

# A Global Generative Model for Chinese Semantic Role Labeling

Haitong Yang and Chengqing Zong

National Laboratory of Pattern Recognition,  
Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China  
{htyang, cqzong}@nlpr.ia.ac.cn

**Abstract.** The predicate and its semantic roles compose a unified entity that conveys the semantics of a given sentence. A standard pipeline of current approaches to semantic role labeling (SRL) is that for a given predicate in a sentence, we can extract features for each candidate argument and then perform the role classification through a classifier. However, this process totally ignores the integrality of the predicate and its semantic roles. To address this problem, we present a global generative model in which a novel concept called Predicate-Arguments-Coalition (PAC) is proposed to encode the relations among individual arguments. Owing to PAC, our model can effectively mine the inherent properties of predicates and obtain a globally consistent solution for SRL. We conduct experiments on the standard benchmarks: Chinese PropBank. Experimental results on a single syntactic tree show that our model outperforms the state-of-the-art methods.

**Keywords:** global generative model, semantic role labeling, Predicate-Arguments-Coalition (PAC).

## 1 Introduction

Semantic Role Labeling (SRL) is a kind of shallow semantic parsing task and its goal is to recognize some related phrases and assign a joint structure (WHO did WHAT to WHOM, WHEN, WHERE, WHY, HOW) to each predicate of a sentence[5]. Because of its ability to encode semantic information, SRL has been applied in many tasks of NLP, such as question and answering[13], information extraction[3, 15] and machine translation[9, 21, 23, 28].

Since the release of FrameNet[1] and PropBank[7, 24], there has been a large amount of work on SRL[5, 10, 11, 18, 20, 22, 25, 26, 29]. When labeling the candidate arguments, a common pipeline schema works as: first extract features from a syntax tree and then independently accomplish the classification for each candidate. An implicit fact behind the process is that there is no interaction among the candidate arguments. However, from linguistic intuition this is not appropriate because an arguments frame of a predicate is a joint structure, with strong dependencies between arguments[19]. For example, if ARG0 is assigned to one argument, then the other arguments are not allowed to be classified into ARG0.

To address the argument dependencies in SRL, there has been some work[6, 19]. Their fundamental view is that the predicate argument structure is a sequence structure. In these approaches, they usually introduce label sequence features into the original model in order to capture the global properties of the arguments. They reported better performance than the original model, which does not consider the argument dependencies.

Different from the above viewpoints, we attempted to directly obtain a global structure for SRL. We introduce a novel concept called Predicate-Argument-Coalition (PAC) to describe the global structure of the predicate and arguments. PAC can naturally catch many linguistic phenomena about the predicates. Based on PAC, we propose a global generative model (GGM) that could obtain a globally consistent structure for SRL directly. Our model works according to the following schema:

- First, we train a base local classifier;
- Second, one binary classifier is trained to distinguish the core arguments from adjunction arguments;
- Third, traverse all of the possible PAC candidates, and then the candidate with the highest score is selected.

The PAC of the predicates could be considered to be one type of prior knowledge about the given predicate and it provides essential assistance for SRL. Our experimental results on Chinese PropBank show that GGM significantly outperforms the state-of-the-art systems and the relative error declines 13.8% compared with the baseline model. Furthermore, after a new feature Word Semantic Class is added, GGM achieves approximately one point of F1 score in improvement.

The remainder of this paper is organized as follows. Section 2 discusses the related work. Section 3 presents our novel concept Predicate-Argument-Coalition. Section 4 presents a standard local classifier as the baseline. Our generative model GGM is described and formulated in Section 5. The experiments and results are presented in Section 6. The conclusions can be found in Section 7.

## 2 Related Work

Since the pioneering work of [5], there has been a large number of studies on automatic semantic role labeling. A variety of approaches[2, 11, 12, 16, 17, 27] have been successfully applied in SRL. Here we give a brief introduction to some typical studies. [25] takes a critical look at features of arguments detection and arguments classification. [17] discusses Chinese semantic role labeling with shallow parsing and [8] explores the joint syntactic and semantic parsing of Chinese to further improve the performance of both syntactic parsing and SRL.

