

# Jointly Modeling WSD and SRL with Markov Logic

Wanxiang Che and Ting Liu

Research Center for Information Retrieval

MOE-Microsoft Key Laboratory of Natural Language Processing and Speech

School of Computer Science and Technology

{car, tliu}@ir.hit.edu.cn

## Abstract

Semantic role labeling (SRL) and word sense disambiguation (WSD) are two fundamental tasks in natural language processing to find a sentence-level semantic representation. To date, they have mostly been modeled in isolation. However, this approach neglects logical constraints between them. We therefore exploit some pipeline systems which verify the automatic all word sense disambiguation could help the semantic role labeling and vice versa. We further propose a Markov logic model that jointly labels semantic roles and disambiguates all word senses. By evaluating our model on the OntoNotes 3.0 data, we show that this joint approach leads to a higher performance for word sense disambiguation and semantic role labeling than those pipeline approaches.

## 1 Introduction

Semantic role labeling (SRL) and word sense disambiguation (WSD) are two fundamental tasks in natural language processing to find a sentence-level semantic representation. Semantic role labeling aims at identifying the relations between predicates in a sentence and their associated arguments. Word sense disambiguation is the process of identifying the correct meaning, or sense of a word in a given context. For example, for the sentence in Figure 1, we can find out that the predicate token “hitting” at position 3 has sense “cause to move by striking” and the sense label is “hit.01”. The argument headed by the token “cat” at position 1 with sense “feline mammal” (cat.01) is referring to the player (A0), and the argument headed by the token “ball” at position 5 with sense

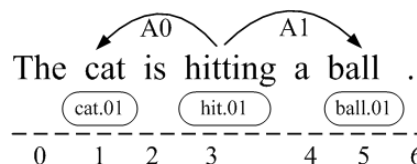


Figure 1: A sample of word sense disambiguation and semantic role labeling.

“round object that is hit in games” (ball.01) is referring to the game object (A1) being hit.

Normally, semantic role labeling and word sense disambiguation are regarded as two independent tasks, i.e., the word sense information is rarely used in a semantic role labeling system and vice versa. A few researchers have used semantic roles to help the verb sense disambiguation (Dang and Palmer, 2005). More people used predicate senses in semantic role labeling (Hajič et al., 2009; Surdeanu et al., 2008). However, both of the pipeline methods ignore possible dependencies between the word senses and semantic roles, and can result in the error propagation problem. The same problem also appears in other natural language processing tasks.

In order to make different natural language processing tasks be able to help each other, jointly modeling methods become popular recently, such as joint Chinese word segmentation and part-of-speech tagging (Kruengkrai et al., 2009; Zhang and Clark, 2008; Jiang et al., 2008), joint lemmatization and part-of-speech prediction (Toutanova and Cherry, 2009), joint morphological segmentation and syntactic parsing (Goldberg and Tsarfaty, 2008), joint text and aspect ratings for sentiment summarization (Titov and McDonald, 2008), and joint parsing and named entity recognition (Finkel and Manning, 2009). For semantic role labeling, Dahlmeier et al. (2009) proposed a method to maximize the joint probability of the seman-

tic role of preposition phrases and the preposition sense.

In order to do better joint learning, a novel statistical relational learning framework, Markov logic (Domingos and Lowd, 2009) was introduced to join semantic role labeling and predicate senses (Meza-Ruiz and Riedel, 2009). Markov logic combines the first order logic and Markov networks, to develop a joint probability model over all related rules. Global constraints (introduced by Punyakanok et al. (2008)) among semantic roles can be easily added into Markov logic. And the more important, the jointly modeling can be realized using Markov logic naturally.

Besides predicates and prepositions, other word senses are also important information for recognizing semantic roles. For example, if we know “cat” is an “agent” of the predicate “hit” in a sentence, we can guess that “dog” can also be an “agent” of “hit”, though it does not appear in the training data. Similarly, the semantic role information can also help to disambiguate word senses. In addition, the predicate sense and the argument sense can also help each other. In the sentence “The cat is hitting a ball.”, if we know “hit” here has a game related sense, we can guess that the “ball” should have the sense “is a round object in games”. In the same way, the correct “ball” sense can help to disambiguate the sense of “hit”. The joint probability, that they are disambiguated correctly simultaneously will be larger than other abnormalities.

