Unsupervised domain adaptation for joint segmentation and POS-tagging

Yang Liu[1] *Yue Zhang*[2]
(1) University of Cambridge
(2) Singapore University of Technology and Design
yang.liu@cantab.net, yue_zhang@sutd.edu.sg

ABSTRACT
Sophisticated models have been developed for joint word segmentation and part-of-speech tagging, with increasing accuracies reported on the Chinese Treebank data. These systems, which rely on supervised learning, typically perform worse on texts from a different domain, for which little annotation is available. We consider self-training and character clustering for domain adaptation. Both methods use only unannotated target-domain data, and are relatively straightforward to implement upon a baseline supervised system. Our results show that both methods can effectively improve target-domain performance. In addition, a combination of the two orthogonal methods leads to further improvement.

TITLE AND ABSTRACT IN CHINESE

分词与词性标注联合模型的领域适应

分词与词性标注的联合模型是一个正在被广泛研究的问题，随着复杂模型的应用，其在宾大中文数据库上的测试精度不断提升。这些方法通常使用有监督学习，致使在不同领域下的效果不如单一领域满意。我们用自学习和字聚类实现领域适应。这两个方法使用未标注领域训练数据，而且易于实现。我们的实验结果表明，这两种方法都可以提高领域适应。同时，这两种方法可以结合使用达到更高性能。

KEYWORDS: Semi-supervised learning, domain adaptation, word segmentation, POS-tagging.
KEYWORDS IN CHINESE: 分词，词性标注，领域适应，半监督学习，聚类，自学习

Proceedings of COLING 2012: Posters, pages 745–754,
COLING 2012, Mumbai, December 2012.

1 Introduction

Joint segmentation and POS-tagging can improve upon a pipelined baseline by reducing error propagation and accommodating features that represent combined word and POS information. Three general approaches have been taken to perform joint inference, namely two-stage ensemble methods (Jiang et al., 2008a; Sun, 2011), reranking (Jiang et al., 2008b; Shi and Wang, 2007) and single joint models with heuristic search (Ng and Low, 2004; Zhang and Clark, 2008; Kruengkrai et al., 2009; Zhang and Clark, 2010), leading to improved accuracies on the Chinese Treebank data.

All these methods rely on supervised learning, and are expected to perform worse when the test domain shifts from CTB to blogs, computer forums, and internet literature, which are written in a different genre, and for which little manual annotation is available. In this paper, we choose internet literature as the target domain, and study domain adaptation for joint segmentation and POS-tagging. We consider the single model approach of Zhang and Clark (2010), trained using the CTB, as our baseline system, and apply self-training and character clustering to improve its performance on our test data from an internet novel.

Much work has been done on domain adaptation for POS-tagging (Blitzer et al., 2006; Daumé III and Marcu, 2006; Jiang and Zhai, 2007). However, relatively little attention has been paid to the domain adaptation for joint segmentation and POS-tagging. Among the range of methods that have been developed for domain adaptation, self-training and character clustering are applicable to a comparatively large number of baseline supervised model types, including feature-based probability models and large-margin discriminative models, and are fairly straightforward to implement. We focus on unsupervised domain adaptation, using fully unannotated data in the target-domain.

We evaluate our system on a set of manually annotated target-domain data. Our baseline system, trained using the CTB, gave an overall segmentation and POS-tagging F-score of 82.20% on this set. Application of self-training and character clustering improved the overall F-score to 83.17% and 82.56%, respectively. Since these two methods are orthogonal, they were combined to further improve the overall F-score to 83.99%.

2 Self-training

Self-training is a general semi-supervised learning approach. It has been applied to several NLP tasks with mixed results reported. Clark et al. (2003) apply self-training to POS-tagging and achieve minor improvements. Steedman et al. (2003) report that self-training can either slightly improve or significantly harm the parsing accuracy. McClosky et al. (2006) achieves improved parsing accuracies using self-training, and Reichart and Rappoport (2007) has obtained significant improvement on small datasets with lexicalized parser.

