CS 388 Natural Language Processing
Homework 3: Statistical Parsing with "Unsupervised" Domain Adaptation

Pengxiang Cheng (EID: pc23445)
pxcheng@cs.utexas.edu

March 31, 2014

1 Introduction

The goal of this homework is to use the PCFG parser provided in The Stanford Parser Package to do transfer learning of statistical parsing, that is, to adapt a system trained on one source domain to perform better on a new target domain, where supervised labeling data is typically not available. Here we use the *WSJ* and *Brown* as two separate domains to do the experiment.

The method we used here is called "Self Training", which can be divided into three phases. The first is to train the model on a labeled *seed* set from source domain. Then, the trained model is used to produce labeled output (parse trees) for an unlabeled *self-training* set from target domain. At last, the automatically annotated "pseudo-supervised" data is combined with the previous *seed* set to *retrain* the model. Finally, the model is evaluated on a separate testing set from the target domain.

2 Implementation

First we need to do some preprocessing on the raw data file to generate the seed set, the self-training set, and the testing set for our purpose. For the *WSJ* corpora, we use Sec 02-22 as the training set and Sec 23 as the testing set. For the *Brown* corpora, we concatenate the first 90% of each genre as the training set and the last 10% as the testing set.

The domain adaptation can be implemented by interacting with the LexicalizedParser and EvaluateTreebank class. LexicalizedParser:makeTreebank is used to generate treebank objects from *.mrg* files, which is then used in LexicalizedParser:trainFromTreebank to train a PCFG parser. EvaluateTreebank:testOnTreebank is used to evaluate the performance of a PCFG parser on a specified treebank. Also, we modified the function EvaluateTreebank:getInputSentence to generate token list from a tree for the annotation on self-training treebank.

F1 scores of evaluating on source domain testing set and target domain testing set are calculated using different size of seed set and self-training set. We first use *WSJ* as source domain and *Brown* as target

domain and then inverting them. Three learning curves for both cases are generated: normal training and testing on source domain, training on source domain and testing on target domain, and unsupervised domain adaptation by training on source domain, self-training on target domain, and then testing on target domain.

3 Results

The results of using *WSJ* as source and *Brown* as target are as shown in Table 1 and Table 2.

Seed Set	1000	2000	3000	4000	5000	7000	10000
Test on Source	71.95	77.09	79.10	80.57	81.49	82.15	83.29
Test on Target	70.15	73.79	75.28	76.70	77.83	78.59	79.89
Domain Adaptation	70.96	73.17	74.27	75.74	76.35	77.02	77.97
Seed Set	13000	16000	20000	25000	30000	35000	
Test on Source	84.03	84.22	84.50	84.84	85.11	85.21	
Test on Target	80.69	81.10	81.35	81.78	82.09	82.38	
Domain Adaptation	79.04	79.38	79.57	80.03	80.21	80.47	

Table 1: Domain Adaptation results from WSJ to Brown with different seed set size.

Self-training Set	1000	2000	3000	4000	5000	7000	10000	13000	17000	21000
Domain Adaptation	76.85	76.86	77.11	77.25	77.28	77.42	77.74	77.9	77.78	77.98

Table 2: Domain Adaptation results from WSJ to Brown with different self-training set size.

The results of using *Brown* as source and *WSJ* as target are as shown in Table 3 and Table 4.

Seed Set	1000	2000	3000	4000	5000	7000	10000	13000	17000	21000
Test on Source	71.32	74.41	76.67	77.53	78.74	79.84	81.06	81.28	81.53	82.29
Test on Target	68.03	71.79	74.33	75.48	76.32	77.07	78.07	78.38	78.54	78.95
Domain Adaptation	70.35	72.92	75.64	76.09	76.58	77.33	77.88	78.13	78.39	78.24

Table 3: Domain Adaptation results from Brown to WSJ with different seed set size.

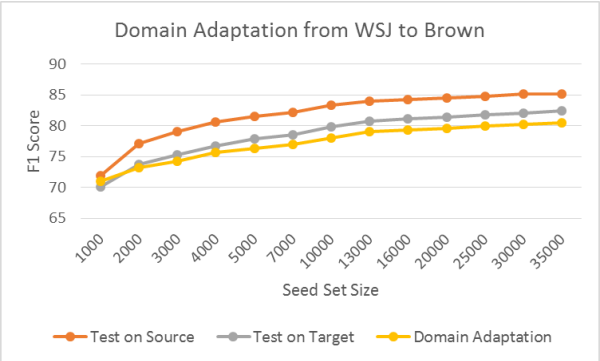
Seed Set	1000	2000	3000	4000	5000	7000	10000
Domain Adaptation	75.75	75.99	76.56	76.86	76.99	77.14	77.23
Seed Set	13000	16000	20000	25000	30000	35000	
Domain Adaptation	77.33	77.45	77.56	77.64	77.76	77.81	

Table 4: Domain Adaptation results from Brown to WSJ with different self-training set size.

