Adapting Self-training for Semantic Role Labeling

Rasoul Samad Zadeh Kaljahi
FCSIT, University of Malaya
50406, Kuala Lumpur, Malaysia.
rsk7945@perdana.um.edu.my

Abstract

Supervised semantic role labeling (SRL) systems trained on hand-crafted annotated corpora have recently achieved state-of-the-art performance. However, creating such corpora is tedious and costly, with the resulting corpora not sufficiently representative of the language. This paper describes a part of an ongoing work on applying bootstrapping methods to SRL to deal with this problem. Previous work shows that, due to the complexity of SRL, this task is not straight forward. One major difficulty is the propagation of classification noise into the successive iterations. We address this problem by employing *balancing* and *preselection* methods for self-training, as a bootstrapping algorithm. The proposed methods could achieve improvement over the base line, which do not use these methods.

1 Introduction

Semantic role labeling has been an active research field of computational linguistics since its introduction by Gildea and Jurafsky (2002). It reveals the event structure encoded in the sentence, which is useful for other NLP tasks or applications such as information extraction, question answering, and machine translation (Surdeanu et al., 2003). Several CoNLL shared tasks (Carreras and Marquez, 2005; Surdeanu et al., 2008) dedicated to semantic role labeling affirm the increasing attention to this field.

One important supportive factor of studying *supervised* statistical SRL has been the existence of hand-annotated semantic corpora for training SRL systems. *FrameNet* (Baker et al., 1998) was the first such resource, which made the emergence of this research field possible by the seminal work of Gildea and Jurafsky (2002). However, this corpus only exemplifies the semantic role assignment by selecting some illustrative examples for annotation. This questions its suitability for statistical learning. *Propbank* was started by Kingsbury and Palmer (2002) aiming at developing a more representative resource of English, appropriate for statistical SRL study.

Propbank has been used as the learning framework by the majority of SRL work and competitions like CoNLL shared tasks. However, it only covers the newswire text from a specific genre and also deals only with verb predicates.

All state-of-the-art SRL systems show a dramatic drop in performance when tested on a new text domain (Punyakanok et al., 2008). This evince the infeasibility of building a comprehensive hand-crafted corpus of natural language useful for training a robust semantic role labeler.

A possible relief for this problem is the utility of *semi-supervised* learning methods along with the existence of huge amount of natural language text available at a low cost. Semi-supervised methods compensate the scarcity of labeled data by utilizing an additional and much larger amount of unlabeled data via a variety of algorithms.

Self-training (Yarowsky, 1995) is a semi-supervised algorithm which has been well studied in the NLP area and gained promising result. It iteratively extend its training set by labeling the unlabeled data using a base classifier trained on the labeled data. Although the algorithm is theoretically straightforward, it involves a large number of parameters, highly influenced by the specifications of the underlying task. Thus to achieve the best-performing parameter set or even to investigate the usefulness of these algorithms for a learning task such as SRL, a thorough experiment is required. This work investigates its application to the SRL problem.

2 Related Work

The algorithm proposed by Yarowsky (1995) for the problem of word sense disambiguation has been cited as the origination of self-training. In that work, he bootstrapped a ruleset from a

Proceedings of the ACL 2010 Student Research Workshop, pages 91–96,
Uppsala, Sweden, 13 July 2010.

Feature Name	Description
Phrase Type	Phrase type of the constituent
Position+Predicate Voice	Concatenation of constituent position relative to verb and verb voice
Predicate Lemma	Lemma of the predicate
Predicate POS	POS tag of the predicate
Path	Tree path of non-terminals from predicate to constituent
Head Word Lemma	Lemma of the head word of the constituent
Content Word Lemma	Lemma of the content word of the constituent
Head Word POS	POS tag of the head word of the constituent
Content Word POS	POS tag of the head word of the constituent
Governing Category	The first VP or S ancestor of a NP constituent
Predicate Subcategorization	Rule expanding the predicate's parent
Constituent Subcategorization *	Rule expanding the constituent's parent
Clause+VP+NP Count in Path	Number of clauses, NPs and VPs in the path
Constituent and Predicate Distance	Number of words between constituent and predicate
Compound Verb Identifier	Verb predicate structure type: simple, compound, or discontinuous compound
Head Word Location in Constituent *	Location of head word inside the constituent based on the number of words in its right and left

Table 1: Features

small number of seed words extracted from an online dictionary using a corpus of unannotated English text and gained a comparable accuracy to fully supervised approaches.

