Joseph Mercedes

Natural Language Processing Final Project

CKY Parser

The entire project was written in python and it contains two main scripts, myCNF.py and parseCKY.py. The first script, myCNF.py, transforms a given context free grammar into chomsky normal form. The second script, parseCKY.py, parses a file filled with sentences according to a chomsky normalized grammar and returns a list of parse trees associated with each of the sentences. The project contains two sets of CFGs. The first one, cfgText.txt, contains a modified grammar from the textbook illustrating some of their sentences in order to illustrate that the system works. The second, mycfg.txt, contains a more complex grammar from various online sources illustrating some different variations of sentences.

In order to run and test the project in Cygwin, you must first execute myCNF.py with two input arguments, a text file containing a context free grammar and a text file that will contain the chomsky normalized form of the grammar. See the following commands to run with the accompanying files:

python myCNF.py cfgText.txt cnfText.txt

python myCNF.py mycfg.txt mycnf.txt

Then, you must execute parseCKY.py with two input arguments, a text file containing a list of sentences and a text file containing an associated chomsky normalized grammar. See the following commands to run with the accompanying files:

python parseCKY.py sentencesText.txt cnfText.txt

python parseCKY.py mySentences.txt mycnf.txt

You will see an output of all the trees for each sentences with the associated grammar.

Comparing the parse trees related to the grammar of the textbook with the parse trees generated by this system, you can see that they line up pretty well. Also, when viewing the parse trees associated with the more complex grammar, it can be seen that the parses are generally comprehensive according to the grammar.