

# Joint Syntactic and Semantic Parsing of Chinese

**Junhui Li and Guodong Zhou**

School of Computer Science & Technology  
Soochow University  
Suzhou, China 215006

{lijunhui, gdzhou}@suda.edu.cn

**Hwee Tou Ng**

Department of Computer Science  
National University of Singapore  
13 Computing Drive, Singapore 117417

nght@comp.nus.edu.sg

## Abstract

This paper explores joint syntactic and semantic parsing of Chinese to further improve the performance of both syntactic and semantic parsing, in particular the performance of semantic parsing (in this paper, semantic role labeling). This is done from two levels. Firstly, an integrated parsing approach is proposed to integrate semantic parsing into the syntactic parsing process. Secondly, semantic information generated by semantic parsing is incorporated into the syntactic parsing model to better capture semantic information in syntactic parsing. Evaluation on Chinese TreeBank, Chinese PropBank, and Chinese NomBank shows that our integrated parsing approach outperforms the pipeline parsing approach on n-best parse trees, a natural extension of the widely used pipeline parsing approach on the top-best parse tree. Moreover, it shows that incorporating semantic role-related information into the syntactic parsing model significantly improves the performance of both syntactic parsing and semantic parsing. To our best knowledge, this is the first research on exploring syntactic parsing and semantic role labeling for both verbal and nominal predicates in an integrated way.

## 1 Introduction

Semantic parsing maps a natural language sentence into a formal representation of its meaning. Due to the difficulty in deep semantic parsing, most previous work focuses on shallow semantic parsing, which assigns a simple structure (such as WHO did WHAT to WHOM, WHEN, WHERE, WHY, HOW) to each predicate in a sentence. In particular, the well-defined semantic role labeling (SRL) task has been drawing increasing attention in recent years due to its importance in natural language processing (NLP) applications, such as question answering (Narayanan and Harabagiu, 2004), information extraction (Surdeanu et al., 2003), and co-reference resolution (Kong et al., 2009). Given a sentence

and a predicate (either a verb or a noun) in the sentence, SRL recognizes and maps all the constituents in the sentence into their corresponding semantic arguments (roles) of the predicate. In both English and Chinese PropBank (Palmer et al., 2005; Xue and Palmer, 2003), and English and Chinese NomBank (Meyers et al., 2004; Xue, 2006), these semantic arguments include core arguments (e.g., Arg0 for agent and Arg1 for recipient) and adjunct arguments (e.g., ArgM-LOC for locative argument and ArgM-TMP for temporal argument). According to predicate type, SRL can be divided into SRL for verbal predicates (verbal SRL, in short) and SRL for nominal predicates (nominal SRL, in short).

With the availability of large annotated corpora such as FrameNet (Baker et al., 1998), PropBank, and NomBank in English, data-driven techniques, including both feature-based and kernel-based methods, have been extensively studied for SRL (Carreras and Màrquez, 2004; Carreras and Màrquez, 2005; Pradhan et al., 2005; Liu and Ng, 2007). Nevertheless, for both verbal and nominal SRL, state-of-the-art systems depend heavily on the top-best parse tree and there exists a large performance gap between SRL based on the gold parse tree and the top-best parse tree. For example, Pradhan et al. (2005) suffered a performance drop of 7.3 in F1-measure on English PropBank when using the top-best parse tree returned from Charniak's parser (Charniak, 2001). Liu and Ng (2007) reported a performance drop of 4.21 in F1-measure on English NomBank.

Compared with English SRL, Chinese SRL suffers more seriously from syntactic parsing. Xue (2008) evaluated on Chinese PropBank and showed that the performance of Chinese verbal SRL drops by about 25 in F1-measure when replacing gold parse trees with automatic ones. Likewise, Xue (2008) and Li et al. (2009) reported a performance drop of about 12 in F1-measure in Chinese NomBank SRL.

While it may be difficult to further improve syntactic parsing, a promising alternative is to perform both syntactic and semantic parsing in an integrated way. Given the close interaction between the two tasks, joint learning not only allows uncertainty about syntactic parsing to be carried forward to semantic parsing but also allows useful information from semantic parsing to be carried backward to syntactic parsing.

This paper explores joint learning of syntactic and semantic parsing for Chinese texts from two levels. Firstly, an integrated parsing approach is proposed to benefit from the close interaction between syntactic and semantic parsing. This is done by integrating semantic parsing into the syntactic parsing process. Secondly, various semantic role-related features are directly incorporated into the syntactic parsing model to better capture semantic role-related information in syntactic parsing. Evaluation on Chinese TreeBank, Chinese PropBank, and Chinese NomBank shows that our method significantly improves the performance of both syntactic and semantic parsing. This is promising and encouraging. To our best knowledge, this is the first research on exploring syntactic parsing and SRL for verbal and nominal predicates in an integrated way.

The rest of this paper is organized as follows. Section 2 reviews related work. Section 3 presents our baseline systems for syntactic and semantic parsing. Section 4 presents our proposed method of joint syntactic and semantic parsing for Chinese texts. Section 5 presents the experimental results. Finally, Section 6 concludes the paper.

