Code

Inside each subfolder of corresponding problem, there are train.py, test.py, and file(s) stored the model.

- a single command line to train the algorithm and save the resulting model
  python3 train .py
- a single command line to run the algorithm on a test partition using a pre-trained model
  python3 test .py

- Trained model:
  "model.txt" for problem 1~3;
  "trained.model" and "tables.txt" for problem 4.

python version: python3

Result

	Accuracy on dev		Accuracy on test		Training time
	overall	unknown	overall	unknown	
Problem 1	93.19%	33.20%	91.40%	39.63%	0 sec
Problem 2	87.40%	27.52%	86.27%	26.13%	0 sec
Problem 3	89.19%	42.92%	88.93%	44.20%	3 sec
Problem 4	84.68%	47.17%	84.14%	48.97%	58min 27 sec

Problem 5

Even though current state-of-the-art in POS tagging performance is already around 97%, the author of this paper believes it is still possible to further improving it.

According to this paper, the token accuracy of state-of-the-art in POS tagging on the WSJ 19-21 development set is 97.28, achieved by a semi-supervised method. It is based on a bidirectional cyclic dependency network tagger from NAACL 2003, and has several improvements on top of it: (1) lowering the support threshold for including rare word features to 5; (2) adds the words two to the left and right as features; (3) adds word shape features; (4) most importantly, adding features based on distributional similarity classes, which were trained separately in an unsupervised way.

To investigate the remaining errors of POS tagging, the author did an error analysis and divided errors to seven classes and count the frequency of each error type. According to the result, most of the errors are from where the gold standard data is just wrong or is inconsistent, and so provides an easiest path for continuing to improve POS tagging. And there is less space left for improvement on the rest types of errors and it is much harder.

On the other hand, people are still using an old distribution of Penn Treebank. Even notice mistakes and deficiencies in the annotation, people are not willing to fix it for several reasons. One important reason is people want to use same dataset so the results are comparable and reproducible. While in other fields such as taxonomic biology, there has been a lot renaming and disruptions to improve the ontological basis of the field. And the author believes this should also happen with the content of treebanks. Thus, the author suggests correcting the treebank and prove a strategy to reduce the error and inconsistency rate in the Penn Treebank, that is, use the syntactic structure of treebank, i.e., tree structure and phrasal categories, to tell the POS tags.