Subsequently, several studies applied the algorithm to other domains of NLP. Reference resolution (Ng and Cardie 2003), POS tagging (Clark et al., 2003), and parsing (McClosky et al., 2006) were shown to be benefited from self-training. These studies show that the performance of self-training is tied with its several parameters and the specifications of the underlying task.

In SRL field, He and Gildea (2006) used self-training to address the problem of unseen frames when using FrameNet as the underlying training corpus. They generalized FrameNet frame elements to 15 thematic roles to control the complexity of the process. The improvement gained by the progress of self-training was small and inconsistent. They reported that the NULL label (non-argument) had often dominated other labels in the examples added to the training set.

Lee et al. (2007) attacked another SRL learning problem using self-training. Using Propbank instead of FrameNet, they aimed at increasing the performance of supervised SRL system by exploiting a large amount of unlabeled data (about 7 times more than labeled data). The algorithm variation was similar to that of He and Gildea (2006), but it only dealt with core arguments of the Propbank. They achieved a minor improvement too and credited it to the relatively poor performance of their base classifier and the insufficiency of the unlabeled data.

3 SRL System

To have enough control over entire the system and thus a flexible experimental framework, we developed our own SRL system instead of using a third-party system. The system works with PropBank-style annotation and is described here.

Syntactic Formalism: A Penn Treebank constituent-based approach for SRL is taken. Syntactic parse trees are produced by the reranking parser of Charniak and Johnson (2005).

Architecture: A two-stage pipeline architecture is used, where in the first stage less-probable argument candidates (samples) in the parse tree are pruned, and in the next stage, final arguments are identified and assigned a semantic role. However, for unlabeled data, a preprocessing stage identifies the verb predicates based on the POS tag assigned by the parser. The joint argument identification and classification is chosen to decrease the complexity of self-training process.

Features: Features are listed in table 1. We tried to avoid features like named entity tags to less depend on extra annotation. Features marked with * are used in addition to common features in the literature, due to their impact on the performance in feature selection process.

Classifier: We chose a *Maximum Entropy* classifier for its efficient training time and also its built-in multi-classification capability. Moreover, the probability score that it assigns to labels is useful in selection process in self-training. The *Maxent Toolkit*[1] was used for this purpose.

[1]http://homepages.inf.ed.ac.uk/lzhang10/maxent_toolkit.html

1- Add the seed example set ***L*** to currently empty training set ***T***.
2- Train the base classifier ***C*** with training set ***T.***
3- Iterate the following steps until the stop criterion ***S*** is met.
   a- **Select *p*** examples from ***U*** into pool ***P*.**
   b- Label pool ***P*** with classifier ***C***
   c- **Select *n*** labeled examples with the highest confidence score whose score meets a certain threshold ***t*** and **add** to training set ***T***.
   d- Retrain the classifier ***C*** with new training set.

Figure 1: Self-training Algorithm

	WSJ Test			Brown Test		
	P	R	F1	P	R	F1
Cur	77.43	68.15	**72.50**	69.14	57.01	**62.49**
Pun	82.28	76.78	**79.44**	73.38	62.93	**67.75**

Table 2: Performances of the current system (Cur) and the state-of-the-art (**Pun**yakanok et al., 2008)

4 Self-training

4.1 The Algorithm

While the general theme of the self-training algorithm is almost identical in different implementations, variations of it are developed based on the characteristics of the task in hand, mainly by customizing several involved parameters. Figure 1 shows the algorithm with highlighted parameters.