## 2 Related Work

Compared to the large body of work on either syntactic parsing (Ratnaparkhi, 1999; Collins, 1999; Charniak, 2001; Petrov and Klein, 2007), or SRL (Carreras and Màrquez, 2004; Carreras and Màrquez, 2005; Jiang and Ng, 2006), there is relatively less work on their joint learning.

Koomen et al. (2005) adopted the outputs of multiple SRL systems (each on a single parse tree) and combined them into a coherent predicate argument output by solving an optimization problem. Sutton and McCallum (2005) adopted a probabilistic SRL system to re-rank the N-best results of a probabilistic syntactic parser. However, they reported negative results, which they blamed on the inaccurate probability estimates from their locally trained SRL model.

As an alternative to the above pseudo-joint learning methods (strictly speaking, they are still pipeline methods), one can augment the syntactic label of a constituent with semantic information, like what function parsing does (Merlo and Musillo, 2005). Yi and Palmer (2005) observed that the distributions of semantic labels could potentially interact with the distributions of syntactic labels and redefined the boundaries of constituents. Based on this observation, they incorporated semantic role information into syntactic parse trees by extending syntactic constituent labels with their coarse-grained semantic roles (core argument or adjunct argument) in the sentence, and thus unified semantic parsing and syntactic parsing. The actual fine-grained semantic roles are assigned, as in other methods, by an ensemble classifier. However, the results obtained with this method were negative, and they concluded that semantic parsing on PropBank was too difficult due to the differences between chunk annotation and tree structure. Motivated by Yi and Palmer (2005), Merlo and Musillo (2008) first extended a statistical parser to produce a richly annotated tree that identifies and labels nodes with semantic role labels as well as syntactic labels. Then, they explored both rule-based and machine learning techniques to extract predicate-argument structures from this enriched output. Their experiments showed that their method was biased against these roles in general, thus lowering recall for them (e.g., precision of 87.6 and recall of 65.8).

There have been other efforts in NLP on joint learning with various degrees of success. In particular, the recent shared tasks of CoNLL 2008 and 2009 (Surdeanu et al., 2008; Hajic et al., 2009) tackled joint parsing of syntactic and semantic dependencies. However, all the top 5 reported systems decoupled the tasks, rather than building joint models. Compared with the disappointing results of joint learning on syntactic and semantic parsing, Miller et al. (2000) and Finkel and Manning (2009) showed the effectiveness of joint learning on syntactic parsing and some simple NLP tasks, such as information extraction and name entity recognition. In addition, attempts on joint Chinese word segmentation and part-of-speech (POS) tagging (Ng and Low, 2004; Zhang and Clark, 2008) also illustrate the benefits of joint learning.

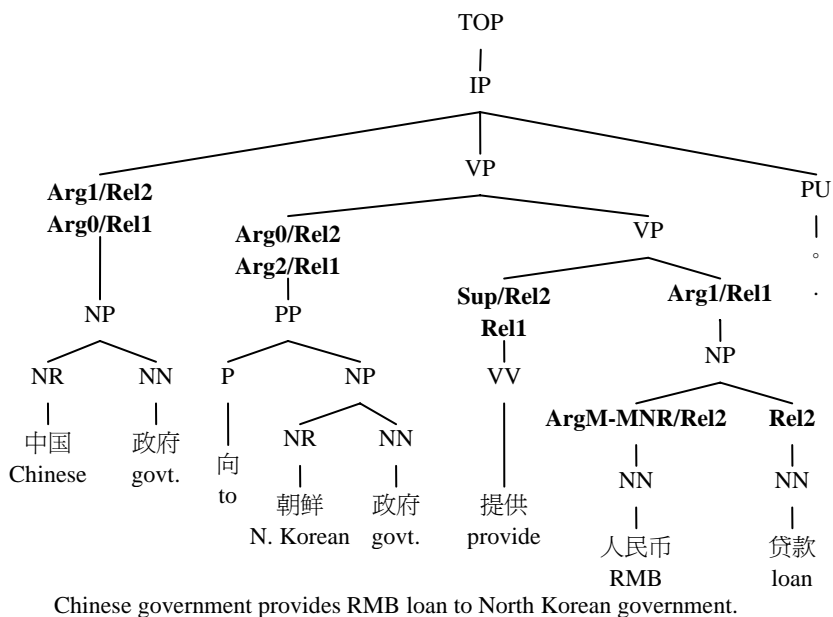


Figure 1: Two predicates (Rel1 and Rel2) and their arguments in the style of Chinese PropBank and NomBank.

### 3 Baseline: Pipeline Parsing on Top-Best Parse Tree

In this section, we briefly describe our approach to syntactic parsing and semantic role labeling, as well as the baseline system with pipeline parsing on the top-best parse tree.

#### 3.1 Syntactic Parsing

Our syntactic parser re-implements Ratnaparkhi (1999), which adopts the maximum entropy principle. The parser recasts a syntactic parse tree as a sequence of decisions similar to those of a standard shift-reduce parser and the parsing process is organized into three left-to-right passes via four procedures, called TAG, CHUNK, BUILD, and CHECK.

**First pass.** The first pass takes a tokenized sentence as input, and uses TAG to assign each word a part-of-speech.

**Second pass.** The second pass takes the output of the first pass as input, and uses CHUNK to recognize basic chunks in the sentence.

