Natural Language Process: COMS4705

Kathy McKeown, Fall 2009
Due: Oct. 21st, 2009, midnight

Homework 2: Parsing (100 points)

Please post any questions concerning this assignment to the Courseworks (*http://courseworks.columbia.edu*) discussion board, under the Homework 2 topic.

0 General Instructions

There are three parts to this homework. In the first part, you are to write a Context Free Grammar (CFG) using the NLTK toolkit. A readme for downloading and using the toolkit can be found at:
nltk_readme_KNL.pdf

The second part of this homework involves downloading and using a robust statistical parser, developed at Stanford and aptly called, The Stanford Parser. You can find information for downloading and using the Stanford parser (download the PCFG version, not the dependency parser) at:
http://nlp.**stanford**.edu/software/lex-**parser**.shtml

The third part of the homework involves answering written questions about statistical and lexical dependency parsing.

You must write the grammar yourself. Don't use publicly available grammars (please refer to the Academic Integrity policy if you have questions in this regard). We will also check that you didn't build the grammar using reverse engineering from the Stanford parser. We expect your grammar to be different.

Your submission must include a README file for Part I as specified in Section 2.2 below. The requirements for what to submit are specified in each part.

1 Context Free Grammar (60 points)

Write a context free grammar in NLTK to handle the (slightly modified) story of Where the Wild Things Are, a story by Maurice Sendak which is about to be released as a major motion picture. You will find the story in
WildthingswithcorrectedPOS.txt

This file contains the story tagged with PennTreebank POS tags. Note that one change you may want to make is to convert different forms of verb (e.g., VBD, VBZ) into just VB and different forms of common nouns (e.g., NNS, NN) into just NN, which would simplify the creation of CFG rules. Whether you do that and exactly how is up to you. You will be creating grammar rules that will allow the system to parse relative clauses (RC), prepositional phrases (PP), Verb phrases (VP) (transitive, intransitive, bitransitive, with an embedded sentence as object), embedded sentences (call them S_Bar), time modifiers (TIME), and you should allow some constructions (NP, VP, S_BAR) to contain conjunction, and sentences to include subordinate conjunction (i.e., two embedded sentences are joined by a subordinate conjunction).

You may find it helpful to consult a grammar of English during this process. C. 12 of our textbook is the best resource. The book, *A Comprehensive Grammar of English*, by Quirk and Greenbaum, is also very helpful and a copy will be placed on reserve in the Engineering Library. Finally, you may also find it useful to look at the annotation guidelines given for the Penn TreeBank POS tags. These guidelines give examples for different parts of speech and reasons why a word in a particular usage falls into that POS. You will find them here: tagguide.pdf

You will be graded in part on whether your rules are syntactically justifiable. You should attempt to make your rules general where possible (i.e., don't make new rules for each and every new string of words you see; a rule should ideally cover several phrases that you see in the input).

You should hand in a file containing your grammar, documented so that it describes why you selected the grammar rules that you did. For example, you may justify using a particular set of rules because they handle multiple constructions or because they follow a rule that you found in C. 12.

We will run your grammar in the NLTK parser environment using the chart parser.

You will be graded on the following elements:

1. A parse tree is produced for each sentence. **(15 points)**
2. Each of the constructions listed above is represented in the grammar **(20 points)**
3. Choices about the particular rules used and the resulting parse are adequately justified. Points will be deducted for parse trees that do not capture a good structure of the language according to your justification **(25 points)**

2 Stanford Parser (20 points)

Download and run the Stanford Parser using the PCFG (not dependency parser) on the training files for HW1. We will provide several files from the training data that you should use for this part of the assignment.

A. Select two sentences where the parse returned is incorrect. One of your choices can be when the parser entirely fails (produces no output), but one should be where a parse is produced, but you don't think it is a good one. In each case explain why the parser failed and what should have been produced. Note: as a comparison point, you might try your own grammar on the same files, though no need to submit this. **(10 points)**

B. Show how you could modify your HW1 QA system to use the results of the Stanford parser. Do this by selecting two patterns that you created for HW1 and show how you would modify them to use features of the parse. Explain why the new patterns would result in a better QA system. **(10 points)**

3 Statistical and Lexical Dependency Grammar (20 points)

A. [10 points] Suppose you use the output of your parser on *Where the Wild Things Are* as a Treebank. Show how you would compute the probabilities, and what they are, for the rules for VP and the rules for NP.

B. [10 points] You will find that your CFG often generates multiple parses for a sentence. This can happen when it's ambiguous for attachment of the PP or when the scope of conjunction is ambiguous (and for other constructions too). Consider the following cases: sentence 18 where "of all wild things" could modify "king" or "made"; sentence 6 where "to bed" could be an argument of the verb or it could be a modifier of the verb; sentence 10 where "for Max" could modify either "boat" or "tumbled by". Sentence 12 where "to the land" could modify "day" or "sailed off" and "of the wild things" could modify "land," "day," or "sailed off". You cannot use your parser output in this case to compute probabilities to do disambiguation. Why not? You will be given access to the Penn Treebank in our class account. Describe what you would need to count, how your rules would be formulated and how you would use them in a probabilistic lexicalized framework in order to do disambiguation in these cases (note: a rough description will do. You do not need to describe a probabilistic version of CKY). Show the counts for disambiguating whether "to a land" attaches to "sailed off" or "day".

UPDATE: The PennTreebank can be found in /home/cs4705/corpora/penntreebank/parsed/wsj/ You will see multiple sections (00,01,02,03,04,05). In each section there are individual reports that have been parsed by hand. These are what you would use to compute counts.

4 Academic Integrity

Copying or paraphrasing someone's work (code included), or permitting your own work to be copied or paraphrased, even if only in part, is not allowed, and will result in an automatic grade of 0 for the entire assignment or exam in which the copying or paraphrasing was done. Your grade should reflect your own work. If you believe you are going to have trouble completing an assignment, please talk to the instructor or TA in advance of the due date.