There are some studies devoted to utilizing the global information of arguments and predicates to improve SRL. [14] constructed an Integer Linear Programming architecture in which the dependency relations among the arguments are implied by the constraints. They summarized ten categories of constraint conditions in which approximately one half of the categories are defined to describe the mutual dependences among the candidate arguments. However, one drawback of their system

is that some constraints must be provided by hand. For example, the 10th constraint of their system is that given the predicate, some argument classes are illegal. To provide the constraint, all predicates must be checked as to whether every argument class is legal for the predicates.

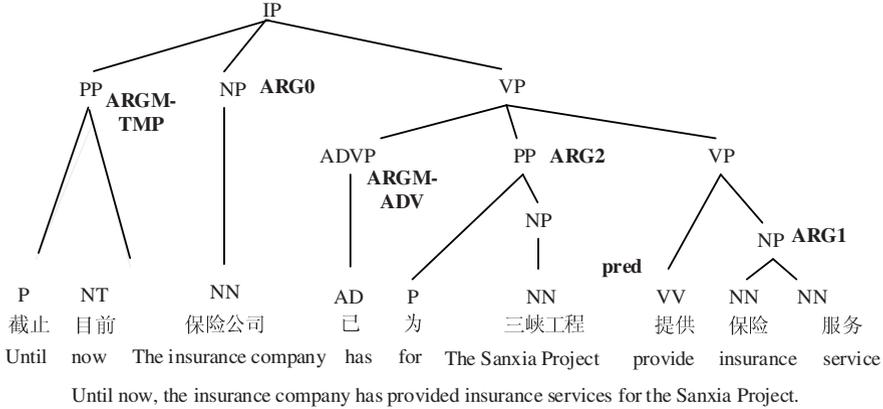
[19] proposes a joint model to explore the global information in the arguments. They first build a local model, then use the local model to generate the  $n$  most likely joint assignments for a sentence, and finally rerank these the  $n$  joint assignments through a joint model in which many global features are defined. In their model, almost all of the global features are expressed in the argument sequence's way such as using whole label sequence features, because in their view this type of sequence structure could handle the structure of the semantic roles. Their experiment results supported their viewpoint. However, we have different opinions about the structure of semantic roles. Unfortunately, according to our investigation, in some cases the change in the position of some particular arguments will not cause a change in the semantic meaning. For example, “明天 我 回去”(Tomorrow I will go back) and “我 明天 回去”(I will go back tomorrow) have the exact meaning and the same semantic roles but their label sequences are different. Therefore, it is debatable to use an ordered sequence structure to express the predicate-argument structure. In addition, the sequence structure is very sparse because there are over 20 types of labels in Prop-Bank. If the sequence length is  $n$ , the number of all possible sequences will be  $20n$ . In the model of [19], there are many lexicalization features that aggravate the sparsity.

It is also worth noting that the predicate is dominant in the predicate-argument structure and all arguments serve the predicate. However, in almost all of the existing approaches, the predicate is treated as only one feature of the discriminative model. In our opinion, every predicate has its own characteristics, and we should mine the intrinsic properties of the predicates to help SRL.

### 3 Predicate-Argument Structure

In the SRL community, the predicate-argument structure is thought to be a composite structure of the predicate and arguments. Figure 1 shows an example for the predicate-argument structure. In the sentence shown, the predicate is ‘提供’ (provide) and is labeled as “pred”. And there are five arguments for ‘提供’ (provide); three of these arguments are core arguments, which are labeled as ARG + a number, and the other two are adjunction arguments which are labeled as ARGM + an adjunction word. These two types of labels have different functions in conveying the meanings of the sentence. The core arguments are essential to constitute the skeleton of the sentence, while the adjunction arguments provide additional information for the predicate, such as Time, Location and so on.

However, there is still not an explicit formulation for representing the predicate-argument structure. In this paper, we propose a novel concept named Predicate-Argument-Coalition (PAC) to represent the predicate-argument structure. And we experimentally demonstrate that PAC is effective to represent the predicate-argument structure.