**Third pass.** The third pass takes the output of the second pass as input, and always alternates between BUILD and CHECK in structural parsing in a recursive manner. Here, BUILD decides whether a subtree will start a new constituent or join the incomplete constituent immediately to its left. CHECK finds the most recently proposed constituent, and decides if it is complete.

#### 3.2 Semantic Role Labeling

Figure 1 demonstrates an annotation example of Chinese PropBank and NomBank. In the figure, the verbal predicate “提供/provide” is annotated with three core arguments (i.e., “NP (中国/Chinese 政府/govt.)” as Arg0, “PP (向/to 朝鲜/N. Korean 政府/govt.)” as Arg2, and “NP (人民币/RMB 贷款/loan)” as Arg1), while the nominal predicate “贷款/loan” is annotated with two core arguments (i.e., “NP (中国/Chinese 政府/govt.)” as Arg1 and “PP (向/to 朝鲜/N. Korean 政府/govt.)” as Arg0), and an adjunct argument (i.e., “NN (人民币/RMB)” as ArgM-MNR, denoting the manner of loan). It is worth pointing out that there is a (Chinese) NomBank-specific label in Figure 1, Sup (support verb) (Xue, 2006), to help introduce the arguments which occur outside the nominal predicate-headed noun phrase. In (Chinese) NomBank, a verb is considered to be a support verb only if it shares at least an argument with the nominal predicate.

##### 3.2.1 Automatic Predicate Recognition

Automatic predicate recognition is a prerequisite for the application of SRL systems. For verbal predicates, it is very easy. For example, 99% of verbs are annotated as predicates in Chinese PropBank. Therefore, we can simply select any word with a part-of-speech (POS) tag of VV, VA, VC, or VE as verbal predicate.

Unlike verbal predicate recognition, nominal predicate recognition is quite complicated. For

example, only 17.5% of nouns are annotated as predicates in Chinese NomBank. It is quite common that a noun is annotated as a predicate in some cases but not in others. Therefore, automatic predicate recognition is vital to nominal SRL. In principle, automatic predicate recognition can be cast as a binary classification (e.g., Predicate vs. Non-Predicate) problem. For nominal predicates, a binary classifier is trained to predict whether a noun is a nominal predicate or not. In particular, any word POS-tagged as NN is considered as a predicate candidate in both training and testing processes. Let the nominal predicate candidate be  $w_0$ , and its left and right neighboring words/POSS be  $w_{-1}/p_{-1}$  and  $w_1/p_1$ , respectively. Table 1 lists the feature set used in our model. In Table 1, local features present the candidate’s contextual information while global features show its statistical information in the whole training set.

Type	Description
local features	$w_0, w_{-1}, w_1, p_{-1}, p_1$ The first and last characters of the candidate
global features	Whether $w_0$ is ever tagged as a verb in the training data? Yes/No Whether $w_0$ is ever annotated as a nominal predicate in the training data? Yes/No The most likely label for $w_0$ when it occurs together with $w_{-1}$ and $w_1$ . The most likely label for $w_0$ when it occurs together with $w_{-1}$ . The most likely label for $w_0$ when it occurs together with $w_1$ .

Table 1: Feature set for nominal predicate recognition

### 3.2.2 SRL for Chinese Predicates

Our Chinese SRL models for both verbal and nominal predicates adopt the widely-used SRL framework, which divides the task into three sequential sub-tasks: argument pruning, argument identification, and argument classification. In particular, we follow Xue (2008) and Li et al. (2009) to develop verbal and nominal SRL models, respectively. Moreover, we have further improved the performance of Chinese verbal SRL by exploring additional features, e.g., voice position that indicates the voice maker (BA, BEI) is before or after the constituent in focus, the rule that expands the parent of the constituent in focus, and the core arguments defined in the predicate’s frame file. For nominal SRL, we simply use the final feature set of Li et al. (2009). As a result, our Chinese verbal and nominal SRL systems achieve performance of 92.38 and 72.67

in F1-measure respectively (on golden parse trees and golden predicates), which are comparable to Xue (2008) and Li et al. (2009). For more details, please refer to Xue (2008) and Li et al. (2009).

### 3.3 Pipeline Parsing on Top-best Parse Tree

Similar to most of the state-of-the-art systems (Pradhan et al., 2005; Xue, 2008; Li et al., 2009), the top-best parse tree is first returned from our syntactic parser and then fed into the SRL system. Specifically, the verbal (nominal) SRL labeler is in charge of verbal (nominal) predicates, respectively. For each sentence, since SRL is only performed on one parse tree, only constituents in it are candidates for semantic arguments. Therefore, if no constituent in the parse tree can map the same text span to an argument in the manual annotation, the system will not get a correct annotation.

## 4 Joint Syntactic and Semantic Parsing

In this section, we first explore pipeline parsing on N-best parse trees, as a natural extension of pipeline parsing on the top-best parse tree. Then, joint syntactic and semantic parsing is explored for Chinese texts from two levels. Firstly, an integrated parsing approach to joint syntactic and semantic parsing is proposed. Secondly, various semantic role-related features are directly incorporated into the syntactic parsing model for better interaction between the two tasks.

