CS47300 Fall 2017 Midterm 1 Solutions, October 11, 2017
Prof. Chris Clifton

Turn Off Your Cell Phone. Use of any electronic device during the test is prohibited. As previously noted, you are allowed notes: Up to two sheets of 8.5x11 or A4 paper, single-sided (or one sheet double-sided).

Time will be tight. If you spend more than the recommended time on any question, go on to the next one. If you can’t answer it in the recommended time, you are either giving too much detail or the question is material you don’t know well. You can skip one or two parts and still demonstrate what I believe to be an A-level understanding of the material.

Note: It is okay to abbreviate in your answers, as long as the abbreviations are unambiguous and reasonably obvious.

In all cases, it is important that you give some idea of how you derived the answer, not simply give an answer. Setting up the derivation correctly, even if you don’t carry out the calculations to get the final answer, is good for nearly full credit.

1 Indexing and Boolean Retrieval (7 minutes, 6 points)
Consider following documents in a corpus:

D1: The weather was pleasant this summer
D2: Hot weather results in crop loss
D3: Summer crop loss in Indiana

A. Create an inverted list for the given corpus with basic stemming with one rule: words-s to word and stopwords “for”, “in”, “is”, “the”, “this”, “was”

weather D1, D2
pleasant D1
summer D1, D3
hot D2
result D2
crop D2, D3
los D2, D3
indiana D3

Scoring: 1 for all relevant words, 1 for no stopwords, 1 for documents

B. Give the documents retrieved by a search query using a boolean retrieval model with the OR operation using query words: crop loss Indiana

D2, D3

Scoring: 1 for correct result

C. Give the documents retrieved by a search query using a boolean retrieval model with the AND operation using query words: Indiana summer crops

D3

Scoring: 1 for correct result

1 for showing evidence of stemming anywhere.
2 TFIDF (6 minutes, 4 points)

Given the following set of documents:

D1: The weather was pleasant this summer
D2: Hot weather results in crop loss
D3: Crop Report: summer crop loss in Indiana

A. Calculate the term frequency of the term “crop”

<table>
<thead>
<tr>
<th>Document</th>
<th>Term Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>D1</td>
<td>0</td>
</tr>
<tr>
<td>D2</td>
<td>1 (or 1/5 or 1/6)</td>
</tr>
<tr>
<td>D3</td>
<td>2 (or 2/6 or 2/7)</td>
</tr>
</tbody>
</table>

*Scoring: 1 for showing for each document, 1 for any reasonable TF weighting*

B. Calculate the Inverse Document Frequency of the term “crop”

\[ \frac{3}{2} \text{ or } \log(\frac{3}{2}) \]

*Scoring: 1 for counting documents containing term, 1 for any reasonable IDF weighting formula capturing “inverse”*

Note that there can be different ways of calculating both TF and IDF, you should show what you are using.

3 Cosine Similarity (8 minutes, 5 points)

Given the following term/document weight matrix:

<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baltimore</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Cake</td>
<td>0</td>
<td>2</td>
<td>2</td>
<td>0</td>
</tr>
<tr>
<td>Crab</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Seafood</td>
<td>1</td>
<td>0</td>
<td>4</td>
<td>1</td>
</tr>
</tbody>
</table>

A. Calculate the cosine similarity between the query “Crab Cake” and Document B.

Since the vectors point in the same direction, the angle is 0, \( \cos(0) = 1 \).

*Scoring: 1 for setting up problem correctly, 1 for correct answer*

B. Determine the top two documents for the query “Crab Cake” and rank them by cosine similarity. Show your work.

B, A, B is figured above, D is 0 (same argument).

\[
\frac{\vec{q} \cdot \vec{A}}{||q|| \cdot ||A||} = \frac{0+0+2+0}{\sqrt{1^2+1^2\cdot\sqrt{1^2+0^2+2^2+1^2}}} = \frac{2}{\sqrt{6}\sqrt{2}}
\]

\[
\frac{\vec{q} \cdot \vec{C}}{||q|| \cdot ||C||} = \frac{0+2+1+0}{\sqrt{1^2+1^2\cdot\sqrt{1^2+2^2+1^2+4^2}}} = \frac{3}{\sqrt{2}\sqrt{22}}
\]

*Scoring: 1 for picking top one, 1 for getting correct second, 1 for evidence of figuring cosine similarity. Admittedly, figuring square roots in your head is a pain. But I asked for the rank by cosine similarity, not to calculate cosine similarity. Since everything is positive,
\( a > b \Rightarrow \sqrt{a} > \sqrt{b} \). Square everything - it is pretty easy to see that 4/12 is greater than 9/44. If you don’t think you should be graded on figuring this out, there is a reason this is important to IR - you could ask your computer to do millions of square roots, or millions of multiplications. Which do you think will be faster, and use less power?

