Wang HL, Zhou GD. Semantic role labeling of Chinese nominal predicates with dependency-driven constituent parse tree structure. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 28(6): 1117–1126 Nov. 2013. DOI 10.1007/s11390-013-1402-9

Semantic Role Labeling of Chinese Nominal Predicates with Dependency-Driven Constituent Parse Tree Structure

Hong-Ling Wang (王红玲), *Member*, *CCF*
and Guo-Dong Zhou* (周国栋), *Senior Member*, *CCF*, *Member*, *ACM*, *IEEE*

Natural Language Processing Lab, School of Computer Science and Technology, Soochow University, Suzhou 215006, China

E-mail: {hlwang, gdzhou} @suda.edu.cn

Received December 10, 2012; revised September 18, 2013.

Abstract This paper explores a tree kernel based method for semantic role labeling (SRL) of Chinese nominal predicates via a convolution tree kernel. In particular, a new parse tree representation structure, called dependency-driven constituent parse tree (D-CPT), is proposed to combine the advantages of both constituent and dependence parse trees. This is achieved by directly representing various kinds of dependency relations in a CPT-style structure, which employs dependency relation types instead of phrase labels in CPT (Constituent Parse Tree). In this way, D-CPT not only keeps the dependency relationship information in the dependency parse tree (DPT) structure but also retains the basic hierarchical structure of CPT style. Moreover, several schemes are designed to extract various kinds of necessary information, such as the shortest path between the nominal predicate and the argument candidate, the support verb of the nominal predicate and the head argument modified by the argument candidate, from D-CPT. This largely reduces the noisy information inherent in D-CPT. Finally, a convolution tree kernel is employed to compute the similarity between two parse trees. Besides, we also implement a feature-based method based on D-CPT. Evaluation on Chinese NomBank corpus shows that our tree kernel based method on D-CPT performs significantly better than other tree kernel-based ones and achieves comparable performance with the state-of-the-art feature-based ones. This indicates the effectiveness of the novel D-CPT structure in representing various kinds of dependency relations in a CPT-style structure and our tree kernel based method in exploring the novel D-CPT structure. This also illustrates that the kernel-based methods are competitive and they are complementary with the feature-based methods on SRL.

Keywords semantic role labeling, Chinese nominal predicate, dependency-driven constituent parse tree, tree kernel

1 Introduction

Semantic role labeling (SRL) has been drawing more and more attention in recent years due to its fundamental role in deep NLP (natural language processing) applications, such as information extraction[1], question answering[2], co-reference resolution[3] and document categorization[4]. Given a sentence and a predicate (either a verb or a noun) in a sentence, SRL recognizes and maps the constituents in the sentence into their corresponding semantic arguments (roles) of the predicate. According to predicate types, SRL can be divided into SRL for verbal predicates (so-called verbal SRL) and SRL for nominal predicates (so-called nominal SRL).

Usually, there are two kinds of methods for SRL. One is feature-based methods, which map a predicate-argument structure into a flat feature vector. The other is tree kernel based methods, which represent a predicate-argument structure as a parse tree and directly measure the similarity between two parse trees instead of the feature vector representations. Although feature-based methods have been consistently performing much better than the tree kernel based methods and represent the state-of-the-art in SRL, the problem with feature-based methods is that it is difficult for them to model syntactic structure information effectively. For example, the path feature widely used in feature-based methods is so sensitive to any change in a parse tree structure that a pair of parse tree structures will be represented by two different path features even if they differ only by one node[5]. In contrast, tree kernel methods have the potential to more effectively capture structured knowledge than feature-based methods. Moreover, Moschitti[6] has said that the kernel ability

Regular Paper
Supported by the National Natural Science Foundation of China under Grant Nos. 61331011 and 61273320, the National High Technology Research and Development 863 Program of China under Grant No. 2012AA011102, and the Natural Science Foundation of Jiangsu Provincial Department of Education under Grant No. 10KJB520016.
*Corresponding Author

to generate large feature sets is useful for quickly modeling new and well understood linguistic phenomena. In other words, kernel methods provide a vital way to explore the relationship governing complex predicate-argument structures and gain useful insights into the underlying linguistic phenomena[7].

Tree kernel methods are usually based on parse tree structure. Currently, there are two widely used representations for the linguistic structure of a sentence: constituent parse tree (CPT) structure and dependency parse tree (DPT) structure. These two kinds of syntactic parse tree structures behave quite differently in capturing different aspects of syntactic phenomena. Specifically, DPT is mainly concerned with the dependency relationship between individual words, instead of the phrase structure in a sentence as done in CPT.

In the literature, some studies[5,8-12] employ tree kernel based methods for SRL and focus on CPT structure. However, there are still some deficiencies with current CPT tree kernels. First, current CPT tree kernels fail to handle similar phrase structures (e.g., "buy a car" vs "buy a yellow car"). Second, the tree kernel fails to differentiate various portions of CPT (e.g., NP in the subject position vs NP in the object position). Finally, semantic information which plays a critical role in SRL has not been studied in above studies. As for DPT, few tree kernel based methods directly employ DPT for SRL due to its sparseness in that DPT only captures the dependency relationship between two words, although some feature-based methods[13-14] have attempted to explore structured information on the structure.