**Fig. 1.** An example from Chinese PropBank

### 3.1 Predicate-Argument-Coalition

Prediate-Argument-Coalition (PAC) is defined as a three-tuple like below.

$$PAC \triangleq \langle Pred, (ArgX : n), (ArgM : n) \rangle$$

The first term is the predicate; the second term is a multi-set that represents the entirety of the core arguments in which  $n$  is the number for  $ArgX$ , which appears in the sentence; the third term is similar to the second term but it represents the entirety of the adjunction arguments.

To provide a clear description, here is an example for the sentence in Figure 1.

$$\langle \text{提供}, (Arg0 : 1, Arg1 : 1, Arg2 : 1), (ArgM - Tmp : 1, ArgM - Adv : 1) \rangle$$

From the definition of PAC, all of the components of the predicate-argument structure are clearly divided into three parts: the predicate, the core arguments and the adjunction arguments. In PAC, the dependency between the predicate and the arguments is implied in  $n$ . The reason why we separate the core arguments from the adjunction arguments is that their functions in the sentence are different and their relationships to the predicate are also different. In fact, in the optimizing process, our model first acquires an optimal solution for the core part and the adjunction part respectively, and then combines the two parts into a whole one, which is the optimal solution for SRL.

### 3.2 Expression Ability of PAC

An eligible concept should embody many linguistic phenomena that are related to the predicate-argument structure. We declare that the Predicate-Argument-Coalition is easy and adequate to express some complex linguistic phenomena. Let us take the examples below to show the expression ability of PAC.

- a) 百分之九十五的 产品 销往 海外。  
95% of the products were sold to abroad.
- b) 经营 成本 降到 最低。  
Operation cost dropped to the minimum
- c) 国际油价 暴跌。  
The global price of oil dropped sharply.
- d) 1994 年 墨西哥 金融 危机 爆发。  
In 1994, Mexico's financial crisis broke out.

A common constraint in SRL system is no duplication for core arguments. To satisfy the constraint, the only necessary measure in PAC is that all of the numbers  $n$  in the ARGX multi-set are not allowed larger than one. In example (a), the predicate is ‘销往’ (be sold to) and according to the literal meaning of ‘销往’ (be sold to), there must be a “location” that indicates the place where products are sold to. Under this circumstance, the “location” is labeled ARG2. Thus, when the word ‘销往’ (be sold to) appears in a sentence, there must be a candidate that is labeled as ARG2. To handle this linguistic phenomenon, we only ensure that the  $n$  of ARG2 in the ARGX multi-set equals one. In examples (c) and (d), the predicates are intransitive. We can see that there is usually ARG0 or ARG1. We just keep the mutual exclusion of ARG0 and ARG1 in the ARGX multi-set. In other words, ARG0 and ARG1 never simultaneously emerge in a sentence with intransitive verbs. In this situation, we just keep the mutual exclusion of ARG0 and ARG1 in the ARGX multi-set.

The above examples and analysis prove much dependency between arguments and the predicate again, and PAC is easily used to represent the dependency. More importantly, we can obtain all predicates’ PAC statistics from training data without much labor.

## 4 Local Model

Following the traditional pipeline, we divide the SRL task into two phases – argument identification and argument classification. Before argument identification, candidate argument pruning is necessary due to a large number of candidates. Here, we implement a state-of-the-art pruning method as in Xue (2008).

### 4.1 Classifier

We adopt a discriminative model - Maximum Entropy model (ME), as our classifier because ME can easily be expanded to incorporate arbitrary features as a discriminative model.

### 4.2 Features

We need to train two classifiers: one for arguments identification and the other for arguments classification. The following lists the features that are utilized in the two classifiers.

The features used in arguments identification:

- Lexicalization features include the head word, the predicate and the predicate’s verb class.
- Syntactic features include the head word’s part-of-speech (POS), the path from the candidate argument to the predicate, and the path from the candidate argument to BA and BEI (Xue, 2008).
- Combination features include the predicate + head word, the predicate + syntactic tag of the candidate argument, the predicate’s verb class + head word, the predicate’s verb class + the syntactic tag of the candidate argument.

