COMS 4705 NLP
Sec 01

HW 2

Name:	Yu Wang
Uni:	yw3025
Date:	Mar. 26, 2018

Upload folder format

--root

-- cfg.counts // temporary counts file, fine to delete

-- cfg.rare.counts // temporary counts file, fine to delete

-- count_cfg_freq.py // file to count frequencies

-- parser.py // the main entrance of the program

-- parse_dev.dat //dev file

-- parse_dev.key // dev key file

-- parse_train_vert.dat // q6 dev file

-- parse_train.dat // train file

-- parse_train.RARE.dat

-- q4.py // code logic for q4

-- q5_prediction_file

-- q5_eval.txt

-- q5.py //code logic for q5 and q6

-- parse_train_vert.RARE.dat

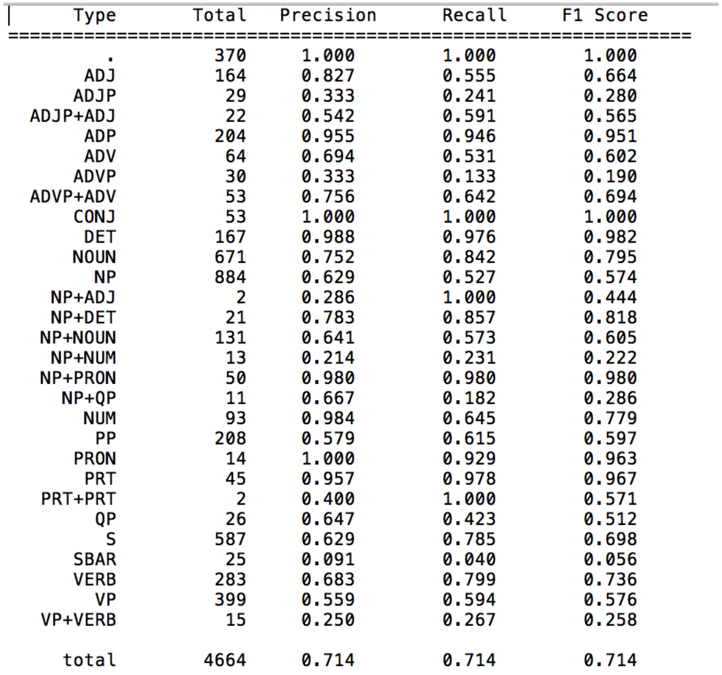
-- q6_prediction_file

-- q6_eval.txt

-- Readme.pdf

Normally, q5 takes ~40 seconds and q6 takes ~70 seconds, and q4 is much faster

Question 5:



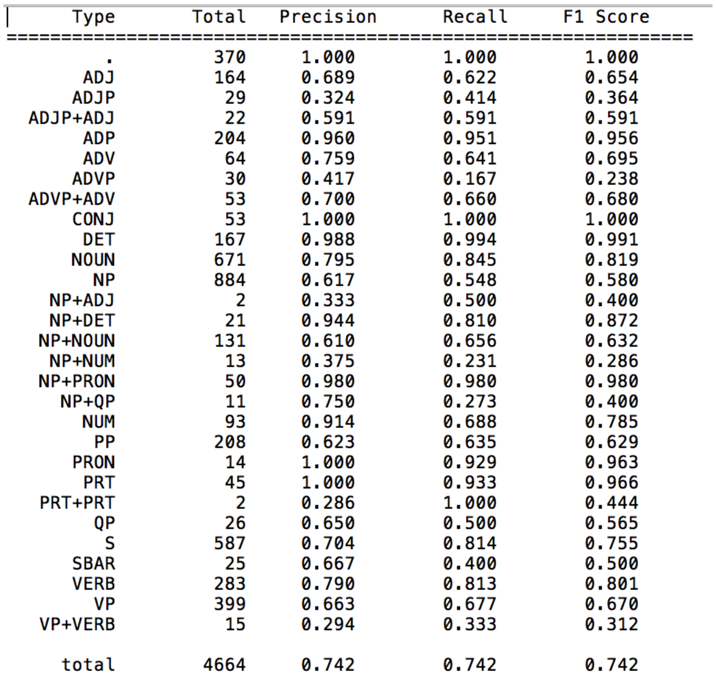
```
|       Type        Total   Precision       Recall      F1 Score
================================================================
           .          370     1.000         1.000         1.000
         ADJ          164     0.827         0.555         0.664
        ADJP           29     0.333         0.241         0.280
    ADJP+ADJ           22     0.542         0.591         0.565
         ADP          204     0.955         0.946         0.951
         ADV           64     0.694         0.531         0.602
        ADVP           30     0.333         0.133         0.190
    ADVP+ADV           53     0.756         0.642         0.694
        CONJ           53     1.000         1.000         1.000
         DET          167     0.988         0.976         0.982
        NOUN          671     0.752         0.842         0.795
          NP          884     0.629         0.527         0.574
      NP+ADJ            2     0.286         1.000         0.444
      NP+DET           21     0.783         0.857         0.818
     NP+NOUN          131     0.641         0.573         0.605
      NP+NUM           13     0.214         0.231         0.222
     NP+PRON           50     0.980         0.980         0.980
       NP+QP           11     0.667         0.182         0.286
         NUM           93     0.984         0.645         0.779
          PP          208     0.579         0.615         0.597
        PRON           14     1.000         0.929         0.963
         PRT           45     0.957         0.978         0.967
     PRT+PRT            2     0.400         1.000         0.571
          QP           26     0.647         0.423         0.512
           S          587     0.629         0.785         0.698
        SBAR           25     0.091         0.040         0.056
        VERB          283     0.683         0.799         0.736
          VP          399     0.559         0.594         0.576
     VP+VERB           15     0.250         0.267         0.258

       total         4664     0.714         0.714         0.714
```