This paper addresses the above mentioned issues by proposing a tree kernel based method for Chinese nominal SRL via a convolution tree kernel. In order to explore more effective representation of syntactic information, a new syntactic parse tree structure, called dependency-driven constituent parse tree (D-CPT), is particularly proposed to combine the benefits of both constituent and dependency parse trees and compensate for their deficiencies. This is done by transforming DPT to a new CPT-style structure, via using dependency relation types instead of phrase labels in the traditional CPT structure. In this way, our tree kernel based method can benefit from the advantages of both DPT and CPT, since D-CPT not only keeps the dependency relationship information in DPT but also retains the basic hierarchical structure of CPT. As for the above mentioned examples, the similar phrase structures (e.g., "buy a car" vs "buy a yellow car") in CPT are different structures in D-CPT. Also this representation could differentiate various portions of parse tree (e.g., NP in the subject position vs NP in the object position) because they are different dependency relationship types in D-CPT. Moreover, several schemes are designed to extract various kinds of necessary information from D-CPT. This largely reduces the noisy information inherent in D-CPT. Finally, a convolution tree kernel is employed to compute the similarity between two parse trees. Besides, we also implement a feature-based method based on D-CPT. Evaluation of Chinese nominal SRL on Chinese NomBank corpus shows the effectiveness of the novel D-CPT structure in representing various kinds of dependency relations in a CPT-style structure and our tree kernel based method in exploring the novel D-CPT structure.

The rest of this paper is organized as follows. Section 2 briefly reviews the related work on SRL. Section 3 introduces our novel dependency-driven constituent parse tree structure. Sections 4 and 5 present our feature-based and tree kernel based methods on D-CPT, respectively. Section 6 presents the experimental results. Finally, Sections 7 draws the conclusion.

2 Related Work

This section mainly reviews the related work on tree kernel based methods and feature-based methods for SRL.

2.1 Tree Kernel Based Methods for SRL

To the best of our knowledge, most of tree kernel based methods for SRL deal with the English language and there are no reported studies on tree kernel based methods for Chinese SRL from either CPT or DPT perspectives.

Moschitti and Bejan[8] pioneered the research of tree kernel based methods for English verbal SRL. In their work, a predicate argument feature (PAF) structure is first extracted from CPT to include salient substructures in the predicate-argument structure. Then, the similarity between two PAFs is computed using a convolution tree kernel, proposed by Collins and Duffy[15]. Motivated by this work, more and more tree kernel based methods have been proposed and explored in SRL since then [5,10-11].

Moschitti *et al.*[10] improved the PAF structure by simply differentiating the node which exactly covers the argument to denote its boundary property. Moschitti[6] experimented with subtree (ST), subset trees (SST) and partial tree (PT) kernels on the classification of semantic roles defined in PropBank. Che *et al*[16] further separated the PAF structure into a path portion and a constituent structure portion. Then, a composite kernel was proposed to combine two convolution tree kernels over these two portions. Zhang *et al.*[11] proposed a grammar-driven convolution tree kernel to better explore grammatical substructures by considering the similarity between those non-identical substructures with similar syntactic properties.

Although there are some studies on tree kernel based methods from the DPT structure perspective, they are only for other NLP tasks rather than for SRL, such as verb classification[17], semantic relation extraction between named entities[18], question classification[19], opinion expression detection[20] and co-reference resolution[21]. Croce *et al.*[17] designed several structures for exploring dependency information, i.e., constituent tree (CT), grammatical relation centered tree (GRCT), and lexical centered tree (LCT). In particular, GRCT and LCT structures are DPT-style, which are expanded by adding POS tags as additional children on DPT. Then they used a structure similar to smoothed partial tree (SPT) to classify verb class from FrameNet and VerbNet. Croce *et al.*[19] merged together convolution dependency tree kernels with lexical similarities for question classification, which can efficiently and effectively measure the similarity between dependency structures. Nguyen *et al.*[18] explored three schemes to extract structured information from DPT, i.e., dependency words (DW) tree, grammatical relation (GR) tree, and grammatical relation and words (GRW) tree.

2.2 Feature-Based Methods for SRL

Compared with tree kernel based SRL, there are much more studies on feature-based SRL. Since this paper focuses on Chinese SRL, here we only review the related work on feature-based methods for Chinese SRL.

Sun and Jurafsky[22] and Pradhan *et al.*[23] pioneered the research on Chinese verbal and nominal SRLs, respectively, on their small private datasets. With the recent release of Chinese PropBank[24] and Chinese NomBank[25] for verbal and nominal predicates of Chinese, respectively, Xue and his colleagues[26-28] systematically explored Chinese verbal and nominal SRLs using feature-based methods, given gold predicates. Among them, Xue and Palmer[26] studied Chinese verbal SRL on Chinese PropBank and achieved the performance of 91.3 and 61.3 in $F1$-measure on gold and automatic CPT structures, respectively. Xue[27] extended their study on Chinese nominal SRL and attempted to improve the performance of nominal SRL by simply including the Chinese PropBank training instances into the training data for nominal SRL. Xue[28] further improved the performance on both verbal and nominal SRLs with a better constituent parser and more features.

Since then, Li *et al.*[29] improved Chinese nominal SRL by integrating various features derived from Chinese verbal SRL via a feature-based method on CPT, and achieved the state-of-the-art performance of 72.67 in $F1$-measure on Chinese NomBank. Li *et al.*[30] further present a feature-based SRL for verbal predicates of Chinese from the views of both CPT and DPT.

