Weakly Supervised Training For Parsing Mandarin Broadcast Transcripts

Wen Wang

Speech Technology and Research Laboratory, SRI International, USA

wwang@speech.sri.com

Abstract

We present a systematic investigation of applying weakly supervised co-training approaches to improve parsing performance for parsing Mandarin broadcast news (BN) and broadcast conversation (BC) transcripts, by iteratively retraining two competitive Chinese parsers from a small set of treebanked data and a large set of unlabeled data. We compare co-training to self-training, and our results show that performance using co-training is significantly better than with self-training and both co-training and self-training with a small seed labeled corpus can improve parsing accuracy significantly over training on the mismatching newswire treebank. We also investigate a variety of example selection approaches for co-training and find that our proposed example selection approach based on maximizing the training utility produces the best parsing accuracy. We also investigate Chinese parsing related issues including character-based parsing and word segmentation for parsing.

Index Terms: weakly-supervised training, co-training, parsing, Mandarin speech, word segmentation

1. Introduction

Parsing is an important research area in natural language processing (NLP), aiming at resolving structural ambiguity. In recent years, there has been much success on corpus-based statistical parsing with Charniak’s parser [1] and Collins’ parser [2] producing about 90% labeled bracketing accuracy on the English Wall Street Journal treebank corpus. Recently, with the great effort of creating a Chinese treebank from LDC and broader applications of Chinese parsing to various tasks, there has been more research on building high quality Chinese parsers, mostly on the traditional newswire text genre. However, under the Defense Advanced Research Projects Agency (DARPA) Global Autonomous Language Exploitation (GALE) program\(^1\), there are new genres besides newswire text, namely, broadcast news (BN), broadcast conversation (BC), newsgroup (NG), and web log (WB). Generating high quality parse trees for Chinese data in these genres can be useful for various tasks within GALE, including syntax-guided translation and reordering models for Chinese-to-English machine translation (MT), name entity detection, and structured language modeling for automatic speech recognition (ASR) on Mandarin BN and BC audio. There has been effort on applying weakly supervised techniques on parsing [3]. However, to our knowledge, there has been no systematic research on employing weakly-supervised learning approaches to improve parsing performance on Chinese speech genres. In this paper, we will explore weakly-supervised learning approaches on parsing Chinese BN and BC transcripts and examine some Chinese parsing related issues such as parsing unsegmented character sequences rather than words and the effect of word segmentation on parsing accuracy. In the rest of the paper, Section 2 describes the co-training and self-training algorithms, as well as example selection approaches used in co-training. Section 3 describes the available treebanks, the small seed annotated corpora, and the large unlabeled corpora used for various parser selection and co-training experiments. Section 4 illustrates the procedure of selecting parsers for our co-training experiments and briefly summarizes our investigations on character-based parsing and the impact of word segmentation on parsing performance. Experimental results, discussions, and conclusions appear in Section 5.

2. Co-training

2.1. General co-training algorithm

First introduced by Blum and Mitchell [4] as a weakly supervised learning method, co-training can be used for bootstrapping a model from a seed corpus of labeled examples, which is typically quite small, augmented with a much larger amount of unlabeled examples, by exploiting redundancy among multiple statistical models that generate different views of the data. Abney [5] proved that a weaker independence assumption on the multiple classifiers than Blum and Mitchell’s quite restrictive assumption could still allow co-training to work well. There has been much effort on investigating the efficacy of co-training in different domains and applications. The co-training algorithm developed by Pierce and Cardie [6] is presented in Algorithm 1.

Informally, co-training can be described as picking multiple classifiers (“views”) of a classification problem, build models for each view and train these models on a small set of labeled data, then on a large set of unlabeled data, sample a subset, label them using the models, select examples from the labeled results, add them to the training pool, and iterate this procedure until the unlabeled set is all labeled.

