CSI 4107
Information Retrieval and the Internet

FINAL EXAMINATION

Length of Examination: 3 hours
Professor: Diana Inkpen

April 21, 2016, 9:30
Page 1 of 11

Family Name: _________________________________
Other Names: _________________________________
Student Number: ________
Signature ____________________________

Important Regulations:
1. Students are allowed to bring in a page of notes (written on one side).
2. Calculators are allowed.
3. Les réponses en français sont acceptées.
4. A student identification cards (or another photo ID and signature) is required.
5. An attendance sheet shall be circulated and should be signed by each student.
6. Please answer all questions on this paper, in the indicated spaces.

Marks:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>9</td>
<td>2</td>
<td>12</td>
<td>10</td>
<td>45</td>
</tr>
</tbody>
</table>
Part A
Short answers and explanations. [14 marks]

1. (3 marks) Explain how can language models be used for Information retrieval. What assumption is made for the query and for the user who inputs the query?

Each document $d$ is represented as a language model $M_d$, based on unigram probabilities (computed based on frequency).

$$P(d \mid q) = P(q \mid d) \times P(d) / P(q) = P(q \mid M_d) \times P(d) / P(q)$$

The documents are ranks based on the probability of relevance to the query $P(d \mid q)$.

The query is assumed to be generated from relevant documents according to their language models. The user is assumed to input a query that uses words that he/she expects to find in relevant documents (otherwise the user would retrieve too many non-relevant documents).

2. (2 marks) Query expansion can be done with synonyms from a thesaurus (dictionary of synonyms) or with similar terms from the collection of documents (for the latter method assume that word similarity is computed for any two terms based on their frequency of co-occurrence). Explain the advantages and disadvantages of the two methods.

Using thesaurus
  Advantages:
   - it is reliable the similar words are correct, they were produces by specialists
   Disadvantages:
   - computationally expensive to lookup in the thesaurus at run-time
   - problem with multiple words senses
   - many words in the text collection will be missing from the thesaurus (for example proper names)

Using similarity matrix derived from text collection
   Advantages:
   - all the words will have synonyms from the collection (including names)
   Disadvantages:
   - similarity is not always reliable
   - computationally intensive to build the matrix, but it can be done offline if global analysis
3. (2 marks)

How can you use supervised classification algorithms to learn to rank documents as answers to queries in an information retrieval system? Assume you have training queries and expected solution/relevance judgements, but then you will need to run the system on other test queries. What kind of features would you use in the classifier?

You can use the training data to learn weights for combining various sets of features that are not related to the query’s topic (such as tf-idf values, document zones, document authority, document date, etc).

4. (2 marks) Latent Semantic Indexing claims to index latent concepts rather than words. Explain in a few sentences how this is accomplished.

LSI uses Singular Value Decomposition to transform the original document by word matrix into a lower-dimensional space. The smaller number of columns corresponds to the latent concepts. Each concept corresponds to a group of synonyms (but it is not known which ones, the concepts are not explicit).
5. (2 marks) In a question answering system, several candidate answers can be produced as an answer to a question. Answer validation means to choose which of the candidate answers is the best choice. Explain how can you do answer validation using results of Web searches.

Assume that q is the relevant part of the question.
Assume that the candidate answers are a1, a2, …

Web search is done for q, a1, a2, … and for combinations q a1, q a2, …
The number of hits from the Web (document counts) is returned by a search engine

Then mutual information score are computed (PMI):
Score(a1) = hits(q a1) / hits(q) hits(a1)
Score(a1) = hits(q a2) / hits(q) hits(a2)
….

The answer with the highest score is selected as the correct answer to the question.

6. (2 marks) The IBM Watson question-answering system used many techniques to improve over other question answering systems, in terms of speed of answering a query and correctness of the results. Enumerate three such improvements.

- Massive parallelism to speed up the answers (map reduce)
- Integration of shallow and deep knowledge
- Preprocessed Wikipedia knowledge
- Better confidence estimation regarding the correctness of the answers
Given a query q, where the relevant documents are d3, d12, d15, d21, d22, and d30, an IR system retrieves the following ranking: d5, d3, d21, d36, d30, d45, d80, d28, d23, and d12.