The release of OntoNotes (Hovy et al., 2006) provides us an opportunity to jointly model all word senses disambiguation and semantic role labeling. OntoNotes is a large corpus annotated with constituency trees (based on Penn Treebank), predicate argument structures (based on Penn PropBank), all word senses, etc. It has been used in some natural language processing tasks, such as joint parsing and named entity recognition (Finkel and Manning, 2009), and word sense disambiguation (Zhong et al., 2008).

In this paper, we first propose some pipeline systems which exploit automatic all word sense disambiguation into semantic role labeling task and vice versa. Then we present a Markov logic model which can easily express useful global con-

straints and jointly disambiguate all word senses and label semantic roles.

Experiments on the OntoNotes 3.0 corpus show that (1) the automatic all word sense disambiguation and semantic role labeling tasks can help each other when using pipeline approaches, and more important, (2) the joint approach using Markov logic leads to higher accuracy for word sense disambiguation and performance ( $F_1$ ) for semantic role labeling than pipeline approaches.

## 2 Related Work

Joint models were often used in semantic role labeling community. Toutanova et al. (2008) and Punyakanok et al. (2008) presented a re-ranking model and an integer linear programming model respectively to jointly learn a global optimal semantic roles assignment. Besides jointly learning semantic role assignment of different constituents for one task (semantic role labeling), their methods have been used to jointly learn for two tasks (semantic role labeling and syntactic parsing). However, it is easy for the re-ranking model to lose the optimal result, if it is not included in the top  $n$  results. In addition, the integer linear programming model can only use hard constraints. A lot of engineering work is also required in both models.

Recently, Markov logic (Domingos and Lowd, 2009) became a hot framework for joint model. It has been successfully used in temporal relations recognition (Yoshikawa et al., 2009), co-reference resolution (Poon and Domingos, 2008), etc. It is very easy to do joint modeling using Markov logic. The only work is to define relevant formulas. Meza-Ruiz and Riedel (2009) have joined semantic role labeling and predicate senses disambiguation with Markov logic.

The above idea, that the predicate senses and the semantic role labeling can help each other, may be inspired by Hajič et al. (2009), Surdeanu et al. (2008), and Dang and Palmer (2005). They have shown that semantic role features are helpful to disambiguate verb senses and vice versa.

Besides predicate senses, Dahlmeier et al. (2009) proposed a joint model to maximize probability of the preposition senses and the semantic role of prepositional phrases.

Except for predicate and preposition senses, Che et al. (2010) explored all word senses for semantic role labeling. They showed that all word senses can improve the semantic role labeling performance significantly. However, the golden word senses were used in their experiments. The results are still unknown when an automatic word sense disambiguation system is used.

In this paper, we not only use all word senses disambiguated by an automatic system, but also make the semantic role labeling results to help word sense disambiguation synchronously with a joint model.

### 3 Markov Logic

Markov logic can be understood as a knowledge representation with a weight attached to a first-order logic formula. Let us describe Markov logic in the case of the semantic role labeling task. We can model this task by first introducing a set of logical predicates such as  $role(p, a, r)$  and  $lemma(i, l)$ , which means that the argument at position  $a$  has the role  $r$  with respect to the predicate at position  $p$  and token at position  $i$  has lemma  $l$  respectively. Then we specify a set of weighted first order formulas that define a distribution over sets of ground atoms of these predicates (or so-called possible worlds).

Ideally, the distribution we define with these weighted formulas assigns high probability to possible worlds where semantic role labeling is correct and a low probability to worlds where this is not the case. For instance, for the sentence in Figure 1, a suitable set of weighted formulas would assign a high probability to the world:

$$lemma(1, cat), lemma(3, hit), lemma(5, ball) \\ role(3, 1, A0), role(3, 5, A1)$$

and low probabilities to other cases.

A Markov logic network (MLN)  $M$  is a set of weighted formulas, i.e., a set of pairs  $(\phi, \omega)$ , where  $\phi$  is a first order formula and  $\omega$  is the real weight of the formula.  $M$  defines a probability distribution over possible worlds:

$$p(y) = \frac{1}{Z} \exp\left(\sum_{(\phi, \omega) \in M} \omega \sum_{c \in C^\phi} f_c^\phi(y)\right)$$

where each  $c$  is a binding of free variables in  $\phi$  to constants. Each  $f_c^\phi$  is a binary feature function that returns 1 if the possible world  $y$  includes the ground formula by replacing the free variables in  $\phi$  with the constants in  $c$  is true, and 0 otherwise.  $C^\phi$  is the set of all bindings for the variables in  $\phi$ .  $Z$  is a normalization constant.