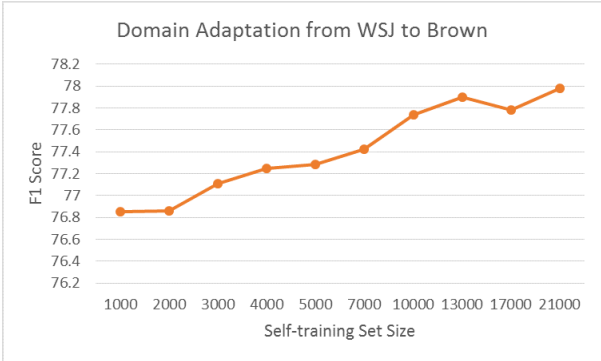
The graphs of different learning curves under different configurations are shown in Fig.1 and Fig.2.

4 Analysis

1. From Fig.1a and Fig.2a, we can see that decrease in accuracy is typically 3% to 4% from in-domain testing to out-of-domain testing.

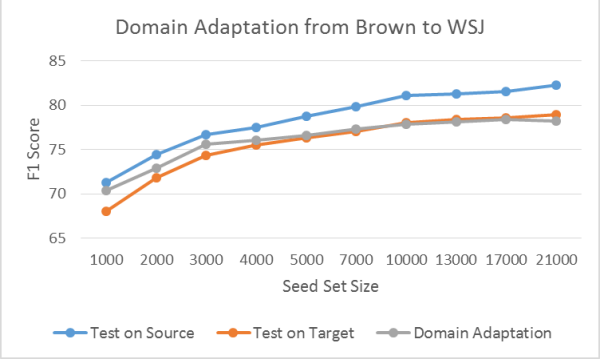


(a) F1 Score v.s. Seed Set Size

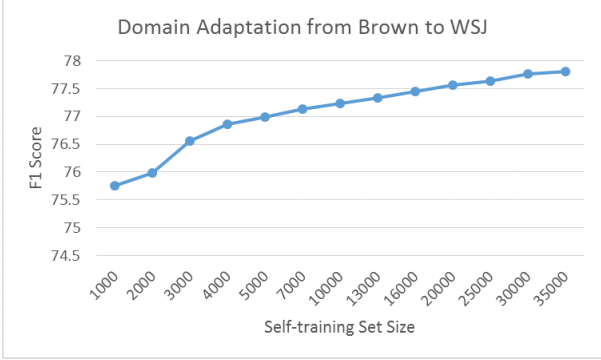


(b) F1 Score v.s. Self-training Set Size

Figure 1: Learning curves of domain adaptation from WSJ to Brown



(a) F1 Score v.s. Seed Set Size



(b) F1 Score v.s. Self-training Set Size

Figure 2: Learning curves of domain adaptation from Brown to WSJ

2. We can not draw a solid conclusion about the impact of unsupervised domain adaptation to the performance of out-of-domain testing. From the "*WSJ* to *Brown*" case shown in Fig.1a, the unsupervised domain adaptation decrease the accuracy of out-of-domain testing rather than increase it, and the reduction becomes larger when the seed set grows. While in the "*Brown* to *WSJ*" case shown in Fig.2a, the unsupervised domain adaptation do introduce improvement of out-of-domain testing when the seed set is small, and the improvement becomes negligible when the seed set grow.

   The phenomenon can be partially explained by that the self-training procedure is in fact a "pseudo-supervised" training, as the labeled parse trees for self-training set are not guaranteed to be consistent with the ground truth. Another way to think about it is that when view the unsupervised domain adaptation as a hard EM process, our method only include one iteration of EM, which would not be likely to reach convergence.

3. From Fig.1a and Fig.2a, increasing the size of seed sets greatly improves parser performance, however this improvement becomes less significant as the seed sets grow larger. From Fig.1b and Fig.2b, increasing the size of self-training sets has very small impact on parser performance, while the performance do improve a little bit as the self-training sets grow very large.

4. Inverting the source and target domain produces similar results, except that the out-of-domain performance from *WSJ* to *Brown* is generally better than that from *Brown* to *WSJ*. Also, in the "*Brown* to *WSJ*" case, the self-training do improve out-of-domain performance when the seed sets from *Brown* is relatively small. Both these results indicate that the *WSJ* corpora might contains more comprehensive lexicalized English grammar than the *Brown* corpora, thus *WSJ* might be a better choice to train a lexicalized parser.

5. Comparing to the results in Reichart and Rappoport paper, the performance of our method given size of seed set equals 2000 is around 72% to 73% for OI experiments, while Reichart and Rappoport's method gave a performance of nearly 80%. Another difference is that Reichart and Rappoport achieved a significant improvement on out-of-domain accuracy after in-domain self-learning, while in our experiment there is no such impact of the self-learning procedure.