The size of seed labeled data set **L** and unlabeled data **U**, and their ratio are the fundamental parameters in any semi-supervised learning. The data used in this work is explained in section 5.1.

In addition to performance, efficiency of the classifier (**C**) is important for self-training, which is computationally expensive. Our classifier is a compromise between performance and efficiency. Table 2 shows its performance compared to the state-of-the-art (Punyakanok et al. 2008) when trained on the whole labeled training set.

Stop criterion (**S**) can be set to a predetermined number of iterations, finishing all of the unlabeled data, or convergence of the process in terms of improvement. We use the second option for all experiments here.

In each iteration, one can label entire the unlabeled data or only a portion of it. In the latter case, a number of unlaleled examples (**p**) are selected and loaded into a *pool* (**P**). The selection can be based on a specific strategy, known as *preselection* (Abney, 2008) or simply done according to the original order of the unlabeled data. We investigate preselection in this work.

After labeling the **p** unlabeled data, training set is augmented by adding the newly labeled data. Two main parameters are involved in this step: *selection* of labeled examples to be added to training set and *addition* of them to that set.

Selection is the crucial point of self-training, in which the propagation of labeling noise into upcoming iterations is the major concern. One can select all of labeled examples, but usually only a number of them (**n**), known as *growth size*, based on a quality measure is selected. This measure is often the confidence score assigned by the classifier. To prevent poor labelings diminishing the quality of training set, a threshold (**t**) is set on this confidence score. Selection is also influenced by other factors, one of which being the balance between selected labels, which is explored in this study and explained in detail in the section 4.3.

The selected labeled examples can be retained in unlabeled set to be labeled again in next iterations (*delibility*) or moved so that they are labeled only once (*indelibility*). We choose the second approach here.

4.2 Preselection

While using a pool can improve the efficiency of the self-training process, there can be two other motivations behind it, concerned with the performance of the process.

One idea is that when all data is labeled, since the growth size is often much smaller than the labeled size, a uniform set of examples preferred by the classifier is chosen in each iteration. This leads to a biased classifier like the one discussed in previous section. Limiting the labeling size to a pool and at the same time (pre)selecting divergence examples into it can remedy the problem.

The other motivation is originated from the fact that the base classifier is relatively weak due to small seed size, thus its predictions, as the measure of confidence in selection process, may not be reliable. Preselecting a set of unlabeled examples more probable to be correctly labeled by the classifier in initial steps seems to be a useful strategy against this fact.

We examine both ideas here, by a random preselection for the first case and a measure of simplicity for the second case. Random preselection is built into our system, since we use randomized

training data. As the measure of simplicity, we propose the number of samples extracted from each sentence; that is we sort unlabeled sentences in ascending order based on the number of samples and load the pool from the beginning.

4.3 Selection Balancing

Most of the previous self-training problems involve a binary classification. Semantic role labeling is a multi-class classification problem with an unbalanced distribution of classes in a given text. For example, the frequency of A1 as the most frequent role in CoNLL training set is 84,917, while the frequency of 21 roles is less than 20. The situation becomes worse when the dominant label NULL (for non-arguments) is added for argument identification purpose in a joint architecture. This biases the classifiers towards the frequent classes, and the impact is magnified as self-training proceeds.

In previous work, although they used a reduced set of roles (yet not balanced), He and Gildea (2006) and Lee et al. (2007), did not discriminate between roles when selecting high-confidence labeled samples. The former study reports that the majority of labels assigned to samples were NULL and argument labels appeared only in last iterations.

To attack this problem, we propose a natural way of balancing, in which instead of labeling and selection based on argument samples, we perform a sentence-based selection and labeling. The idea is that argument roles are distributed over the sentences. As the measure for selecting a labeled sentence, the average of the probabilities assigned by the classifier to all argument samples extracted from the sentence is used.

5 Experiments and Results

In these experiments, we target two main problems addressed by semi-supervised methods: the performance of the algorithm in exploiting unlabeled data when labeled data is scarce and the domain-generalizability of the algorithm by using an out-of-domain unlabeled data.