In this paper, we focus on the use of self-training for unsupervised domain adaptation. Self-training has been applied to the domain adaptation of several NLP tasks, including parsing (Roark and Bacchiani, 2003; Sagae, 2010), POS-tagging (Jiang and Zhai, 2007) and cross-language text classification (Shi et al., 2010). It improves system performance on the target domain by simultaneously modelling annotated source-domain data and unannotated target-domain data in the training process. Theoretically, self-training has a strong relationship with the EM algorithm, where tagging unlabeled data corresponds to the expectation step, and supervised parameter estimation corresponds to the maximization step. There are various factors that affects the effectiveness of self-training, such as the difference in the distributions of

Data set	chap. IDs	# of sen.	# of words
Training	1-270, 400-931, 1001-1151	18089	493939
Development	301-325	350	6821
Test	271-300	348	8008

Table 1: CTB training, development and test data.

labeled and unlabeled data, the supervised training algorithm, and additional reranking and filtering of output predictions.

Modifications can be made to the standard self-training process for domain adaptation to address the difference in source and target distributions (Margolis, 2011). In Tan et al. (2009), the weights on the target-domain data is increased at each iteration; in Saerens et al. (2002), EM is applied to the target-domain only, and the source data is used for an initial estimation. In this paper, we apply the standard self-training process, but with target-domain data point selection (Rehbein, 2011; Søgaard, 2011).

3 Character clustering

Word/character clustering is an unsupervised approach that groups similar words/characters according to their context. Clusters can be used as features instead of the original words/characters for the reduction of data sparsity. Word clustering has been applied to many NLP problems (Miller et al., 2004; Liang, 2005; Koo et al., 2008).

For our domain adaptation problem, clusters are created from large unannotated target-domain data, and applied as features in our joint segmentor and POS-tagger during both training and testing. The weights of the cluster features are estimated during training using source-domain data. During testing, they can help to alleviate the out-of-vocabulary (OOV) problem in the target-domain when a rare input has not been seen in the training data but belongs to a known cluster.

We use Liang's implementation (Liang, 2005) of the bottom-up agglomerative Brown algorithm (Brown et al., 1992) to generate character clusters, choosing the numbers of clusters according to development experiments.

4 Experiments

Software We use ZPar (Zhang and Clark, 2010, 2011) as the baseline system[1]. The system uses a single discriminative model for joint segmentation and tagging, trained using the generalized perceptron algorithm. Standard beam search is applied to ensure efficient decoding.

Source-domain data We use the CTB 5 for source-domain training, making the same training, development and test sections as Kruengkrai et al. (2009) (Table 1).

Target-domain data We collect the target-domain data from a Chinese Internet novel "*Jade dynasty*"[2] (also known as "*Zhuxian*") by Ding Xiao. The first 18 chapters (927K words in 25413 sentences) have been collected. Section 1 of chapter 6 is used as the development data, and

[1] www.sourceforge.net/project/zpar; version 0.4

[2] An electronic version of the book is free for download from the Internet.

Data set	chap. IDs	# of sen.	# of words
Training	1-5, 8-18	25413	927405
Development	6.1	159	5077
Test	7.2	226	5173

Table 2: Target-domain training, development and test data.

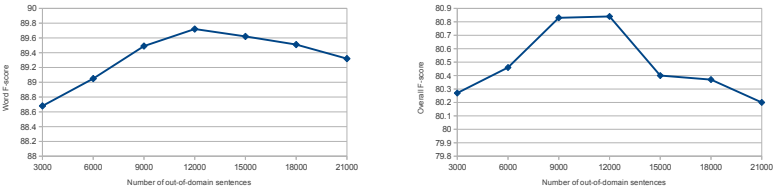

Figure 1: Development word F-scores (left) and overall word and POS F-scores (right) of self-training.

section 2 of chapter 7 is used as the test data. The remaining 16 chapters are used as the training data, as shown in Table 2. We manually annotate the development and test data to produce the gold standard reference.

