds terms to the query, typically synonyms or otherwise related terms. This would result in additional documents being retrieved, or perhaps some documents ranked highly that would not have been ranked highly in the original query.

   - Precision
     Since precision is already 0, if any of the retrieved documents are relevant, precision will improve. But this may not always be true, as it could also result in more irrelevant documents being retrieved.
   - Recall
     Recall is likely to improve, as the synonyms may retrieve relevant documents that are not retrieved by the original query, improving recall. If we fix the number of documents retrieved, it is possible that we'll retrieve fewer relevant documents, lowering recall - but in general, we expect that the numerator (relevant retrieved) will go up, and the denominator (relevant documents in corpus) is unchanged, so recall goes up.

4 Relevance Feedback

Given a similar case to your search engine in Question 3.

1. Explain why relevance feedback might be difficult to use. (Please list two reasons, each of them in one sentence is enough.)

   - **Users sometimes are reluctant to provide explicit feedback.**

- **Makes it harder to understand why a particular document was retrieved.**

2. Suppose that a user's initial query is inverse document frequency is importance of a term in a corpus. The user examines two documents, d1 and d2.

   d1 (relevant): inverse document frequency importance in corpus

   d2 (irrelevant): document essential corpus

   Assume that we are using direct term frequency (with no scaling and no document frequency). There is no need to length-normalize vectors. Using Rocchio relevance feedback, what would the revised query vector be after relevance feedback? Assume $\alpha = 0.75$, $\beta = 0.25$. (Note: If using the version given in the books, use $\alpha = 1$, $\beta = 0.75$, $\gamma = 0.25$; the books list a slightly different notation that introduces an additional parameter. If you already did it using the version in the slides with $\alpha = 1$, $\beta = 0.75$, and ignoring γ, then that is fine. Although for the theory to work out right, the parameters for relevant and non-relevant documents should sum to 1.)

   Documents: Inverse document frequency importance in corpus essential. Assume α is 0.75, β is 0.25, using Rocchio relevance feedback.
   $q' = q + \alpha * (1/|R|) * (\sum_{d_i \in R} d_i) - \beta * (1/|NR|) * (\sum_{d_i \in R} d_i)$**,**
   $q' = [1, 1, 1, 1, 1, 1, 0] + 0.75[1, 1, 1, 1, 1, 1, 0] - 0.25 * [0, 1, 0, 0, 0, 1, 1]$**,**
   then, $q' = [1.75, 1.50, 1.75, 1.75, 1.75, 1.50, -0.25]$

5 Text Categorization

For this question, you will be presented with training data documents, with category labels "P"etLover or "M"otorhead. Perform KNN classification on the new document DN with n = 2, using TF-IDF vectorization of the documents. Please show your work.

DocID	Text	Category
D1:	i love cats	PetLover
D2:	i love dogs	PetLover
D4:	i love cars	Motorhead
D5:	i hate cars	PetLover
D6:	i hate cats	Motorhead

New document to categorize (DN):
DN: i love cats dogs

Solution:

IDF ($log_2(N/n)$) values based on training data.

i	love	cats	dogs	cars	hate
0	0.737	1.322	2.322	1.322	1.322

TF-IDF values for train documents.

DocID	i	love	cats	dogs	cars	hate
D1	0	0.737	1.322	0	0	0
D2	0	0.737	0	2.322	0	0
D4	0	0.737	0	0	1.322	0
D5	0	0	0	0	1.322	0
D6	0	0	1.322	0	0	1.322

TF-IDF values of DN

DocID	i	love	cats	dogs	cars	hate
DN	0	0.737	1.322	2.322	0	0

Compute cosine similarity between DN (i.e., vector) and all documents (i.e., vectors) in the training.

Score: D1:0.546 D2:0.879 D4:0.129 D5:0 D6:0.337

Closest two neighbour documents of DN are D1, and D2. Label of D1, and D2 is PetLover, thus DN is classified as PetLover.

Note that, depending on the IDF formula, stopword removal, or stemming, the TF-IDF values might differ. However, that should not impact the final outcome.