CS 421 – Natural Language Processing – Spring 2015
Homework 2 - Part A
Part of speech tagging

General Information

Deadline (Part A) : Tue March 3, 11:59pm
Worth: 50 points (out of 1000 total for the class) (Part B will be worth around 20-30 points)
What and how to hand it in: You will upload all your answers on blackboard.

1 Automatic POS tagging (20 points)

Tag the following three sentences with the two POS taggers listed below, they both use the Penn Treebank inventory of tags (Fig. 5.6 p. 131 in the book).

1. sentence 5 from exercise 5.2 from the book (p. 171) ;
2. the following tweet (taken from a corpus of tweets we assembled in the NLP lab); don't correct any spelling mistakes or abbreviations: *Wearn a suit tomorrow for da interview ;+) dats how serious da money is*
3. a sentence at least 15 word long that you can take from any written document you choose, such as a newspaper article, a book, etc. It can be the sentence you used for the translation exercise in Homework 1, but not one of the 15 sentences from part 2 of this homework.

The two taggers to use are:

1. The SNOW POS tagger, whose demo is available on line at
   http://cogcomp.cs.illinois.edu/demo/pos/index.php
2. the POS tagger that comes with the Stanford parser, available at
   http://nlp.stanford.edu:8080/parser/index.jsp
   For the Stanford parser, disregard the parse trees and just look at the POS tagged sentence.

You will have three pairs of results. Carefully examine each pair. Answer the following questions:

1. Do the two taggers agree on every single tag? If not, where do they disagree?

2. Do they make mistakes in your opinion? **To answer this question, you need to CAREFULLY LOOK AT EACH SINGLE TAG, and FIGURE 5.6, to recognize the mistakes they make**. Speculate on why the POS tagger(s) make the mistakes you notice.

   NOTE. In case you find ony n mistakes, where $n < 3$, speculate about **how you would disambiguate** $3 - n$ ambiguous words in one of the three sentences – meaning, which specific knowledge you as a human would use to disambiguate.

Your answer will include: all SNOW analyses, all STANFORD analyses, and answers to all questions above.

2 Train your own POS tagger (30 points)

Now take a newspaper article, or a novel. Feed its first 15 sentences (but not the title) to SNOW, or to the STANFORD parser (only one of the two). Save the results. This will be your training corpus, 15 sentences that are tagged with POS tags (if you notice mistakes made by the tagger, don't correct them). Your test set will consists of a new sentence of at least 8 words, that you will make up using **only** the words contained in those 15 sentences.

1. Write a simple program in a language of your choosing to compute **all** the lexical likelihood probabilities and **all** the tag bigram probabilities, as if we were training an HMM from this toy corpus.

2. Use your language model (i.e., all the probabilities you computed) to compute the most likely sequence of tags for the new sentence you built (use bigrams of tags). You can do this part manually or automatically, as you prefer. You are not asked to implement the Viterbi algorithm, but to deal with the new sentence as we did in the examples in class: i.e., given your corpus, which words are ambiguous? which probabilities differ when we instantiate the formula

$$argmax_{T \in \tau} \Pi_{i=1}^{n} P(w_i|t_i) \, \Pi_{i=1}^{n} P(t_i|t_{i-1})$$

Upload your training corpus (i.e. the 15 sentences with their tags); your code and the results, i.e., all the probabilities your program has computed; the new sentence you built, its most likely sequence of tags and its probability. Each probability must have an indication of which probability it is, e.g. `P(take|VB) = 0.0034`, and `P(VB|NN) = 0.05`.

3 Uploading files

Please follow the link under Assignments / Homework 2A to upload your file.
Important notes.

- Package all your files in a single .zip or .tar file, and upload that, not individual files.
- Upload only when you are really ready to do so. Whereas you are allowed two uploading attempts, better to upload only one.