Tree Conditional Random Fields for Japanese Semantic Role Labeling

Shilpa Arora SHILPAA@CS.CMU.EDU
Frank Lin FRANK+@CS.CMU.EDU
Hideki Shima HIDEKI@CS.CMU.EDU
Mengqiu Wang MENGQIU@CS.CMU.EDU
Language Technologies Institute, Carnegie Mellon University, Pittsburgh, PA 15213 USA

Abstract

Semantic Role Labeling (SRL) is an important and challenging task in natural language processing. Traditional approaches treat the problem as a pair-wise classification problem. That is, given the predicate, we make local classification decisions for each word to decide if it is an argument, independent of the other words. However, this independence assumption does not always hold on real data. In this paper, we propose a new approach in which we make joint prediction of the predicate and arguments using Markov random fields. The most commonly seen linear-chain conditional random fields (CRFs) are not optimal for this task since it cannot capture the long-distance dependencies between predicate and its arguments. We propose using tree-CRFs for this task, and derive the graph structure from dependency parse trees. We focus on the SRL problem for Japanese language, which has been very little studied. We report experimental results on a newly available Japanese SRL corpus. We show that tree-CRFs significantly and consistently outperformed linear-chain CRFs.

1. Introduction

Semantic Role Labeling (SRL) is the task of extracting information about *who* did *what* to *whom, when* and *how* and has been widely adopted in many NLP applications such as Question Answering (Stenchikova et al. 2005; Narayanan and Harabagiu, 2004) Semantic Role Labeling is a well-studied problem in English, and recently SRL has also been done for Chinese (Xue & Palmer, 2005). The usual approach to Semantic Role Labeling is to treat it as a multi-class classification problem using algorithms such as SVM and Maximum Entropy (Pradhan et. al., 2004; Xue & Palmer, 2004; Hacioglu, 2004).

In this paper, we present our work on Semantic Role Labeling for Japanese. To our knowledge, there has not been any work done in Semantic Role Labeling for Japanese aside from a somewhat related task of deleted case marker prediction (Suzuki & Toutanova, 2006, Toutanova & Suzuki, 2007). Japanese is linguistically quite different from English or Chinese, and some interesting characteristics of the language may potentially make SRL easier and others may make it more difficult. More details of Japanese language and its presented in section 2.

The Semantic Role Labeling task has been traditionally defined, by CoNLL Shared Task 2005 (Carreras & Màrquez, 2005), as the task of finding the arguments for a given predicate. And if a sentence has multiple predicates, arguments for each predicate are predicted in isolation. However, this approach presents a couple of drawbacks. Firstly, when actually annotating a corpus with semantic roles, it is unrealistic to assume that all the predicates in a sentence are always given or can be gathered easily with high precision and recall. Secondly, predicting the arguments of a given predicate in isolation from all other predicates and arguments, makes the questionable assumption that the labeling of a predicate and its arguments is independent of the labeling of other predicates and their arguments in the same sentence.

In this paper, we present a unique approach to Semantic Role Labeling. As a two-step process: first, all the arguments and predicates in a sentence are predicted at the same time; then the arguments would be linked to the predicates as a second step. In this paper we describe our work on the first task of predicating all predicates and arguments.

Conditional Random Fields (CRFs) are undirected graphical models that allow large numbers of arbitrary, overlapping features in predicting labels for a given observation. In the various labeling tasks that it has been applied to (McCallum & Li, 2003; Lafferty, McCallum & Pereira, 2001; Sha & Pereira, 2003), the underlying graphical model is usually implemented as a linear chain, where observations come in a linear sequence and dependencies exist between two consecutive observations. However, in many languages, arguments and predicates are not found next to each other, modeling long distance

dependencies is important for identifying arguments of a predicate. Skip-chain CRFs was used to model long distance relationships as in (Sutton & McCallum, 2004), but exact inference is not possible with skip-chain CRFs and determining which nodes to connect seems to be rather arbitrary.