### 4.1 Pipeline Parsing on N-best Parse Trees

The pipeline parsing approach employed in this paper is largely motivated by the general framework of re-ranking, as proposed in Sutton and McCallum (2005). The idea behind this approach is that it allows uncertainty about syntactic parsing to be carried forward through an N-best list, and that a reliable SRL system, to a certain extent, can reflect qualities of syntactic parse trees. Given a sentence  $x$ , a joint parsing model is defined over a semantic frame  $F$  and a parse tree  $t$  in a log-linear way:

$$\begin{aligned} \text{Score}(F, t | x) \\ = (1 - \alpha) \log P(F | t, x) + \alpha \log P(t | x) \end{aligned} \quad (1)$$

where  $P(t|x)$  is returned by a probabilistic syntactic parsing model, e.g., our syntactic parser, and  $P(F/t, x)$  is returned by a probabilistic semantic parsing model, e.g. our verbal & nominal

---

**Algorithm 1.** The algorithm integrating syntactic parsing and SRL.

---

Assume:

*t*: constituent which is complete with “YES” decision of CHECK procedure*P*: number of predicates*P<sub>i</sub>*: *i*<sup>th</sup> predicate*S*: SRL result, set of predicates and its arguments

---

```
BEGIN
  srl_prob = 0.0;
  FOR i=1 to P DO
    IF t covers Pi THEN
      T = number of children of t;
      FOR j=1 to T DO
        IF t's jth child Chj does not cover Pi THEN
          Run SRL given predicate Pi and constituent Chj to get their semantic role
          lbl and its probability prob;
          IF lbl does not indicate non-argument THEN
            srl_prob += log(prob);
            S = S ∪ {(Pi, Chj, lbl)};
          END IF
        END IF
      END FOR
    END IF
  END FOR
  return srl_prob;
END
```

---

SRL systems. In our pipeline parsing approach,  $P(t/x)$  is calculated as the product of all involved decisions' probabilities in the syntactic parsing model, and  $P(F/t, x)$  is calculated as the product of all the semantic role labels' probabilities in a sentence (including both verbal and nominal SRL). That is to say, we only consider those constituents that are supposed to be arguments. Here, the parameter  $\alpha$  is a balance factor indicating the importance of the semantic parsing model.

In particular,  $(F^*, t^*)$  with maximal  $Score(F, t/x)$  is selected as the final syntactic and semantic parsing results. Given a sentence, N-best parse trees are generated first using the syntactic parser, and then for each parse tree, we predict the best SRL frame using our verbal and nominal SRL systems.

## 4.2 Integrated Parsing

Although pipeline parsing on N-best parse trees could relieve severe dependence on the quality of the top-best parse tree, there is still a potential drawback: this method suffers from the limited scope covered by the N-best parse trees since the items in the parse tree list may be too similar, especially for long sentences. For example, 50-best parse trees can only represent a combination of 5 to 6 binary ambiguities since  $2^5 < 50 < 2^6$ .

Ideally, we should perform SRL on as many parse trees as possible, so as to enlarge the search scope. However, pipeline parsing on all possible parse trees is time-consuming and thus unrealistic. As an alternative, we turn to integrated parsing, which aims to perform syntactic and semantic parsing synchronously. The key idea is to construct a parse tree in a bottom-up way so that it is feasible to perform SRL at suitable moments, instead of only when the whole parse tree is built. Integrated parsing is practicable, mostly due to the following two observations: (1) Given a predicate in a parse tree, its semantic arguments are usually siblings of the predicate, or siblings of its ancestor. Actually, this special observation has been widely employed in SRL to prune non-arguments for a verbal or nominal predicate (Xue, 2008; Li et al., 2009). (2) SRL feature spaces (both in feature-based method and kernel-based method) mostly focus on the predicate-argument structure of a given (predicate, argument) pair. That is to say, once a predicate-argument structure is formed (i.e., an argument candidate is connected with the given predicate), there is enough contextual information to predict their SRL relation.

As far as our syntactic parser is concerned, we invoke the SRL systems once a new constituent covering a predicate is complete with a “YES” decision from the CHECK procedure. Algorithm

1 illustrates the integration of syntactic and semantic parsing. For the example shown in Figure 2, the CHECK procedure predicts a “YES” decision, indicating the immediately proposed constituent “VP (提供/provide 人民币/RMB 贷款/loan)” is complete. So, at this moment, the verbal SRL system is invoked to predict the semantic label of the constituent “NP (人民币/RMB 贷款/loan)”, given the verbal predicate “VV (提供/provide)”. Similarly, “PP (向/to 朝鲜/N. Korean 政府/govt.)” would also be semantically labeled as soon as “PP (向/to 朝鲜/N. Korean 政府/govt.)” and “VP (提供/provide 人民币/RMB 贷款/loan)” are merged into a bigger VP. In this way, both syntactic and semantic parsing are accomplished when the root node TOP is formed. It is worth pointing out that all features (Xue, 2008; Li et al., 2009) used in our SRL model can be instantiated and their values are same as the ones when the whole tree is available. In particular, the probability computed from the SRL model is interpolated with that of the syntactic parsing model in a log-linear way (with equal weights in our experiments). This is due to our hypothesis that the probability returned from SRL model is helpful to joint syntactic and semantic parsing, considering the close interaction between the two tasks.

