Lecture 1a: Exam

William Webber (william@williamwebber.com)

COMP90042, 2014, Semester 1, Lecture 21b

Final exam

- 10th June
- Royal Exhibition Building (East)
- Starts 3:15pm

Please verify these details from whatever the official source is!

Reading time

- 15 minutes reading time begins at official exam time
- I.e., 3:15pm to 3:30pm is reading time
- Exam proper runs 3:30pm to 5:30pm

Exam format

- The exam is **closed book**
  - No materials may be referred to during the exam
- The exam consists of 6 questions, each of which has parts to it
- All questions should be attempted
- Answers include:
  - Python code
  - Mathematical formulae
  - Short written answers
- **Advice:** There are lots of short questions; don't get stuck on one!

Examinable material

- Course material from Lecture 1 ("Text, terms, and bags of words") up to and including Lecture 16 ("Advanced topics in text classification") are examinable
  - Probabilistic topic models (pLSI, LDA) are **not** examinable
- There will be questions directly addressing work done in the first project
- Materials in "additional reading" are not examinable, except as covered in the lecture notes
- Novel material covered in the worksheets is not examinable, except as covered in the lecture notes

That is, the lecture notes and the first project should be your main focus for revision

Example questions (1)

Question

An alternative to unit-length normalization is pivoted document-length normalization. What is the motivation for pivoted document-length normalization?

Sample answer

Unit-length normalization may give too much weight to short documents, and too little to long ones. Pivoted document-length normalization decreases the weight of terms in short documents, and increases that in long documents, relative to unit-length normalization.

Example questions (2)

Question

Consider the following Python function skeleton:

```
def inv_idx_eval(inv_idx, qry):
    """
Evaluate a query against an inverted index.

Arguments:
  - inv_idx: a dictionary of { term : dvec }, where each
      dvec is a dictionary of { docid : wt }, giving the
      documents the term appears in and their weight.
  - qrt: a list of query terms

Return value:
  Returns a { doc : sim } dictionary, giving the similarity
  between the query and each document (considering only
  documents that at least one query term appears in.
  """
  pass
```

Implement this function

Example questions (3)

Question

Consider the following probability statement:

$$P(w_1, \dots, w_i, \dots, w_m) = ? \tag{1}$$

Fill in "?" for an n-gram language model (where n could be greater than 1)

Answer

$$\begin{aligned} P(w_1, \dots, w_m) = P(w_1) \times P(w_2 | w_1) \times \dots \\ \times P(w_i | w_{i-1}, \dots, w_{i-(n-1)}) \times \dots \\ \times P(w_m | w_{m-1}, \dots, w_{m-(n-1)}) \end{aligned} \tag{2}$$