

Improving Chinese Semantic Role Labeling with Rich Syntactic Features

Weiwei Sun*

Department of Computational Linguistics, Saarland University
German Research Center for Artificial Intelligence (DFKI)
D-66123, Saarbrücken, Germany
wsun@coli.uni-saarland.de

Abstract

Developing features has been shown crucial to advancing the state-of-the-art in Semantic Role Labeling (SRL). To improve Chinese SRL, we propose a set of additional features, some of which are designed to better capture structural information. Our system achieves 93.49 F-measure, a significant improvement over the best reported performance 92.0. We are further concerned with the effect of parsing in Chinese SRL. We empirically analyze the two-fold effect, grouping words into constituents and providing syntactic information. We also give some preliminary linguistic explanations.

1 Introduction

Previous work on Chinese Semantic Role Labeling (SRL) mainly focused on how to implement SRL methods which are successful on English. Similar to English, parsing is a *standard* pre-processing for Chinese SRL. Many features are extracted to represent constituents in the input parses (Sun and Jurafsky, 2004; Xue, 2008; Ding and Chang, 2008). By using these features, semantic classifiers are trained to predict whether a constituent fills a semantic role. Developing features that capture the right kind of information encoded in the input parses has been shown crucial to advancing the state-of-the-art. Though there has been some work on feature design in Chinese SRL, information encoded in the syntactic trees is not fully exploited and requires more research effort. In this paper, we propose a set of additional

*The work was partially completed while this author was at Peking University.

features, some of which are designed to better capture structural information of sub-trees in a given parse. With help of these new features, our system achieves 93.49 F-measure with hand-crafted parses. Comparison with the best reported results, 92.0 (Xue, 2008), shows that these features yield a significant improvement of the state-of-the-art.

We further analyze the effect of syntactic parsing in Chinese SRL. The main effect of parsing in SRL is two-fold. First, grouping words into constituents, parsing helps to find argument candidates. Second, parsers provide semantic classifiers plenty of syntactic information, not to only recognize arguments from all candidate constituents but also to classify their detailed semantic types. We empirically analyze each effect in turn. We also give some preliminary linguistic explanations for the phenomena.

2 Chinese SRL

The Chinese PropBank (CPB) is a semantic annotation for the syntactic trees of the Chinese TreeBank (CTB). The arguments of a predicate are labeled with a contiguous sequence of integers, in the form of AN (N is a natural number); the adjuncts are annotated as such with the label AM followed by a secondary tag that represents the semantic classification of the adjunct. The assignment of semantic roles is illustrated in Figure 1, where the predicate is the verb “调查/investigate”. E.g., the NP “事故原因/the cause of the accident” is labeled as *A1*, meaning that it is the *Patient*.

In previous research, SRL methods that are successful on English are adopted to resolve Chinese SRL (Sun and Jurafsky, 2004; Xue, 2008; Ding and Chang, 2008, 2009; Sun et al., 2009; Sun, 2010). Xue (2008) produced complete and systematic research on full parsing based methods.

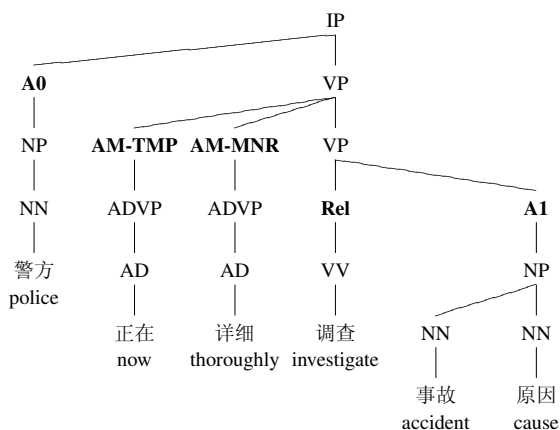


Figure 1: An example sentence: *The police are thoroughly investigating the cause of the accident.*

Their method divided SRL into three sub-tasks: 1) pruning with a heuristic rule, 2) Argument Identification (AI) to recognize arguments, and 3) Semantic Role Classification (SRC) to predict semantic types. The main two sub-tasks, AI and SRC, are formulated as two classification problems. Ding and Chang (2008) divided SRC into two sub-tasks in sequence: Each argument should first be determined whether it is a core argument or an adjunct, and then be classified into fine-grained categories. However, delicately designed features are more important and our experiments suggest that by using rich features, a better SRC solver can be directly trained without using hierarchical architecture. There are also some attempts at relaxing the necessity of using full syntactic parses, and semantic chunking methods have been introduced by (Sun et al., 2009; Sun, 2010; Ding and Chang, 2009).