We use the CoNLL 2005 shared task data and setting for testing and evaluation purpose. The evaluation metrics include *precision*, *recall*, and their harmonic mean, *F1*.

5.1 The Data

The labeled data are selected from Propbank corpus prepared for CoNLL 2005 shared task. Our learning curve experiments on varying size of labeled data shows that the steepest increase in F1 is achieved by $1/10^{th}$ of CoNLL training data. Therefore, for training a base classifier as high-performance as possible, while simulating the labeled data scarcity with a reasonably small amount of it, 4000 sentence are selected randomly from the total 39,832 training sentences as seed data (L). These sentences contain 71,400 argument samples covering 38 semantic roles out of 52 roles present in the total training set.

We use one unlabeled training set (U) for in-domain and another for out-of-domain experiments. The former is the remaining portion of CoNLL training data and contains 35,832 sentences (698,567 samples). The out-of-domain set was extracted from Open American National Corpus [2] (OANC), a 14-million words multi-genre corpus of American English. The whole corpus was preprocessed to prune some problematic sentences. We also excluded the *biomed* section due to its large size to retain the domain balance of the data. Finally, 304,711 sentences with the length between 3 and 100 were parsed by the syntactic parser. Out of these, 35,832 sentences were randomly selected for the experiments reported here (832,795 samples).

Two points are worth noting about the results in advance. First, we do not exclude the argument roles not present in seed data when evaluating the results. Second, we observed that our predicate-identification method is not reliable, since it is solely based on POS tags assigned by parser which is error-prone. Experiments with gold predicates confirmed this conclusion.

5.2 The Effect of Balanced Selection

Figures 2 and 3 depict the results of using unbalanced and balanced selection with WSJ and OANC data respectively. To be comparable with previous work (He and Gildea, 2006), the growth size (n) for unbalanced method is 7000 samples and for balanced method is 350 sentences, since each sentence roughly contains 20 samples. A probability threshold (t) of 0.70 is used for both cases. The F1 of base classifier, best-performed classifier, and final classifier are marked.

When trained on WSJ unlabeled set, the balanced method outperforms the other in both WSJ (68.53 vs. 67.96) and Brown test sets (59.62 vs. 58.95). A two-tail t-test based on different random selection of training data confirms the statistical significance of this improvement at $p<=0.05$ level. Also, the self-training trend is