2.2. Example selection approaches for co-training

In Algorithm 1, when calling the classifier that provides additional training data for the opposite classifier the student, since the labeled output from both classifiers \(h_1\) and \(h_2\) is noisy, an important question is which newly labeled examples from the teacher should be added to the training data pool of the student. This issue of example selection plays an important role in the learning rate of co-training and the performance of resulting classifiers. In this paper, we investigate four example selection approaches. The first is naive co-training, which simply adds all examples in the cache labeled by the teacher to the training data pool of the student. The single parameter that needs to be optimized (on the development data set) for this example selection approach on the classification accuracy is the cache size, \(a\).

The second approach, agreement-based co-training [7], is to select the subset of the labeled cache that maximizes the...
Input: $S$ is a seed set of labeled data.
$L_0$, is labeled training data for $h_1$.
$L_{\text{max}}$ is labeled training data for $h_2$.
$U$ is the unlabeled data set.
$C$ is the cache holding a small subset of $U$.

1. $L_0 \leftarrow S$
2. $L_{\text{max}} \leftarrow S$
3. Train classifier $h_1$ on $L_{\text{max}}$
4. Train classifier $h_2$ on $L_{\text{max}}$

repeat
6. Randomly partition $U$ into $C$ where $|C| = n$ and $U/\prime$
7. Apply $h_1$, $h_2$ to assign labels for all examples in $C$
8. Select examples labeled by $h_1$ and add to $L_{\text{max}}$
9. Train $h_2$ on $L_{\text{max}}$
10. Select examples labeled by $h_2$ and add to $L_{\text{max}}$
11. Train $h_1$ on $L_{\text{max}}$
12. $U \leftarrow U/\prime$

until $U$ is empty

Algorithm 1: General co-training algorithm.

The effectiveness of co-training is largely based on the assumption that the two views are sufficient and conditionally independent given classes. Although Abney has proved that this assumption can be relaxed, our preliminary research shows that it is important to make sure that the two classifiers are as complementary as possible and generate significantly different error patterns. We used Chinese Treebank 5.2 released by LDC (denoted as CTB) for parser selection to fulfill this requirement. CTB is also part of the training data for co-training experiments. Chinese Treebank 5.2 contains 500K words, 800K characters, 18K sentences, and 900 data files, including articles from the Xinhua news agency (China Mainland), Information Services Department of HKSAR (Hong Kong), and Sinorama magazine (Taiwan). The format of CTB is similar to the English Penn Treebank and it is carefully annotated. Since the CTB corpus was collected covering a wide span of epoques from different sources with a diversity of articles, in order to obtain a representative split of training, development, and test sets for parsing, we divide the whole corpus into blocks of 10 files by sorted order. For each block, the first file is used for development, the second file for test, and the remaining 8 files for training. Finally, the training set consists of 14,925 sentences and 404,844 words, the dev set consists of 1,904 sentences and 51,243 words, and the test set consists of 1,975 sentences and 52,900 words.

Under the GALE program, the BN genre follows its tradition and consists of "talking head" style broadcasts, i.e., generally one person reading a news script. The BC genre, by contrast, is more conversational and spontaneous, consisting of talk shows, interviews, call-in programs and roundtables. The evaluation of co-training for parsing Mandarin BN and BC transcripts is conducted on the GALE OntoNotes released Mandarin BN and BC treebanks. The BN treebank is from the Mandarin TDT4 collection and the BC treebank is from GALE Mandarin BC data and translations from English BC data. The Mandarin BN treebank includes 300K words and 814 data files, and the BC treebank 100K words and 16 data files. To create a seed corpus and a test set for evaluating parsing accuracy, for BN and BC respectively, we divided the whole BN/BC treebank into blocks of 10 files by sorted order and within each block, the first file is used for co-training development, the second for testing parsing accuracy, and the rest 8 files are used as part of the seed annotated corpus for co-training. The resulting BN test set is denoted BN-test and the seed annotated corpus BN-seed, and the BC test set is denoted BC-test and the BC seed annotated corpus BC-seed. BN-test includes 31K words and 1,565 sentences, BC-test includes 11K words and 1,482 sentences. The large set of unlabeled data for BN parsing includes Hub4 1997 Mandarin BN acoustic transcripts, LDC Chinese TDT[2,3,4] corpora, Chinese Gigaword 3.0, and all GALE released BN audio transcripts, denoted BN-unlabeled. For BC parsing, we added all GALE released BC audio transcripts denoted BC-unlabeled.

