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

Hongling Wang, Bukang Wang, and Guodong Zhou

NLP Lab, School of Computer Science & Technology, Soochow University, Suzhou, China
{hlwang,20094227014,gdzhou}@suda.edu.cn

Abstract. This paper explores a tree kernel-based method for nominal semantic role labeling (SRL). In particular, a new dependency-driven constituent parse tree (D-CPT) structure is proposed to better represent the dependency relations in a CPT-style structure, which employs dependency relation types instead of phrase labels in CPT. In this way, D-CPT not only keeps the dependency relationship information in the dependency parse tree (DPT) structure but also retains the basic structure of CPT. 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 . Evaluation on Chinese NomBank shows that our tree kernel-based method on D-CPT achieves comparable performance with the state-of-art feature-based ones. This indicates the effectiveness of the novel D-CPT structure for better representation of dependency relations in tree kernel-based methods. To our knowledge, this is the first research of tree kernel-based SRL on effectively exploring dependency relationship information, which achieves comparable performance with the state-of-the-art feature-based ones.

Keywords: Semantic Role Labeling, Dependency 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 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 type, SRL can be divided into SRL for verbal predicates (verbal SRL) and SRL for nominal predicates (nominal SRL).

Usually, there are two kinds of methods for SRL. One is feature-based methods, which map a predicate-argument structure to 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 predicate-argument parse trees instead of the feature vector representations. Although feature-based methods have been

M. Zhou et al. (Eds.): NLPCC 2012, CCIS 333, pp. 97–109, 2012.

consistently performing much better than kernel-based methods and represent the state-of-the-art in SRL, tree kernel-based methods have the potential in better capturing structured knowledge in the parse tree structure, which is critical for the success of SRL, than feature-based methods. In the literature, however, there are only a few studies [5-8] employing tree kernel-based methods for SRL and most of them focus on the constituent parse tree (CPT) structure.

Although some feature-based methods [9-10] have attempted to explore structured information in the dependency parse tree (DPT) structure, few tree kernel-based methods directly employ DPT due to its sparseness in that DPT only captures the dependency relationship between two words. While both DPT and CPT are widely used to represent the linguistic structure of a sentence, however, there still exist some important differences between them. For example, DPT mainly concerns with the dependency relationship between individual words, instead of the phrase structure in a sentence as done in CPT. Therefore, these two kinds of syntactic parse tree structures may behave quite differently in capturing different aspects of syntactic phenomena.

In this paper, we explore a tree kernel-based method for Chinese nominal SRL using a new syntactic parse tree structure, called dependency-driven constituent parse tree (D-CPT). This is done by transforming DPT to a new CPT-style structure, 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 structure of CPT. Evaluation of Chinese nominal SRL on Chinese NomBank shows that our tree kernel-based method achieves comparable performance with the state-of-the-art feature-based methods.

The rest of this paper is organized as follows: Section 2 briefly reviews the related work on SRL. Section 3 introduces our tree kernel-based method over the novel D-CPT structure. Section 4 presents the experimental results. Finally, Sections 5 draws the conclusion.

2 Related Work

Since this paper focuses on tree kernel-based methods for SRL, this section only overviews the related work on tree kernel-based methods for SRL. For an overview on feature-based methods for SRL, please refer to Xue [11] and Li et al [12].

- **Tree Kernel-Based Methods for SRL**

Moschitti [5] pioneers the research of tree kernel-based methods for English verbal SRL. In his work, a Predicate Argument Feature (PAF) structure is 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 [13]. Motivated by this work, more and more tree kernel-based methods are proposed and explored in SRL since then [7,8,14].

Moschitti et al [7] improves the PAF structure by simply differentiating the node which exactly covers the argument to denote its boundary property. Che et al [14] further separates the PAF structure into a path portion and a constituent structure portion. Then, a composite kernel is proposed to combine two convolution tree kernels over these two portions. Zhang et al [8] proposes a grammar-driven convolution tree kernel to better explore grammatical substructures by considering the similarity between those non-identical substructures with similar syntactic properties.