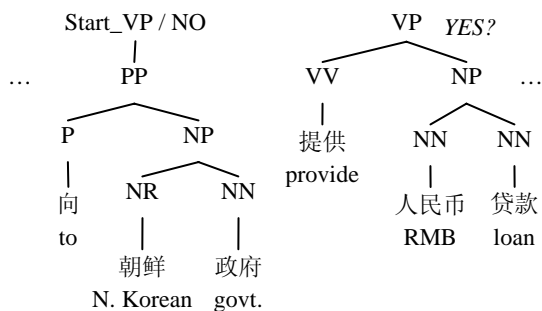


Figure 2: An application of CHECK with YES as the decision. Thus, VV (提供/provide) and NP (人民币/RMB 贷款/loan) reduce to a big VP.

### 4.3 Integrating Semantic Role-related Features into Syntactic Parsing Model

The integrated parsing approach as shown in Section 4.2 performs syntactic and semantic parsing synchronously. In contrast to traditional syntactic parsers where no semantic role-related information is used, it may be interesting to investigate the contribution of such information in the syntactic parsing model, due to the availability of such information in the syntactic parsing

process. In addition, it is found that 11% of predicates in a sentence are speculatively attached with two or more core arguments with the same label due to semantic parsing errors (partly caused by syntactic parsing errors in automatic parse trees). This is abnormal since a predicate normally only allows at most one argument of each core argument role (i.e., Arg0-Arg4). Therefore, such syntactic errors should be avoidable by considering those arguments already obtained in the bottom-up parsing process. On the other hand, taking those expected semantic roles into account would help the syntactic parser. In terms of our syntactic parsing model, this is done by directly incorporating various semantic role-related features into the syntactic parsing model (i.e., the BUILD procedure) when the newly-formed constituent covers one or more predicates.

For the example shown in Figure 2, once the constituent “VP (提供/provide 人民币/RMB 贷款/loan)”, which covers a verbal predicate “VV (提供/provide)”, is complete, the verbal SRL model would be triggered first to mark constituent “NP (人民币/RMB 贷款/loan)” as ARG1, given predicate “VV (提供/provide)”. Then, the BUILD procedure is called to make the BUILD decision for the newly-formed constituent “VP (提供/provide 人民币/RMB 贷款/loan)”. Table 2 lists various semantic role-related features explored in our syntactic parsing model and their instantiations with regard to the example shown in Figure 2. In Table 2, feature sf4 gives the possible core semantic roles that the focus predicate may take, according to its frame file; feature sf5 presents the semantic roles that the focus predicate has already occupied; feature sf6 indicates the semantic roles that the focus predicate is expecting; and SF1-SF8 are combined features. Specifically, if the current constituent covers  $n$  predicates, then  $14 * n$  features would be instantiated. Moreover, we differentiate whether the focus predicate is verbal or nominal, and whether it is the head word of the current constituent.

**Feature Selection.** Some features proposed above may not be effective in syntactic parsing. Here we adopt the greedy feature selection algorithm as described in Jiang and Ng (2006) to select useful features empirically and incrementally according to their contributions on the development data. The algorithm repeatedly selects one feature each time which contributes the most, and stops when adding any of the remain-

ing features fails to improve the syntactic parsing performance.

Feat.	Description
sf1	Path: the syntactic path from C to P. ( $VP > VV$ )
sf2	Predicate: the predicate itself. ( <i>提供/provide</i> )
sf3	Predicate class (Xue, 2008): the class that P belongs to. ( $C3b$ )
sf4	Possible roles: the core semantic roles P may take. ( $Arg0, Arg1, Arg2$ )
sf5	Detected roles: the core semantic roles already assigned to P. ( $Arg1$ )
sf6	Expected roles: possible semantic roles P is still expecting. ( $Arg0, Arg2$ )
SF1	For each already detected argument, its role label + its path from P. ( $Arg1 + VV < VP > NP$ )
SF2	sf1 + sf2. ( $VP > VV + 提供/provide$ )
SF3	sf1 + sf3. ( $VP > VV + C3b$ )
SF4	Combined possible argument roles. ( $Arg0 + Arg1 + Arg2$ )
SF5	Combined detected argument roles. ( $Arg1$ )
SF6	Combined expected argument roles. ( $Arg0 + Arg2$ )
SF7	For each expected semantic role, sf1 + its role label. ( $VP > VV + Arg0, VP > VV + Arg2$ )
SF8	For each expected semantic role, sf2 + its role label. ( $提供/provide + Arg0, 提供/provide + Arg2$ )

Table 2: SRL-related features and their instantiations for syntactic parsing, with “VP (*提供/provide* 人民币/RMB 贷款/loan)” as the current constituent C and “*提供/provide*” as the focus predicate P, based on Figure 2.

## 5 Experiments and Results

We have evaluated our integrated parsing approach on Chinese TreeBank 5.1 and corresponding Chinese PropBank and NomBank.

### 5.1 Experimental Settings

This version of Chinese PropBank and Chinese NomBank consists of standoff annotations on the file (chtb 001 to 1151.fid) of Chinese Penn TreeBank 5.1. Following the experimental settings in Xue (2008) and Li et al. (2009), 648 files (chtb 081 to 899.fid) are selected as the training data, 72 files (chtb 001 to 040.fid and chtb 900 to 931.fid) are held out as the test data, and 40 files (chtb 041 to 080.fid) are selected as the development data. In particular, the training, test, and development data contain 31,361 (8,642), 3,599 (1,124), and 2,060 (731) verbal (nominal) propositions, respectively.

