CS371R: Sample Solution to Midterm Exam

October 17, 2019

NAME: _____

1. (13 points) Corpus C consists of the following three documents:

   "new york times"
   "new york post"
   "los angeles times"

   Assuming that term frequencies are normalized by the maximum frequency in a given document, calculate the TF-IDF weighted term vectors for all documents in C. Assume that words in the vectors are ordered alphabetically.

Answer:

Term frequencies:

	angeles	los	new	post	times	york
"new york times"	0	0	1	0	1	1
"new york post"	0	0	1	1	0	1
"los angeles times"	1	1	0	0	1	0

Inverse document frequencies:

angeles	los	new	post	times	york
$\log_2 \frac{3}{1}$	$\log_2 \frac{3}{1}$	$\log_2 \frac{3}{2}$	$\log_2 \frac{3}{1}$	$\log_2 \frac{3}{2}$	$\log_2 \frac{3}{2}$

Since $\log_2 3 = 1.5850$ and $\log_2 \frac{3}{2} = 0.5850$, we have the following TF-IDF weighted term vectors:

	angeles	los	new	post	times	york
"new york times"	0	0	0.5850	0	0.5850	0.5850
"new york post"	0	0	0.5850	1.5850	0	0.5850
"los angeles times"	1.5850	1.5850	0	0	0.5850	0

2. (14 points) Given the following query:

   "chai pumpkin spice pumpkin muffin"

   calculate the TF weighted query vector (no IDF factor), and compute the score of each document below using the cosine similarity measure. Assume that term frequencies are normalized by the maximum frequency in a given query.

   Document vectors:

	chai	latte	muffin	pumpkin	spice	tea
"chai tea latte"	1	1	0	0	0	1
"pumpkin spice latte"	0	1	0	1	1	0
"chai latte muffin"	1	1	1	0	0	0

Answer:

Fill in the values of the query vector in the table below:

chai	latte	muffin	pumpkin	spice	tea
0.5	0	0.5	1	0.5	0

The vector lengths for the query and the documents are:

"chai pumpkin spice pumpkin muffin"	$\sqrt{0.5^2 + 0.5^2 + 1^2 + 0.5^2} = 1.3229$
"chai tea latte"	$\sqrt{1^2 + 1^2 + 1^2} = 1.7321$
"pumpkin spice latte"	$\sqrt{1^2 + 1^2 + 1^2} = 1.7321$
"chai latte muffin"	$\sqrt{1^2 + 1^2 + 1^2} = 1.7321$

Hence, the scores of the documents in terms of cosine similarity are:

"chai tea latte"	$(0.5 \times 1)/(1.3229 \times 1.7321) = 0.2182$
"pumpkin spice latte"	$(1 \times 1 + 0.5 \times 1)/(1.3229 \times 1.7321) = 0.6546$
"chai latte muffin"	$(0.5 \times 1 + 0.5 * 1)/(1.3229 \times 1.7321) = 0.4364$

3. (13 points) A user makes the query "cheap austin flights" a document corpus and gets the ranked results in the table below. The document vectors for each document is next to the document. The stop word "in" has been removed.

	austin	cheap	flights	kayak	rental
1. "kayak cheap flights" (Cosine Similarity: 2/3)	0	1	1	1	0
2. "cheap kayak rental" (Cosine Similarity: 1/3)	0	1	0	1	1
3. "kayak in austin" (Cosine Similarity: 1/3)	1	0	0	1	0

The query vector for "cheap austin flights" is:

austin	cheap	flights	kayak	rental
1	1	1	0	0

Answer:

The highest ranked among the irrelevant documents is "kayak cheap flights" according to the results above. Therefore, the query is reformulated by subtracting the document vector for "kayak cheap flights" from the query vector:

$$\begin{aligned}
\overrightarrow{q_m} &= \alpha \overrightarrow{q} - \gamma \max_{non-relevant} (\overrightarrow{d_j}) \\
&= \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ 1 \\ 1 \\ 1 \\ 0 \end{bmatrix} \\
&= \begin{bmatrix} 1 \\ 0 \\ 0 \\ -1 \\ 0 \end{bmatrix}
\end{aligned}$$

4. (14 points) Assume that an IR system returns a ranked list of 10 total documents for a given query. Assume that according to a gold-standard labelling there are 4 relevant documents for this query, and that the relevant documents in the ranked list are in the 1st, 3rd, 5th, and 7th positions in the ranked results. Calculate and clearly show the interpolated precision value for each of the following standard recall levels: {0.0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0} for this individual query.