Tree-CRF, based on parse trees, models the inherent syntactic structure in a sentence and thus would make a logical choice as the graphical structure in identifying predicates and arguments. Tree CRF has been used for Semantic Role Labeling in English (Cohn & Philip Blunsom, 2005), in which a pruned constituency parse tree is used as the model structure. Constituency parse tree are deep trees leading to a large number of random variables resulting in long training times (1 month for Cohn & Philip), and the resulting performance showed little improvement over traditional Maximum Entropy classifiers.

We used dependency parse tree to model Tree-CRF. Dependency parse trees are a much shallower, compact representation of the syntactic structure of a sentence than constituency parse trees; giving us several advantages over constituency parse trees: first, there is no need for pruning; second, a smaller number of random variables resulting in faster training; third, shallower trees leads to shorter paths between nodes that are potentially predicates and arguments, leading to better modeling of dependencies among labels. Details of Tree-CRF are presented in section 3.

2. Semantic Role Labeling task for Japanese

SRL tasks in English and Chinese, commonly uses the Propbank (Palmer, Gildea, & Kingsbury, 2005), where the annotation scheme consists of the predicate, the core arguments (ARG0 to ARG5) and the adjunctive arguments (ARGM-TMP, ARGM-LOC, etc). ARG0 is normally the subject of the predicate and ARG1 the object. SRL work in Japanese has only begun recently, and the arguments are named after Japanese case markers, i.e. "GA", "NI", "O", due to the argument's strong relation with the case marker. A case marker, or a case marking particle, is a free or bound morpheme that indicates the grammatical function of the marked word or sentence. Case markers "GA", "O", "NI" are analogous to nominative, accusative and dative cases, which are strongly correlated with ARG0, ARG2, and ARG1, respectively.

Japanese sentences can be split into a series of "bunsetsu" chunks which are smallest meaningful units in Japanese. *Bunsetsu* is usually made of one content word followed by any number of function words. By utilizing dependencies among multiple bunsetsu, we can build trees where each node represents a bunsetsu. Note that in terms of both accuracy and speed of SRL task, Tree-CRF can benefit from this structure better than from constituent based dependencies since we can represent the relational information more densely.

Properties of Japanese language make the SRL task different from English. For example, Japanese is a head final language i.e. it always places the head at the end of its clause.

Another property of Japanese is its relatively free word-ordered compared to English or Chinese, and this makes our task very difficult. Another interesting characteristic of Japanese is that it has Zero-Anaphora. Zero anaphora is the use of a gap in a phrase or a clause that refers back to an expression in the previous clause or sentence. Zero-anaphora makes the semantic role labeling task difficult too as we need to map the gap to the actual phrase/clause it's referring to.

One apparent task-specific advantage of Japanese language is the fact that it has rich morphology. Functional words in a bunsetsu chunk, especially case markers such as が(GA), に(NI), を(O) give strong evidence to the semantic roles they play in a sentence. Non-case-marking particles such as は(WA) are also informative. Even so, case markers can be arbitrarily left out from a bunsetsu (Case Ellipsis is common phenomena where particles are omitted from text) and often there is not a one-to-one mapping from case markers to semantic roles. For instance, the sentence in figure 1 has GA, NI and O particle but it only has one bunsetsu with the GA label. Therefore, this "advantage" can also become a disadvantage – once a learning algorithm has determined the strong correlation between semantic roles and particular case markers, it becomes difficult to train the algorithm to recognize examples where roles don't correspond to case markers. These "false leads" make improvement of the labeling performance especially challenging after reaching a certain threshold.

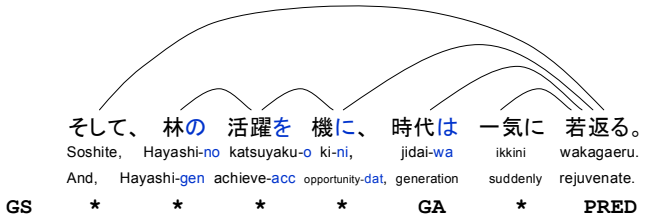

Figure 1. Example sentence of a dependency-parsed sentence with gold standard labels on each node. In English, it reads *"Then suddenly, the generation of Shougi rejuvenates by a young player Hayashi's achievement"*

