Task #3 part 1
Producing a Context-Free Grammar for NLTK

Thuong-Hai Pham

March 9, 2017

For **task 1**, we parsed the sentences with Stanford parser [1] [1], then converted these parsed structures to Context-free grammar (CFG) rules in **grammar.py** using **nltk.tree** and **nltk.grammar.CFG**. The output file is **grammar.cfg**.

The Chomsky normal form (CNF) was also achieved in grammar.py paralleling with non-CNF CFG. The function **t.chomsky_normal_form()** allows us to convert non-CNF tree to CNF tree. This can be double checked with **is_chomsky_normal_form()** function of the CFG object. Our CNF CFG is stored in **grammar_cnf.cfg**

Parsing raw sentences (**task 2**) with grammar.cfg was implemented in **parse.py**, in which we initiated **nltk.parse.EarleyChartParser** with loaded grammar and parsed the sentences after being tokenised by **nltk.tokenize.WordPunctTokenizer**. It is important to note that we have to specify the grammar start non-terminal

```
grammar._start = Nonterminal('ROOT')
```

or else the grammar object will be loaded with starting non-terminal assigned by the first rule.

References

[1] D. Klein and C. D. Manning, "Accurate unlexicalized parsing," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*, pp. 423–430, Association for Computational Linguistics, 2003.

[1] http://nlp.stanford.edu:8080/parser/index.jsp