Neural Dependency Parsing with Unsupervised Domain Adaptation

Pandian Raju (pandian@cs.utexas.edu, UT ID: pr22695)

April 2017

1 Introduction

The statistical parsers can be quite specific to the genre over which they are trained on. For instance, a parser trained on one corpus, say WSJ, might give significantly lesser accuracies while testing over other genres, say Brown. The task of domain adaptation or transfer learning is to adapt a system trained on one 'source' domain to perform better on a new 'target' domain. This project aims at exploiting "unsupervised domain adaptation" to adapt a statistical parser trained on a source domain to a different target domain. While doing so, the performance obtained, specifically the LAS (Labelled Attachment Score) scores are compared, plotted and analyzed between the baseline, in-domain and out-of-domain testing (with and without domain adaptation) and the reasons for any variations are discussed.

2 Background

- The work by Reichart and Rappoport [2] achieves unsupervised domain adaptation using self-training and it compares the accuracies obtained for various models like II, IO, OI etc where I referes to In-Domain and O refers to Out-of-Domain, with respect to the test data.
- This project aims at getting similar measures as obtained in [2] with the only difference being that the parser used in [2] was Collins parser while this project uses the latest neural dependency parser by Stanford [1].
- The stanford neural parser [1] can be trained over a train set and tested on a test set using train() and testCoNLL() APIs respectively. The labelled training data is given in the conllx format.
- The parser can be configured to run for different number of iterations, the results presented in Section 5 are corresponding to 500 iterations (Although more number of iterations might give better results, 500 iterations are chosen for computational tractability).

3 Methods

3.1 Preprocessing

3.1.1 Preprocessing for WSJ

- WSJ corpus has the labelled conllx files in 25 sub-directories (00 to 24). For the purpose of this experiment, data from directories 02 to 10 (wsj_02_10.conllx) is taken as the training set and directory 23 (wsj_23.conllx) is taken as the test set.
- For all the required seed/self-training sizes of WSJ, the first required number of sentences are read from the WSJ training file wsj_02_10.conllx and written to corresponding individual wsj seed/self-training files.

3.1.2 Preprocessing for Brown

- Brown corpus contains different genres (8 genres) and each of the genre has different number of sentences. For the purpose of this experiment, the Brown corpus is split into 90% for the seed/self-training set and 10% for the test set. While doing so, the first 90% of sentences from each genre are chosen for the training set and the remaining 10% of sentences from each genre are chosen for the test set. This ensures uniformity among the different genres.
- For different seed/self-training sizes of Brown, a similar selection of sentences is done. For each size T, T sentences are chosen from the training set collated in the previous step, distributed across all the genres not equally but proportional to the size of the genres. Formally, let T be the target number of sentences to choose, N be the total number of sentences in the Brown training set and M be the number of sentences from a particular genre, then $(M/N) * T$ sentences are chosen from that genre. This is done to ensure that each genre in Brown corpus is given fair weightage proportional to the amount of training corpus in that genre.

3.2 Processing

- Once the preprocessing is complete, all the required files for seed or self-training or testing are available.
- The Stanford Neural Parser provides APIs to train a DependencyParser given a training file path and test over a test file.
- For the self-training, the model trained on source corpus is tested over the self-training set from the target domain and the annotations obtained are combined with the seed set. The combined file is then used to train the model again and then it is tested over the test set from the target domain.

4 Experiments

Four set of experiments are conducted to analyze the LAS scores for different sets of source and target corpuses and for different models like in-domain testing and OI:

1. **Source domain: WSJ; Target domain: Brown; Varying the seed set size** - For different seed sizes (number of sentences in seed set) of WSJ [1000, 2000, 3000, 4000, 5000, 7000, 10000, 12000, 14000],
   (a) The parser is trained over WSJ with corresponding seed size and tested over WSJ test set (wsj_23.conllx).
   (b) The parser is trained over WSJ with corresponding seed size and tested over Brown test set (last 10% sentences in each genre of Brown).
   (c) The parser is trained over WSJ with corresponding seed size, self-trained over Brown training set (first 90% sentences in each genre of Brown) and then tested over Brown test set.
2. **Source domain: WSJ; Target domain: Brown; Varying the self training set size** - For different self training sizes of Brown training set [1000, 2000, 3000, 4000, 5000, 7000, 10000, 13000, 17000, 21000], the parser is trained over WSJ of seed size 10000, self-trained over Brown with corresponding self training size and tested over Brown test set.
3. **Source domain: Brown; Target domain: WSJ; Varying the seed set size** - For different seed sizes of Brown [1000, 2000, 3000, 4000, 5000, 7000, 10000, 13000, 17000, 21000],
   (a) The parser is trained over Brown with corresponding seed size and tested over Brown test set.
   (b) The parser is trained over Brown with corresponding seed size and tested over WSJ test set.

(c) The parser is trained over Brown with corresponding seed size, self-trained over WSJ training set (wsj_02_10.conllx) and then tested over WSJ test set.