3 Dependency-Driven Constituent Parse Tree Structure

Syntactic parsing aims at identifying the grammatical structure in a sentence. There are two main paradigms for representing structured information: constituent parsing and dependency parsing, which produce different parse tree structures. In particular, the DPT structure encodes grammatical dependency relations between words in a sentence, with the words as nodes and corresponding dependency types as edges. An edge from a word to another word represents a grammatical dependency relation between these two words. Every word in a dependency parse tree has exactly one parent except for the root.

Fig.1 shows an example of the DPT structure for sentence (中国 进出口 银行 与 企业 加强 合作/The Import & Export Bank of China and the enterprise strengthen the cooperation). It also shows a nominal predicate with all its respective arguments annotated. Specifically, the nominal predicate “合作/cooperation” with “加强/strengthen” as the support verb has a argument, “中国 进出口 银行 与 企业/the Import & Export Bank and the enterprise”, as Arg0. In addition, W, R and G denote the word itself, its dependency relation with the head argument, and its part-of-speech (POS), respectively. In this section, we describe how to construct the D-CPT structure. In the next two sections, we explore how to exploit such D-CPT structure from both feature and tree kernel perspectives. Besides, a composite kernel is employed to combine the tree kernel and the feature-based linear kernel to further improve performance.

The D-CPT structure is achieved by transforming the DPT structure into a new CPT-style structure, using dependency types instead of phrase labels in the traditional CPT structure. CPT-style structure is different from the DPT-style structure, such as GRCT and LCT by Croce *et al.*[17] which retains the DPT structure and adds POS Tags, while the former retains the CPT structure and adds dependency relation. In particular, two transformations are done to achieve the D-CPT structure from the DPT structure:

1) For each node in DPT, create a new node by moving its contained word W and part-of-speech G as its left-most child while only keeping its contained dependency relation type R. Fig.2(a) illustrates an example of the resulted parse tree, corresponding to Fig.1.

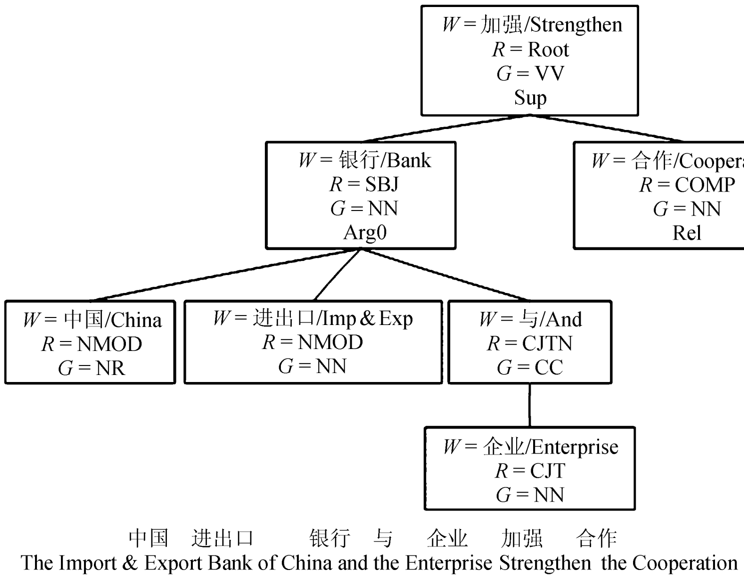


Fig.1. Example of the DPT structure with a nominal predicate and all its related arguments annotated.

2) For each terminal node, create a new node by moving its contained word W as its (only) child while only keeping its contained part-of-speech. Fig.2(b) illustrates an example of the resulted parse tree, corresponding to Fig.2(a).

Just as described in the introduction, both DPT and CPT have their own advantages. Fig.2(b) indicates that the new D-CPT structure can combine the advantages of both DPT and CPT structure since D-CPT not only keeps the dependency relationship information in DPT but also retains the similar hierarchical phrase structure of CPT.

4 Feature-Based SRL on D-CPT

A simple way to explore the novel D-CPT structure is via a feature vector, as verified in the SRL literature. While tree kernel based methods prefer to use the structured information of a parse tree, feature-based methods depend more on feature engineering of various lexical, syntactic, and semantic information.

However, since our main focus is not on feature-based methods, we do not employ complex feature engineering. Instead, we only use those traditional features widely used in CoNLL-2008 and CoNLL-2009 shared tasks, which aim at performing and evaluating SRL using a dependency-based representation for both syntactic and semantic dependencies on English and other languages. Table 1 shows a list of features extracted from the D-CPT structure.

Here, in order to save training time, we use a simple pruning strategy to filter out the nodes that are unlikely to be semantic arguments of the predicate according to the specific characteristics of Chinese NomBank corpus. In particular, given the nominal predicate as the current node, we only keep its father, grandfather, grandfather's siblings, grandfather's children, siblings, siblings'