### 4 Model

We divide our system into two stages: word sense disambiguation and semantic role labeling. For comparison, we can process them with pipeline strategy, i.e., the word sense disambiguation results are used in semantic role labeling or the semantic role labeling results are used in word sense disambiguation. Of course, we can jointly process them with Markov logic easily.

We define two hidden predicates for the two stages respectively. For word sense disambiguation, we define the predicate  $sense(w, s)$  which indicates that the word at position  $w$  has the sense  $s$ . For semantic role labeling, the predicate  $role(p, a, r)$  is defined as mentioned in above.

Different from Meza-Ruiz and Riedel (2009), which only used sense number as word sense representation, we use a triple (lemma, part-of-speech, sense num) to represent the word sense  $s$ . For example, (hit, v, 01) denotes that the verb ‘‘hit’’ has sense number 01. Obviously, our representation can distinguish different word senses which have the identical sense number. In addition, we use one argument classification stage with predicate  $role$  to label semantic roles as Che et al. (2009). Similarly, no argument identification stage is used in our model. The approach can improve the recall of the system.

In addition to the hidden predicates, we define observable predicates to represent the information available in the corpus. Table 1 presents these predicates.

#### 4.1 Local Formula

A local formula means that its groundings relate any number of observed ground atoms to exactly one hidden ground atom. For example

$$lemma(p, +l_1) \wedge lemma(a, +l_2) \Rightarrow role(p, a, +r)$$

Predicates	Description
$word(i, w)$	Token $i$ has word $w$
$pos(i, t)$	Token $i$ has part-of-speech $t$
$lemma(i, l)$	Token $i$ has lemma $l$
$chdpos(i, t)$	The part-of-speech string of token $i$ 's all children is $t$
$chdddep(i, d)$	The dependency relation string of token $i$ 's all children is $t$
$firstLemma(i, l)$	The leftmost lemma of a subtree rooted by token $i$ is $l$
$lastLemma(i, l)$	The rightmost lemma of a subtree rooted by token $i$ is $l$
$posFrame(i, fr)$	$fr$ is a part-of-speech frame at token $i$
$dep(h, a, de)$	The dependency relation between an argument $a$ and its head $h$ is $de$
$isPredicate(p)$	Token $p$ is a predicate
$posPath(p, a, pa)$	The part-of-speech path between a predicate $p$ and an argument $a$ is $pa$
$depPath(p, a, pa)$	The dependency relation path between a predicate $p$ and an argument $a$ is $pa$
$pathLen(p, a, le)$	The path length between a predicate $p$ and an argument $a$ is $le$
$position(p, a, po)$	The relative position between a predicate $p$ and an argument $a$ is $po$
$family(p, a, fa)$	The family relation between a predicate $p$ and an argument $a$ is $fa$
$wsdCand(i, t)$	Token $i$ is a word sense disambiguation candidate, here $t$ is "v" or "n"
$unique(r)$	For a predicate, semantic role $r$ can only appear once

Table 1: Observable Predicates.

means that if the predicate lemma at position  $p$  is  $l_1$  and the argument lemma at position  $a$  is  $l_2$ , then the semantic role between the predicate and the argument is  $r$  with some possibility.

The  $+$  notation signifies that Markov logic generates a separate formula and a separate weight for each constant of the appropriate type, such as each possible pair of lemmas  $(l_1, l_2, r)$ . This type of "template-based" formula generation can be performed automatically by a Markov logic engine, such as the *thebeast*<sup>1</sup> system.

The local formulas are based on features employed in the state-of-the-art systems. For word sense disambiguation, we use the basic features mentioned by Zhong et al. (2008). The semantic role labeling features are from Che et al. (2009),

<sup>1</sup><http://code.google.com/p/thebeast/>

Features	SRL	WSD
Lemma	•	•
POS	•	•
FirstwordLemma	•	
HeadwordLemma	•	
HeadwordPOS	•	
LastwordLemma	•	
POSPath	•	
PathLength	•	
Position	•	
PredicateLemma	•	
PredicatePOS	•	
RelationPath	•	
DepRelation	•	
POSUpPath	•	
POSFrame	•	
FamilyShip	•	
BagOfWords		•
Window3OrderedWords		•
Window3OrderedPOSS		•

Table 2: Local Features.

the best system of the CoNLL 2009 shared task. The final features are listed in Table 2.