4. **Source domain: Brown; Target domain: WSJ; Varying the self training set size** - For different self training sizes of WSJ training set [1000, 2000, 3000, 4000, 5000, 7000, 10000, 12000, 14000], the parser is trained over Brown of seed size 10000, self-trained over WSJ with corresponding self training size and tested over WSJ test set.

5 Results

5.1 Varying seed size

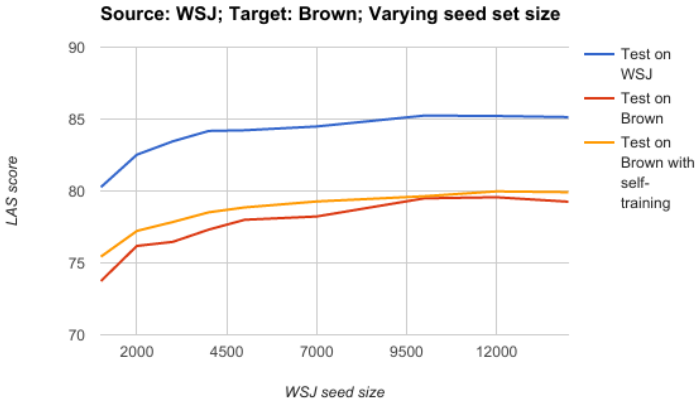

Figure 1: LAS scores for in-domain testing, baseline out-of-domain and OI with WSJ as source and Brown as target

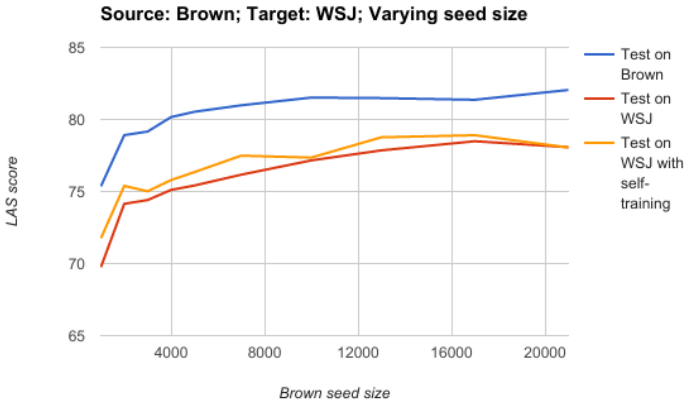

Figure 2: LAS scores for in-domain testing, baseline out-of-domain and OI with Brown as source and WSJ as target

Figures 1 and 2 show the LAS scores obtained for the three different models: in-domain testing, baseline out-of-domain (seed) and OI (with self-training), as the size of the seed set is increased. The following observations can be made:

- It is very clear that the in-domain testing has higher LAS score compared to out-of-domain training (OI and baseline). The reason is that the source and target are different domains

and hence training on one domain need not necessarily represent the target domain. For example, there can be many tokens in the target set which did not even appear in the source domain. Hence the performance drops by around 5%-7% from in-domain to out-of-domain testing.

- We can also observe that the unsupervised domain adaptation helps in increasing the LAS score by around 1%-3% while testing on the target domain. This is a consequence of the result illustrated by [2].
- An other observation is that as the seed set size increases, the LAS score also increases in all three cases. This is just a result of training the parser on enough training data (otherwise it will be underfitted).
- The relative trend is same on inverting the source and target domains but the actual LAS scores goes down by roughly 2%-4% on inverting the source and target. This is because Brown corpus contains a wide variety of genres within itself and so the parser is trained over different genres with lesser amount of training data in each genre. Hence, adapting that to the target domain (WSJ) doesn't give as good performance as the vice-versa.

5.2 Varying self training size

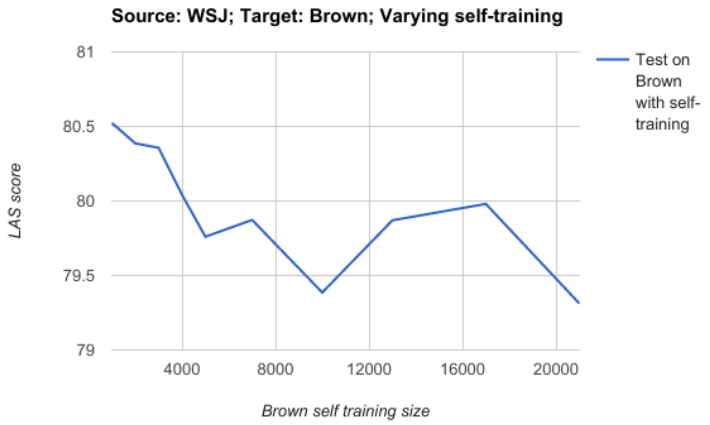

Figure 3: LAS scores for OI with self-training with WSJ as source and Brown as target