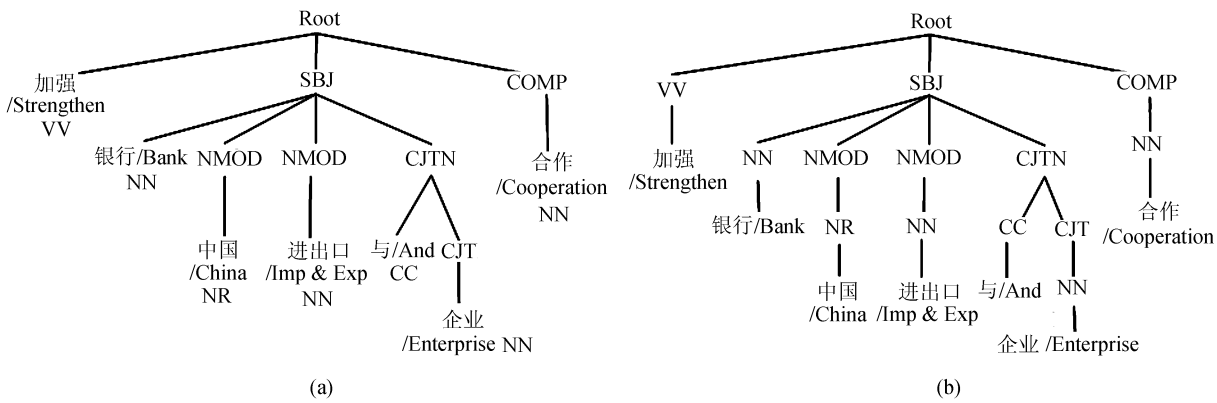


Fig.2. Example of achieving the D-CPT structure from the DPT structure.

Table 1. Features and Remarks Explored in D-CPT

Features	Remarks
Dependent word and its POS tag	Modifying word and its POS tag in the dependency relation (企业/enterprise, NN)
Dependency relation type	Type of the dependency relation (CJT)
Predicate word and its POS tag	Current predicate and its POS tag (合作/cooperation, NN)
Head word and its POS tag	Modified (head) word and its POS tag in the dependency relation (与/and, CC)
DepSubCat	Subcategorization frame of the predicate (COMP->-)
DeprelPath	Path from predicate to argument concatenating dependency labels with the direction of the edge (CJT↑CJTN↑SBJ↑ROOT↓COMP)
POSPath	The same as DeprelPath, but dependency labels are exchanged for POS tags (NN↑CC↑NN↑VV↓NN)
Path	Path from the modifying word to the nominal predicate (NN↑CJT↑CJTN↑SBJ↑ROOT↓COMP↓NN)
Family membership	Indicating how the dependency relation is related to the predicate in the family (siblings' grandchildren)
ChildDepSet	Set of dependency labels of the children of the predicate (None)
ChildPOSSet	Set of POS tags of the children of the predicate (None)
SiblingDepSet	Set of dependency labels of the siblings of the predicate (SBJ)
SiblingPOSSet	Set of POS tags of the siblings of the predicate (NN)
Position	Position of the argument with respect to the predicate (before)

Note: for nominal predicate "合作/Cooperation" and argument candidate "企业/Enterprise" with regard to Fig.2(b).

children, siblings' grandchildren, children, grandchildren with respect to the D-CPT structure. As a result, our pruning strategy effectively reduces the number of instances for semantic role labeling by approximately 2∼3 folds at the risk of 2% loss of semantic arguments.

5 Tree Kernel Based SRL on D-CPT

An alternative way to explore the novel D-CPT structure is via a tree kernel.

5.1 Tree Kernel

Given a parse tree structure, this paper employs the well-known convolution tree kernel[15] to compute the similarity between two parse trees. In principle, the convolution tree kernel works by counting the number of common subset trees as the syntactic similarity between two parse trees. Thus, this tree kernel implicitly defines a large feature space.

$$K_C(T_1, T_2) = \sum_{n_1 \in N_1, n_2 \in N_2} \Delta(n_1, n_2),$$

where N_j is the set of nodes in tree T_j, and $\Delta(n_1, n_2)$ evaluates the common sub-trees rooted at n_1 and n_2 and is computed recursively as follows:

1) If the context-free productions (context-free grammar (CFG) rules) at n_1 and n_2 are different, $\Delta(n_1, n_2) = 0$; otherwise go to step 2.

2) If both n_1 and n_2 are POS tags, $\Delta(n_1, n_2) = 1 \times \lambda$; otherwise go to step 3.

3) Calculate $\Delta(n_1, n_2)$ recursively as:

$$\Delta(n_1, n_2) = \lambda \prod_{k=1}^{\#ch(n_1)} (1 + \Delta(ch(n_1, k), ch(n_2, k))),$$

where $\#ch(n)$ is the number of children of node n, $ch(n, k)$ is the k-th child of node n and λ ($0 < \lambda < 1$) is the decay factor in order to make the kernel value less variable with respect to different sub-tree sizes.

Besides, in order to capture the complementary nature between feature-based methods and tree kernel based methods, we combine them via a composite kernel, which has been proved to be effective in the literature[11].

In particular, our composite kernel is combined by linearly interpolating a convolution tree kernel K_T over a parse tree structure and a feature-based linear kernel K_L as follow:

$$CK = \alpha \times K_L + (1 - \alpha) \times K_T,$$

where α is a coefficient for K_L.

5.2 Extraction Schemes

Given a predicate and an argument candidate, the key is to extract an appropriate portion of the D-CPT structure in covering necessary information to determine their semantic relationship. Generally, the more substructures of the tree are included, the more structured information would be provided at the risk of more noisy information.

In our study, we examine three schemes for this purpose, considering the specific characteristics of nominal SRL. Since D-CPT takes the advantages of both CPT and DPT, these schemes can directly encode the argument structure of lexical units populated at their nodes through corresponding dependency relations.