To our knowledge, there are no reported studies on tree kernel-based methods for SRL from the DPT structure perspective. However, there are a few related studies in other NLP tasks, such as semantic relation extraction between named entities [15] and co-reference resolution [16], which employ DPT in tree kernel-based methods and achieve comparable performance to the ones on CPT. For example, Nguyen et al [15] explore three schemes to extract structured information from DPT: dependency words (DW) tree, grammatical relation (GR) tree, and grammatical relation and words (GRW) tree.

- **SRL on Chinese**

With recent release of Chinese PropBank and Chinese NomBank for verbal and nominal predicates of Chinese, respectively, Xue and his colleagues [11,17,18] systematically explore Chinese verbal and nominal SRLs using feature-based methods, given golden predicates. Among them, Xue and Palmer [17] study Chinese verbal SRL on Chinese PropBank and achieve the performance of 91.3 and 61.3 in F1-measure on golden and automatic CPT structures, respectively. Xue [18] extends their study on Chinese nominal SRL and attempts to improve the performance of nominal SRL by simply including the Chinese PropBank training instances into the training data for nominal SRL. Xue [11] further improves the performance on both verbal and nominal SRLs with a better constituent parser and more features.

Since then, Li et al [12] improve Chinese nominal SRL by integrating various features derived from Chinese verbal SRL via a feature-based method on CPT, and achieve the state-of-art performance of 72.67 in F1-measure on Chinese NormBank. Li et al [19] further present a feature-based SRL for verbal predicates of Chinese from the views of both CPT and DPT.

To our knowledge, there are no reported studies on tree kernel-based methods for Chinese SRL from either CPT or DPT perspectives.

3 Tree Kernel-Based Nominal SRL on D-CPT

Syntactic parsing aims at identifying the grammatical structure in a sentence. There are two main paradigms for representing the structured information: constituent and dependency parsing, which produces different parse tree structures. In particular, the DPT structure encodes grammatical 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 relation between these two words. Every word in a dependency tree has exactly one parent except 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 and 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 first describe how to construct the D-CPT structure. Then, we explore different ways to extract necessary structured information from this new parse tree structure. Finally, we briefly present the convolution tree kernel for computing the similarity between two parse trees and its combination with a feature-based linear kernel via a composite kernel for further performance improvement.

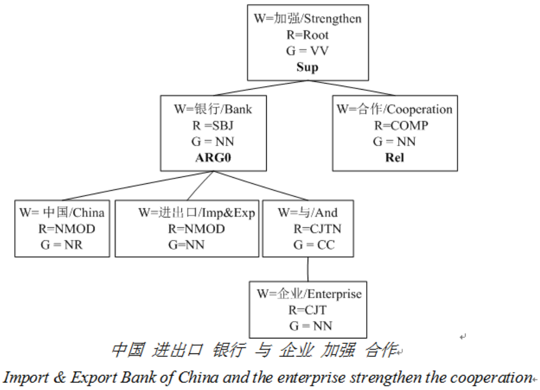


Fig. 1. Example of DPT structure with nominal predicate and its related arguments annotated

3.1 D-CPT

Just as described in the introduction, both DPT and CPT have their own advantages. The new D-CPT structure benefits from the advantages of both DPT and CPT since

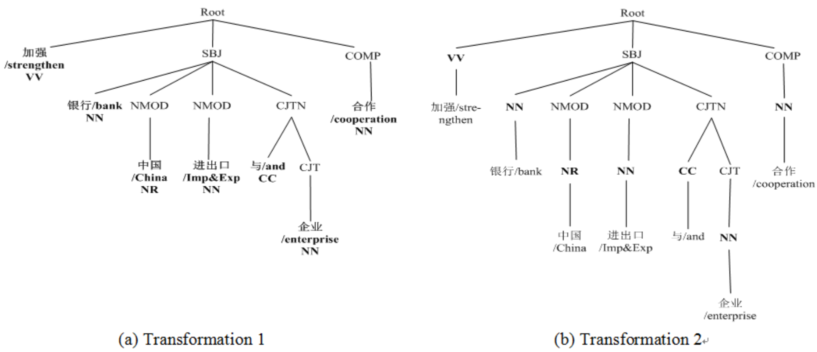


Fig. 2. Example of achieving D-CPT structure from DPT structure

D-CPT not only keeps the dependency relationship information in DPT but also retains the basic structure of CPT. This is done by transforming the DPT structure to a new CPT-style structure, using dependency types instead of phrase labels in the traditional CPT structure. 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.
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).