What follows are some simple examples in order to explain how we implement each feature as a formula (or a set of formulas).

Consider the "Position" feature. We first introduce a predicate  $position(p, a, po)$  that denotes the relative position between predicate  $p$  and argument  $a$  is  $po$ . Then we add a formula

$$position(p, a, +po) \Rightarrow role(p, a, +r)$$

for all possible combinations of  $position$  and  $role$  relations.

The "BagOfWords" feature means that the sense of a word  $w$  is determined by all of lemmas in a sentence. Then, we add the following formula set:

$$\begin{aligned} wsdCand(w, +t_w) \wedge lemma(w, +l_w) \wedge lemma(1, +l_1) &\Rightarrow sense(w, +s) \\ wsdCand(w, +t_w) \wedge lemma(w, +l_w) \wedge lemma(2, +l_2) &\Rightarrow sense(w, +s) \\ \dots &\dots \\ wsdCand(w, +t_w) \wedge lemma(w, +l_w) \wedge lemma(n, +l_n) &\Rightarrow sense(w, +s) \end{aligned}$$

where, the  $w$  is the position of current word and  $t_w$  is its part-of-speech tag,  $l_w$  is its lemma.  $l_i$  is the lemma of token  $i$ . There are  $n$  tokens in a sentence totally.

## 4.2 Global Formula

Global formulas relate more than one hidden ground atoms. We use this type of formula for two purposes:

1. To capture the global constraints among different semantic roles;

2. To reflect the joint relation between word sense disambiguation and semantic role labeling.

Punyakanok et al. (2008) proposed an integer linear programming (ILP) model to get the global optimization for semantic role labeling, which satisfies some constraints. This approach has been successfully transferred into dependency parse tree based semantic role labeling system by Che et al. (2009). The final results must satisfy two constraints which can be described with Markov logic formulas as follows:

C1: Each word should be labeled with one and only one label.

$$role(p, a, r_1) \wedge r_1 \neq r_2 \Rightarrow \neg role(p, a, r_2)$$

The same unique constraint also happens on the word sense disambiguation, i.e.,

$$sense(w, s_1) \wedge s_1 \neq s_2 \Rightarrow \neg sense(w, s_2)$$

C2: Some roles (A0~A5) appear only once for a predicate.

$$role(p, a_1, r) \wedge unique(r) \wedge a_1 \neq a_2 \Rightarrow \neg role(p, a_2, r)$$

It is also easy to express the joint relation between word sense disambiguation and semantic role labeling with Markov logic. What we need to do is just adding some global formulas. The relation between them can be shown in Figure 2. Inspired by CoNLL 2008 (Surdeanu et al., 2008) and 2009 (Hajič et al., 2009) shared tasks, where most of successful participant systems used predicate senses for semantic role labeling, we also model that the word sense disambiguation implicates the semantic role labeling.

Here, we divide the all word sense disambiguation task into two subtasks: predicate sense disambiguation and argument sense disambiguation. The advantages of the division method approach lie in two aspects. First, it makes us distinguish the contributions of predicate and argument word sense disambiguation respectively. Second, as previous discussed, the predicate and argument sense disambiguation can help each other. Therefore, we can reflect the help with the division and use Markov logic to represent it.

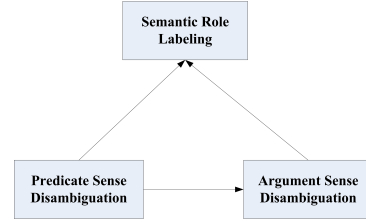


Figure 2: Global model between word sense disambiguation and semantic role labeling.

Finally, we use three global formulas to implement the three lines with direction in Figure 2. They are:

$$sense(p, +s) \Rightarrow role(p, a, +r)$$

$$sense(a, +s) \Rightarrow role(p, a, +r)$$

$$sense(p, +s) \Rightarrow sense(a, +s)$$

## 5 Experiments

### 5.1 Experimental Setting

In our experiments, we use the OntoNotes Release 3.0<sup>2</sup> corpus, the latest version of OntoNotes (Hovy et al., 2006). The OntoNotes project leaders describe it as “a large, multilingual richly-annotated corpus constructed at 90% inter-annotator agreement.” The corpus has been annotated with multiple levels of annotation, including constituency trees, predicate argument structure, word senses, co-reference, and named entities. For this work, we focus on the constituency trees, word senses, and predicate argument structures. The corpus has English, Chinese, and Arabic portions, and we just use the English portion, which has been split into four sections: broadcast conversation (bc), broadcast news (bn), magazine (mz), and newswire (nw). There are several datasets in each section, such as cnn and voa.