[2] http://www.americannationalcorpus.org/OANC

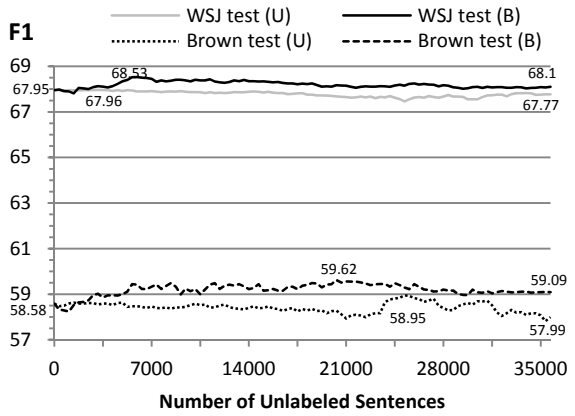

Figure 2: Balanced (B) and Unbalanced (U) Selection with WSJ Unlabeled Data

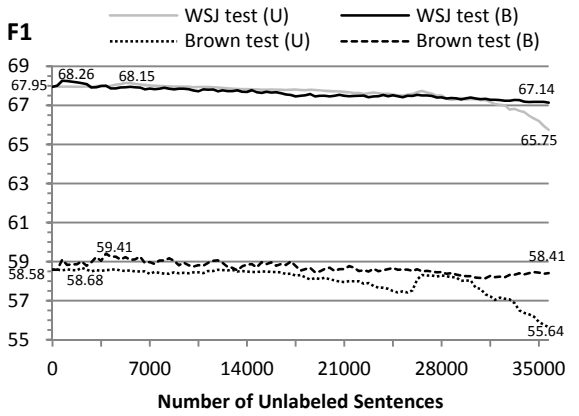

Figure 3: Balanced (B) and Unbalanced (U) Selection with OANC Unlabeled Data

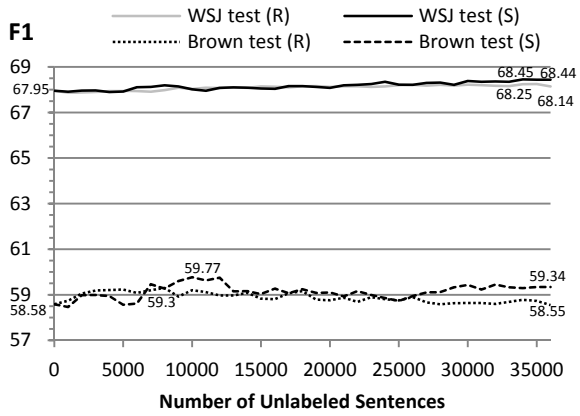

Figure 4: Random (R) and Simplicity (S) Pre-selection with WSJ Unlabeled Data

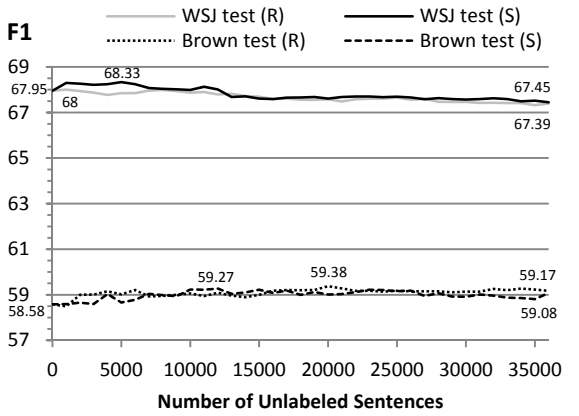

Figure 5: Random (R) and Simplicity (S) Pre-selection with OANC Unlabeled Data

more promising with both test sets. When trained on OANC, the F1 degrades with both methods as self-training progress. However, for both test sets, the best classifier is achieved by the balanced selection (68.26 vs. 68.15 and 59.41 vs. 58.68). Moreover, balanced selection shows a more normal behavior, while the other degrades the performance sharply in the last iterations (due to a swift drop of recall).

Consistent with previous work, with unbalanced selection, non-NULL-labeled unlabeled samples are selected only after the middle of the process. But, with the balanced method, selection is more evenly distributed over the roles.

A comparison between the results on Brown test set with each of unlabeled sets shows that in-domain data generalizes even better than out-of-domain data (59.62 vs. 59.41 and also note the trend). One apparent reason is that the classifier cannot accurately label the out-of-domain unlabeled data successively used for training. The lower quality of our out-of-domain data can be another reason for this behavior. Furthermore, the parser we used was trained on WSJ, so it negatively affected the OANC parses and consequently its SRL results.

5.3 The Effect of Preselection

Figures 4 and 5 show the results of using pool with random and simplicity-based preselection with WSJ and OANC data respectively. The pool size (p) is 2000, and growth size (n) is 1000 sentences. The probability threshold (t) used is 0.5.

Comparing these figures with the previous figures shows that preselection improves the self-training trend, so that more unlabeled data can still be useful. This observation was consistent with various random selection of training data.

Between the two strategies, simplicity-based method outperforms the random method in both self-training trend and best classifier F1 (68.45 vs. 68.25 and 59.77 vs. 59.3 with WSJ and 68.33 vs. 68 with OANC), though the t-test shows that the F1 difference is not significant at $p<=0.05$. This improvement does not apply to the case of using OANC data when tested with Brown data

(59.27 vs. 59.38), where, however, the difference is not statistically significant. The same conclusion to the section 5.2 can be made here.