3. Tree-CRF Algorithm

The conditional probability for the labels given the observation in CRF is defined as below:

$$p(y \mid x) = \frac{1}{Z(x)} \exp\left(\sum_{c \in C} \sum_{k} \lambda_k f_k(c, y_c, x)\right)$$

where C is the set of cliques, λ_k are the feature weights and $f_k(.)$ are the feature function for the clique. $Z(.)$ is the normalization function. The probability distribution can be rewritten as below, separating the node cliques and from edge cliques (between different nodes).

$$p(y \mid x) = \frac{1}{Z(x)} \exp\{\sum_{u \in C_1} \sum_{k} \lambda_k g_k(u, y_u, x)$$

$$+ \exp\left(\sum_{v \in C_2} \sum_{k} \lambda_k h_j(u, v, y_u, y_v, x)\right)\}$$

C_1 is the set of node cliques and C_2 is the set of edge cliques. In case of Linear Chain CRF, edge cliques are the edges between two adjacent nodes where as for Tree-CRF, edge cliques are edges between parent and child nodes. We used the dependency parse structure to model the Tree-CRF. The arrows in the example sentence in Figure 1 shows the dependencies between the nodes.

4. Experiment Setup

4.1 Dataset

Our dataset comes from a recently released NAIST Text Corpus (Citation needed), a corpus for Semantic Role Labeling in Japanese. The NIAST corpus contains newswire stories from the 1995 corpus of Mainichi News, and was hand-annotated with labels for predicates and three argument types, GA, O, and NI.

After cleaning up the corpus (removing non-sentences and sentences without any predicates or arguments), we are left with about 38,384 sentences from 2,929 articles. The dataset is divided into 2,000 sentences for development, 2000 sentences for testing, and a training set with varying size of training data from 2,000 to 34,000 sentences.

4.2 Preprocessing Tools

To obtain dependency parse and morphological analysis, we used CaboCha (Citation needed), a high performance (about 90% in F1 score on newswire text) non-projective dependency parser that uses a cascaded chunking model and SVM. The output from CaboCha also includes morpheme segmentation, morphological POS tags, and Named Entity tags.

4.3 Software Package

We used GRMM (Graphical Models in Mallet) toolkit (Sutton, 2006) for training a classifier over Tree-CRF. GRMM is a general machine learning software package in graphical models, and we slightly modified the software to model arbitrary tree structures and suit our experimental needs. We used the Tree Belief Propagation algorithm for inference and L-BFGS algorithm for optimization.

4.4 Classifiers

In our experiment we implemented three classifiers for labeling semantic roles. The first is a baseline classifier that uses simple hand-written rules to label semantic roles based on function words and POS tags. The second is a CRF classifier using linear chain as its graphical model. The third is a CRF classifier using dependency parse tree as its graphical model.

5. Observation Features

Since we are working with *bunsetsu* chunks in Japanese, features are defined over the chunks, which also correspond to nodes in the dependency parse tree.

We experimented with two different feature settings. One where we used very basic feature set based on the words, Part of Speech (POS), functional word etc. Our objective was to compare Tree-CRF and Linear chain-CRF without any influence of the features and show that Tree-CRF does better than Linear-CRF because of its inherent graphical structure. The base features we used are listed in table 1.

We also experimented with more informative tree-features (features that exploit the tree structure). These features capture information about a node's parents, its siblings, for e.g., "is my parent a verb", "what is my position in the tree", "number of hops to the root". These features are semantically and structurally very informative and we expect to improve our performance with these. The tree features we used are listed in Table 1.

