Natural Language Systems Coursework 2: named-entity recognition and sentiment analysis of movie reviews

1. Named-entity recognition:

a) Run and compare the outputs of two NER methods - NLTK named entity classifier and Stanford named entity recognizer - for the PERSON and ORGANIZATION classes. Explore the differences between the two approaches – which of the tools seems better in getting the bounders of named entities right? Provide some examples. Specifically, report in how many cases the tools fully agree on the mentions of named entities (exact match), and in how many cases they have a partial overlap.

2. Sentiment analysis of movie reviews:

a) With the popularity of social media, building and maintaining a sentiment/polarity lexicon is a huge challenge. In the first part of this task, you will build a sentiment lexicon using a semi-supervised approach by bootstrapping the process, starting from a small lexicon of adjectives (see at the end of the document) and corpus 2 (see Data below). Write a program that will collect more adjectives to populate the lexicon and assign them with the likely polarity. For example, adjectives conjoined by "and" are likely to be of the same polarity (e.g. corrupt and brutal), while adjectives conjoined by "but" are not (fair but brutal). Consider (and implement) other possible patterns; consider how you could assign a polarity if an adjective appears several times in the corpus (and, for example, you have conflicting polarity signals). Evaluate the outcome – how many of the proposed adjectives have been properly classified (according to your judgement of their typical polarity)? Provide explanations for any errors in the report.

b) The two files in corpus 2 represent positive and negative examples of movie reviews. Build a sentiment classifier and evaluate it using k-fold cross validation using this corpus. Instead of a simple bag-of-words approach, you should consider a richer feature set – for example, whether any of the words in the review come from a polarity lexicon (e.g. the MPQA Subjectivity Cues Lexicon), whether they are negated, etc. For training your classifier, use any available machine-learning framework (e.g. Weka (http://www.cs.waikato.ac.nz/ml/weka/) or scikit-learn (http://scikit-learn.org/)). Compare the results of your classifier to a base-line classifier that counts whether there are more positive or negative words in a comment (use the MPQA lexicon, see below). Explain briefly what you have done and discuss the results in the report.