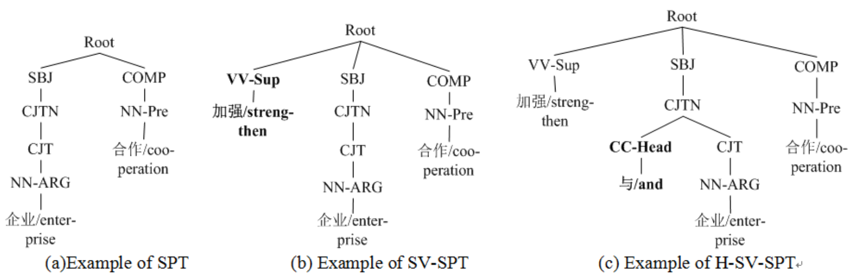
3.2 Extraction Schemes

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 encodes the argument structure of lexical units populated at their nodes through corresponding dependency relations.

1) 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".



(a)Example of SPT (b) Example of SV-SPT (c) Example of H-SV-SPT

Fig. 3. Extraction schemes

2) 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 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.

3) 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.

3.3 Kernels

Given a parse tree structure, this paper employs the well-known convolution tree kernel [13] to compute the similarity between two parse trees. In principle, the convolution tree kernel works by counting the number of common sub-trees as the syntactic

Table 1. Features explored in the feature-based linear kernel

Feature   Remarks (Feature instance with regard to Fig. 1)
Dependent word and its POS tag: the modifying word and its POS tag in the dependency relation. (企业/enterprise, NN)
Dependency relation type: the type of the dependency relation. (CJT)
Predicate word and its POS tag: the current predicate and its POS tag. (合作/ cooperation, NN)
Head word and its POS tag: the modified (head) word and its POS tag in the dependency relation. (与/and, CC)
DepSubCat: the subcategorization frame of the predicate. (COMP->-)
DeprelPath: the path from predicate to argument concatenating dependency labels with the direction of the edge. (CJT ↑ CJTN ↑ SBJ ↑ ROOT ↓ COMP)
POSPath: same as DeprelPath, but dependency labels are exchanged for POS tags. (NN ↑ CC ↑ NN ↑ VV ↓ NN)
Family membership: indicating how the dependency relation is related to the predicate in the family. (siblings' grandchildren)
ChildDepSet: the set of dependency labels of the children of the predicate. (None)
ChildPOSSet: the set of POS tags of the children of the predicate. (None)
SiblingDepSet: the set of dependency labels of the siblings of the predicate. (SBJ)
SiblingPOSSet: the set of POS tags of the siblings of the predicate. (NN)
Position: the position of the argument with respect to the predicate. (before)

similarity between two parse trees. Thus, this tree kernel implicitly defines a large feature space.

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 proven effective in the literature [8]. In particular, our composite kernel is combined by linearly interpolating a convolution tree kernel KT over a parse tree structure and a feature-based linear kernel KL as follow:

$$CK = \alpha \cdot K_L + (1\text{-}\alpha) \cdot K_T \qquad \text{where } \alpha \text{ is a coefficient for } K_L$$

Table 1 shows a list of features in the feature-based linear kernel extracted from the DPT structure. Here, we only select those 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.

4 Experimentation

4.1 Experimental Setting

Following the experimental setting in Xue [11] and Li et al [12], 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) as the development data, with 8642, 1124, and 731 propositions, respectively.

To save training time, we use a simple pruning strategy to filter out the dependency nodes that are less likely to be semantic arguments to the predicate according to the specific characteristics of Chinese NomBank. In particular, given the nominal predicate as the current node, we only keep its father, grandfather, grandfather's siblings, grandfather's children, siblings, siblings' children, siblings' grandchildren, children, grandchildren with respect to the DPT 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. After pruning, we first do argument identification for those remaining candidates, and then classify the positive ones into their corresponding semantic roles.