For the evaluation measurement on syntactic parsing, we report labeled recall, labeled precision, and their F1-measure. Also, we report re-

call, precision, and their F1-measure for evaluation of SRL on automatic predicates, combining verbal SRL and nominal SRL. An argument is correctly labeled if there is an argument in manual annotation with the same semantic label that spans the same words. Moreover, we also report the performance of predicate recognition. To see whether an improvement in F1-measure is statistically significant, we also conduct significance tests using a type of stratified shuffling which in turn is a type of compute-intensive randomized tests. In this paper, ‘>>>’, ‘>>’, and ‘>’ denote p-values less than or equal to 0.01, in-between (0.01, 0.05], and bigger than 0.05, respectively.

We are not aware of any SRL system combining automatic predicate recognition, verbal SRL and nominal SRL on Chinese PropBank and NomBank. Xue (2008) experimented independently with verbal and nominal SRL and assumed correct predicates. Li et al. (2009) combined nominal predicate recognition and nominal SRL on Chinese NomBank. The CoNLL-2009 shared task (Hajic et al., 2009) included both verbal and nominal SRL on dependency parsing, instead of constituent-based syntactic parsing. Thus the SRL performances of their systems are not directly comparable to ours.

## 5.2 Results and Discussions

**Results of pipeline parsing on N-best parse trees.** While performing pipeline parsing on N-best parse trees, 20-best (the same as the heap size in our syntactic parsing) parse trees are obtained for each sentence using our syntactic parser as described in Section 3.1. The balance factor  $\alpha$  is set to 0.5 indicating that the two components in formula (1) are equally important. Table 3 compares the two pipeline parsing approaches on the top-best parse tree and the N-best parse trees. It shows that the approach on N-best parse trees outperforms the one on the top-best parse tree by 0.42 (>>>) in F1-measure on SRL. In addition, syntactic parsing also benefits from the N-best parse trees approach with improvement of 0.17 (>>>) in F1-measure. This suggests that pipeline parsing on N-best parse trees can improve both syntactic and semantic parsing.

It is worth noting that our experimental results in applying the re-ranking framework in Chinese pipeline parsing on N-best parse trees are very encouraging, considering the pessimistic results of Sutton and McCallum (2005), in which the re-ranking framework failed to improve the performance on English SRL. It may be because,

unlike Sutton and McCallum (2005),  $P(F, t/x)$  defined in this paper only considers those constituents which are identified as arguments. This can effectively avoid the noises caused by the predominant non-argument constituents. Moreover, the huge performance gap between Chinese semantic parsing on the gold parse tree and that on the top-best parse tree leaves much room for performance improvement.

Method	Task	R (%)	P (%)	F1
Pipeline on top -best parse tree	Syntactic	76.68	79.12	77.88
	SRL	62.96	65.04	63.98
	Predicate	94.18	92.28	93.22
	V-SRL	65.33	68.52	66.88
	V-Predicate	89.52	93.12	91.29
	N-SRL	49.58	48.19	48.88
	N-Predicate	86.83	71.76	78.58
Pipeline on 20 -best parse trees	Syntactic	76.89	79.25	78.05
	SRL	62.99	65.88	64.40
	Predicate	94.07	92.22	93.13
	V-SRL	65.41	69.09	67.20
	V-Predicate	89.66	93.02	91.31
	N-SRL	49.24	49.46	49.35
	N-Predicate	86.65	72.15	78.74
Integrated parsing	Syntactic	77.14	79.01	78.07
	SRL	62.67	67.67	65.07
	Predicate	93.97	92.42	93.19
	V-SRL	65.37	70.27	67.74
	V-Predicate	90.08	92.87	91.45
	N-SRL	48.02	52.83	50.31
Integrated parsing with semantic role-related features	N-Predicate	85.41	73.23	78.85
	Syntactic	77.47	79.58	78.51
	SRL	63.14	68.17	65.56
	Predicate	93.97	92.52	93.24
	V-SRL	65.74	70.98	68.26
	V-Predicate	89.86	93.17	91.49
N-SRL	48.80	52.67	50.66	
N-Predicate	85.85	72.78	78.78	

Table 3: Syntactic and semantic parsing performance on test data (using gold standard word boundaries). “V-” denotes “verbal” while “N-”denotes “nominal”.

**Results of integrated parsing.** Table 3 also compares the integrated parsing approach with the two pipeline parsing approaches. It shows that the integrated parsing approach improves the performance of both syntactic and semantic parsing by 0.19 (>) and 1.09 (>>>) respectively in F1-measure over the pipeline parsing approach on the top-best parse tree. It is also not surprising to find out that the integrated parsing approach outperforms the pipeline parsing approach on 20-best parse trees by 0.67 (>>>) in F1-measure on SRL, due to its exploring a larger

search space, although the integrated parsing approach integrates the SRL probability and the syntactic parsing probability in the same manner as the pipeline parsing approach on 20-best parse trees. However, the syntactic parsing performance gap between the integrated parsing approach and the pipeline parsing approach on 20-best parse trees is negligible.