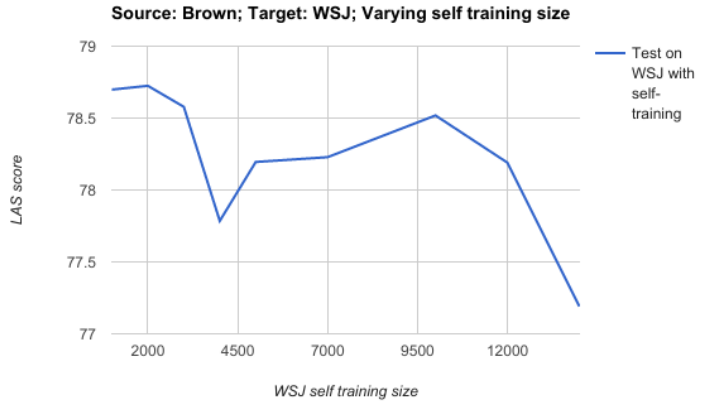

Figure 4: LAS scores for OI with self-training with Brown as source and WSJ as target

Figures 3 and 4 show the LAS scores obtained for out-of-domain training with in-domain self-training, as the size of the self training set is increased. The following observations can be made:

- As the size of the self-training size is increased, the LAS score of the target domain starts decreasing. The optimal self training size for Brown is 1000 while the optimal value for WSJ is 2000.
- As we increase the self training size beyond the optimal value, the model is overfitted leading to loss of accuracy. When more self-training data is chosen, it means that the combined corpus for the re-training can contain more noise (since the initial annotations over target corpus can contain many incorrect predictions) making the overall model input noisy.
- As the source and target are inverted, WSJ has a different optimal value for self training size. This is just a property of the corpus itself and it depends on the seed size as well. But in general, testing on Brown with seed as WSJ has little higher accuracy over testing on WSJ with Brown as seed set and the reason corresponds to diverse genres of Brown as explained above.

5.3 Comparision with results from [2]

- It can be seen that the F-score for the testing (with in-domain seed, out-of-domain baseline and OI) increases with the seed size of the out-of-domain corpus in [2] and similar trend can be seen in Figures 1 and 2.
- Also, with self-training (OI setting), [2] achieves a higher F-score compared to the baseline (without self-training) and it can be inferred from Figures 1 and 2 as well.
- Hence, while testing on a set with out-of-domain seed, doing self-training on a small amount of in-domain training data helps in increasing the accuracy which is depicted both in this project and [2].

5.4 Number of iterations

Figure 5: LAS scores for OI with self-training with WSJ as source and Brown as target

- Figure 5 shows the LAS scores obtained for OI model with self-training when the parser is trained for different number of iterations.
- It is clear that as the number of iterations is increased, better LAS scores are obtained since for lower number of iterations the model is underfitted.
- Also, the curve for 500 iterations is very smooth while the curve for 200 iterations is very irregular and is due to insufficicent training of the model.

5.5 Runtime performance

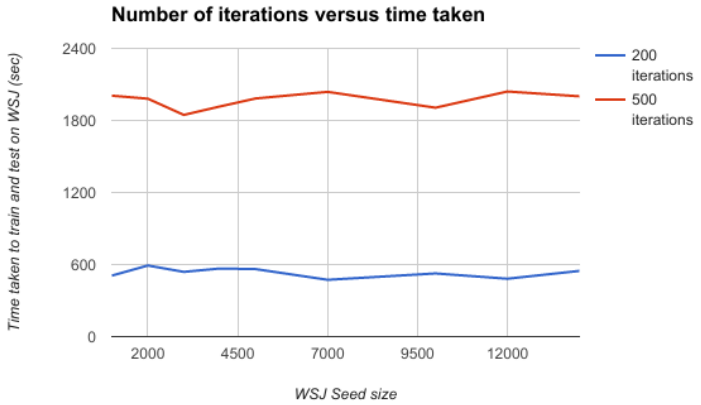

Figure 6: Time taken for in-domain testing with WSJ as source

- Figure 6 shows the time taken by the parser to train and test for an in-domain testing model for 200 and 500 iterations. It can be seen that increasing the number of iterations directly increases the training time taken by the parser, although it increases the LAS scores as well as shown by Figure 5.

Figure 7: Time taken for OI model and OI model with self-training with WSJ as source and Brown as target

- Figure 7 shows that using self-training for an OI model also increases the time taken to train the parser since it needs to re-train over the combined corpus. Although this takes more time, it increases LAS scores over normal OI model and hence should be used when higher accuracies are required.

6 Conclusion

Unsupervised domain adaptation is experimented and the effects of self-training are analyzed. Using different source and target domains, the LAS scores obtained for different models are analyzed and it is concluded that using a small set of the target domain for self-training gives an improvement in LAS by a tangible amount in the field of NLP.

References

[1] Danqi Chen and Christopher D. Manning. "A Fast and Accurate Dependency Parser using Neural Networks". In: *EMNLP* (2014). URL: `https://nlp.stanford.edu/software/nndep.shtml`.

[2] Roi Reichart and Ari Rappoport. "Self-Training for Enhancement and Domain Adaptation of Statistical Parsers Trained on Small Datasets". In: *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics* (2007), pp. 616–623.