Nooreldean Koteb

Professor Anastasopoulos

April 2, 2021

Part 1

a.

a. Unique Rules: **752**

b. Top 5 most frequent Rules:

```
Top 5 Frequent Rules:
    PUNC -> . # 346
    TO -> to # 241
    PP -> IN NP_NNP # 239
    IN -> from # 218
    PP -> IN NP # 197
```

b. Top 5 Highest Probabability Rules:

```
Top 5 Probability Rules:
    PRT_RP -> <unk> #  1.0
    WRB -> How #  1.0
    NP_CD -> one #  1.0
    VP_VBN -> served #  1.0
    TO -> to #  1.0
```

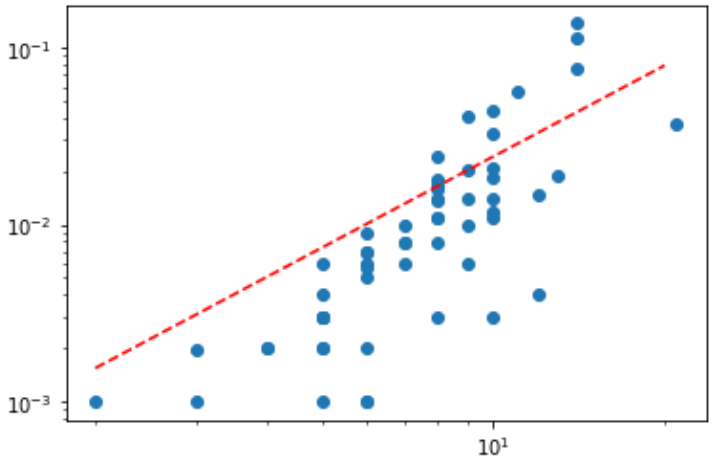
Part 2

a. Done

b.

```
First 5 Parsed lines of dev.strings:

(TOP (S (NP (DT The) (NN flight)) (VP (MD should) (VP (VB be) (NP (NP* (CD eleven) (RB a.m)) (NN tomorrow))))) (PUNC .)) PROB:-17.745297989080456

 PROB:0.0

(TOP (SBARQ (WHNP (WHNP_WDT Which) (PP (IN of) (NP_DT these))) (SQ_VP (VBP serve) (NP_NN dinner))) (PUNC ?)) PROB:-8.775120525449587

(TOP (SBARQ (WHNP (WDT Which) (NNS ones)) (SQ_VP (VBP stop) (PP (IN in) (NP_NNP Nashville)))) (PUNC ?)) PROB:-10.223445425164012

 PROB:0.0
```

c. The curve_fit function returned ~1.7 for k and 0.000562 for c. This is the image in loglog scale. K is about half of 3. I think this might be because the sentences are too short.



d.

```
../data/dev.parses.post 275 brackets
../data/dev.trees       474 brackets
matching        264 brackets
precision       0.96
recall  0.5569620253164557
F1      0.7049399198931909
```

e.

```
First 3 Parsed lines of test.strings:

(TOP (S (NP (DT The) (NN flight)) (VP (MD should) (VP (VP* (VBP arrive) (PP (IN at) (NP (CD eleven) (RB a.m))))
(NP_NN tomorrow)))) (PUNC .)) PROB:-18.975458724111768

 PROB:0.0

(TOP (S_VP (VP* (VB Show) (NP_PRP me)) (NP (NP* (NP (DT the) (NNS flights)) (PP (IN from) (NP_NNP Newark))) (PP
(TO to) (NP (NNP Los) (NNP Angeles))))) (PUNC .)) PROB:-11.998376330814533
```

Sources

I used this file for reference on the data structure and method of tracing the CKY table backwards: https://github.com/xianc/Weighted-CYK-Probabilistic-Context-Free-Grammar/blob/master/pcfg.py