We will do our experiments on all of the OntoNotes 3.0 English datasets. For each dataset, we aimed for roughly a 60% train / 20% development / 20% test split. See Table 3 for the detailed statistics. Here, we use the human annotated part-of-speech and parse trees provided by OntoNotes. The lemma of each word is extracted using WordNet tool<sup>3</sup>.

<sup>2</sup><http://www ldc.upenn.edu/Catalog/CatalogEntry.jsp?catalogId=LDC2009T24>

<sup>3</sup><http://wordnet.princeton.edu/>

		Training		Developing		Testing	
bc	cctv	1,042	(0000-0003)	328	(0004-0004)	333	(0005-0005)
	cnn	2,927	(0000-0004)	963	(0005-0006)	880	(0007-0008)
	msnbc	2,472	(0000-0003)	1,209	(0004-0005)	1,315	(0006-0007)
	phoenix	590	(0000-0001)	240	(0002-0002)	322	(0003-0003)
bn	abc	594	(0001-0040)	146	(0041-0054)	126	(0057-0069)
	cnn	1,610	(0001-0234)	835	(0235-0329)	1,068	(0330-0437)
	mnb	309	(0001-0015)	111	(0016-0020)	114	(0021-0025)
	nbc	281	(0001-0023)	128	(0024-0031)	78	(0032-0039)
	pri	1,104	(0001-0068)	399	(0069-0090)	366	(0091-0112)
	voa	1,159	(0001-0159)	315	(0160-0212)	315	(0213-0265)
mz	sinorama	5,051	(1001-1048)	1,262	(1049-1063)	1,456	(1064-1078)
nw	wsj	8,138	(0020-1446)	2,549	(1447-1705)	3,133	(1730-2454)
	xinhua	2,285	(0001-0195)	724	(0196-0260)	670	(0261-0325)
<b>All</b>		27,562		9,209		10,176	

Table 3: Training, developing and testing set sizes for the datasets in sentences. The file ranges (in parenthesis) refer to the numbers within the names of the original OntoNotes 3.0 files. Here, we remove 4,873 sentences without semantic role labeling annotation.

Because we used semantic role labeling system which is based on dependence syntactic trees, we convert the constituency trees into dependence trees with an Constituent-to-Dependency Conversion Tool<sup>4</sup>.

The *thebeast* system is used in our experiment as Markov logic engine. It uses cutting planes inference technique (Riedel, 2008) with integer linear programming. The weights are learned with MIRA (Crammer and Singer, 2003) online learning algorithm.

To our knowledge, this is the first word sense disambiguation and semantic role labeling experiment on OntoNotes 3.0 corpus. In order to compare our joint model with previous work, we build several systems:

**Baseline:** There are two independent baseline systems: word sense disambiguation and semantic role labeling. In each of baseline systems, we only use the local formulas (Section 4.1) and the global formulas which only express the global constraints (Section 4.2).

**Pipeline:** In a pipeline system, we use additional features outputted by preceded stages. Such as in semantic role labeling pipeline system, we use word sense as features, i.e., we set  $sense(w, s)$  as an observable predicate and add  $sense(p, s) \Rightarrow role(p, a, r)$  and  $sense(a, s) \Rightarrow role(p, a, r)$  formulas into semantic role labeling task. As for word sense disambiguation

task, we add a set of formulas  $role(p, a_i, r) \Rightarrow sense(p, s)$ , where  $a_i$  is the  $i^{th}$  argument of the predicate at position  $p$ , and a formula  $role(p, a, r) \Rightarrow sense(p, s)$  for the argument at position  $a$  respectively.

**Jointly:** We use all global formulas mentioned in Section 4.2. With Markov logic, we can add global constraints and get the word sense disambiguation and the semantic role labeling results simultaneously.

## 5.2 Results and Discussion

The performance of these systems on test set is shown in Table 4. All of the parameters are fine tuned on the development set.