Evaluation We follow Zhang and Clark (2008) and Kruengkrai et al. (2009), and use standard F-scores to measure both the word segmentation accuracy and the overall word segmentation and POS tagging accuracy. The F-score is $TF = \frac{2pr}{p+r}$ where p is the precision and r is the recall. The precision p is calculated as the percentage of correct tokens in the output, and the recall r as the percentage of golden-standard tokens that are correctly identified by the program. For word F-score, a correct token is identified as a word with the correct word boundary. For overall word and POS F-score, both the word boundary and the POS tag must be correct to make the word a correct token.

4.1 Self-training development experiments

We produce different amounts of target-domain training data by taking the first n sentences of the internet novel, with n ranging from 3000 to 21000. The sentences are automatically annotated and then combined with the CTB training data. Development test results achieved with the optimal numbers of training iterations are shown in Figure 1. The results are consistently higher than the baseline (79.64%), and the best accuracy (80.83%) is achieved with 12000 target-domain sentences (424K words).

Figure 1 also suggests that more raw text does not always lead to improved target-domain test accuracies for self-training. When the number of target-domain sentences exceeds 12000, the accuracies start to decrease. Similar observations have been reported for a cross-domain parsing task (Zhang et al., 2010). Possible reasons include the difference between source- and target-domain texts, and the intrinsic nature of self-training. To further study the problem, we conduct data-point selection (Søgaard, 2011; Rehbein, 2011), choosing to use those target-

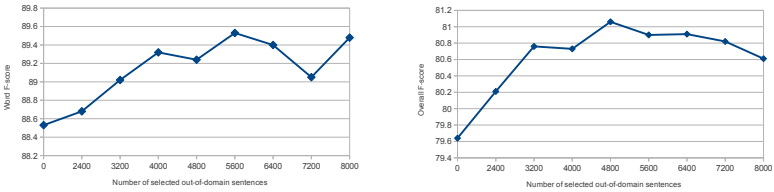

Figure 2: Development word F-scores (left) and overall word and POS F-scores (right) of self-training with data-point selection.

domain sentences that are most similar to the source-domain data for self-training, so that we can separate out the effect of text dissimilarity to some extent. To measure similarity, we use the source-domain training data to train a trigram character language model, and use perplexity per character to measure the similarity to source-domain data for target-domain sentences.

We produce different amounts of training data by selecting the top n sentences from the target domain with the lowest perplexity per character, with n ranging from 2400 to 8000, and combining them with the CTB training data. Figure 2 shows our development test results with respect to the number of target-domain sentences. The best result (F-score = 81.06%) is achieved with 4800 selected target domain sentences, which is slightly better than our previous result of self-training (F-score = 80.83%). The "self-training (perplexity)" rows in Tables 4 and 5 show the development and final test results of this method.

As Figure 2 shows, the F-scores increase when the amount of target-domain unannotated data increases from 0 to 4800, demonstrating the effect of sentences that are most similar to the source-domain data. When the amount of data increases, the perplexity of the additional data starts to increase, and the target-domain sentences are less similar to the source-domain sentences. After the peak point, the accuracy of self-training starts to decrease with more unannotated sentences. These observations suggest that data distribution does influence the effect of self-training on domain adaptation, and also partly explains why more unannotated data do not necessarily lead to improved target-domain accuracies in previous experiments.

4.2 Clustering development experiments

To include cluster information in the tagger, we add 10 cluster-based features to the feature templates used by ZPar, as shown in Table 3. Templates 1-6 contain only word information and templates 7-10 contain both word and POS information. w, t and c represent a word, a POS tag and a cluster bit-string, respectively. The subscripts in the templates are based on the current character, e.g. w_{-2} is the second word to the left of the current character. All templates are instantiated when the current character starts a new word. We select these feature templates based on the feature templates of Zhang and Clark (2010) and our development experiments.