All of the above features are contained in the arguments classification. In addition, there are some other features:

- Position: the relative position of the candidate argument to the predicate.
- Subcat frame: the syntactic rule that expands the parent of the verb
- Phrase type: the syntactic tag of the candidate argument
- The first and the last word of the candidate argument
- Subcat frame+: the frame that consists of the NPs (Xue, 2008).

## 5 Proposed Global Generative Model

In Section 3, we propose a compact representation PAC for the predicate-argument structure. Based on PAC, we construct a generative model for SRL in this section.

### 5.1 Core Arguments and Free Arguments

Before our model’s formulation, we take a second look at the arguments which are divided into the core arguments and the adjunction arguments. Here, we investigate their relationships with the predicate.

**Core Arguments:** These arguments are obviously sensitive with respect to the predicate. For example, the phrase ‘保險公司’ (the insurance company) in Figure 1 is a core argument. If the predicate ‘提供’ (provide) is not given, then it is difficult to label the argument ARG0.

**Adjunction Arguments:** In Figure 1, there are two adjunction arguments. The first argument is the phrase ‘截止目前’ (until now), which has the label ARGM-TMP, which manifests the time of the word ‘提供’ (provide); the second argument is the word ‘已’ (has), which has the label ARGM-ADV, which is an adverbial modifier of the word ‘提供’ (provide). For these two arguments, even though we do not know the predicate, we can classify the two arguments into ARGM-TMP and ARGM-ADV.

Based on the above observation, we make an assumption that adjunction arguments are independent from the predicate and we prefer calling these arguments Free Arguments (in the remaining parts, we keep calling them free arguments). Moreover, the independence assumption of the adjunction arguments is also helpful for simplifying our model.

## Separate the Core Arguments from Free Arguments

Here we also implement an ME model to accomplish binary classification. The features include all of the argument classification features in the local model, and there are some additional features:

- The syntactic tag of the predicate's parent
- The syntactic tag of the candidate argument's parent
- The syntactic tag of the predicate

### 5.2 Formulations

Our model's generative story is described as follows:

- 1) Generate a candidate PAC.
- 2) For a candidate PAC, one solution is obtained by assigning every label of the core-multiset and free-multiset of PAC to candidate arguments
- 3) Repeat (1) and (2).

We take SRL as the problem with structured output. The above generative process can be formulated as.

$$\begin{aligned} structure^* &= \arg \max P(structure \mid Cand, pred) \\ &= \arg \max P(structure \mid Cand, PAC)P(PAC \mid pred) \end{aligned}$$

in which,  $Cand$  stands for all candidate arguments and consists of core arguments  $Cand_{core}$  and free arguments  $Cand_{free}$ ;  $pred$  stands for the predicate.

Due to free arguments' independence from the predicate, we use the classifier described in subsection 5.1 to separate core arguments from free argument. And then the deduction below is obtained:

$$\begin{aligned} structure^* &= \arg \max P(structure \mid PAC, Cand_{free})P(structure \mid PAC, Cand_{core})P(PAC \mid pred) \\ &= \arg \max P(structure_{free} \mid PAC_{free}, Cand_{free})P(structure_{core} \mid PAC_{core}, Cand_{core}) \\ &\quad P(PAC_{core}, PAC_{free} \mid pred) \\ &= \arg \max P(structure_{free} \mid PAC_{free}, Cand_{free})P(structure_{core} \mid PAC_{core}, Cand_{core}) \\ &\quad P(PAC_{core} \mid pred) \end{aligned}$$

In the above deduction, besides separating core candidates from free candidates, we also factor structure into  $structure_{core}$  and  $structure_{free}$ ,  $PAC$  into  $PAC_{core}$  and  $PAC_{free}$ .

### 5.3 Inference

There are three parts in the ultimate formulation. Here we give detailed description about how to infer the optimal solution.

The first part is to obtain  $structure_{free}$  of the free arguments. Because there is no constraint about the free arguments in the optimization process, the optimization solution for  $structure_{free}$  is exactly the same as that in the local model.

