School of Computing and Information Systems
The University of Melbourne
COMP90049
Knowledge Technologies (Semester 1, 2019)
Workshop exercises: Week 5

Suppose that we have observed the token `lended`, and we have a dictionary as follows:

```
addendum
blenders
commodity
deaden
end
leader
leant
lent
lemonade
pleading
```

1. Assuming that the "correct" (intended) dictionary entry was `lent`, calculate the precision of the following methods of finding approximate entries from the dictionary.

   (a) Neighbourhood search, with a neighbourhood of 1

   - There weren't any results returned from the dictionary, so precision isn't well-defined ($\frac{0}{0}$)

   (b) Neighbourhood search, with a neighbourhood of 2

   - There was one entry returned from the dictionary (`leader`), but it wasn't `lent`, so the precision is $\frac{0}{1} = 0$.

   (c) Neighbourhood search, with a neighbourhood of 3

   - There were five entries returned from the dictionary, and `lent` was one of them. The precision of this system is the number of correct responses (1) out of the total number of attempted responses (5), $\frac{1}{5} = 20\%$

   (d) Global Edit Distance, with a parameter $[m, i, d, r] = [1, -1, -1, -1]$

   - There were two (tied) results from the dictionary (`blenders` and `leader`), but no `lent`, so the precision is $\frac{0}{2} = 0$

   (e) Local Edit Distance, with a parameter $[m, i, d, r] = [1, -1, -1, -1]$

   - There was just a single result (`blenders`) which wasn't `lent`, so the precision is 0

   (f) N-gram Distance, where n is 2 (without padding with terminals)

   - There was a single result (`end`) which wasn't `lent`, so the precision is $\frac{0}{1}$

   (g) Using the Soundex transformation, and then looking for exact matches

   (h) Using the Soundex transformation, and then permitting a 1-neighbourhood

   - There weren't any exact matches with the Soundex code of `lended`, so precision isn't well defined
   - Allowing approximate matches of the Soundex code meant that there were four results, including `lent`, so the precision is $\frac{1}{4} = 25\%$

&

---

2. What is the difference between "data retrieval" and "information retrieval"? Why is the latter a knowledge task, but the former is not?

TF-IDF

TF - term Frequency (more weight when query terms appear many times)

IDF - inversed document frequency

Less weight when terms appear in many documents

Less weight is given when documents that have many terms.

$$W_{d,t} = \begin{cases} 1 + \log_2 f_{d,t}, & \text{if } f_{d,t} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$W_{q,t} = \begin{cases} \log_2\left(1 + \frac{N}{f_t}\right) & \text{if } f_{q,t} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$W_{d,t}$	Weight of a term t in document d
$f_{d,t}$	Frequency of term t in document d
$W_{q,t}$	weight of a term t in query
N	Number of documents
f_t	Number of documents containing term t
$f_{q,t}$	Frequency of term t in query (0 or 1)

Term Weight

Doc ID	TF → apple	ibm	lemon	sun	as vector
Doc 1	4 → 1+log₂4 = 3	0 ⇒ 0	0 ⇒ 0	1 ⇒ 1	<3, 0, 0, 1>
Doc 2	5 → 3.32	0 ⇒ 0	5 ⇒ 3.32	0 ⇒ 0	<3.32, 0, 3.32, 0>
Doc 3	2 → 2	5 ⇒ 3.32	0 ⇒ 0	0 ⇒ 0	<2, 3.32, 0, 0>
Doc 4	1 ⇒ 1	2 ⇒ 2	1 ⇒ 1	7 ⇒ 3.81	<1, 2, 1, 3.81>
Doc 5	1 ⇒ 1	1 ⇒ 1	3 ⇒ 2.58	0 ⇒ 0	<1, 1, 2.58, 0>

Query: apple, lemon.