Our clusters are extracted from the combined source- and target-domain data using the Brown algorithm. Following Koo et al. (2008) and Miller et al. (2004), we use specific prefixes of the cluster hierarchy to produce clusterings of varying granularity. Koo et al. (2008) used short

ID	Features	ID	Features
1	$w_{-2}c\_start(w_{-1})$	6	$c_{-2}c_{-1}c_0$
2	$w_{-1}c_{-1}$	7	$c_0t_{-1}t_0$
3	$w_{-1}c_0$	8	$c_0t_{-2}t_{-1}t_0$
4	c_{-1}	9	$c_{-1}t_0$
5	$c_{-1}c_0$	10	$c\_start(w_{-1})t_{-2}$

Table 3: Cluster-based feature templates.

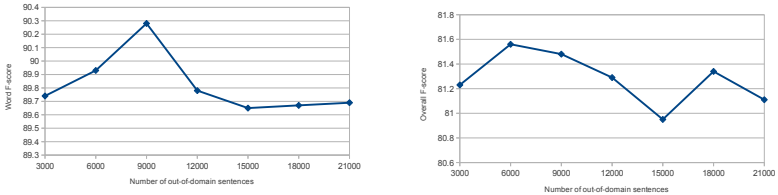

Figure 3: Development word F-scores (left) and overall word and POS F-scores (right) of the combined method.

bit-string and full bit prefixes for dependency parsing; Miller et al. (2004) used longer prefixes (12 to 20 bits) for the named-entity tagging task. In our case, we try every possible prefix length ranging from 4 to 18, both individually and jointly, and choose to use the combination of 14- and 16-bit prefixes.

Our development tests suggest that the best accuracy is achieved with the clustering extracted from the combined dataset consisting of 1000 clusters. We achieve an overall F-score of 80.26% on the Internet literature development data, which is higher than the baseline and proves the effectiveness of character clustering on this task.

4.3 Combining the two methods

Since the two methods are orthogonal to each other, they can be combined to achieve further improvement. In each of the following experiments, the same target-domain sentence set is used for both self-training and clustering. Figure 3 shows the development test results with respect to the amount of target-domain data. The highest accuracy (81.49%) is achieved with 9000 target-domain sentences, which we choose to use in our final test. Table 4 gives a summary of our development experiments.

4.4 Final test results

Table 5 shows our final test results. Similar to the development experiments, both self-training and character clustering improve the performance of the system on the target-domain, and the combined method achieves further improvement. Character clustering gives less improvements over the baseline than self-training in both the development tests and the final test. We give discussions on possible reasons in the next section.

	F-score
baseline	79.64
character clustering	80.26
self-training	80.83
self-training (perplexity)	81.06
combined method	81.49

Table 4: Development test summary.

	F-score
baseline	82.20
character clustering	82.56
self-training	83.17
self-training (perplexity)	83.32
combined method	83.99

Table 5: Final test results.

5 Discussions

In this section we give error analysis and some intuitions about the effect of the methods that have been applied in our experiments.

Self-training The most important improvement with self-training is the more accurate handling of proper nouns such as the names of persons or locations. For example, the following sentence is from the target-domain dataset: "萧逸才转头对田不易道 (Yicai Xiao turns his head towards Buyi Tian and says)". The correct segmentation and tagging of the sentence should be:

"萧逸才_NR (Yicai Xiao) 转头_VV (turn one's head) 对_P (towards) 田不易_NR (Buyi Tian) 道_VV (say)"

The output from the baseline system is:

"萧逸_NR (XiaoYi) 才_AD (Cai) 转头_VV (turn one's head) 对_P (towards) 田_NN (Tian) 不_AD (Bu) 易道_VV (Yi say),"

with both segmentation and POS-tagging errors. Using the self-trained model, the output sentence becomes:

"萧逸才_NN (Yicai Xiao) 转头_VV (turn one's head) 对_P (towards) 田不易_NR (Buyi Tian) 道_VV (say),"

which contains only one tagging error and no segmentation error — a significant improvement over the baseline result.