The second part is  $P(structure_{core}|PAC_{core}, Cand_{core})$  is the probability of obtaining  $structure_{free}$  given  $PAC_{core}$  and  $Cand_{core}$  and also reflects the probability of assigning one tag of  $PAC_{core}$  to one candidate of  $Cand_{core}$  without repetition. Since the  $PAC_{core}$  is a multi-set, we need to traverse all of the possible tag sequences to obtain the maximum. The detailed computation is given as follows:

$$P(structure_{core} | PAC_{core}, Cand_{core}) = \max_{\text{all sequences of } PAC_{core}} \prod p(Arg_i | cand_i)$$

in which  $p(Arg_i | cand_i)$  is the probability of assigning the  $i$ -th tag of the sequence to the  $i$ -th candidate argument according to the local model.

The third part is  $P(PAC_{core} | pred)$  is the probability of a candidate multi-set  $PAC_{core}$  given the predicate  $pred$ . The maximum likelihood estimation for  $P(PAC_{core} | pred)$  can be computed by the following equation:

$$\begin{aligned} P(PAC_{core} | pred) \\ = \frac{\text{count}(pred, PAC_{core})}{\sum \text{count}(pred, PAC_{core}')} \end{aligned}$$

In inferring  $structure_{core}$ , the second part stands for the local property while the third part reflects the inherent prior property of the predicate. The two parts provide a solution for  $structure_{core}$ . In summary, we obtain  $structure_{free}$  through the first part and  $structure_{core}$  through the second and the third part. After  $structure_{free}$  and  $structure_{core}$  are obtained, the two parts constitute a whole structure and the optimal  $structure^*$  is our solution.

## 6 Experiments

### 6.1 Experiments Setup

We use Chinese Proposition Bank 1.0 in this experiment. According to the traditional division (Xue, 2008; Sun et al., 2009), all of the data are divided into three parts. 648 files (from chtb\_081.fid to chtb\_899.fid) are used as the training set. The second part includes 40 files from chtb\_041.fid to chtb\_080.fid as the development set. The test set is 72 files, which are chtb\_001.fid to chtb\_040.fid and chtb\_900.fid to chtb\_931.fid. We adopt the Berkeley parser to perform auto parsing for SRL and re-train the parser on the training set.

## 6.2 Results

We have compared our GGM with the local model in Table 1 and the evaluation criterion is F1. From Table 1, we can see that for the core arguments, our model significantly outperforms the local model by approximately 0.8 points. For free arguments, GGM’s score is higher than the baseline, which benefits from core and free arguments separation stage. The overall performance has been improved from 74.04 to 74.73.

To provide a further description, Table 2 lists the detailed numbers that are related to the F1 value. “False” means the number of null arguments that are distinguished as arguments. “Miss” means the number of missing arguments. “Right” and “Error” means the numbers of arguments that are classified correct and wrong respectively. Because both GGM and the local model take the same pruning and argument identification steps, the “False” number and the “Miss” number are the same. However, GGM’s error number declines by 13.8% compared with the local model.

**Table 1.** Comparison with the Local Model

	Num	Local	GGM
ARG0	2023	67.55	<b>68.35</b>
ARG1	2649	78.63	<b>79.43</b>
ARG2	359	62.79	65.41
ARG3	28	50.00	55.32
ARG4	5	54.55	72.73
ARGM	3023	74.81	<b>75.10</b>
all	8432	74.04	<b>74.73</b>

**Table 2.** Comparison on detailed “False”, “Right”, “Error” and “Miss” numbers with the Local Model

Method	False	Right	Error	Miss
Local	1159	5897	<b>377</b>	2208
GGM	1159	5932	<b>322</b>	2208

## 6.3 Advanced Features for Free Arguments

It is noted that in Table 1, the performance of free arguments is not improved as core arguments since our GGM focuses on core arguments. In [4], they thought that different features were necessary to capture the crucial information for classifying the core arguments and the free arguments. In their system, they first discriminate as to whether arguments belong to either core arguments or free arguments by a binary classifier, and then, they label core candidates and free candidates with different features. Motivated by their approach, we define a new feature called Word Semantic Class (WSC) for classifying free arguments.