4 Latent Semantic Indexing (7 minutes, 5 points)

Using Latent Semantic Indexing, the cosine similarity between three documents and a query in the LSI concept space has been computed as:

- Document A: 0.20
- Document B: -0.60
- Document C: 0.02

A. Rank the documents in terms of how good a match they are to the query.

   Document A, Document B, Document C

   Scoring: 1 for correct rank

B. Does this tell you anything about how good the best match is? Explain.

   Not really, as it is measuring an angle in the concept space. If the query isn’t a good match for any of the concepts, the similarity to a document may be low even if the document really is a good match for the query.

   Scoring: 1 for “no”, 1 for explanation

C. Does the negative similarity value for document B suggest that there may be an error in the singular value decomposition?

   No, it is perfectly normal for Singular Value Decomposition to introduce negative values in the matrix, giving the possibility of a negative cosine similarity.

   Scoring: 1 for “no”, 1 for discussion that negatives can occur.

5 Probabilistic Retrieval (8 minutes, 6 points)

With a probabilistic retrieval model, we often compute \( P(\tilde{d}|R=1, \vec{q}) \): The conditional probability of a document given the query and the assumption it is relevant.

A. If we assume term independence, give a formula to calculate the above probability given that we know for each term \( i \), \( P(d_i|R=1, q) \).

   \[ \prod_i P(d_i|R=1, q_i) \]

   Scoring: 1 for multiplication of terms, 1 for completely correct.

B. If we have \( P(\tilde{d}|R=1, \vec{q}) = 0.08 \), does this mean the document \( \tilde{d} \) is a good match for the query, a bad match for the query, or can we not determine this? Explain.

   We really can’t determine if this means a good match or not. This is the probability that given the query and that a document is relevant, it is this particular document. Out of all possible documents. This would seem like a pretty low probability even for a relevant document, given the number of possible documents, and how many of them could be relevant.
Scoring: 1 for “can not determine”, 1 for probabilities likely to be low, it is rank that matters, or discussion of odds ratio.

C. In probabilistic retrieval models, we often multiply the probabilities of individual terms. If a term is missing from a document (i.e., it’s probability of occurrence in the document is 0), this would give us a 0 for the overall probability. What do we do to prevent such 0 probabilities? Explain.

Smoothing. There are various techniques, but they generally add some small value to every value (e.g., assume every document starts with 1 of every word, and then add the actual numbers to that.)

Scoring: 1 for “smoothing”, 1 for formula or solid explanation of what smoothing does.

6 Evaluating ad-hoc information retrieval (10 minutes, 8 points)

With pseudo-relevance feedback, we used the results of a query to do query expansion. Pick three ways of measuring the quality of the result of a query, and for each, say if you would expect pseudo-relevance feedback to improve the results, give worse results, or result in no change. Explain your answers.

A. Recall. Pseudo-relevance feedback will expand the search based on words in the top-ranked documents, and negatively weight words found in the bottom ranked documents; this will likely give additional documents not found by the original query. As Recall is \[
\frac{\text{relevant retrieved}}{\text{total relevant}}
\], and total relevant doesn’t change, retrieving more documents is likely to increase relevant retrieved and increase recall.

B. Precision. Since both the numerator (relevant retrieved) and denominator (total retrieved) change, it is difficult to predict the impact, but precision could easily go down.

C. There are many other good answers - mean average precision, F1 score, ... There were some very impressive answers involving some of the more complex metrics, particularly mathematical proofs of why these would or would not be changed by pseudo-relevance feedback.

Scoring: 1 for each correct metric, 1 for each reasonable explanation (even if it was a good explanation of something that wasn’t an evaluation metric), 1 for showing understanding of relevance feedback across all three, 1 for mathematical explanation in any of the three answers.