1. (2 marks) What are the precision and recall for this ranking?

<table>
<thead>
<tr>
<th>Document</th>
<th>Recall</th>
<th>Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>d5</td>
<td>0/6 = 0</td>
<td>0/1 = 0</td>
</tr>
<tr>
<td>d3</td>
<td>1/6 = 0.166</td>
<td>1/2 = 0.50</td>
</tr>
<tr>
<td>d21</td>
<td>2/6 = 0.33</td>
<td>2/3 = 0.66</td>
</tr>
<tr>
<td>d36</td>
<td>2/6 = 0.33</td>
<td>2/4 = 0.50</td>
</tr>
<tr>
<td>d30</td>
<td>3/6 = 0.50</td>
<td>3/5 = 0.60</td>
</tr>
<tr>
<td>d45</td>
<td>3/6 = 0.50</td>
<td>3/6 = 0.50</td>
</tr>
<tr>
<td>d80</td>
<td>3/6 = 0.50</td>
<td>3/7 = 0.42</td>
</tr>
<tr>
<td>d28</td>
<td>3/6 = 0.50</td>
<td>3/8 = 0.37</td>
</tr>
<tr>
<td>d23</td>
<td>3/6 = 0.50</td>
<td>3/9 = 0.33</td>
</tr>
<tr>
<td>d12</td>
<td>4/6 = 0.66</td>
<td>4/10 = 0.40</td>
</tr>
</tbody>
</table>

2. (2 marks) Interpolate the precision scores at 11 recall levels. Note: The interpolated precision at the j-th standard recall level is the maximum known precision at any recall level between the j-th and (j+1)-th level:

\[ P(r_j) = \max_{r_j \leq r \leq r_{j+1}} P(r) \]

<table>
<thead>
<tr>
<th>Recall</th>
<th>Interpolated Precision</th>
</tr>
</thead>
<tbody>
<tr>
<td>0%</td>
<td>0.66</td>
</tr>
<tr>
<td>10%</td>
<td>0.66</td>
</tr>
<tr>
<td>20%</td>
<td>0.66</td>
</tr>
<tr>
<td>30%</td>
<td>0.66</td>
</tr>
<tr>
<td>40%</td>
<td>0.6</td>
</tr>
<tr>
<td>50%</td>
<td>0.6</td>
</tr>
<tr>
<td>60%</td>
<td>0.4</td>
</tr>
<tr>
<td>70%</td>
<td>0</td>
</tr>
<tr>
<td>80%</td>
<td>0</td>
</tr>
<tr>
<td>90%</td>
<td>0</td>
</tr>
<tr>
<td>100%</td>
<td>0</td>
</tr>
</tbody>
</table>
3. (1 mark) What is the R-precision? (precision at first \( R \) retrieved documents where \( R \) is the total number of relevant documents)

| R-Precision | 3/6 = 0.5 |

4. (4 marks) If we have two users, the first user knew prior to the search that \( d_{12}, d_{15}, d_{21} \) are relevant to the query, and the second user knew that \( d_{3}, d_{12}, d_{15}, d_{21}, d_{22} \) are relevant to the query, what is the coverage ratio and the novelty ratio for this user? (Remember that the coverage ratio is the proportion of relevant items retrieved out of the total relevant documents known to a user prior to the search. The novelty ratio is the proportion of retrieved items, judged relevant by the user, of which they were previously unaware.)

<table>
<thead>
<tr>
<th>Coverage ratio</th>
<th>Novelty ratio</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1</td>
<td>2/3 = 0.66</td>
</tr>
<tr>
<td>User 2</td>
<td>3/5 = 0.60</td>
</tr>
</tbody>
</table>
Part C [2 marks]
Assume we have documents with two zones: titles and bodies. We want to learn how to score documents for the degree of matching queries, using training data to automatically learn the optimal weight $w$ in the following scoring formula.

$$score(d, q) = w \cdot S_T(d, q) + (1 - w)S_B(d, q)$$

where $S_T(d, q)$ is 1 if query $q$ matches document $d$ in the title and zero otherwise. $S_B(d, q)$ is 1 if query $q$ matches document $d$ in the body and zero otherwise.

