School of Computing and Information Systems
The University of Melbourne
COMP90042 WEB SEARCH AND TEXT ANALYSIS (Semester 1, 2017)

Workshop exercises: Week 6

Discussion

1. For the following term co-occurrence matrix (suitably interpreted):

	`cup`	`not(cup)`
`world`	55	225
`not(world)`	315	1405

   (a) Find the Point-wise Mutual Information (PMI) between these two terms in this collection.

   (b) What does the value from (a) tell us about **distributional similarity**?

2. In the `WSTA_N9_distributional_semantics` iPython notebook, a document–term matrix is built for the purposes of IR-style document retrieval.

   (a) What is the Singular Value Decomposition (SVD) method used for here? Why is this helpful?

   (b) What is the significance of the `transform_query()` function?

3. What is a **word embedding** and how does it relate to **distributional similarity**?

   (a) What is the difference between a **skip-gram** model and a **CBOW** model?

   (b) How are the above models trained?

Programming

1. Use the `distributional_semantics` iPython notebook to find some interesting collocations, using PMI.

   (a) Write a wrapper function which finds the 10 collocations with the greatest PMI, amongst all of bi-grams in the collection. (Note that you might want to be careful about your strategy for doing this in a very large collection!)

   (b) NLTK has an in-built method `collocations()` (of a `Text` object) — does it come up with the same collocations as PMI? Why do you think this is the case?

2. Run the `word_vector_learning` iPython notebook.

   (a) Does the model come up with reasonable similarity estimates of the given words? Change the size of the corpus, and re-build the model — how sensitive is it to the size of the corpus?

   (b) What are the most similar words to *woman*? What is the outcome of *king-man+woman*? What do you think is happening in the model?

   (c) Check some of the predictions made for the `question-words` problem. Are there some kinds of problems that this model is better on? Why might this be? (You might try a different corpus for comparison.)

Catch-up

- What is the **cosine similarity** and how is it calculated?
- What is **entropy** and how is it calculated? What is entropy attempting to measure?
- Revise the difference between **prior**, **joint**, and **posterior** probabilities.
- Recall the definition of **independence** in a probabilistic sense. How can it be formally demonstrated?
- What is a **term–document matrix**? How is it different to an **inverted index**?
- What is a TF-IDF model? What are its intuitions and how do they appear in a typical model (formula)?

Get ahead

- In the notebook `WSTA_N9_distributional_semantics`:
  - Try doing the same calculations on the collection without the SVD method. How much time does the truncation step save?
  - Try different values for truncating the decomposition; at what point do the results seem to become noticeably worse?
  - How important is the TF-IDF step? Try the retrieval without it; do the results change? What if you omit it from only the document matrix, or only the query vector?
  - Try to find some queries where the results are different with and without the TF-IDF transformation.