Here, we only consider the noun and verb word sense disambiguation, which cover most of multi-sense words. Therefore, the word sense disambiguation performance means the accuracy of all nouns and verbs in the test set. The performance of semantic role labeling is calculated using the semantic evaluation metric of the CoNLL 2009 shared task scorer<sup>5</sup>. It measures the precision, recall, and  $F_1$  score of the recovered semantic dependencies. The  $F_1$  score is used as the final performance metric. A semantic dependency is created for each predicate and its arguments. The label of such dependency is the role of the argument. The same with the CoNLL 2009 shared task, we assume that the predicates have been identified

<sup>4</sup>[http://nlp.cs.lth.se/software/treebank\\_converter/](http://nlp.cs.lth.se/software/treebank_converter/)

<sup>5</sup><http://ufal.mff.cuni.cz/conll2009-st/eval09.pl>

		WSD	SRL
Most Frequent Sense		85.58	—
Baseline		89.37	83.97
Pipeline	PS	89.53	84.17
	AS	89.41	83.94
	PS + AS	—	84.24
Jointly	PS $\Rightarrow$ SRL	89.53	84.27
	AS $\Rightarrow$ SRL	89.49	84.16
	PS $\Rightarrow$ AS	89.45	—
	PS + AS $\Rightarrow$ SRL	89.54	84.34
	Fully	89.55	84.36

Table 4: The results of different systems. Here, PS means predicate senses and AS means argument senses.

correctly.

The first row of Table 4 gives the word sense disambiguation result with the most frequent sense, i.e., the #01 sense of each candidate word which normally is the most frequent one in a balance corpus.

The second row shows the baseline performances. Here, we note that the 89.37 word sense disambiguation accuracy and the 83.97 semantic role labeling  $F_1$  we obtained are comparable to the state-of-the-art systems, such as the 89.1 word sense disambiguation accuracy given by Zhong et al. (2008) and 85.48 semantic role labeling performance given by Che et al. (2010) on OntoNotes 2.0 respectively, although the corpus used in our experiments is upgraded version of theirs<sup>6</sup>. Additionally, the performance of word sense disambiguation is higher than that of the most frequent sense significantly ( $z$ -test<sup>7</sup> with  $\rho < 0.01$ ). Therefore, the experimental results show that the Markov logic can achieve considerable performances for word sense disambiguation and semantic role labeling on the latest OntoNotes 3.0 corpus.

There are two kinds of pipeline systems: word sense disambiguation (WSD) based on semantic role labeling and semantic role labeling (SRL) based on word sense disambiguation. For the using method of word senses, we first only exploit predicate senses (PS) as mentioned by Surdeanu et al. (2008) and Hajič et al. (2009). Then, in or-

<sup>6</sup>Compared with OntoNotes 2.0, the version 3.0 incorporates more corpus.

<sup>7</sup><http://www.dimensionresearch.com/resources/calculators/ztest.html>

der to examine the contribution of word senses except for predicates, we use argument senses (AS) in isolation. Finally, all word senses (PS + AS) were considered.

We can see that when the predicate senses (PS) are used to label semantic role, the performance of semantic role labeling can be improved from 83.97 to 84.17. The conclusion, that the predicate sense can improve semantic role labeling performance, is similar with CoNLL 2008 (Surdeanu et al., 2008) and 2009 (Hajič et al., 2009) shared tasks. However, the improvement is not significant ( $\chi^2$ -test<sup>8</sup> with  $\rho < 0.1$ ). Additionally, the semantic role labeling can improve the predicate sense disambiguation significantly from 89.37 to 89.53 ( $z$ -test with  $\rho < 0.1$ ). The same conclusion was obtained by Dang and Palmer (2005).

However, when we only use argument senses (AS), both of the word sense disambiguation and semantic role labeling performances are almost unchanged (from 89.37 to 89.41 and from 83.97 to 83.94 respectively). For the semantic role labeling task, the reason is that the original lemma and part-of-speech features have been able to describe the argument related information. This kind of sense features is just reduplicate. On the other hand, the argument senses cannot be determined only by the semantic roles. For example, the semantic role “A1” cannot predict the argument sense of “ball” exactly. The predicates must be considered simultaneously.