**Results of integrated parsing with semantic role-related features.** After performing the greedy feature selection algorithm on the development data, features {SF3, SF2, sf5, sf6, SF4} as proposed in Section 4.3 are sequentially selected for syntactic parsing. As what we have assumed, knowledge about the detected semantic roles and expected semantic roles is helpful for syntactic parsing. Table 3 also lists the performance achieved with those selected features. It shows that the integration of semantic role-related features in integrated parsing significantly enhances both the performance of syntactic and semantic parsing by 0.44 (>>>) and 0.49 (>>) respectively in F1-measure. In addition, it shows that it outperforms the widely-used pipeline parsing approach on top-best parse tree by 0.63 (>>>) and 1.58 (>>>) in F1-measure on syntactic and semantic parsing, respectively. Finally, it shows that it outperforms the widely-used pipeline parsing approach on 20-best parse trees by 0.46 (>>>) and 1.16 (>>>) in F1-measure on syntactic and semantic parsing, respectively. This is very encouraging, considering the notorious difficulty and complexity of both the syntactic and semantic parsing tasks.

Table 3 also shows that our proposed method works well for both verbal SRL and nominal SRL. In addition, it shows that the performance of predicate recognition is very stable due to its high dependence on POS tagging results, rather than syntactic parsing results. Finally, it is not surprising to find out that the performance of predicate recognition when mixing verbal and nominal predicates is better than the performance of either verbal predicates or nominal predicates.

### 5.3 Extending the Word-based Syntactic Parser to a Character-based Syntactic Parser

The above experimental results on a word-based syntactic parser (assuming correct word segmentation) show that both syntactic and semantic parsing benefit from our integrated parsing approach. However, observing the great challenge of word segmentation in Chinese informa-



tion processing, it is still unclear whether and how much joint learning benefits character-based syntactic and semantic parsing. In this section, we extended the Ratnaparkhi parser (1999) to a character-based parser (with automatic word segmentation), and then examined the effectiveness of joint learning.

Given the three-pass process in the word-based syntactic parser, it is easy to extend it to a character-based parser for Chinese texts. This can be done by only replacing the TAG procedure in the first pass with a POSCHUNK procedure, which integrates Chinese word segmentation and POS tagging in one step, following the method described in (Ng and Low 2004). Here, each character is annotated with both a boundary tag and a POS tag. The 4 possible boundary tags include “B” for a character that begins a word and is followed by another character, “M” for a character that occurs in the middle of a word, “E” for a character that ends a word, and “S” for a character that occurs as a single-character word. For example, “北京市/Beijing city/NR” would be decomposed into three units: “北/north/B\_NR”, “京/capital/M\_NR”, and “市/city/E\_NR”. Also, “是/is/VC” would turn into “是/is/S\_VC”. Through POSCHUNK, all characters in a sentence are first assigned with POS chunk labels which must be compatible with previous ones, and then merged into words with their POS tags. For example, “北/north/B\_NR”, “京/capital/M\_NR”, and “市/city/E\_NR” will be merged as “北京市/Beijing/NR”, “是/is/S\_VC” will become “是/is/VC”. Finally the merged results of the POSCHUNK are fed into the CHUNK procedure of the second pass.

Using the same data split as the previous experiments, word segmentation achieves performance of 96.3 in F1-measure on the test data. Table 4 lists the syntactic and semantic parsing performance by adopting the character-based parser.

Table 4 shows that integrated parsing benefits syntactic and semantic parsing when automatic word segmentation is considered. However, the improvements are smaller due to the extra noise caused by automatic word segmentation. For example, our experiments show that the performance of predicate recognition drops from 93.2 to 90.3 in F1-measure when replacing correct word segmentations with automatic ones.

Method	Task	R (%)	P (%)	F1
Pipeline on top-best parse tree	Syntactic	82.23	84.28	83.24
	SRL	60.40	62.75	61.55
Pipeline on 20-best parse trees	Syntactic	82.25	84.29	83.26
	SRL	60.17	63.63	61.85
Integrated parsing with semantic role-related features	Syntactic	82.51	84.31	83.40
	SRL	60.09	65.35	62.61

Table 4: Performance with the character-based parser<sup>1</sup> (using automatically recognized word boundaries).

## 6 Conclusion

In this paper, we explore joint syntactic and semantic parsing to improve the performance of both syntactic and semantic parsing, in particular that of semantic parsing. Evaluation shows that our integrated parsing approach outperforms the pipeline parsing approach on N-best parse trees, a natural extension of the widely-used pipeline parsing approach on the top-best parse tree. It also shows that incorporating semantic information into syntactic parsing significantly improves the performance of both syntactic and semantic parsing. This is very promising and encouraging, considering the complexity of both syntactic and semantic parsing.

To our best knowledge, this is the first successful research on exploring syntactic parsing and semantic role labeling for verbal and nominal predicates in an integrated way.

## Acknowledgments

The first two authors were financially supported by Projects 60683150, 60970056, and 90920004 under the National Natural Science Foundation of China. This research was also partially supported by a research grant R-252-000-225-112 from National University of Singapore Academic Research Fund. We also want to thank the reviewers for insightful comments.