2.1 Our System

We implement a three-stage (i.e. pruning, AI and SRC) SRL system. In the pruning step, our system keeps all constituents (except punctuations) that c-command¹ current predicate in focus as argument candidates. In the AI step, a lot of syntactic features are extracted to distinguish argument and non-argument. In other words, a binary classifier is trained to classify each argument candidate as either an argument or not. Finally, a multi-class classifier is trained to label each argument recognized in the former stage with a specific semantic role label. In both AI and SRC, the main job is to select strong syntactic features.

¹See (Sun et al., 2008) for detailed definition.

3 Features

A majority of features used in our system are a combination of features described in (Xue, 2008; Ding and Chang, 2008) as well as the word formation and coarse frame features introduced in (Sun et al., 2009), the c-command thread features proposed in (Sun et al., 2008). We give a brief description of features used in previous work, but explain new features in details. For more information, readers can refer to relevant papers and our source codes² that are well commented. To conveniently illustrate, we denote a candidate constituent c_k with a fixed context $w_{i-1}[c_k w_i \dots w_h \dots w_j]w_{j+1}$, where w_h is the head word of c_k , and denote predicate in focus with a context $w_{-2}^v w_{-1}^v w^v w_{+1}^v w_{+2}^v$, where w^v is the predicate in focus.

3.1 Baseline Features

The following features are introduced in previous Chinese SRL systems. We use them as baseline.

Word content of w^v , w_h , w_i , w_j and w_{i+j} ; *POS tag* of w^v , w_h ; *subcategorization frame*, *verb class* of w^v ; *position*, *phrase type* c_k , *path* from c_k to w^v (from (Xue, 2008; Ding and Chang, 2008))

First character, *last character* and *word length* of w^v , *first character+length*, *last character+word length*, *first character+position*, *last character+position*, *coarse frame*, *frame+ w^v* , *frame+left character*, *frame+verb class*, *frame+ c_k* (from (Sun et al., 2009)).

Head word POS, *head word* of PP phrases, *category* of c_k 's left and right siblings, *CFG rewrite rule* that expands c_k and c_k 's parent (from (Ding and Chang, 2008)).

3.2 New Word Features

We introduce some new features which can be extracted without syntactic structure. We denote them as word features. They include:

Word content of w_{-1}^v , w_{+1}^v , w_{i-1} and w_{j+1} ; *POS tag* of w_{-1}^v , w_{+1}^v , w_{-2}^v , w_{+2}^v , w_{i-1} , w_i , w_j , w_{j+1} , w_{i+2} and w_{j-2} .

Length of c_k : how many words are there in c^k .

Word before "LC": If the POS of w_j is "LC" (localizer), we use w_{j-1} and its POS tag as two new features.

NT: Does c_k contain a word with POS "NT" (temporal noun)?

²Available at <http://code.google.com/p/csrlr/>.

Combination features: w_i 's POS+ w_j 's POS, w_v +Position

3.3 New Syntactic Features

Taking complex syntax trees as inputs, the classifiers should characterize their structural properties. We put forward a number of new features to encode the structural information.

Category of c_k 's parent; head word and POS of head word of parent, left sibling and right sibling of c_k .

Lexicalized Rewrite rules: Conjunction of rewrite rule and head word of its corresponding RHS. These features of candidate (lrw-c) and its parent (lrw-p) are used. For example, this *lrw-c* feature of the NP “事故原因” in Figure 1 is $NP \rightarrow NN + NN(\text{原因})$.

Partial Path: Path from the c_k or w^v to the lowest common ancestor of c_k and w^v . One *path* feature, hence, is divided into *left path* and *right path*.

Clustered Path: We use the manually created clusters (see (Sun and Sui, 2009)) of categories of all nodes in the *path* (cpath) and *right path*.

C-commander thread between c_k and w^v (cct): (proposed by (Sun et al., 2008)). For example, this feature of the NP “警方” in Figure 1 is $NP + ADVP + ADVP + VV$.

Head Trace: The sequential container of the head down upon the phrase (from (Sun and Sui, 2009)). We design two kinds of traces (htr-p, htr-w): one uses POS of the head word; the other uses the head word word itself. E.g., the head word of 事故原因 is “原因” therefore these feature of this NP are $NP \downarrow NN$ and $NP \downarrow \text{原因}$.

Combination features: *verb class*+ c_k , w_h + w^v , w_h +Position, w_h + w^v +Position, *path*+ w^v , w_h +*right path*, w^v +*left path*, *frame*+ w^v + w_h , and w^v +cct.