Therefore, we use the last strategy (PS + AS), which combines the predicate sense and the argument sense together to predict semantic roles. The results show that the performance can be improved significantly ( $\chi^2$ -test with  $\rho < 0.05$ ) from 83.97 to 84.24. Accordingly, the experiment proves that automatic all word sense disambiguation can further improve the semantic role labeling performance. Different from Che et al. (2010), where the semantic role labeling can be improved with correct word senses about  $F_1 = 1$ , our improvement is much lower. The main reason is that the performance of our word sense disambiguation with the most basic features is not high enough. Another limitation of the pipeline strat-

<sup>8</sup><http://graphpad.com/quickcalcs/chisquared1.cfm>

egy is that it is difficult to predict the combination between predicate and argument senses. This is an obvious shortcoming of the pipeline method.

With Markov logic, we can easily join different tasks with global formulas. As shown in Table 4, we use five joint strategies:

1.  $PS \Rightarrow SRL$ : means that we jointly disambiguate predicate senses and label semantic roles. Compared with the pipeline PS system, word sense disambiguation performance is unchanged. However, the semantic role labeling performance is improved from 84.17 to 84.27. Compared with the baseline’s 83.97, the improvement is significant ( $\chi^2$ -test with  $\rho < 0.05$ ).

2.  $AS \Rightarrow SRL$ : means that we jointly disambiguate argument senses and label semantic roles. Compared with the pipeline AS system, both of word sense disambiguation and semantic role labeling performances are improved (from 89.41 to 89.49 and from 83.94 to 84.16 respectively). Although, the improvement is not significant, it is observed that the joint model has the capacity to improve the performance, especially for semantic role labeling, if we could have a more accurate word sense disambiguation.

3.  $PS \Rightarrow AS$ : means that we jointly disambiguate predicate word senses and argument senses. This kind of joint model does not influence the performance of semantic role labeling. The word sense disambiguation outperforms the baseline system from 89.37 to 89.45. The result verifies our assumption that the predicate and argument senses can help each other.

4.  $PS + AS \Rightarrow SRL$ : means that we jointly disambiguate all word senses and label semantic roles. Compared with the pipeline method which uses the PS + AS strategy, the joint method can further improve the semantic role labeling (from 84.24 to 84.34). Additionally, it can obtain the predicate and argument senses together. The all word sense disambiguation performance (89.54) is higher than the baseline (89.37) significantly ( $z$ -test with  $\rho < 0.1$ ).

5. Fully: finally, we use all of the three global formulas together, i.e., we jointly disambiguate predicate senses, argument senses, and label semantic roles. It fully joins all of the tasks. Both of all word sense disambiguation and semantic role

labeling performances can be further improved. Although the improvements are not significant compared with the best pipeline system, they significantly ( $z$ -test with  $\rho < 0.1$  and  $\chi^2$ -test with  $\rho < 0.01$  respectively) outperform the baseline system. Additionally, the performance of the fully joint system does not outperform partly joint systems significantly. The reason seems to be that there is some overlap among the contributions of the three joint systems.

## 6 Conclusion

In this paper, we presented a Markov logic model that jointly models all word sense disambiguation and semantic role labeling. We got the following conclusions:

1. The baseline systems with Markov logic is competitive to the state-of-the-art word sense disambiguation and semantic role labeling systems on OntoNotes 3.0 corpus.

2. The predicate sense disambiguation is beneficial to semantic role labeling. However, the automatic argument sense disambiguation itself is harmful to the task. It must be combined with the predicate sense disambiguation.

3. The semantic role labeling not only can help predicate sense disambiguation, but also argument sense disambiguation (a little). In contrast, because of the limitation of the pipeline model, it is difficult to make semantic role labeling to help predicate and argument sense disambiguation simultaneously.

4. It is easy to implement the joint model of all word sense disambiguation and semantic role labeling with Markov logic. More important, the joint model can further improve the performance of the all word sense disambiguation and semantic role labeling than pipeline systems.

## Acknowledgement

This work was supported by National Natural Science Foundation of China (NSFC) via grant 60803093, 60975055, the “863” National High-Tech Research and Development of China via grant 2008AA01Z144, and Natural Scientific Research Innovation Foundation in Harbin Institute of Technology (HIT.NSRIF.2009069).