4. Selecting Parsers for Co-training

We investigated four publicly available parsers, namely, Charniak’s maximum-entropy inspired parser with the MaxEnt reranker [1], the Stanford unlexicalized parser [10], Berkeley parser [11], and Dan Bikel’s reimplementation of Michael Collins’ Model 2 parser [12]. To select two from them in our co-training setup, we considered two important factors, accuracy and mutual complementariness. To evaluate parser accuracy, we consider the F-measure (the F1 measure to be specific) based on labeled Precision (LP) and labeled Recall (LR). LP agreement of the two classifiers on unlabeled data. The student classifier is the one being retrained and the teacher classifier is the one remaining static. During the agreement-based selection procedure, we repeatedly sample from all possible subsets of the cache, by first choosing the size of the subset and then randomly choosing examples from the labeled cache based on the size. In this algorithm, if $h_2$ is trained on the updated $L_{\text{max}}$, after adding output from $h_1$, then the most recent version of $h_1$ is used to measure agreement and vice versa. Hence, this approach aims to improve the performance of the two classifiers alternatively, instead of simultaneously. Note that the agreement rate on $U$, denoted $A$, is the number of times each token in the unlabeled set $U$ is assigned the same label by both classifiers $h_1$ and $h_2$.

Besides naïve co-training and the agreement-based example selection approach, we proposed two different methods in [8]. One method is to select the top $n$ examples with the highest scores (based on a scoring function) when labeled by the teacher to add to the training pool of the student. This approach has been employed in many co-training applications. We denote it max-score. The underlying intuition is to select examples that are reliably labeled by the teacher for the student. To combine accuracy and training utility, we defined another example selection criterion, which selects examples with scores within the $m$ percent of top high-scoring labeled examples by the teacher and within the $n$ percent of bottom low-scoring labeled examples by the student [8]. We denote it max-s-min-s. The intuition for this approach is that the newly labeled data should not only be reliably labeled by the teacher but also should be as useful and compensatory as possible for the student. During empirical evaluations of these example selection methods, control parameters, e.g., $n$ and $m$, in these approaches, are optimized on the development data set with respect to the performance of resulting classifiers after co-training. We also compare the performance of co-training to self-training. There are a variety of definitions of self-training in the literature and we adopted that of Nigran and Ghani [9]. Self-training in this work simply adds all examples in the labeled cache to the training pool in each iteration.
is the number of correct constituents divided by the number of constituents found by the parser and LR is the number of correct constituents divided by the number of constituents in the gold parse. F-measure is defined as $F_1 = \frac{2 \times PR \times LR}{PR + LR}$. Using the train/dev/test split described in Section 3, Table 1 shows the F-measure of all four parsers on the test set.

Table 1: F-measure of all four parsers on the CTB test set. The train/dev/test split was described in Section 3. Both F-measure from parsing the original word-segmented treebank and the converted character-based treebank, described in Section 4, are presented.

<table>
<thead>
<tr>
<th>Parser</th>
<th>F-measure, Word</th>
<th>F-measure, Character</th>
</tr>
</thead>
<tbody>
<tr>
<td>Charniak</td>
<td>83.9%</td>
<td>75.5%</td>
</tr>
<tr>
<td>Stanford</td>
<td>82.0%</td>
<td>–</td>
</tr>
<tr>
<td>Berkeley</td>
<td>83.5%</td>
<td>75.6%</td>
</tr>
<tr>
<td>Bikel</td>
<td>82.9%</td>
<td>–</td>
</tr>
</tbody>
</table>