Answer:

Document Number	Recall	Precision
1	$1/4 = 0.25$	$1/1 = 1.0$
3	$2/4 = 0.5$	$2/3 = 0.67$
5	$3/4 = 0.75$	$3/5 = 0.6$
7	$4/4 = 1.0$	$4/7 = 0.57$

Table 1: Precision-Recall values corresponding to relevant documents positions

Recall	Precision
0.0	1.0
0.1	1.0
0.2	1.0
0.3	0.67
0.4	0.67
0.5	0.67
0.6	0.6
0.7	0.6
0.8	0.57
0.9	0.57
1.0	0.57

Table 2: Interpolated Precision-Recall values

5. (13 points) Given a corpus that consists of the following two documents:

   "new orleans"
   "new hampshire"

   Compute a normalized association matrix that quantifies term correlations in terms of how frequently they co-occur. Order terms in the matrix alphabetically.

Answer:

The *unnormalized* association matrix C is as follows:

	hampshire	new	orleans
hampshire	1	1	0
new		2	1
orleans			1

By applying $s_{ij} = c_{ij}/(c_{ii} + c_{jj} - c_{ij})$, we have the following normalized association matrix:

	hampshire	new	orleans
hampshire	$\frac{1}{1+1-1} = 1$	$\frac{1}{1+2-1} = 0.5$	0
new		$\frac{2}{2+2-2} = 1$	$\frac{1}{1+2-1} = 0.5$
orleans			$\frac{1}{1+1-1} = 1$

6. (12 points) What is the Levenshtein distance between the following pairs of strings? List the edit operations you used to transform the first string into the second string to find the Levenshtein distance.

"beauracracy" and "bureaucracy"

Answer:
The Levenshtein distance is 4. Sample edit operations to convert 1st string to 2nd: delete first 'e', delete first 'a', insert 'e' after first 'r', insert 'u' before first 'c'

"Levenshtein" and "Levanstine"

Answer:

The Levenshtein distance is 4. Sample edit operations to convert 1st string to 2nd: substitute 'a' for second 'e', delete 'h', delete last 'e', add 'e' at the end.

7. (21 points) Provide short answers (1-3 sentences) for each of the following questions:

- Why is vector inner product (dot product) alone not a good similarity metric for vector space retreival, i.e. why is normalizing by the length of the vectors important?

  Answer: (Retrieval models slide 22)

  - Inner product is unbounded
  - Favors long documents with a large number of unique terms
  - Measures how many terms are matched but not how many are *not* matched.

- What are the two major limitations of the Porter Stemmer?

  Answer: (VSR Implementation slides 8-9)

  - May conflate words that are actually semantically distinct (organize, organ both stem to organ)
  - May not recognize the true common root of morphologically complex terms (create, creation not share the same stem)

- When evaluating relevance feedback, or any machine learning method that uses training data, what must be true about the data used to test the system?

  Answer: The test data must be disjoint from the training data to avoid "testing on the training data."

- List three other items that exhibit a Zipfian distribution other than the frequency of words in text and number of in-links on the web?

  Answer: (Text Properties slide 7)

  - Wealth of individuals
  - Population of cities
  - Popularity of books or movies

- How should one order the items in the queue in a web spider in order to implement a "directed/focused" crawler?

  Answer: Sort the queue by a heuristic comparison function that prefers one page over another based on some property of interest.

- How does one construct a random directed graph that, like the web, is "scale free" where the number of edges into a node exhibits a Zipfian power-law distribution?

  Answer:

  Preferential attachment, a version of "rich get richer," where an edge into a node is probabilistically added in proportion to its existing number of incoming edges.

- In Matt Lease's automatic fact checking system, given a claim, relevant article headlines, and the trustworthiness of the source of each article, what are the two values that the system predicts?

  Answer:

(a) Headline stance - Is the article for, against, or observing/neutral about the claim?

(b) Claim veracity - Is the claim true or false?

- (Extra credit) From what conference was Tim Berner Lee's first paper about the world-wide-web rejected?

  Answer: ACM Hypertext Conference

- (Extra credit) Herbert Simon, one of the founders of AI and investigators of the cause of Zipfian distributions, is the only recipient of both of what two major scientific awards (be specific)?

  Answer: The Nobel Memorial Prize in Economic Sciences and the Turing Award