## References

- Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The Berkeley FrameNet Project. In *Proceedings of COLING-ACL 1998*.
- Xavier Carreras and Lluís Màrquez. 2004. Introduction to the CoNLL-2004 Shared Task: Semantic Role Labeling. In *Proceedings of CoNLL 2004*.

<sup>1</sup> POS tags are included in evaluating the performance of a character-based syntactic parser. Thus it cannot be directly compared with the word-based one where correct word segmentation is assumed.

- Xavier Carreras and Lluís Màrquez. 2005. Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling. In *Proceedings of CoNLL 2005*.
- Eugene Charniak. 2001. Immediate-Head Parsing for Language Models. In *Proceedings of ACL 2001*.
- Michael Collins. 1999. Head-Driven Statistical Models for Natural Language Parsing. Ph.D. thesis, University of Pennsylvania.
- Jenny Rose Finkel and Christopher D. Manning. 2009. Joint Parsing and Named Entity Recognition. In *Proceedings of NAACL 2009*.
- Jan Hajic, Massimiliano Ciaramita, Richard Johansson, et al. 2009. The CoNLL-2009 Shared Task: Syntactic and Semantic Dependencies in Multiple Languages. In *Proceedings of CoNLL 2009*.
- Zheng Ping Jiang and Hwee Tou Ng. 2006. Semantic Role Labeling of NomBank: A Maximum Entropy Approach. In *Proceedings of EMNLP 2006*.
- Fang Kong, Guodong Zhou, and Qiaoming Zhu. 2009. Employing the Centering Theory in Pronoun Resolution from the Semantic Perspective. In *Proceedings of EMNLP 2009*.
- Peter Koomen, Vasin Punyakanok, Dan Roth, Wen-tau Yih. 2005. Generalized Inference with Multiple Semantic Role Labeling Systems. In *Proceedings of CoNLL 2005*.
- Junhui Li, Guodong Zhou, Hai Zhao, Qiaoming Zhu, and Peide Qian. 2009. Improving Nominal SRL in Chinese Language with Verbal SRL information and Automatic Predicate Recognition. In *Proceedings of EMNLP 2009*.
- Chang Liu and Hwee Tou Ng. 2007. Learning Predictive Structures for Semantic Role Labeling of NomBank. In *Proceedings of ACL 2007*.
- Paola Merlo and Gabriele Mussillo. 2005. Accurate Function Parsing. In *Proceedings of EMNLP 2005*.
- Paola Merlo and Gabriele Musillo. 2008. Semantic Parsing for High-Precision Semantic Role Labeling. In *Proceedings of CoNLL 2008*.
- Adam Meyers, Ruth Reeves, Catherine Macleod, Rachel Szekely, Veronika Zielinska, Brian Young, and Ralph Grishman. 2004. Annotating Noun Argument Structure for NomBank. In *Proceedings of LREC 2004*.
- Scott Miller, Heidi Fox, Lance Ramshaw, and Ralph Weischedel. 2000. A Novel Use of Statistical Parsing to Extract Information from Text. In *Proceedings of ANLP 2000*.
- Srini Narayanan and Sanda Harabagiu. 2004. Question Answering based on Semantic Structures. In *Proceedings of COLING 2004*.
- Hwee Tou Ng and Jin Kiat Low. 2004. Chinese Part-of-Speech Tagging: One-at-a-Time or All-at-Once? Word-Based or Character-Based? In *Proceedings of EMNLP 2004*.
- Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The Proposition Bank: An Annotated Corpus of Semantic Roles. *Computational Linguistics*, 31, 71-106.
- Slav Petrov and Dan Klein. 2007. Improved Inference for Unlexicalized Parsing. In *Proceedings of NAACL 2007*.
- Sameer Pradhan, Kadri Hacioglu, Valerie Krugler, Wayne Ward, James H. Martin, and Daniel Jurafsky. 2005. Support Vector Learning for Semantic Argument Classification. *Machine Learning*, 2005, 60:11-39.
- Adwait Ratnaparkhi. 1999. Learning to Parse Natural Language with Maximum Entropy Models. *Machine Learning*, 34, 151-175.
- Mihai Surdeanu, Sanda Harabagiu, John Williams and Paul Aarseth. 2003. Using Predicate-Argument Structures for Information Extraction. In *Proceedings of ACL 2003*.
- Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The CoNLL-2008 Shared Task on Joint Parsing of Syntactic and Semantic Dependencies. In *Proceedings of CoNLL 2008*.
- Charles Sutton and Andrew McCallum. 2005. Joint Parsing and Semantic Role Labeling. In *Proceedings of CoNLL2005*.
- Nianwen Xue and Martha Palmer. 2003. Annotating the Propositions in the Penn Chinese TreeBank. In *Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing*.
- Nianwen Xue. 2006. Annotating the Predicate-Argument Structure of Chinese Nominalizations. In *Proceedings of LREC 2006*.
- Nianwen Xue. 2008. Labeling Chinese Predicates with Semantic Roles. *Computational Linguistics*, 34(2):225-255.
- Szu-ting Yi and Martha Palmer. 2005. The Integration of Syntactic Parsing and Semantic Role Labeling. In *Proceedings of CoNLL 2005*.
- Yue Zhang and Stephen Clark. 2008. Joint Word Segmentation and POS Tagging Using a Single Perceptron. In *Proceedings of ACL 2008*.