The co-training principle requires the two views to be conditionally independent or weakly conditionally independent. This means that we need to select parsers that are complementary on their learning patterns and error types. To measure the structural complementariness between parsers, we adapted the measure of structural consistency between parsers and modified the objective function for maximizing the structural complementariness between parsers to be selecting parsers with the minimal structural consistency. Note that to measure the structural consistency between the bracketing parses from parsers and gold standard parses, Black, Garside, and Leech [13] defined the metric average crossing brackets (ACB), the mean number of times per sentence that a bracketed sequence from one parser overlaps with the gold standard from the treebank such that neither is properly contained in the other. Although ACB does not account for all types of conflicting constituency, it is a practical measure for the structural consistency between two sets of parse trees. By using the output from one parser as the gold set, we can calculate the pair-wise $ACB_{A,B}$ of parser $A$ on parser $B$. The $ACB$ values on the CTB test set among all six pairs from the four parsers are ordered as: {Charniak, Stanford}, 2.11; {Berkeley, Stanford}, 2.09; {Charniak, Bikel}, 2.05; {Berkeley, Bikel}, 2.01; {Charniak, Berkeley}, 1.99; {Bikel, Stanford}, 1.87. Since we need to achieve the best combination of maximizing parsers’ accuracy and their mutual complementariness, we selected Charniak’s parser and Berkeley parser for co-training.

Words in Chinese are not delimited by white spaces. Before conducting parsing on Chinese, word segmentation needs to be complete so that each word is isolated by the word boundary information. There has been significant research effort on Chinese word segmentation but finding which word segmentation schemes are most beneficial for parsing is still an open question. Hence, in this work, we investigated character-based parsing by transforming the original word-based treebank into a character-based treebank and then training and applying parsers as we did on the word level. For any pre-terminal $X$ in the original CTB, we add four tags: $X_t$, $X_{c1}$, $X_{c2}$, $X_c$, for each character in the word $w$ dominated by $X$. $X_t$ is the POS tag for the first character of a multi-character word $w$. $X_{c1}$ is the last character of it, $X_{c2}$ is any character between $X_1$ and $X_c$. To discriminate between multi-character words and single-character words, we also introduced $X_s$ as the POS tag for all single character words. Then the original pre-terminal $X$ is converted to a non-terminal and each character of the original word is assigned with its POS tag and inserted as a descendant of the node $X$ in the new tree. After this procedure, the original word-based CTB is converted to a character-based treebank and parsers trained on this treebank can be applied to parse unsegmented Chinese text. Before applying co-training, we examined this character-based parsing strategy on Charniak’s parser and Berkeley parser on the converted character-based CTB. Results shown in Table 1 demonstrated that parsing unsegmented text will lose about 8% absolutely on F-measure compared to parsing the original word-segmented treebank. Hence, in this work, we conducted all parser training and testing experiments based on the word level.

We also found that it is essential to ensure consistent word segmentations between the treebank used for training parsers and the word-segmented text data for parsing. In one experiment, we trained Charniak’s parser on the training set of CTB, resegmented the CTB test set using the LDC word segmenter without adapting the word segmenter on the CTB corpus, then employed the parser to parse this new CTB test set and scored the parsing accuracy. The resulting F-measure is 68.1%, a 15.8% absolute degradation over the 83.9% F-measure when training and testing the parser use the same CTB word segmentation. Since CTB is the major component for training parsers in this work, we picked up the Stanford Chinese word segmenter [14] trained on the CTB corpus and used it to segment the large unlabeled BN-unlabeled and BC-unlabeled data. After word segmentation, the BN-unlabeled data includes 760M words and 34M sentences; the BC-unlabeled data includes 11M words and 1M sentences.