In the ArgM-TMP arguments of PropBank, there must be a word that means the time and for the ArgM-LOC arguments, there must be a word that means the location. Because these words are usually nouns, we could easily extract the time and location words from the corpus, and these are two types of Word Semantic Classes. For the other free arguments, there are some exclusive words. For example, in ArgM-Dir, there is always a preposition that indicates a direction such as ‘朝着’ ( face to ). Moreover, these words often appear at the beginning of the argument. Thus, we can extract these exclusive words according to the first word of the arguments. Some typical examples of WSC are shown in Table 3.

**Table 3.** Examples of Word Semantic Class

	WSC
春天 (spring)	TMP
埃及 (Egypt)	LOC
为了 (to/for)	PRP
朝着 (face to)	DIR
但是 (but)	DIS
和 (and)	CRD
根据 (accord to)	MNR
即使 (although)	CND

We have evaluated the effect of the WSC feature for SRL. The results are shown in Table 4. We divide the local model’s classification features into two types: base features and pred features. The pred features are related to the predicate and the base features are the other features. We can see that if we only use the base features, the performance drops to 71.86 from 74.04. After the WSC features are added, the performance rises to 74.94 sharply. After we add the pred features into the model, the performance is improved slightly.

**Table 4.** Evaluation Results on the Feature WSC

	Base	+WSC	+Pred
GGM	71.86	74.94	75.01

## 6.4 Comparison with Other Methods

We also compared GGM with other methods and results are shown in Table 5. It can be seen that the approaches that incorporate argument dependencies (Toutanova et al., 2008; GGM; GGM+) are better than others. Meanwhile, our approach outperforms the state-of-the-art method by 0.5 F1 points. It is generally believed that Chinese language generation is more complex and arbitrary than English. Therefore, the sequence structure in Toutanova’s joint model is not adequate to catch the global properties of the candidate arguments while our PAC can handle these in an effective way.

**Table 5.** Comparison with Other Methods. GGM+ means adding the new feature WSC into GGM

Methods	F <sub>1</sub>
Xue(2008)	71.9
Sun(2009)	74.12
Toutanova(2008)	74.50
GGM	<b>74.73</b>
GGM+	<b>75.01</b>

## 7 Conclusion and Future Work

In this paper, to address the argument dependencies in SRL, we propose a global generative model in which a novel concept Predicate-Argument-Coalition is defined to represent the predicate-argument structure. The existing approaches treat the predicate-argument structure as a sequence structure but in some languages such as Chinese, the sequence structure is not adequate to describe a complex predicate-argument structure. Unlike them, our model can effectively mine the inherent properties of the predicates through PAC, which is helpful for SRL. Experiment results on Chinese PropBank and English PropBank demonstrate the superiority of our approach.

**Acknowledgments.** We thank the three anonymous reviewers for their helpful comments and suggestions. The research work has been partially funded by the Natural Science Foundation of China under Grant No.61333018 and the International Science & Technology Cooperation Program of China under Grant No.2014DFA11350, and also the High New Technology Research and Development Program of Xinjiang Uyghur Autonomous Region under Grant No.201312103 as well.

## References

1. Baker, C., Fillmore, C., Lowe, J.: The berkeley framenet project. In: Proceedings of COLING-ACL 1998 (1998)
2. Cohn, T., Blunsom, P.: Semantic Role Labeling with Tree Conditional Random Fields. In: Proceedings of CONLL 2005 (2005)
3. Christensen, J., Mausam, Soderland, S., Etzioni, O.: Semantic Role Labeling for Open Information Extraction. In: Proceedings of ACL 2010 (2010)
4. Ding, W., Chang, B.: Improving Chinese Semantic Role Classification with Hierarchical Feature Selection Strategy. In: Proceedings of EMNLP 2008 (2008)
5. Gildea, D., Jurafsky, D.: Automatic labeling for semantic roles. *Computational Linguistics* 28(3), 245–288 (2002)
6. Jiang, Z., Li, J., Ng, H.T.: Semantic Argument Classification Exploiting Argument Interdependence. In: Proceedings of IJCAI 2005 (2005)
7. Kingsbury, P., Palmer, M.: From Treebank to PropBank. In: Proceedings of LREC 2002 (2002)