We can notice that

1) The result of `.`, `ADP`, `DET`, `PRON` and `PRT` is fairly good since there is no much ambiguity in such words.
2) The result of `SBAR`, `ADVP`, `ADJP` is relatively bad since it's because such words may have some dependency on words long before or after them (e.g., SBAR involves long distance dependencies).
3) Rare words (count < 5) don't perform much well, but not that bad, so sparsity lead to some problem but rare counts may help to mitigate this.

Question 6:



```
|      Type        Total   Precision       Recall      F1 Score
=================================================================
         .          370     1.000         1.000         1.000
       ADJ          164     0.689         0.622         0.654
      ADJP           29     0.324         0.414         0.364
  ADJP+ADJ           22     0.591         0.591         0.591
       ADP          204     0.960         0.951         0.956
       ADV           64     0.759         0.641         0.695
      ADVP           30     0.417         0.167         0.238
  ADVP+ADV           53     0.700         0.660         0.680
      CONJ           53     1.000         1.000         1.000
       DET          167     0.988         0.994         0.991
      NOUN          671     0.795         0.845         0.819
        NP          884     0.617         0.548         0.580
    NP+ADJ            2     0.333         0.500         0.400
    NP+DET           21     0.944         0.810         0.872
   NP+NOUN          131     0.610         0.656         0.632
    NP+NUM           13     0.375         0.231         0.286
   NP+PRON           50     0.980         0.980         0.980
     NP+QP           11     0.750         0.273         0.400
       NUM           93     0.914         0.688         0.785
        PP          208     0.623         0.635         0.629
      PRON           14     1.000         0.929         0.963
       PRT           45     1.000         0.933         0.966
   PRT+PRT            2     0.286         1.000         0.444
        QP           26     0.650         0.500         0.565
         S          587     0.704         0.814         0.755
      SBAR           25     0.667         0.400         0.500
      VERB          283     0.790         0.813         0.801
        VP          399     0.663         0.677         0.670
   VP+VERB           15     0.294         0.333         0.312

     total         4664     0.742         0.742         0.742
```

We can notice that

1) There is around ~0.03 increase with regard to q5, vertical markovization works in this problem.
2) The type `SBAR`(and `ADVP`, `ADJP`) works fairly bad in q5 improves a lot here, mainly because we incorporate more contextual information and dependency by vertical markovization (use syntactic information from its parent node) instead of the strong independency assumption.
3) The accuracy of type ` ADVP+ADV', 'NP+ADJ` decreases. As mentioned in the HW2 instruction, the vertical markovization greatly increases the number of rules N in the grammar potentially causing data sparsity issues when estimating the probability model q, and some of such non-terminal have little occurrences in the test corpus, which makes their accuracy decreases.