Discussion

Pre-workshop

- What is “entropy”? What it is attempting to measure? How is it calculated?
- What is difference between “classification” and “regression”, in the Machine Learning domain?
- Revisit the “Naive Bayes” model for Machine Learning. What assumptions does it make and how does this affect the model in practice?
- What are some common evaluation metrics in the Machine Learning domain?

Workshop

1. What are the benefits and downsides of $k$-Nearest Neighbour, relative to other common classification regimes?
2. What is a “logistic regression”?
   (a) What assumptions does it make about the data? How does this compare to Naive Bayes?
   (b) In Natural Language Processing, logistic regression models are very popular, and are called “maxent models”. What properties of natural language data make these kinds of models particularly suitable?
3. What is a “POS tag”?
   (a) What models of natural language is “POS tagging” trying to encode? (Mostly out-of-scope for this subject!)
   (b) What are some common approaches to POS tagging? What aspects of the data might allow us to predict POS tags systematically? What difficulties might we have?
   (c) POS tag (by hand) the following sentence: Pierre Vinken, 61 years old, will join the board as a nonexecutive director Nov. 29.

Post-workshop

- Compare logistic regression models with some other machine learning models, e.g. $k$-Nearest Neighbour, Decision Trees, SVMs, etc.
- What are some NLP tasks where POS tagging might be useful? How might this be related to “language models”?
- How might you have made use of POS tag information in Project 1?
Programming

Pre-workshop

• There are numerous pre-implemented logistic regression models, for example the OpenNLP version at http://maxent.sourceforge.net — download a package and use it to classify some data, for example, from the UCI Machine Learning Repository at http://archive.ics.uci.edu/ml/.

• NLTK has extensive support for POS tagging. Have a read through Chapter 5 of the NLTK book (http://www.nltk.org/book/ch05.html).

Workshop

1. Use some different NLTK in-built mechanisms to POS tag the phrases time flies like an arrow and fruit flies like a banana. For example, nltk.pos_tag(text) and nltk.RegexpTagger(patterns).

2. POS tag the above two sentences based on some pre-tagged data: try the Brown corpus (nltk.corpus.brown.tagged_words or nltk.corpus.nps_chat.tagged_words). Build a “Unigram tagger” and a “Bigram Tagger”. Does it correctly tag the sentences above? What about a “Bigram Tagger”?

Post-workshop

• Think conceptually about how you might have included POS tag information in your Project 1 system. With what kinds of queries might it have helped?

• Some types of words are notorious for not “obeying the rules” with respect to POS tagging. In English, the most striking example of this is colours (e.g. black, green, etc.). Consider the following code snippet:

```python
import nltk
tws = nltk.corpus.brown.tagged_words()
for i in range(len(tws)):
    if tws[i][0] == black:
        context = tws[i-3:i+3]
        print ' '.join(map(lambda (w,t): w+'/'+t, context))
```

Observe some instances of when black is an adjective (JJ) and when it is a noun (NN) — what are some regularities of these sentences which would allow your system to recognise these differences?