4 Experiments and Analysis

4.1 Experimental Setting

To facilitate comparison with previous work, we use CPB 1.0 and CTB 5.0, the same data setting with (Xue, 2008). The data is divided into three parts: files from 081 to 899 are used as training set; files from 041 to 080 as development set; files from 001 to 040, and 900 to 931 as test set. Nearly all previous research on constituency based SRL evaluation use this setting, also including (Ding and Chang, 2008, 2009; Sun

et al., 2009; Sun, 2010). All parsing and SRL experiments use this data setting. To resolve classification problems, we use a linear SVM classifier SVM_{lin}^3 , along with One-Vs-All approach for multi-class classification. To evaluate SRL with automatic parsing, we use a state-of-the-art parser, Bikel parser⁴ (Bikel, 2004). We use gold segmentation and POS as input to the Bikel parser and use its parsing results as input to our SRL system. The overall LP/LR/F performance of Bikel parser is 79.98%/82.95%/81.43.

4.2 Overall Performance

Table 1 summarizes precision, recall and F-measure of AI, SRC and the whole task (AI+SRC) of our system respectively. The fourth line is the best published SRC performance reported in (Ding and Chang, 2008), and the sixth line is the best SRL performance reported in (Xue, 2008). Other lines show the performance of our system. These results indicate a significant improvement over previous systems due to the new features.

Test	P(%)	R(%)	F/A
AI	98.56	97.91	98.24
SRC	--	--	95.04
(Ding and Chang, 2008)	--	--	94.68
AI + SRC	93.80	93.18	93.49
(Xue, 2008)	93.0	91.0	92.0

Table 1: SRL performance on the test data with gold standard parses.

4.3 Two-fold Effect of Parsing in SRL

The effect of parsing in SRL is two-fold. On the one hand, SRL systems should group words as argument candidates, which are also constituents in a given sentence. Full parsing provides boundary information of all constituents. As arguments should c-command the predicate, a full parser can further prune a majority of useless constituents. In other words, parsing can effectively supply SRL with argument candidates. Unfortunately, it is very hard to rightly produce full parses for Chinese text. On the other hand, given a constituent, SRL systems should identify whether it is an argument and further predict detailed semantic types if

³<http://people.cs.uchicago.edu/~vikass/svmlin.html>

⁴<http://www.cis.upenn.edu/~dbikel/software.html>

Task	Parser	Bracket	Feat	P(%)	R(%)	F/A
AI	--	Gold	W	82.44	86.78	84.55
	CTB	Gold	W+S	98.69	98.11	98.40
	Bikel	Bikel	W+S	77.54	71.62	74.46
SRC	--	Gold	W	--	--	93.93
	CTB	Gold	W+S	--	--	95.80
	Bikel	Gold	W+S	--	--	92.62

Table 2: Classification performance on development data. In the *Feat* column, *W* means word features; *W+S* means word and syntactic features.

it is an argument. For the two classification problems, parsing can provide complex syntactic information such as *path* features.

4.3.1 The Effect of Parsing in AI

In AI, full parsing is very important for both grouping words and classification. Table 2 summarizes relative experimental results. Line 2 is the AI performance when gold candidate boundaries and word features are used; Line 3 is the performance with additional syntactic features. Line 4 shows the performance by using automatic parses generated by Bikel parser. We can see that: 1) word features only cannot train good classifiers to identify arguments; 2) it is very easy to recognize arguments with good enough syntactic parses; 3) there is a severe performance decline when automatic parses are used. The third observation is a similar conclusion in English SRL. However this problem in Chinese is much more serious due to the state-of-the-art of Chinese parsing.

Information theoretic criteria are popular criteria in variable selection (Guyon and Elisseeff, 2003). This paper uses empirical mutual information between each variable and the target, $I(X, Y) = \sum_{x \in X, y \in Y} p(x, y) \log \frac{p(x, y)}{p(x)p(y)}$, to roughly rank the importance of features. Table 3 shows the ten most useful features in AI. We can see that the most important features all based on full parsing information. Nine of these top 10 useful features are our new features.

Rank	Feature	Rank	Feature
1	w^v_cct	2	$\ddagger w_h + w^v + \text{Position}$
3	htr-w	4	htr-p
5	path	6	$\ddagger w_h + w^v$
7	cpath	8	cct
9	path+ w^v	10	lrw-p

Table 3: Top 10 useful features for AI. \ddagger means *word features*.

4.3.2 The Effect of Parsing in SRC

The second block in Table 2 summarizes the SRC performance with gold argument boundaries. Line 5 is the accuracy when word features are used; Line 6 is the accuracy when additional syntactic features are added; The last row is the accuracy when syntactic features used are extracted from automatic parses (*Bikel+Gold*). We can see that different from AI, word features only can train reasonable good semantic classifiers. The comparison between Line 5 and 7 suggests that with parsing errors, automatic parsed syntactic features cause noise to the semantic role classifiers.