The two OOV personal names contribute to all baseline errors. The three characters of the second name, "田不易", are more likely to be used individually (田→ field, 不→ not, 易→ easy), which is a possible explanation to the errors made by the baseline model. In the automatically annotated target-domain data, however, in-vocabulary local context can lead "田不易" to be

tagged as a single word most of the time, which could then improve the self-trained model.

Character clustering Compared to self-training, character clustering helps joint word segmentation and POS tagging in a different way, by improving the recognition of words containing rare characters and single-character words.

For example, the rare character "鳌", which stands for a legendary animal, was tagged as FW (foreign word) by the baseline tagger. The cluster-based tagger, on the other hand, is able to identify it as a noun, since it appears in the same cluster with many nouns, which indicates its syntactical similarity with nouns.

For another example, the character "而" can appear as a part of an adverb (AD). Such cases including the word "然而 (however)" or "从而 (so that)". It could also be used as a single-character conjunction or part of a conjunction, e.g. "而 (and)" and "而且 (besides)", with the POS tag "CC". Yet another less frequent use of "而" is as an auxiliary that connects an adverb to a verb, as in "侃侃 (confidently) 而 (auxiliary) 谈 (talk)", whose POS tag is "MSP". Since the first two cases are more likely to happen, the baseline model mostly treats "而" as a conjunction or an adverb, rather than an auxiliary. With the clustering information, the tagger receives more information from single-character words, therefore could tag the character "而" as "MSP" rather than "CC" or "AD" when it appears alone.

The combined method One explanation for the comparatively less effect of the character clustering method compared to the self-training method is that, although data sparsity is reduced, the weights to the cluster-based features are trained on annotated data, therefore capturing the distribution of source-domain data. When the two methods are combined, some target-domain data are used to train the feature weights of the clusters, and therefore they can play a better role in improving target-domain accuracies.

The combined method does combine the advantages of both self-training and clustering. We find that both the handling of personal names and the identification of rare characters are improved.

6 Conclusion

We studied the domain adaptation problem for joint segmentation and POS-tagging. Trained using the Chinese Treebank, the baseline system gave significantly lower accuracies on test data from internet literature. We applied self-training and unsupervised clustering to improve target-domain accuracies, both of which require comparatively small changes to the supervised baseline system, and use fully unannotated target-domain data. We observed positive results using both methods, and a combination of the methods led to further improvements. Future work remain to further reduce the gap between in-domain and out-of-domain performances for joint segmentation and tagging.

References

Blitzer, J., McDonald, R., and Pereira, F. (2006). Domain adaptation with structural correspondence learning. In *Proceedings of EMNLP*, pages 120–128, Sydney, Australia.

Brown, P. F., Pietra, V. J. D., deSouza, P. V., Lai, J. C., and Mercer, R. L. (1992). Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Clark, S., Curran, J., and Osborne, M. (2003). Bootstrapping POS-taggers using unlabelled data. In *Proceedings of CoNLL-2003*, pages 49–55.

Daumé III, H. and Marcu, D. (2006). Domain adaptation for statistical classifiers. *Artificial Intelligence Research*, 26:101–126.

Jiang, J. and Zhai, C. (2007). Instance weighting for domain adaptation in nlp. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 264–271, Prague, Czech Republic. Association for Computational Linguistics.

Jiang, W., Huang, L., Liu, Q., and Lü, Y. (2008a). A cascaded linear model for joint Chinese word segmentation and part-of-speech tagging. In *Proceedings of ACL/HLT*, pages 897–904, Columbus, Ohio.

Jiang, W., Mi, H., and Liu, Q. (2008b). Word lattice reranking for Chinese word segmentation and part-of-speech tagging. In *Proceedings of COLING*, pages 385–392, Manchester, UK.

Koo, T., Carreras, X., and Collins, M. (2008). Simple semi-supervised dependency parsing. In *Proceedings of ACL/HLT*, pages 595–603, Cambridge, MA.