We use the SVM-light toolkit with the convolution tree kernel function SVMlight–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 vs. others strategy to implement multi-class classification, which builds multiple classifiers so as to separate one class from all 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 [10]. 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, '>>>', '>>', and '>'

denote p-values less than or equal to 0.01, in-between (0.01, 0.05], and bigger than 0.05, respectively.

4.2 Experimental Results on Golden Parse Trees

Table 2 shows the performance of our tree-kernel-based method using different extraction schemes on the D-CPT structure. Here, the golden CPT structure is converted into the DPT structure using the same conversion toolkit as adopted by the CoNLL-2009 shared task.

Table 2. Performance of our tree-kernel-based method using different extraction schemes on the D-CPT structure of golden parse trees

Scheme	P(%)	R(%)	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 2 shows that:

1) SPT achieves the performance of 65.98 in F1-measure with a much lower recall of only 58.26%, compared to 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-mesure. This means that SV-SPT performs significantly better than SPT by 3.91 (>>>) 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. Comparison of different kernels on golden parse trees

Kernel	P(%)	R(%)	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**

Table 3 illustrates the performance comparison with different kernel setups on golden 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 (>>). 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 that the combination of them via a simple composite kernel improves the performance by 4.72 (>>>) and 2.97(>>>) in F1-measure over the feature-based linear kernel and the tree kernel.

4.3 Experimental Results on Automatic Parse Trees

In previous subsection, we assume the availability of golden 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. Table 4 and Table 5 present the performance on automatic parse trees.

Table 4. Performance of our tree-kernel-based method using different extraction schemes on the D-CPT structure of automatic parse trees

Scheme	P(%)	R(%)	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 5. Comparison of different kernels on automatic parse trees

Kernel	P(%)	R(%)	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 4 and Table 5 show that:

1) For each extraction scheme on D-CPT of automatic parse trees, our tree kernel-based method shows the performance tendency similar to golden 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 golden parse trees. This indicates the dependency of Chinese nominal SRL on the performance of syntactic parsing.

4.4 Comparison with Other Tree Kernel-Based Methods on DPT Structure

Nguyen et al [15] propose 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 6. Comparison with other tree kernel-based on DPT structure

structure	P(%)	R(%)	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

Table 6 compares our D-CPT structure with the DW, GR and GRW trees on Chinese nominal SRL, using the same convolution tree kernel on golden parse trees. Table 6 shows that even SPT, extracted from the D-CPT using the simplest scheme, significantly outperforms the GR (>>>), DW (>>>) and GRW (>>>) 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.

4.5 Comparison with Other Systems

Finally, Table 7 compares our proposed method with the state-of-the-art ones on Chinese NomBank, Xue [11] and Li et al [12]. Both of them are feature-based ones with various features derived from the CPT structure via extensive feature engineering.

Table 7 shows that our tree kernel-based method achieves comparable performance with the state-of-the-art feature-based ones on either golden 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 propose 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 7. Comparison to the state-of-the-art systems

System	Golden (F1)	Auto (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[11]:feature-based	69.6	57.60
Li et al [12]:feature-based	70.63	58.66

4.6 Experimentation on the CoNLL-2009 Chinese Corpus

To further illustrate the effectiveness of the novel DR-CPT structure for better representation of dependency relations in tree kernel-based methods, we also do the experimentation on the CoNLL-2009 Chinese corpus.

Table 8. Performance of our tree-kernel-based 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[20]:feature-based	78.60
Meza-Ruiz and Riedel[21]:feature-based	77.73

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 [20], who obtained the best F1 score on the Chinese corpus.

Table 8 compares the performance of different kernel setups on the CoNLL-2009 Chinese corpus. It shows that:

1) Our tree-kernel method achieves comparable performance with Meza-Ruiz and Riedel [21], who obtained the second best performance on the Chinese Corpus. It further denotes the effectiveness of our DR-CPT structure in a tree kernel-based method on SRL of verbal predicates.
2) Our composite kernel (without global re-ranking) achieves comparable performance with Bjorkelund et al [20], who employed a global re-ranking strategy and obtained the best performance on the Chinese Corpus.