8. Li, J., Zhou, G., Ng, H.T.: Joint Syntactic and Semantic Parsing of Chinese. In: Proceedings of ACL 2010 (2010)
9. Liu, D., Gildea, D.: Semantic role features for machine translation. In: Proceedings of COLING 2010 (2006)
10. Moschitti, A., Pighin, D., Basili, R.: Semantic role labeling via tree kernel joint inference. In: Proceedings of CONLL 2006 (2006)
11. Moschitti, A., Pighin, D., Basili, R.: Tree Kernels for Semantic Role Labeling. *Computational Linguistics* 34(2), 193–224 (2008)
12. Mitsumori, T., Murata, M., Fukuda, Y., Doi, K., Doi, H.: Semantic Role Labeling Using Support Vector Machines. In: Proceedings of CONLL 2005 (2005)
13. Narayanan, S., Harabagiu, S.: Question Answering based on Semantic Structures. In: Proceedings of COLING 2004 (2004)
14. Punyakanok, V., Roth, D., Yih, W., Zimak, D.: Semantic Role Labeling via Integer Linear Programming Inference. In: Proceedings of COLING 2004 (2004)
15. Surdeanu, M., Harabagiu, S., Williams, J., Aarseth, P.: Using Predicate-Argument Structures for Information Extraction. In: Proceedings of ACL 2003 (2003)
16. Surdeanu, M., Turmo, J.: Semantic Role Labeling Using Complete Syntactic Analysis. In: Proceedings of CONLL 2005 (2005)
17. Sun, W., Sui, Z., Wang, M., Wang, X.: Chinese Semantic Role Labeling with Shallow Parsing. In: Proceedings of ACL 2009(2009)
18. Toutanova, K., Haghighi, A., Manning, C.D.: Joint learning improves semantic role labeling. In: Proceedings of ACL 2005(2005)
19. Toutanova, K., Haghighi, A., Manning, C.D.: A Global Joint Model for Semantic Role Labeling. *Computational Linguistics* 34(2), 161–191 (2008)
20. Tsai, T.-H., Wu, C.-W., Lin, Y.-C., Hsu, W.-L.: Exploiting Full Parsing Information to Label Semantic Roles Using an Ensemble of ME and SVM via Integer Linear Programming. In: Proceedings of CONLL 2005 (2005)
21. Wu, D., Fung, P.: Can semantic role labeling improve smt. In: Proceedings of EAMT 2009 (2009)
22. Wu, S., Palmer, M.: Semantic mapping using automatic word alignment and semantic role labeling. In: Proceedings of the Fifth Workshop on Syntax, Semantics and Structure in Statistical Translation (2011)
23. Deyi, X., Min, Z., Haizhou, L.: Modeling the Translation of Predicate-Argument Structure for SMT. In: Proceedings of ACL 2012 (2012)
24. Xue, N., Palmer, M.: Annotating the Propositions in the Penn Chinese Treebank. In: The Proceedings of the 2nd SIGHAN Workshop on Chinese Language Processing (2003)
25. Xue, N., Palmer, M.: Calibrating Features for Semantic Role Labeling. In: Proceedings of EMNLP 2004 (2004)
26. Xue, N.: Labeling Chinese Predicates with Semantic Roles. *Computational Linguistics* 34(2), 225–255 (2008)
27. Yi, S., Palmer, M.: The Integration of Syntactic Parsing and Semantic Role Labeling. In: Proceedings of CONLL 2005 (2005)
28. Zhai, F., Zhang, J., Zhou, Y., Zong, C.: Machine Translation by Modeling Predicate-Argument Structure Transformation. In: Proceedings of COLING 2012 (2012)
29. Tao, Z., Zong, C.: A Minimum Error Weighting Combination Strategy for Chinese Semantic Role Labeling. In: Proceedings of EMNLP 2010 (2010)