5. Experimental Results

Table 2 shows the parsing accuracy F-measure (%) on BN-test under various parser training conditions. As can be seen from the table, training Charniak’s parser and Berkeley parser using only the small training set of BN treebank, i.e., BN-seed, resulted in relatively poor parsing performance, as 76.5% F1 for Charniak’s parser and 75.2% for Berkeley parser. Using the larger full CTB corpus for training improves parsing performance significantly and adding BN-seed to CTB brought additional gain. However, both self-training and co-training using CTB plus BN-seed as the initial training pool significantly improve the performance of the two parsers over directly training on CTB plus BN-seed, with co-training strongly outperforming self-training. Note for self-training and co-training carried out in these experiments, we used cache size as 10K sentences. Among the four example selection approaches, our proposed max-t-min-s approach yields the best accuracy from resulting parsers and it is much more computationally efficient compared to the agreement-based co-training method, which ranked the second on performance. Between agreement-based co-training and naive co-training, consistent with the findings from Clark et al. [7], agreement-based co-training is superior to naive co-training, since at each iteration this approach dynamically selects the examples that can improve the agreement rate and rejects ones that cannot fulfill the goal. In contrast, naive co-training adds all new examples in the cache which might accumulate noise during learning. On the other hand, the number of iterations of retraining that the agreement-based approach requires is generally an order of magnitude larger than that of naive co-training. Also, Table 2 demonstrates that max-t-min-s approach outperforms max-score. This shows that although max-t-min-s might let in many examples with errorful labels, the training utility of these examples for the student outweighs
the cost of errors introduced by these examples into the training data pool of the student. This observation of importance of training utility is consistent with the finding in active learning. Overall, by applying co-training, we improved F-measure on BN-test by 2.2% to 2.6% absolutely from the two parsers.

Table 2: Overall parsing accuracy F-measure (%) on the Mandarin BN treebank test set, BN-test, after applying self-training and co-training.

<table>
<thead>
<tr>
<th>Training Condition</th>
<th>F-measure (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Charniak</td>
</tr>
<tr>
<td>BN-seed</td>
<td>76.5</td>
</tr>
<tr>
<td>CTB</td>
<td>79.8</td>
</tr>
<tr>
<td>CTB+BN-seed</td>
<td>81.1</td>
</tr>
<tr>
<td>self-training</td>
<td>81.8</td>
</tr>
<tr>
<td>co-training</td>
<td></td>
</tr>
<tr>
<td>naive</td>
<td>82.0</td>
</tr>
<tr>
<td>agreement-based</td>
<td>83.0</td>
</tr>
<tr>
<td>max-score</td>
<td>82.7</td>
</tr>
<tr>
<td>max-t-min-s</td>
<td>83.3</td>
</tr>
</tbody>
</table>

Table 3 shows the F-measure from the two parsers on BC-test under various training conditions. The condition BN-co-trained denotes the final annotated BN data after applying max-t-min-s co-training as shown in Table 2, which includes the full BN treebank and the automatic annotations on the BN-unlabeled text generated through co-training. As can be seen, BN-co-trained outperforms CTB, indicating greater similarity between the two speech genres compared to CTB vs BC. Combining BN-co-trained and CTB achieved further gain on parsing performance. Consistent with Table 2, it is always quite helpful to add the small in-genre seed treebank into training, as CTB+BN-co-trained+BC-seed outperforms CTB+BN-co-trained. The comparisons between self-training and various co-training example selection approaches are also consistent with those observed in Table 2. Overall, we gained 2.4% to 2.5% absolutely on F-measure on BC-test over the two parsers. We also investigated the contribution from the small BN-seed and BC-seed corpora for co-training and observed that adding BN-seed and BC-seed to the initial training pool for co-training always outperforms using only the CTB corpus, by 1% on BN and 1.4% on BC. This preliminary study indicates the importance of creating a small in-genre/in-domain annotated seed corpus for co-training.

In conclusion, we have shown that co-training can be effectively applied to bootstrap parsers for parsing Mandarin BN and BC transcripts by combining labeled and unlabeled data. The computationally efficient example selection approach, which is based on maximizing training utility, produces co-trained parsers with the best parsing accuracy on BN and BC test sets. We also found that parsing unsegmented text is still quite inferior to parsing on the word level and it is essential to use a consistent word segmentation model for training the parsers and applying them for parsing text. The parsing performance on Mandarin BC transcripts is significantly lower than that on Chinese newswire and BN genres, suggesting that this difficult genre imposes serious challenges and requires much research effort.

6. Acknowledgements

The authors thank Mary Harper for discussions on Chinese word segmentation and parsing. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR0011-06-C-0023 (approved for public release, distribution unlimited). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of DARPA.

7. References