Kruengkrai, C., Uchimoto, K., Kazama, J., Wang, Y., Torisawa, K., and Isahara, H. (2009). An error-driven word-character hybrid model for joint Chinese word segmentation and POS tagging. In *Proceedings of ACL/AFNLP*, pages 513–521, Suntec, Singapore.

Liang, P. (2005). Semi-supervised learning for natural language. Master's thesis, Massachusetts Institute of Technology.

Margolis, A. (2011). A literature review of domain adaptation using unlabeled data.

McClosky, D., Charniak, E., and Johnson, M. (2006). Effective self-training for parsing. In *Proceedings of HLT-NAACL*, pages 152–159.

Miller, S., Guinness, J., and Zamanian, A. (2004). Name tagging with word clusters and discriminative training. In *Proceedings of HLT-NAACL*, Cambridge, MA.

Ng, H. T. and Low, J. K. (2004). Chinese part-of-speech tagging: One-at-a-time or all-at-once? word-based or character-based? In *Proceedings of EMNLP*, Barcelona, Spain.

Rehbein, I. (2011). Data point selection for self-training. In *Proceedings of the Second Workshop on Statistical Parsing of Morphologically Rich Languages*.

Reichart, R. and Rappoport, A. (2007). Self-training for enhancement and domain adaptation of statistical parsers trained on small datasets. In *Proceedings of ACL*.

Roark, B. and Bacchiani, M. (2003). Supervised and unsupervised pcfg adaptation to novel domains. In *HLT-NAACL'03*, pages –1–1.

Saerens, M., Latinne, P., and Decaestecker, C. (2002). Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, pages 21–41.

Sagae, K. (2010). Self-training without reranking for parser domain adaptation and its impact on semantic role labeling. In *Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing*, pages 37–44, Uppsala, Sweden. Association for Computational Linguistics.

Shi, L., Mihalcea, R., and Tian, M. (2010). Cross language text classification by model translation and semi-supervised learning. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 1057–1067, Cambridge, MA. Association for Computational Linguistics.

Shi, Y. and Wang, M. (2007). A dual-layer CRF based joint decoding method for cascade segmentation and labelling tasks. In *Proceedings of IJCAI*, Hyderabad, India.

Søgaard, A. (2011). Data point selection for cross-language adaptation of dependency parsers. In *49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (ACL-HLT)*.

Steedman, M., Osborne, M., Sarkar, A., Clark, S., Hwa, R., Hockenmaier, J., Ruhlen, P., Baker, S., and Crim, J. (2003). Bootstrapping statistical parsers from small datasets. In *Proceedings of EACL*, Budapest, Hungary.

Sun, W. (2011). A stacked sub-word model for joint Chinese word segmentation and part-of-speech tagging. In *Proceedings of ACL/HLT*, pages 1385–1394, Portland, Oregon, USA. Association for Computational Linguistics.

Tan, S., Cheng, X., Wang, Y., and Xu, H. (2009). Adapting naive bayes to domain adaptation for sentiment analysis. In *ECIR'09*, pages 337–349.

Zhang, Y., Ahn, B.-G., Clark, S., Van Wyk, C., Curran, J. R., and Rimell, L. (2010). Chart pruning for fast lexicalised-grammar parsing. In *Coling 2010: Posters*, pages 1471–1479, Beijing, China. Coling 2010 Organizing Committee.

Zhang, Y. and Clark, S. (2008). Joint word segmentation and POS tagging using a single perceptron. In *Proceedings of ACL/HLT*, pages 888–896, Columbus, Ohio.

Zhang, Y. and Clark, S. (2010). A fast decoder for joint word segmentation and POS-tagging using a single discriminative model. In *Proceedings of EMNLP*, pages 843–852, Cambridge, MA.

Zhang, Y. and Clark, S. (2011). Syntactic processing using the generalized perceptron and beam search. *Computational Linguistics*, 37(1):105–151.