## References

- Che, Wanxiang, Zhenghua Li, Yongqiang Li, Yuhang Guo, Bing Qin, and Ting Liu. 2009. Multilingual dependency-based syntactic and semantic parsing. In *Proceedings of CoNLL 2009: Shared Task*, pages 49–54, June.
- Che, Wanxiang, Ting Liu, and Yongqiang Li. 2010. Improving semantic role labeling with word sense. In *NAACL 2010*, pages 246–249, June.
- Crammer, Koby and Yoram Singer. 2003. Ultraconservative online algorithms for multiclass problems. *Journal of Machine Learning Research*, 3:951–991.
- Dahlmeier, Daniel, Hwee Tou Ng, and Tanja Schultz. 2009. Joint learning of preposition senses and semantic roles of prepositional phrases. In *Proceedings of EMNLP 2009*, pages 450–458, August.
- Dang, Hoa Trang and Martha Palmer. 2005. The role of semantic roles in disambiguating verb senses. In *Proceedings of ACL 2005*, pages 42–49, Morristown, NJ, USA.
- Domingos, Pedro and Daniel Lowd. 2009. *Markov Logic: An Interface Layer for Artificial Intelligence*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers.
- Finkel, Jenny Rose and Christopher D. Manning. 2009. Joint parsing and named entity recognition. In *Proceedings of NAACL 2009*, pages 326–334, June.
- Goldberg, Yoav and Reut Tsarfaty. 2008. A single generative model for joint morphological segmentation and syntactic parsing. In *Proceedings of ACL 2008*, pages 371–379, June.
- Hajič, Jan, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of CoNLL 2009: Shared Task*, pages 1–18, June.
- Hovy, Eduard, Mitchell Marcus, Martha Palmer, Lance Ramshaw, and Ralph Weischedel. 2006. Ontonotes: The 90% solution. In *Proceedings of NAACL 2006*, pages 57–60, June.
- Jiang, Wenbin, Liang Huang, Qun Liu, and Yajuan Lü. 2008. A cascaded linear model for joint chinese word segmentation and part-of-speech tagging. In *Proceedings of ACL 2008*, pages 897–904, June.
- Kruengkrai, Canasai, Kiyotaka Uchimoto, Jun'ichi Kazama, Yiou Wang, Kentaro Torisawa, and Hitoshi Isahara. 2009. An error-driven word-character hybrid model for joint chinese word segmentation and pos tagging. In *Proceedings of ACL-IJCNLP 2009*, pages 513–521, August.
- Meza-Ruiz, Ivan and Sebastian Riedel. 2009. Jointly identifying predicates, arguments and senses using markov logic. In *Proceedings of NAACL 2009*, pages 155–163, June.
- Poon, Hoifung and Pedro Domingos. 2008. Joint unsupervised coreference resolution with Markov Logic. In *Proceedings of EMNLP 2008*, pages 650–659, October.
- Punyakanok, Vasin, Dan Roth, and Wen tau Yih. 2008. The importance of syntactic parsing and inference in semantic role labeling. *Computational Linguistics*, 34(2).
- Riedel, Sebastian. 2008. Improving the accuracy and efficiency of map inference for markov logic. In *Proceedings of UAI 2008*, pages 468–475. AUAI Press.
- Surdeanu, Mihai, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The conll 2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of CoNLL 2008*, pages 159–177, August.
- Titov, Ivan and Ryan McDonald. 2008. A joint model of text and aspect ratings for sentiment summarization. In *Proceedings of ACL 2008*, pages 308–316, June.
- Toutanova, Kristina and Colin Cherry. 2009. A global model for joint lemmatization and part-of-speech prediction. In *Proceedings of ACL-IJCNLP 2009*, pages 486–494, August.
- Toutanova, Kristina, Aria Haghighi, and Christopher D. Manning. 2008. A global joint model for semantic role labeling. *Computational Linguistics*, 34(2).
- Yoshikawa, Katsumasa, Sebastian Riedel, Masayuki Asahara, and Yuji Matsumoto. 2009. Jointly identifying temporal relations with markov logic. In *Proceedings of ACL-IJCNLP 2009*, pages 405–413, August.
- Zhang, Yue and Stephen Clark. 2008. Joint word segmentation and POS tagging using a single perceptron. In *Proceedings of ACL 2008*, pages 888–896, June.
- Zhong, Zhi, Hwee Tou Ng, and Yee Seng Chan. 2008. Word sense disambiguation using OntoNotes: An empirical study. In *Proceedings of EMNLP 2008*, pages 1002–1010, October.