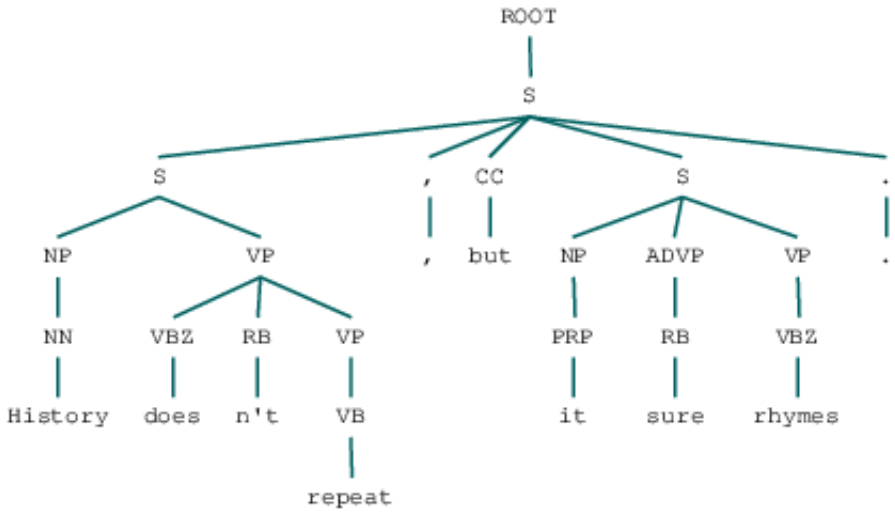
Wk3_SLP1_Parsing with a Stanford Parser in Python

March 10, 2021

[ ]:
```
student_name = "Nikki Fitzherbert"
student_id = "13848336"
```

[1]:
```
from nltk.parse import CoreNLPParser
from IPython.display import display
def my_parse_tree(sentence):
    parser = CoreNLPParser(url='http://localhost:9000')
    parsed_tree = next(parser.raw_parse(sentence))
    display(parsed_tree) #brew install ghostscript
    parsed_list = list(parser.raw_parse(sentence))
    print(parsed_list)
```

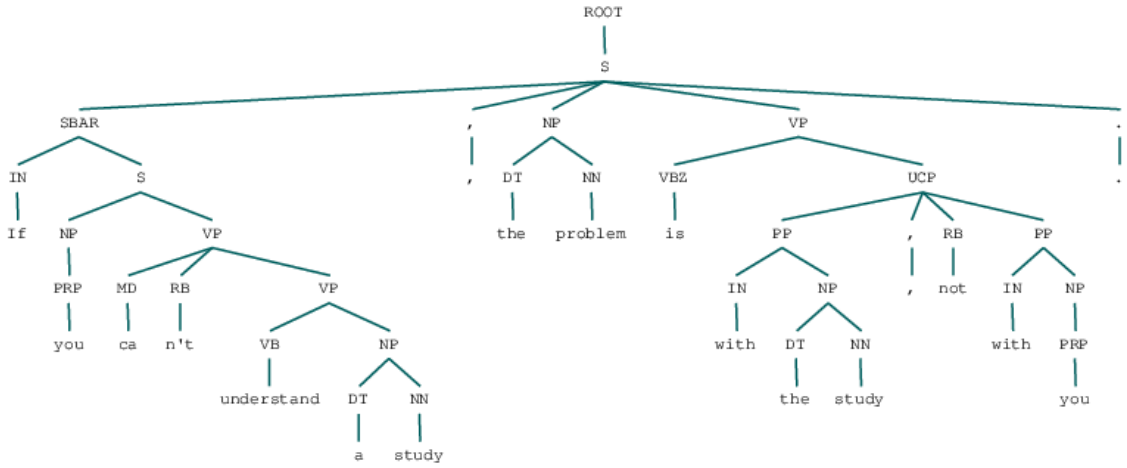
[2]:
```
my_parse_tree("History doesn't repeat, but it sure rhymes.")
```




```
[Tree('ROOT', [Tree('S', [Tree('S', [Tree('NP', [Tree('NN', ['History'])]),
Tree('VP', [Tree('VBZ', ['does']), Tree('RB', ["n't"]), Tree('VP', [Tree('VB',
['repeat'])])])]), Tree(',', [',']), Tree('CC', ['but']), Tree('S', [Tree('NP',
[Tree('PRP', ['it'])]), Tree('ADVP', [Tree('RB', ['sure'])]), Tree('VP',
[Tree('VBZ', ['rhymes'])])]), Tree('.', ['.'])])])]
```

This is the sentence I was using in a previous activity. I found it interesting that the parser was splitting the sentence into two sub-clauses connected by the coordinating conjunction 'but' and the second noun-phrase only included the personal pronoun 'it' and not the adverb and verb until I did some more research into the types of phrase structures possible in the Penn Treebank.

[4]:
```
my_parse_tree("If you can't understand a study, the problem is with the study,␣
↪not with you.")
```




```
[Tree('ROOT', [Tree('S', [Tree('SBAR', [Tree('IN', ['If']), Tree('S',
[Tree('NP', [Tree('PRP', ['you'])]), Tree('VP', [Tree('MD', ['ca']), Tree('RB',
["n't"]), Tree('VP', [Tree('VB', ['understand']), Tree('NP', [Tree('DT', ['a']),
Tree('NN', ['study'])])])])])]), Tree(',', [',']), Tree('NP', [Tree('DT',
['the']), Tree('NN', ['problem'])]), Tree('VP', [Tree('VBZ', ['is']),
Tree('UCP', [Tree('PP', [Tree('IN', ['with']), Tree('NP', [Tree('DT', ['the']),
Tree('NN', ['study'])])]), Tree(',', [',']), Tree('RB', ['not']), Tree('PP',
[Tree('IN', ['with']), Tree('NP', [Tree('PRP', ['you'])])])])]), Tree('.',
['.'])])])]
```

This is a more complicated sentence that I found interesting. The subject was data science, but it could easily be applied to any field such as economics or psychology. Again, I had to do some additional research to work out what some of the unfamilar tags like 'SBAR', 'MD', and 'UCP' were.