5 Conclusion and Future Work

This paper systematically explores a tree kernel-based method on a novel D-CPT structure, which employs dependency types instead of phrase labels in the traditional CPT structure for nominal SRL. In particular, we propose 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 structure of CPT. Furthermore, several extraction schemes are designed to extract various kinds of necessary information for nominal SRL and verbal SRL (CoNLL-2009 corpus). Evaluation shows the effectiveness of D-CPT both on Chinese NomBank and CoNLL-2009 corpus.

To our knowledge, this is the first research on tree kernel-based SRL on effectively exploring dependency relationship information, which achieves comparable performance with the state-of-the-art feature-based ones.

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

Acknowledgements. This research is supported by Project BK2011282 under the Natural Science Foundation of Jiangsu Province, Project 10KJB520016 and key Project 11KJA520003 under the Natural Science Foundation of Jiangsu Provincial Department of Education.

Reference

1. Surdeanu, M., Harabagiu, S., Williams, J., Aarseth, P.: Using Predicate-argument Structures for Information Extraction. In: Proceedings of NAACL 2004, pp. 8–15 (2004)
2. Narayanan, S., Harabagiu, S.: Question Answering based on Semantic Structures. In: Proceedings of COLING 2004, Article No. 693 (2004)
3. Ponzetto, S.P., Strube, M.: Semantic Role labeling for Coreference Resolution. In: Proceedings of EACL 2006, pp. 143–146 (2006)
4. Persson, J., Johansson, R., Nugues, P.: Text Categorization Using Predicate-Argument Structures. In: Proceedings of NODALIDA 2009, pp. 142–149 (2009)
5. Moschitti, A., Bejan, C.A.: A Semantic Kernel for Predicate Argument Classification. In: Proceedings of CoNLL 2004, pp. 17–24 (2004)
6. Moschitti, A.: A Study on Convolution Kernels for Shallow Semantic Parsing. In: Proceedings of ACL 2004, Article No. 335 (2004)
7. Moschitti, A., Pighin, D., Basili, R.: Tree Kernel Engineering in Semantic Role Labeling Systems. In: Proceedings of EACL 2006, pp. 49–56 (2006)
8. 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: Proceedings of ACL 2007, pp. 200–207 (2007)
9. Hacioglu, K.: Semantic Role Labeling Using Dependency Trees. In: Proceedings of COLING 2004, Article No. 1273 (2004)
10. Johansson, R., Nugues, P.: Dependency-based Semantic Role Labeling of PropBank. In: Proceedings of EMNLP 2008, pp. 69–78 (2008)
11. Xue, N.W.: Labeling Chinese Predicates with Semantic Roles. Computational Linguistics 34(2), 225–255
12. 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: Proceedings of EMNLP 2009, pp. 1280–1288 (2009)
13. Collins, M., Duffy, N.: Convolution Kernels for Natural Language. In: Proceedings of NIPS 2001 (2001)
14. Che, W.X., Zhang, M., Liu, T., Li, S.: A Hybrid Convolution Tree Kernel for Semantic Role Labeling. In: Proceedings of ACL 2006, pp. 73–80 (2006)

15. Nguyen, T.T., Moschitti, A., Riccardi, G.: Convolution Kernels on Constituent, Dependency and Sequential Structures for Relation Extraction. In: Proceedings of ACL 2009, pp. 1378–1387 (2009)
16. Kong, F., Zhou, G.D., Qian, L.H., Zhu, Q.M.: Dependency-driven Anaphoricity Determination for Coreference Resolution. In: Proceedings of COLING 2010, pp. 599–607 (2010)
17. Xue, N.W., Palmer, M.: Automatic Semantic Role Labeling for Chinese verbs. In: Proceedings of IJCAI 2005, pp. 1160–1165 (2005)
18. Xue, N.W.: Semantic Role Labeling of Nominalized Predicates in Chinese. In: Proceedings of HLT-NAACL, pp. 431–438 (2006)
19. 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: Proceedings of COLING 2010, pp. 665–673 (2010)
20. Bjorkelund, A., Hafdell, L., Nugues, P.: Multilingual Semantic Role Labeling. In: Proceedings of CoNLL 2009, pp. 43–48 (2009)
21. Meza-Ruiz, I., Riedel, S.: Multilingual Semantic Role Labeling with Markov Logic. In: Proceedings of CoNLL 2009, pp. 85–90 (2009)