Shortest Path Tree (*SPT*). This extraction scheme only includes the nodes occurring in the shortest path connecting the predicate and the argument candidate, via the nearest commonly-governing node. Fig.3(a) shows an example of SPT for nominal predicate “合作/cooperation” and argument candidate “企业/enterprise”.

SV-SPT (*Chinese*). NomBank adopts the same predicate-specific approach in representing the core arguments of a predicate as (Chinese) PropBank, with special treatment for nominal predicate-specific phenomena, such as support verbs, which cover much useful information in determining the semantic relationship between the nominal predicate and the argument candidate. Specifically, there is a specific label, “Sup”, to indicate the support verb of the nominal predicate. Fig.1 includes an example of support verb “加强/strengthen”, in helping introduce the arguments of the nominal predicate “合作/cooperation”. Normally, a verb is marked as a support verb only when it shares some arguments with the nominal predicate. Statistics on NomBank and Chinese NomBank shows that about 20% and 22% of arguments are introduced via a support verb, respectively. This indicates the importance of support verb in nominal SRL. Since the support verb of a nominal predicate normally pivots outside the nominal predicate and its arguments in the D-CPT structure, e.g., the one as shown in Fig.2(b), it is necessary to include the support verb information in nominal SRL. Fig.3(b) shows an example of SPT after retaining the support verb information. We call the new structure as SV-SPT.

H-SV-SPT. It is well proven that the head argument of the argument candidate plays a critical role in verbal SRL. In our study, we also consider the head argument information in nominal SRL. Fig.3(c) illustrates an example after attaching the head argument information to SV-SPT. We call the new structure as H-SV-SPT.

6 Experimentation and Discussion

We systematically evaluate our approach on Chinese NomBank.

6.1 Experimental Setting

Following the experimental setting in [28] and [29], 648 files (chtb 081∼899.fid) are selected as the training data, 72 files (chtb 001∼040.fid and chtb 900∼931.fid) are held out as the test data, and 40 files (chtb 041∼080.fid) as the development data, with 8 642, 1 124, and 731 propositions, respectively.

At first, we prune the parse trees in order to save training time. After pruning, we then do argument identification for those remaining candidates, and classify the positive ones into their corresponding semantic roles at last. Here, for argument identification we use the same tree structure, kernel, and configuration as for role classification. And the only difference is that we use binary classifier in argument identification.

We use the SVM-light toolkit with the convolution tree kernel function SVM-light-TK as the classifier. In particular, the training parameters C (SVM) and λ (tree kernel) are fine-tuned to 4.0 and 0.5 respectively. For the composite kernel, the coefficient α is fine-tuned to 0.5. Since SVM is a binary classifier, we apply the one versus other strategies to implement multi-class classification, which builds multiple classifiers so as to separate one class from all the others. The final decision of an instance in the multiple binary classifications is determined by the class which has the maximal SVM output.

To have a fair comparison of our system with the state-of-the-art ones, we use the widely used segment-based evaluation algorithm, proposed by Johansson and Nugues[14]. 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 computation-intensive randomized tests. In this paper, “$\ggg$”, “$\gg$”, and “$>$” denote p-values less than or equal to 0.01, in-between (0.01, 0.05], and bigger than 0.05, respectively.

6.2 Experimental Results on Gold Standard Parse Trees

Table 2 shows the performance of our feature-based method using different structures of gold parse trees. It shows that the performance on D-CPT is better than that on DPT since D-CPT contains all of the information in DPT, while it is not so good as the one on CPT

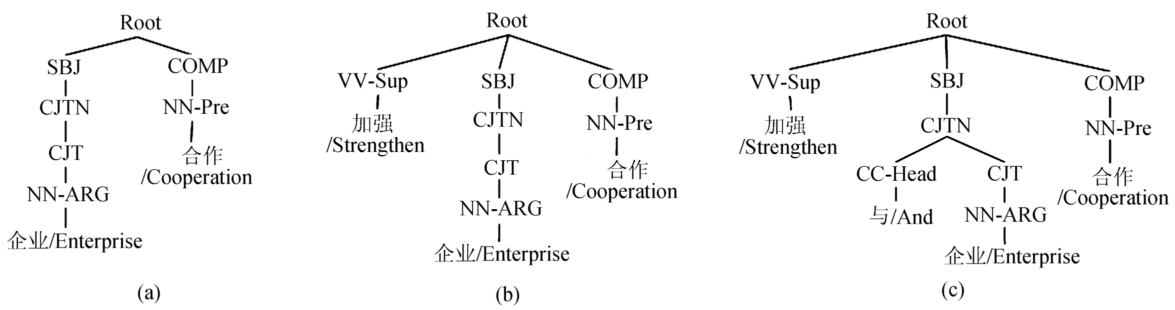


Fig.3. Extraction schemes. (a) Example of SPT. (b) Example of SV-SPT. (c) Example of H-SV-SPT.

due to the lack of phrase labels in D-CPT. It also shows that the precision on D-CPT is much higher than that on CPT, while the recall on D-CPT is far lower than that on CPT.

Table 2. Performance of Our Feature-Based Method Using Different Structures of Gold Parse Trees

Structure	Precision (%)	Recall (%)	$F1$
CPT	67.86	73.63	**70.63**
DPT	79.61	58.92	67.72
D-CPT	**79.96**	60.02	68.57

Table 3 shows the performance of our tree kernel based method using different extraction schemes on the D-CPT structure of gold parse trees. Table 3 shows that:

1) SPT achieves the performance of 65.98 in $F1$-measure with a much lower recall of only 58.26%, compared with 76.07% in precision. This indicates the necessity of incorporating more structured information into SPT.

