Fall Semester 2016
Iowa State University

ENGL 520 - Computational Analysis of English

Assignment 5
Submission Deadline: 15 NOV 2016, end of the day

Instructions: This assignment consists of three questions. Each question carries 5 marks. First question does not involve programming. Second and third question involve programming and you can use any programming language you want. Create a zip file with three components (a pdf file for first question, 2 perl or python files). If any of the programs does not run and throws errors, you cannot get a credit for that unless you defend it during the office hours. Some times, a programming question may have different instructions for perl and python programmers.

Question 1

Here is the link to Stanford Parser's online demo. `http://nlp.stanford.edu:8080/parser/`. Now, do the following:

1. Create a list of 10 sentences written by English learners at an Elementary proficiency (You can use any sentences from corpus resources you already have. Or pick something from `http://cblle.tufs.ac.jp/llc/icci/search.php?menulang=en`. Or contact me for any other learner corpus resources). You can use the same 10 sentences from previous assignment.

2. Run the parser on these 10 sentences, and study the phrase structure tree and Universal dependencies representation there (You should also read a little bit on what those tags mean). Now, write what you think about the robustness of both the representations to such ungrammatical input (and misspelt words). You should write about about both phrase structure and dependency representations , and compare their performance. Make sure you have two types of sentences in your chosen list: a) sentences that are incorrect (mis-spelt or ungrammatical), yet, do not affect POS tagging. b) sentences that are incorrect and affect POS tagging.

Note: Your commentary can be up to 2 pages long. List the sentences you tried with at the end of this document.

Question 2

Write a set of pattern matching rules, to identify noun phrases and preposition phrases from the output of a POS tagger (you can use the one from Assignment 3). Incorporate these rules into a program which takes a sentence as input, performs POS tagging, and returns the Noun phrases and Preposition phrases in the sentence, one per line. Again, the expectation is that you will think and try to solve the problem. I don't expect anyone to come up with something that works with 100% precision.

Question 3

Use any off the self English parser implementation in Perl or Python, and write a program that takes a sentence of text as input and outputs the following:

- Syntactic parse tree of the sentence
- Number of NPs in the sentence
- Average length of the NP
- Number of Verbs in the sentence
- Number of Preposition Phrases in the sentence.

Python users: NLTK comes with several parser implementations. Go through the examples in Chapter 7 and Chapter 8 and pick any one of them. Else, try to figure out how to make Stanford parser work in Python and use that. Perl users: Perl also has several parser implementations. Search on CPAN for text parsers in LINGUA or NLP libraries. My recommendation is Stanford parser for you too. Figure out how to make this work.