4.4 Why Word Features Are Effective for SRC?

Rank	Feature	Rank	Feature
1	$\ddagger \text{frame} + w_h + w^v$	2	$\ddagger w_h + w^v + \text{position}$
3	$\ddagger w_h + w^v$	4	$w^v + \text{cct}$
5	lrw-p	6	$\ddagger w_i + w_j$
7	lrw-c	8	$\ddagger w_h + \text{Position}$
9	$\ddagger \text{frame} + w^v$	10	htr-p

Table 4: Top 10 useful features for SRC.

Table 4 shows the ten most useful features in SRC. We can see that two of these ten features are word features (denoted by \ddagger). Namely, word features play a more important role in SRC than in AI. Though the other eight features are based on full parsing, four of them (denoted by \ddagger) use the head word which can be well approximated by word features, according to some language specific properties. The head rules described in (Sun and Jurafsky, 2004) are very popular in Chinese parsing research, such as in (Duan et al., 2007; Zhang and Clark, 2008). From these head rules, we can see that head words of most phrases in Chinese are located at the first or the last position. We implement these rules on Chinese Tree Bank and find that 84.12%⁵ nodes realize their heads as either their first or last word. Head position suggests that boundary words are good approximation of head word features. If head words have good approximation word features, then it is not strange that the four features denoted by \ddagger can be effectively represented by word features. Similar with feature effect in AI, most of most useful features in SRC are our new features.

⁵This statistics excludes all empty categories in CTB.

5 Conclusion

This paper proposes an additional set of features to improve Chinese SRL. These new features yield a significant improvement over the best published performance. We further analyze the effect of parsing in Chinese SRL, and linguistically explain some phenomena. We found that (1) full syntactic information plays an essential role only in AI and that (2) due to the head word position distribution, SRC is easy to resolve in Chinese SRL.

Acknowledgments

The author is funded both by German Academic Exchange Service (DAAD) and German Research Center for Artificial Intelligence (DFKI).

The author would like to thank the anonymous reviewers for their helpful comments.

References

- Daniel M. Bikel. 2004. A distributional analysis of a lexicalized statistical parsing model. In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 182–189. Association for Computational Linguistics, Barcelona, Spain.
- Weiwei Ding and Baobao Chang. 2008. Improving Chinese semantic role classification with hierarchical feature selection strategy. In *Proceedings of the EMNLP 2008*, pages 324–333. Association for Computational Linguistics, Honolulu, Hawaii.
- Weiwei Ding and Baobao Chang. 2009. Fast semantic role labeling for Chinese based on semantic chunking. In *ICCPOL '09: Proceedings of the 22nd International Conference on Computer Processing of Oriental Languages. Language Technology for the Knowledge-based Economy*, pages 79–90. Springer-Verlag, Berlin, Heidelberg.
- Xiangyu Duan, Jun Zhao, and Bo Xu. 2007. Probabilistic models for action-based Chinese dependency parsing. In *ECML '07: Proceedings of the 18th European conference on Machine Learning*, pages 559–566. Springer-Verlag, Berlin, Heidelberg.
- Isabelle Guyon and André Elisseeff. 2003. An introduction to variable and feature selection. *Journal of Machine Learning Research*, 3:1157–1182.
- Honglin Sun and Daniel Jurafsky. 2004. Shallow semantic parsing of Chinese. In Daniel Marcu, Susan Dumais and Salim Roukos, editors, *HLT-NAACL 2004: Main Proceedings*.
- Weiwei Sun. 2010. Semantics-driven shallow parsing for Chinese semantic role labeling. In *Proceedings of the ACL 2010*.
- Weiwei Sun and Zhifang Sui. 2009. Chinese function tag labeling. In *Proceedings of the 23rd Pacific Asia Conference on Language, Information and Computation*. Hong Kong.
- Weiwei Sun, Zhifang Sui, and Haifeng Wang. 2008. Prediction of maximal projection for semantic role labeling. In *Proceedings of the 22nd International Conference on Computational Linguistics*.
- Weiwei Sun, Zhifang Sui, Meng Wang, and Xin Wang. 2009. Chinese semantic role labeling with shallow parsing. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 1475–1483. Association for Computational Linguistics, Singapore.
- Nianwen Xue. 2008. Labeling Chinese predicates with semantic roles. *Comput. Linguist.*, 34(2):225–255.
- Yue Zhang and Stephen Clark. 2008. A tale of two parsers: Investigating and combining graph-based and transition-based dependency parsing. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 562–571. Association for Computational Linguistics, Honolulu, Hawaii.