2) SV-SPT achieves the performance of 69.89 in $F1$-measure. This means that SV-SPT performs significantly better than SPT by 3.91 ($\ggg$) in $F1$-measure, much due to the gain in both precision and recall. This indicates the discriminative ability of the support verb in determining the semantic relationship between the nominal predicate and the argument candidate.

3) H-SV-SPT further slightly improves the performance by 0.43 ($>$) in $F1$-measure, due to considering the head argument information, which has been proven useful in feature-based methods.

Table 3. Performance of Our Tree Kernel Based Method Using Different Extraction Schemes on the D-CPT Structure of Gold Parse Trees

Scheme	Precision (%)	Recall (%)	$F1$
SPT	76.07	58.26	65.98
SV-SPT	79.64	62.27	69.89
H-SV-SPT	79.79	62.86	**70.32**

Table 4 illustrates the performance comparison with different kernel setups on gold parse trees. It shows that:

1) Our tree kernel on the new D-CPT structure using the extraction scheme of H-SV-SPT performs much better than a popular feature-based linear kernel by 1.75 ($\gg$). This denotes the effectiveness of our D-CPT structure in representing the dependency relations in a tree kernel based method, which may perform well without complicated feature engineering.

2) The tree kernel and the feature-based linear kernel is quite complementary since the combination of them via a simple composite kernel improves the performance by 4.72 ($\ggg$) and 2.97($\ggg$) in $F1$-measure over the feature-based linear kernel and the tree kernel.

Table 4. Comparison of Different Kernels on Gold Parse Trees

Kernel	Precision (%)	Recall (%)	$F1$
Linear kernel	79.96	60.02	68.57
Tree kernel	79.79	62.86	70.32
Composite kernel	80.85	67.03	**73.29**

6.3 Experimental Results on Automatic Parse Trees

In the previous subsection, we assumed the availability of gold parse trees during the testing process. In this subsection, we evaluate the performance using automatic parse trees. In this paper, we firstly get the CPT structure using the word-based Berkeley parser and then convert it to the DPT structure using the same conversion toolkit as adopted by the CoNLL-2009 shared task. Tables 5∼7 present the performance on automatic parse trees.

From Table 5 we can see that the performance tendency is similar to gold parse trees. Due to the new errors introduced during the transition from CPT to DPT structure, the performance gap further widens between CPT and D-CPT.

Table 5. Performance of Our Feature-Based Method Using Different Structures of Automatic Parse Trees

Structure	Precision (%)	Recall (%)	$F1$
CPT	55.95	66.74	60.87
DPT	64.36	49.37	55.88
D-CPT	66.89	48.90	56.50

Table 6. Performance of Our Tree Kernel Based Method Using Different Extraction Schemes on the D-CPT Structure of Automatic Parse Trees

Scheme	Precision (%)	Recall (%)	$F1$
SPT	63.17	46.30	53.44
SV-SPT	66.06	50.17	57.03
H-SV-SPT	67.51	50.91	58.04

Table 7. Comparison of Different Kernels on Automatic Parse Trees

Kernel	Precision (%)	Recall (%)	$F1$
Linear kernel	66.89	48.90	56.50
Tree kernel	67.51	50.91	58.04
Composite kernel	66.59	55.07	60.28

Table 6 and Table 7 show that:

1) For each extraction scheme on D-CPT of automatic parse trees, our tree kernel based method shows

performance tendency similar to that of gold parse trees. For example, our tree kernel based method achieves the best performance of 58.04 in $F1$-measure when including the support verb and the head argument into SPT.

2) For each kernel, the performance on automatic parse trees drops by about 12 in $F1$-measure, compared with that on gold parse trees. This indicates the dependency of Chinese nominal SRL on the performance of syntactic parsing.

6.4 Comparison with Other Tree Kernel Based Methods

Nguyen *et al.*[19] proposed a dependency words (DW) tree, a grammatical relation (GR) tree, and a grammatical relation and words (GRW) tree, extracted from the DPT structure, to a similar task of semantic relation extraction between named entities. In their work, the DW tree is simply constituted by keeping the words in the DPT structure. The GR tree is generated by replacing the words in the DW tree with their dependency relations. The GRW tree is formed by combining the DW and GR trees, where the latter is inserted as a father node of the former.

Table 8 compares our D-CPT structure with the DW, GR and GRW trees on Chinese nominal SRL, using the same convolution tree kernel on gold parse trees. Table 6 shows that even SPT, extracted from the D-CPT structure using the simplest scheme, significantly outperforms the GR ($\ggg$), DW ($\ggg$) and GRW ($\ggg$) trees. This indicates the effectiveness of our D-CPT structure in that D-CPT not only keeps the dependency information of the DPT structure but also retains the CPT structure.

Table 8. Comparison with Other Tree Kernel Based Methods

Structure	Precision (%)	Recall (%)	$F1$
GR	79.42	28.17	41.59
DW	77.80	52.72	62.85
GRW	77.47	54.22	63.79
D-CPT (SPT)	76.07	58.26	**65.98**

6.5 Comparison with Other Systems

Finally, Table 9 compares our proposed method with two state-of-the-art ones on Chinese NomBank[28-29]. Both of them are feature-based ones with various features derived from the CPT structure via extensive feature engineering.

