Discussion

Pre-workshop

• What is “POS tagging”? What kinds of information can be used to POS tag a sentence?

• What is an “HMM”? How can we use one to perform POS tagging? What are its limitations?

Workshop

1. Consider the following humourous dialogue from *The Importance of Being Earnest*, by Oscar Wilde:

   Jack: I have lost both my parents.
   Lady Bracknell: To lose one parent, Mr. Worthing, may be regarded as a misfortune; to lose both looks like carelessness.

   (a) What is “polysemy”? Give an example. (Why is the dialogue humourous?)
   (b) What is “word sense disambiguation” (WSD)? What are some properties that one might use to disambiguate word senses?
   (c) What is the significance of the “one sense per collocation” heuristic? What is the significance of the “one sense per discourse” heuristic?

2. What is “shallow parsing” (or “chunking")?

   (a) Chunk the two sentences in the dialogue above.
   (b) How can we use an HMM to perform the task of shallow parsing? Re-chunk the sentences to be a format that might be output by an HMM.
   (c) What factors control the chunk boundaries in a sentence? Are the limitations of an HMM suitable for solving this problem? Do HMM shallow parsers perform well?

Post-workshop

• What is the “most frequent sense” baseline in WSD? Why is it so difficult to beat?

• Identify some cases where the “one sense per...” heuristics fail to hold. Consider how you would leverage the heuristics to help build a WSD system.

• In what way(s) does the Unsupervised method for performing WSD resemble $k$-Nearest Neighbour? In what way(s) does it resemble query expansion?

• Why are we interested in shallow parsing? Is shallow parsing useful?
Programming

Pre-workshop

- Take a look at Trevor’s python implementation of the main HMM problems.

Workshop

1. Build a simple shallow parser using a RegexpParser.
2. Evaluate your shallow parser on the CONLL 2000 test data, included in NLTK as conll2000.chunked_sents('test_sents').

Post-workshop

- Alter your HMM from last week, or Trevor’s HMM to perform chunk tagging. You might need to start with some pre-chunked sentences: in NLTK, you can use conll2000.chunked_sents('train.txt').
- Evaluate this shallow parser on the CONLL 2000 test data. Do you think this model could make a competitive shallow parser? What could you do to improve it?