query <1, 0, 1.42, 0>

$W_{q,apple} = \log_2\left(1 + \frac{5}{5}\right) = 1$

$W_{q,ibm} = 0$

$W_{q,lemmon} = \log_2\left(1 + \frac{5}{3}\right) = 1.42$

$W_{q,sun} = 0$

Cosine similarity: $\cos(Doc, q) = \frac{Doc \cdot q}{|Doc| \cdot |q|}$

$$\cos(Doc1, q) = \frac{Doc1 \cdot q}{|Doc1| \cdot |q|} = \frac{3\times1 + 0\times0 + 0\times1.42\times1\times0}{\sqrt{3^2+0^2+0^2+1^2} \cdot \sqrt{1^2+0^2+1.42^2+0^2}} \approx 0.\text{从}$$

$\cos(Doc2, q) = 0.99$

$\cos(Doc3, q) = 0.30$

$\cos(Doc4, q) = 0.31$

$\cos(Doc5, q) = 0.91$

$\therefore 2 > 5 > 1 > 4 > 3$

- The main difference here is the existence of people users. Because people are wildly divergent, the notion of a relevant result in information retrieval depends on contextualising the data to the particular user (which may be very difficult, because we have an imperfect model of the user and indeed the user has an imperfect model of their needs!). Whereas with data retrieval, there is a particular unit of data (bitstream) that we need to access in memory or on a hard drive, and there is generally no ambiguity (for different users).

3. (Extension) How many books are there in an average library? How many words are there in an average library? How many documents are there on the World Wide Web? How many words?

   - These aren't straightforward questions to answer. But, to an order of magnitude, a small city library might have about 10K books; a larger one, maybe 30K. I might estimate the word count of a typical book to be about 50K (many are longer; many are shorter; there are varying definitions of "word"), which would situate a library as carrying roughly 1G words. The US Library of Congress catalogues about 2.3M books, so perhaps 100G words.
   - As of 2008, Google claimed to index 1T unique urls (`http://googleblog.blogspot.com.au/2008/07/we-knew-web-was-big.html`); by 2012, this had supposedly risen to 30T (`http://www.google.com/insidesearch/howsearchworks/thestory/`); now, it would hypothetically be much larger. But maybe take all of this with a grain of salt! :-) Estimating the number of words on the Web is even harder Google tells me that the mean document size is about 400KB, but much of that isn't going to be text. I might ballpark about 1000 words (roughly 6KB of the 400KB) per document, which might be upwards of 10000T words! (Or more! But probably less!)

4. Identify some different types of "informational needs."

   - Rehashing the lecture slides:
     - Informational, where we want to know more about a topic, e.g. "global warming"
     - Factoid, where the response is a single piece of unambiguous information, e.g. "melting point of lead"
     - Topic tracking, where the user wants information relevant to today, e.g. "Dutch elections" [as in, the most recent ones]
     - Navigational, where we want to browse to a specific web page, e.g. "University of Melbourne home page"
     - Service or transactional, which can be defined a couple of different ways, but typically querying an underlying database, e.g. "Mac powerbook"
     - Geospatial, where the information refers to a physical location, e.g. "Carlton restaurant"
   - This isn't an exhaustive list. Nor is it non-overlapping: for example, most queries can be construed as being navigational in nature (as the user is likely to click through to a relevant document), and many are informational as well.

Evaluation Metrics

① For Approximate String Search.

(1) one or more probably misspelled tokens of interest
(2) System returns one or more items from dictionary
(3) We examine whether the returned dictionary items are correct.
   ① Accuracy
   ② Precision
   ③ recall.

② For Information Retrieval

(1) One or more queries
(2) System returns many documents from the collection
(3) We examine whether the returned documents are relevant to meet user's information need.
   ① Precision
   ② Recall.

Precision: $\frac{\text{number of returned relevant results}}{\text{number of returned results}}$

Recall: $\frac{\text{number of returned relevant results}}{\text{total number of relevant results}}$ (often useless in IR context)

Precision at K (P@K): $\frac{\text{number of returned relevant results in top K}}{K}$.

Average Precision (AP): $\frac{1}{R}\sum_{i=1}^{d}\left(\frac{r_i}{i}\cdot\sum_{j=1}^{i} r_j\right)$

R: known relevant items.
di: items

$\begin{cases} r_i = 1, & \text{if ith item relevant} \\ r_i = 0, & \text{not relevant} \end{cases}$

Mean Average Precision (MAP).