Table 9 shows that our tree kernel based method achieves comparable performance with the state-of-the-art feature-based ones on either gold parse trees or auto parse trees. One advantage of our proposed tree kernel based method on the novel D-CPT structure lies in its simplicity and effectiveness. Another advantage is its flexibility for further performance improvement. In this paper, we have proposed three simple extraction schemes to extract necessary information from D-CPT. It will be easy to incorporate other useful information, such as competitive information from other argument candidates.

Table 9. Comparison with the State-of-the-Art Systems

System	Gold Parse Trees ($F1$)	Automatic Parse Trees ($F1$)
Linear kernel (ours): feature-based	68.57	56.50
Tree kernel (ours): D-CPT	70.32	58.04
Composite kernel (ours)	**73.29**	60.28
Xue[28]: feature-based	69.60	57.60
Li *et al.*[29]: feature-based	70.63	58.66

6.6 Experimentation on CoNLL-2009 Chinese Corpus

To further illustrate the effectiveness of the novel DR-CPT structure in representing various kinds of dependency relations in tree kernel based methods, we also do experiments on the CoNLL-2009 Chinese corpus.

Since most predicates in the CoNLL-2009 Chinese corpus are verbal and do not have the support verbs, here we only apply the SPT and H-SPT extraction schemes. Furthermore, we only select those simple features widely used in CoNLL-2008 and CoNLL-2009 shared tasks in the composite Kernel.

Predicate disambiguation is a sub-task of the CoNLL-2009 shared task. In order to better compare the results of SRL-only, we simply employ the predicate disambiguation module as proposed by Bjorkelund *et al.*[31] who obtained the best $F1$ score on the Chinese corpus.

Table 10 compares the performance of different kernel setups on the CoNLL-2009 Chinese corpus. It shows that:

1) Our tree kernel method achieves performance comparable with the work of Meza-Ruiz and Riedel[32],

Table 10. Performance of Our Tree Kernel Based Method on the CoNLL-2009 Chinese Corpus

System	$F1$
Tree kernel (ours): SPT	76.88
Tree kernel (ours): H-SPT	77.43
Composite kernel (ours)	**78.47**
Bjorkelund *et al.*[31]: feature-based	78.60
Meza-Ruiz and Riedel[32]: feature-based	77.73

who obtained the second best performance on the Chinese corpus. It further denotes the effectiveness of our D-CPT structure in a tree kernel based method on SRL of verbal predicates.

2) Our composite kernel (without global re-ranking) achieves performance comparable with the work of Bjorkelund *et al.*[31] who employed a global re-ranking strategy and obtained the best performance on the Chinese Corpus.

7 Conclusions

This paper proposed a novel D-CPT structure, which employs dependency types instead of phrase labels in the traditional CPT structure. Based on the D-CPT structure, we systematically explored a tree kernel based method and a feature-based method for Chinese nominal SRL. In particular, we proposed a simple strategy, which transforms the DPT structure into a CPT-style structure. Generally, D-CPT takes the advantages of both DPT and CPT by not only keeping the dependency relationship information in DPT but also retaining the basic hierarchical structure of CPT. Furthermore, several extraction schemes were designed to extract various kinds of necessary information. Evaluation on Chinese NomBank corpus shows the effectiveness of our approach for nominal SRL. Evaluation on the CoNLL-2009 Chinese corpus also justifies the effectiveness of our approach for both nominal and verbal SRL.

Our experimental results show the approach has two contributions. One is that kernel-based methods are competitive, which achieve performance comparable with the state-of-the-art feature-based ones. The other is that kernel-based methods are complementary with feature-based methods, which has been verified by the performance of composite kernel.

In the near future, we will explore more necessary structured information in the novel D-CPT structure. Besides, we will extend this structure to similar tasks, such as semantic relation extraction between named entities and coreference resolution.

References

[1] Surdeanu M, Harabagiu S, Williams J, Aarseth P. Using predicate-argument structures for information extraction. In *Proc. the 41st Annual Meeting on Association for Computational Linguistics*, July 2003, Vol.1, pp.8-15.

[2] Narayanan S, Harabagiu S. Question answering based on semantic structures. In *Proc. the 20th Int. Conf. Computational Linguistics*, August 2004, Article No. 693.

[3] Ponzetto S P, Strube M. Semantic role labeling for coreference resolution. In *Proc. the 11th EACL*, April 2006, pp.143-146.

[4] Persson J, Johansson R, Nugues P. Text categorization using predicate-argument structures. In *Proc. the 17th NODALIDA*, May 2009, pp.142-149.

[5] Moschitti A, Pighin D, Basili R. Tree kernels for semantic role labeling. *Computational Linguistics*, 2008, 34(2): 193-224.

[6] Moschitti A. Efficient convolution kernels for dependency and constituent syntactic trees. In *Proc. the 17th ECML*, September 2006, pp.318-329.

[7] Punyakanok V, Roth D, Yih W. The necessity of syntactic parsing for semantic role labeling. In *Proc. the 19th IJCAI*, July 30-August 5, 2005, pp.1117-1123.

[8] Moschitti A, Bejan C A. A semantic kernel for predicate argument classification. In *Proc. the 8th CoNLL*, May 2004, pp.17-24.

[9] Moschitti A. A study on convolution kernels for shallow semantic parsing. In *Proc. the 42nd ACL*, July 2004, Article No. 335.