Assume we have the 7 training examples from the table below. We want to minimize the difference between the scores on the training data and the relevance judgments. After building an equation to measure this difference, we get the the $w$ for which the difference is minimal by deriving the formula and obtain that the optimal value for $w$ is:

$$w = \frac{n_{10r} + n_{01i}}{n_{10r} + n_{10i} + n_{01r} + n_{01i}},$$

where $n_{jk}$ is the number of training examples that have $S_T = j$ and $S_B = k$ for the class Relevant and $n_{jk}$ is the number of training examples that have $S_T = j$ and $S_B = k$ for the class Non-relevant (irrelevant).

<table>
<thead>
<tr>
<th>Example</th>
<th>DocID</th>
<th>Query</th>
<th>$S_T$</th>
<th>$S_B$</th>
<th>Judgment</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Phi_1$</td>
<td>37</td>
<td>linux</td>
<td>1</td>
<td>1</td>
<td>Relevant</td>
</tr>
<tr>
<td>$\Phi_2$</td>
<td>37</td>
<td>penguin</td>
<td>0</td>
<td>1</td>
<td>Non-relevant</td>
</tr>
<tr>
<td>$\Phi_3$</td>
<td>238</td>
<td>system</td>
<td>0</td>
<td>1</td>
<td>Relevant</td>
</tr>
<tr>
<td>$\Phi_4$</td>
<td>238</td>
<td>penguin</td>
<td>0</td>
<td>0</td>
<td>Non-relevant</td>
</tr>
<tr>
<td>$\Phi_5$</td>
<td>1741</td>
<td>kernel</td>
<td>1</td>
<td>1</td>
<td>Relevant</td>
</tr>
<tr>
<td>$\Phi_6$</td>
<td>2094</td>
<td>driver</td>
<td>0</td>
<td>1</td>
<td>Relevant</td>
</tr>
<tr>
<td>$\Phi_7$</td>
<td>3191</td>
<td>driver</td>
<td>1</td>
<td>0</td>
<td>Non-relevant</td>
</tr>
</tbody>
</table>

What is optimal value of $w$ in this case? (hint compute first the $n_{jk}$ values)

$$w = \frac{0 + 1}{0 + 1 + 2 + 1} = \frac{1}{4} = 0.25$$

$n_{00r} = 0 \quad n_{00i} = 1$
$n_{01r} = 2 \quad n_{01i} = 1$
$n_{10r} = 0 \quad n_{10i} = 1$
$n_{11r} = 2 \quad n_{11i} = 0$
Part D

[12 marks]

D. 1. Assume the following documents are in the training set, classified into two classes:

- **Music**: “romantic night”
- **Music**: “silent night”
- **Movie**: “romantic comedy”
- **Movie**: “real action comedy”

a. (3 marks) Apply the Rocchio algorithm to classify a new document: “real night”

Use tf without idf and normalization, with cosine similarity. Assume the words in the vectors are ordered alphabetically. Show the prototype vectors for the two classes, and their similarities to the new document. (Remember that a prototype is the sum of the vectors in that class.)

<table>
<thead>
<tr>
<th></th>
<th>action</th>
<th>comedy</th>
<th>night</th>
<th>real</th>
<th>romantic</th>
<th>silent</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>d3</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>d4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>q</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Music = p1 = d1 + d2 = <0,0,2,0,1,1>
Movie = p2 = d3 + d4 = <1,2,0,1,1,0>

\[
\text{cosSim}(q, p1) = \frac{2}{\sqrt{2} \sqrt{6}} \Rightarrow \text{New document is Music (more similar to p1)}
\]

\[
\text{cosSim}(q, p2) = \frac{1}{\sqrt{2} \sqrt{7}}
\]

b. (3 marks) Apply kNN with k=3 to classify the new document: “real night”

Use tf without idf and normalization, with cosine similarity.

