Phrase Structure Analysis of NPs

Introduction to NLP, ACS 2012, Assignment 2
Lecturer: Ann Copestake

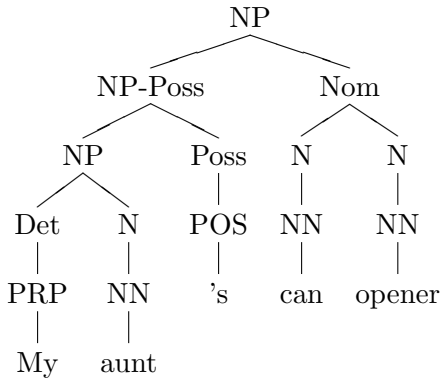
1 Task

Choose 2 sentences from each of the 4 sets below (8 total) and bracket all the noun phrases (NPs) in each sentence. Then for each NP found, draw a phrase-structure tree (PST) using non-terminal labels (NP, AP etc). Base your non-terminal labels on those used in the handout (see also Jurafsky and Martin ch12, p422f, p428f and other references on handout. You may also find it useful to look at Section 1 of the 'Theories of Syntax, Semantics and Discourse Interpretation' handout.) You can invent your own labels for constituents motivated by distributional analysis as necessary, and base your PST on the tokenization and PoS tags assigned in the first handout (if you decide to assume a different PoS tag, mention this in the notes and give reasons). (Marked copies of Assignment 1 should be available for collection from graduate student admin on Wednesday. I will send email when they are ready.)

For instance the PST analysis of the first two NPs in:

My aunt's can opener can open a drum

should look something like this:



Write up / draw your answers and hand them in to graduate student administration by the deadline given on the assignment section of the module webpage. Include BRIEF notes on any difficulties or issues you had with specific cases. It is more important to understand and be able to explain your reasoning than to get every constituent right. Be prepared to discuss the difficult cases during the session. Please feel free to work on the task in groups, but the final selection of sentences and their analyses should be your own. I'd recommend trying some new sentences in this assignment, especially if you are finding the analyses easy.

Important: Earlier hand-in would be appreciated, especially if you have difficulties with the assignment. Keep a copy of your completed assignment and bring it to the lectures.

2 Sentences

(1) a The old car broke down in the car park
   b At least two men broke in and stole my TV
   c The horses were broken in and ridden in two weeks
   d Kim and Sandy both broke up with their partners

(2) a The horse which Kim sometimes rides is more bad tempered than mine
   b The horse as well as the rabbits which we wanted to eat have escaped
   c It was my aunt's car which we sold at auction last year in February
   d The only rabbit that I ever liked was eaten by my parents one summer
   e The veterans who I thought that we would meet at the reunion were dead

(3) a Natural disasters – storms, flooding, hurricanes – occur infrequently but cause devastation that strains resources to breaking point
   b Letters delivered on time by old-fashioned means are increasingly rare, so it is as well that that is not the only option available
   c It won't rain but there might be snow on high ground if the temperature stays about the same over the next 24 hours
   d The long and lonely road to redemption begins with self-reflection: the need to delve inwards to deconstruct layers of psychological obfuscation
   e My wildest dream is to build a POS tagger which processes 10K words per second and uses only 1MB of RAM, but it may prove too hard

(4) a English also has many words of more or less unique function, including interjections (oh, ah), negatives (no, not), politeness markers (please, thank you), and the existential 'there' (there are horses but not unicorns) among others.

b Making these decisions requires sophisticated knowledge of syntax; tagging manuals (Santorini, 1990) give various heuristics that can help human coders make these decisions and that can also provide useful features for automatic taggers.

c The Penn Treebank tagset was culled from the original 87-tag tagset for the Brown Corpus. For example the original Brown and C5 tagsets include a separate tag for each of the different forms of the verbs *do* (e.g. C5 tag VDD for *did* and VDG tag for *doing*), *be* and *have*.

d The slightly simplified version of the Viterbi algorithm that we present takes as input a single HMM and a sequence of observed words $O = (o_1, o_2, ...o_T)$ and returns the most probable state/tag sequence $Q = (q_1, q_2, q_T)$ together with its probability.

e Thus the EM-trained "pure HMM" tagger is probably best suited to cases where no training data is available, for example, when tagging languages for which no data was previously hand-tagged.