[10] Moschitti A, Pighin D, Basili R. Tree kernel engineering in semantic role labeling systems. In *Proc. the 11st EACL*, April 2006, pp.49-56.

[11] Zhang M, Che W X, Aw A, Tan C L, Zhou G D, Liu T, Li S. A grammar-driven convolution tree kernel for semantic role classification. In *Proc. the 45th Annual Meeting of the Association of Computational Linguistics*, June 2007, pp.200-207.

[12] Zhou G D, Li J H, Fan J X, Zhu Q M. Tree kernel-based semantic role labeling with enriched parse tree structure. *Information Processing and Management*, 2011, 47(3): 349-362.

[13] Hacioglu K. Semantic role labeling using dependency trees. In *Proc. the 20th COLING*, August 2004, Article No. 1273.

[14] Johansson R, Nugues P. Dependency-based semantic role labeling of PropBank. In *Proc. the 5th EMNLP*, Oct. 2008, pp.69-78.

[15] Collins M, Duffy N. New ranking algorithms for parsing and tagging: Kernels over discrete structures, and the voted perception. In *Proc. the 40th Annual Meeting on Association for Computational Linguistics*, July 2002, pp.263-270.

[16] Che W X, Zhang M, Liu T *et al.* A hybrid convolution tree kernel for semantic role labeling. In *Proc. the 44th ACL*, July 2006, pp.73-80.

[17] Croce D, Moschitti A, Basili R, Palmer M. Verb classification using distributional similarity in syntactic and semantic structures. In *Proc. the 50th ACL*, July 2012, pp.263-272.

[18] Nguyen T T, Moschitti A, Riccardi G. Convolution kernels on constituent, dependency and sequential structures for relation extraction. In *Proc. the 2009 Conf. Empirical Methods in Natural Language Processing*, Aug. 2009, Vol.3, pp.1378-1387.

[19] Croce D, Moschitti A, Basili R. Semantic convolution kernels over dependency trees: Smoothed partial tree kernel. In *Proc. the 20th CIKM*, October 2011, pp.2013-2016.

[20] Johansson R, Moschitti A. Syntactic and semantic structure for opinion expression detection. In *Proc. the 14th CoNLL*, July 2010, pp.67-76.

[21] Kong F, Zhou G D, Qian L H *et al.* Dependency-driven anaphoricity determination for coreference resolution. In *Proc. the 23rd Int. Conf. Computational Linguistics*, August 2010, pp.599-607.

[22] Sun HL, Jurafsky D. Shallow semantic parsing of Chinese. In *Proc. the 2004 NAACL*, May 2004, pp.249-256.

[23] Pradhan S, Sun H L, Ward W, Martin H J, Jurafsky D. Parsing arguments of nominalizations in English and Chinese. In *Proc. HLT-NAACL*, May 2004, pp.141-144.

[24] Xue N W, Palmer M. Annotating the propositions in the Penn Chinese Treebank. In *Proc. the 2nd SIGHAN Workshop on Chinese Language Processing*, July 2003, Vol.17, pp.47-54.

[25] Xue N W. Annotating the predicate-argument structure of Chinese nominalizations. In *Proc. the 2006 LREC*, May 2006, pp.1382-1387.

[26] Xue N W. Palmer M. Automatic semantic role labeling for Chinese verbs. In *Proc. the 19th IJCAI*, July 30-August 5,

2005, pp.1160-1165.

[27] Xue N W. Semantic role labeling of nominalized predicates in Chinese. In *Proc. the 2006 HLT-NAACL*, June 2006, pp.431-438.

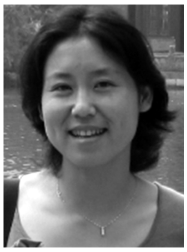
[28] Xue N W. Labeling Chinese predicates with semantic roles. *Computational Linguistics*, 2008, 34(2): 225-255.

[29] Li J H, Zhou G D, Zhao H, Zhu Q M, Qian P D. Improving nominal SRL in Chinese language with verbal SRL information and automatic predicate recognition. In *Proc. EMNLP*, August 2009, Vol.3, pp.1280-1288.

[30] Li S Q, Lu Q, Zhao T J, Liu P Y, Li H J. Combing constituent and dependency syntactic views for Chinese semantic role labeling. In *Proc. the 23rd COLING*, August 2010, pp.665-673.

[31] Bjørkelund A, Hafdell L, Nugues P. Multilingual semantic role labeling. In *Proc. the 13th CoNLL*, June 2009, pp.43-48.

[32] Meza-Ruiz I, Riedel S. Multilingual semantic role labeling with Markov logic. In *Proc. the 13th CoNLL*, June 2009, pp.85-90.



Hong-Ling Wang received the Ph.D. degree in computer application technology from Soochow University, Suzhou, China, in 2009. Currently, she is an associate professor at the university. Her research interests include natural language processing and information extraction. She has been a member of CCF since 2009.

Guo-Dong Zhou received the Ph.D. degree in computer science from the National University of Singapore in 1999. He joined the Institute for Infocomm Research, Singapore, in 1999, and had been an associate scientist, scientist and associate lead scientist at the institute until August 2006. Currently, he is a distinguished professor at the School of Computer Science and Technology, Soochow University, Suzhou. His research interests include natural language processing, information extraction and machine learning. He has been a senior member of CCF since 2008 and a member of ACM and IEEE since 1999.