Would the result be the same if k=1? Why?

\[
\text{cosSim}(q, d1) = \frac{1}{\sqrt{2} \sqrt{2}} = \frac{1}{2}
\]

\[
\text{cosSim}(q, d2) = \frac{1}{\sqrt{2} \sqrt{2}} = \frac{1}{2}
\]

\[
\text{cosSim}(q, d3) = 0
\]

\[
\text{cosSim}(q, d4) = \frac{1}{\sqrt{2} \sqrt{3}}
\]

With K=3 the closest documents are d1, d2, d3 => New document is classified as Music
With K=1 the closest document is d1 => New document is classified as Music
D. 2. (6 marks) Cluster the following documents using K-means with K=2 and cosine similarity.

Doc1: “run dog run”
Doc2: “run horse”
Doc3: “dog dog”
Doc4: “horse dog”
Doc5: “run horse run”

Assume Doc1 and Doc4 are chosen as initial seeds. Use tf (without idf and normalization) and cosine similarity. Assume the words in the vectors are ordered alphabetically.

Show the clusters and their centroids for each iteration. (Remember that a centroid is the average of the vectors in that cluster.) How many iterations are needed for the algorithm to converge?

<table>
<thead>
<tr>
<th></th>
<th>dog</th>
<th>horse</th>
<th>run</th>
<th>length</th>
</tr>
</thead>
<tbody>
<tr>
<td>d1</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>sqrt(5)</td>
</tr>
<tr>
<td>d2</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>sqrt(2)</td>
</tr>
<tr>
<td>d3</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>sqrt(4)</td>
</tr>
<tr>
<td>d4</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>sqrt(2)</td>
</tr>
<tr>
<td>d5</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>sqrt(5)</td>
</tr>
</tbody>
</table>

---------------------------------------

Iteration 1

\[ c_1 = d_1 \]
\[ c_2 = d_4 \]

Compute cosSim => d1,d2,d5 in c1 \ d3,d4 in c2

---------------------------------------

Iteration 2

\[ c'_1 = \frac{d_1 + d_2 + d_5}{3} = \langle \frac{1}{3}, \frac{2}{3}, \frac{5}{3} \rangle \]
\[ c'_2 = \frac{d_3 + d_4}{2} = \langle \frac{3}{2}, 1/2, 0 \rangle \]

Compute cosSim => d1,d2,d5 in c1 \ d3,d4 in c2

The algorithm converged.
Part E

Consider the following web pages and the links between them:
- Page A points to pages B, C, and D.
- Page B points to page E.
- Page C points to pages B and D.
- Page D points to page E.
- Page E points to pages A and C.

E. 1. (5 marks) Run the Hubs and Authorities algorithms on this subgraph of pages. Show the authority and hub scores for each page for two iterations. Present the results in the order A,B,,D,E.

Remember that the Hubs and Authorities algorithms can be described in pseudo-code as:
- Initialize for all \( p \in S \): \( a_p = h_p = 1 \)
- For \( i = 1 \) to \( \text{No}\_\text{iterations} \):
  - For all \( p \in S \): update authority scores
  - For all \( p \in S \): update hub scores

\[
\begin{align*}
\text{In} & \quad \text{Out} & \quad \text{It 0} & \quad \text{It 1} & \quad \text{Normalized} & \quad \text{It 2} & \quad \text{Normalized} \\
A & \quad E & \quad B,C,D & 1 & 1 & 1 & 3 & 0.11 & 0.33 & 0.22 & 0.66 & 0.10 & 0.35 \\
B & \quad A,C & \quad E & 1 & 1 & 2 & 1 & (1/9) & (3/9) & 0.22 & 0.11 & 0.55 & 0.22 & 0.26 & 0.11 \\
C & \quad A,E & \quad B, D & 1 & 1 & 2 & 2 & (2/9) & (1/9) & 0.22 & 0.22 & 0.55 & 0.44 & 0.26 & 0.23 \\
D & \quad A,C & \quad E & 1 & 1 & 2 & 1 & (2/9) & (1/9) & 0.22 & 0.11 & 0.55 & 0.22 & 0.26 & 0.11 \\
E & \quad B,D & \quad A,C & 1 & 1 & 2 & 2 & (2/9) & (2/9) & 0.22 & 0.22 & 0.22 & 0.33 & 0.10 & 0.17 \\
\text{Sum} & \quad 9 & \quad 9 & \quad \text{Sum} & \quad 2.09 & \quad \text{Sum} & \quad 1.87 \\
\end{align*}
\]
D.2. (5 marks) For the same graph, run the PageRank algorithm for two iterations.