6 Conclusion and Future Work

This work studies the application of self-training in learning semantic role labeling with the use of unlabeled data. We used a balancing method for selecting newly labeled examples for augmenting the training set in each iteration of the self-training process. The idea was to reduce the effect of unbalanced distribution of semantic roles in training data. We also used a pool and examined two preselection methods for loading unlabeled data into it.

These methods showed improvement in both classifier performance and self-training trend. However, using out-of-domain unlabeled data for increasing the domain generalization ability of the system was not more useful than using in-domain data. Among possible reasons are the low quality of the used data and the poor parses of the out-of-domain data.

Another major factor that may affect the self-training behavior here is the poor performance of the base classifier compared to the state-of-the-art (see Table 2), which exploits more complicated SRL architecture. Due to high computational cost of self-training approach, bootstrapping experiments with such complex SRL approaches are difficult and time-consuming.

Moreover, parameter tuning process shows that other parameters such as pool-size, growth number and probability threshold are very effective. Therefore, more comprehensive parameter tuning experiments than what was done here is required and may yield better results.

We are currently planning to port this setting to co-training, another bootstrapping algorithm. One direction for future work can be adapting the architecture of the SRL system to better match with the bootstrapping process. Another direction can be adapting bootstrapping parameters to fit the semantic role labeling complexity.

References

Abney, S. 2008. *Semisupervised Learning for Computational Linguistics.* Chapman and Hall, London.

Baker, F., Fillmore, C. and Lowe, J. 1998. The Berkeley FrameNet project. In *Proceedings of COLING-ACL*, pages 86-90.

Charniak, E. and Johnson, M. 2005. Coarse-to-fine n-best parsing and MaxEnt discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 173-180.

Carreras, X. and Marquez, L. 2005. Introduction to the CoNLL-2005 shared task: Semantic role labeling. In *Proceedings of the 9th Conference on Natural Language Learning (CoNLL)*, pages. 152-164.

Clark S., Curran, R. J. and Osborne M. 2003. Bootstrapping POS taggers using Unlabeled Data. In *Proceedings of the 7th Conference on Natural Language Learning At HLT-NAACL 2003*, pages 49-55.

Gildea, D. and Jurafsky, D. 2002. Automatic labeling of semantic roles. *CL*, 28(3):245-288.

He, S. and Gildea, H. 2006. Self-training and Co-training for Semantic Role Labeling: Primary Report. TR 891, University of Colorado at Boulder

Kingsbury, P. and Palmer, M. 2002. From Treebank to PropBank. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002)*.

Lee, J., Song, Y. and Rim, H. 2007. Investigation of Weakly Supervised Learning for Semantic Role Labeling. In *Proceedings of the Sixth international Conference on Advanced Language Processing and Web information Technology (ALPIT 2007)*, pages 165-170.

McClosky, D., Charniak, E., and Johnson, M. 2006. Effective self-training for parsing. In *Proceedings of the Main Conference on Human Language Technology Conference of the North American Chapter of the ACL*, pages 152-159.

Ng, V. and Cardie, C. 2003. Weakly supervised natural language learning without redundant views. In *Proceedings of the 2003 Conference of the North American Chapter of the ACL on Human Language Technology*, pages 94-101.

Punyakanok, V., Roth, D. and Yi, W. 2008. The Importance of Syntactic Parsing and Inference in Semantic Role Labeling. *CL*, 34(2):257-287.

Surdeanu, M., Harabagiu, S., Williams, J. and Aarseth, P. 2003. Using predicate argument structures for information extraction. In *Proceedings of the 41st Annual Meeting of the ACL*, pages 8-15.

Surdeanu, M., Johansson, R., Meyers, A., Marquez, L. and Nivre, J. 2008. The CoNLL 2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the 12th Conference on Natural Language Learning (CoNLL)*, pages 159-177.

Yarowsky, E. 1995. Unsupervised Word Sense Disambiguation Rivaling Supervised Methods. In *proceeding of the 33rd Annual Meeting of ACL*, pages 189-196.