- Page A points to pages B, C, and D.
- Page B points to page E.
- Page C points to pages B and D.
- Page D points to page E.
- Page E points to pages A and C.

Remember that one way to describe the algorithm is:

\[ PR(A) = (1 - d) + d \left( \frac{PR(T_1)}{C(T_1)} + \ldots + \frac{PR(T_n)}{C(T_n)} \right) \]

where \( T_1 \ldots T_n \) are the pages that point to a page A (the incoming links), \( d \) is damping factor (usually \( d = 0.85 \)), \( C(A) \) is number of links going out of a page A and \( PR(A) \) is the PageRank of a page A. NOTE: the sum of all pages’ PageRank is 1. Include the normalization step (after each iteration, divide each score by the sum of the scores for all the pages). Trace the algorithm for two iterations. Use initial PR values 1 for all nodes. Use \( d = 0.85 \). What is the order of the pages after the two iterations?

\[
\begin{align*}
P(A) &= 0.15 + 0.85 \times \frac{P(E)}{2} \\
P(B) &= 0.15 + 0.85 \times \frac{P(A)}{3} + \frac{P(C)}{2} \\
P(C) &= 0.15 + 0.85 \times \left( \frac{P(A)}{3} + \frac{P(E)}{2} \right) \\
P(D) &= 0.15 + 0.85 \times \left( \frac{P(A)}{3} + \frac{P(C)}{2} \right) \\
P(E) &= 0.15 + 0.85 \times \left( \frac{P(B)}{3} + \frac{P(D)}{2} \right)
\end{align*}
\]

a) Initial values:

\[
P(A) = 1 \quad P(B) = 1 \quad P(C) = 1 \quad P(D) = 1 \quad P(E) = 1
\]

Iteration 1

\[
\begin{align*}
P(A) &= 0.15 + 0.85 \times \frac{1}{2} = 0.575 \\
P(B) &= 0.15 + 0.85 \times \left( \frac{1}{3} + \frac{1}{2} \right) = 0.8583 \\
P(C) &= 0.15 + 0.85 \times \left( \frac{1}{3} + \frac{1}{2} \right) = 0.8583 \\
P(D) &= 0.15 + 0.85 \times \left( \frac{1}{3} + \frac{1}{2} \right) = 0.8583 \\
P(E) &= 0.15 + 0.85 \times (1 + 1) = 1.85 \\
\text{Normalization: } P(A) &= 0.575 / 5.0005 = 0.1149 \\
P(B) &= 0.8583 / 5.0005 = 0.1716 \\
P(C) &= 0.1716 \quad P(D) = 0.1716 \quad P(E) = 0.3699
\end{align*}
\]

-------

Iteration 2

\[
\begin{align*}
P(A) &= 0.15 + 0.85 \times \left( \frac{0.3699}{2} \right) = 0.3072 \\
P(B) &= 0.15 + 0.85 \times \left( \frac{0.1149}{3} + \frac{0.1716}{2} \right) = 0.2554 \\
P(C) &= 0.15 + 0.85 \times \left( \frac{0.1149}{3} + \frac{0.3699}{2} \right) = 0.3397 \\
P(D) &= 0.15 + 0.85 \times \left( \frac{0.1149}{3} + \frac{0.1716}{2} \right) = 0.2554 \\
P(E) &= 0.15 + 0.85 \times (0.1716 + 0.1716) = 0.4417 \\
\text{Normalization: } P(A) &= 0.3072 / 1.5994 = 0.1920 \quad P(B) = 0.1596 \quad P(C) = 0.2123 \\
P(D) &= 0.1596 \quad P(E) = 0.2761 \quad \text{Order: E C A B & D}
\end{align*}
\]