Table 1. Base Features & Tree Features

BASE FEATURES	
POS	
General POS	
Named Entity	
Word	
Functional words	
Punctuations	
TREE FEATURES	
isFirst	parentVerbANDpossfromLeft
isLast	parentVerbANDpossfromRight
isRoot	parentVerbANDRootANDpossfro
numSibling	mLeft
wordParent	parentVerbANDRootANDpossfro
sposParent	mRight

lemmaParent	leftSibStr
neParent	sposLeftSib
wordGrandParent	lemmaLeftSib
sposGrandParent	rightSibStr
lemmaGrandParent	sposRightSib
neGrandParent	lemmaRightSib
numChildren	gaSib
parentIsRoot	oSib
parentIsVerb	niSib
possfromLeft	closestVerb
possfromRight	closestVerbIsRoot
	numhops2Root

6. Experiment Results

For the baseline system, we implemented a simple rule based Semantic Role Labeling system, which uses the particles in the word to identify the arguments and uses the POS to identify the predicate. Table 2 shows the rules used for predicting the case for a word.

Table 2. Baseline: Rule based SRL system

RULE	PREDICTED CASE
PARTICLE=が (GA) は (WA)	GA
PARTICLE=に(NI)	NI
PARTICLE=を(O)	O
PARTICLE= 動詞 (verb)	PRED

The Tree-CRF and Chain-CRF, were trained on different sets of training data with 2K, 4K, 10K, 20K and 34K sentences each. A comparison of the Baseline, Tree-CRF and Chain-CRF (training until convergence) over the full data set with only base features is shown in Table 3. The Tree-CRF does the best on Precision, Recall and F-measure for most of the arguments and the predicate.

A comparison of performance of Tree-CRF and Linear chain-CRF (at 100 training iterations) with different amounts of training data is shown in figure 2. As can be seen, Tree-CRF outperforms chain-CRF for all the arguments and the predicate. The difference is more when training data is lesser. This implies that Tree-CRF is able to learn the concept better than the chain-CRF with little data.

We used the Bootstrapping resampling significance tests (Hoeffding, 1952) of 10,000 iterations to evaluate how significant the improvement of Tree-CRF over Linear chain-CRF is. The P and R in figure 2 indicate whether the Precision and Recall was significantly more for Tree-CRF. As can be seen from the figure 2, the Recall was always significant but in most of the cases except NI, the precision was also significant.

Table 3. Comparison of Linear & Tree CRF with Rule Based baseline using complete Training Data (training until convergence)

RULE-BASED	P	R	F
PRED	0.8291397	0.723244	0.7761919
GA	0.7044834	0.6592484	0.681865
O	**0.8486239**	0.71036753	0.7794957
NI	0.5533101	**0.8291806**	0.6914930
LINEAR-CRF	P	R	F
PRED	0.8393957	0.88273	0.8610649
GA	0.738748	0.7365924	0.7376705
O	0.8145110	0.7081733	0.7613421
NI	0.7045454	0.6802507	0.6923981
TREE-CRF	P	R	F
PRED	**0.867418**	**0.884824**	**0.8761212**
GA	**0.742744**	**0.756293**	**0.7495189**
O	0.8277363	**0.7301151**	**0.7789257**
NI	**0.7079303**	0.7648902	**0.7364103**

Using tree features with Tree CRF improves its performance for GA, O and PRED. The improvement is especially more for O. However, for NI, tree features decrease the performance of Tree-CRF. NI is normally the most difficult and ambiguous case to predict. A comparison of Tree-CRF with and without tree features is shown in figure 3. In this figure, we also show a comparison with the Baseline System. The Baseline system has a very high recall for O and NI and hence a very good F-measure. For argument "O", the Baseline system outperforms the Tree-CRF without tree features but with tree features it does much better than Baseline. For NI, both Tree-CRF with and without tree features perform better than the baseline but the tree features hurt the performance for NI.

Figure 4 shows an example sentence where Linear Chain CRF mislabels the arguments but Tree-CRF gets them correct. The word *katsuyaku* has an *O* particle but since it is not dependent on any predicate in the dependency parse tree, it is not an argument of any predicate in the sentence. However, since Linear Chain-CRF cannot capture the long distance dependency, it mislabels this node with the case "O".

Training time: For 100 training iterations, it takes about 2 hours for the training data set of 2000 sentences and about 12 hours for the complete training data set of 34,000 sentences. For training until convergence, it took about 3 days for the data set of 34,000 sentences.

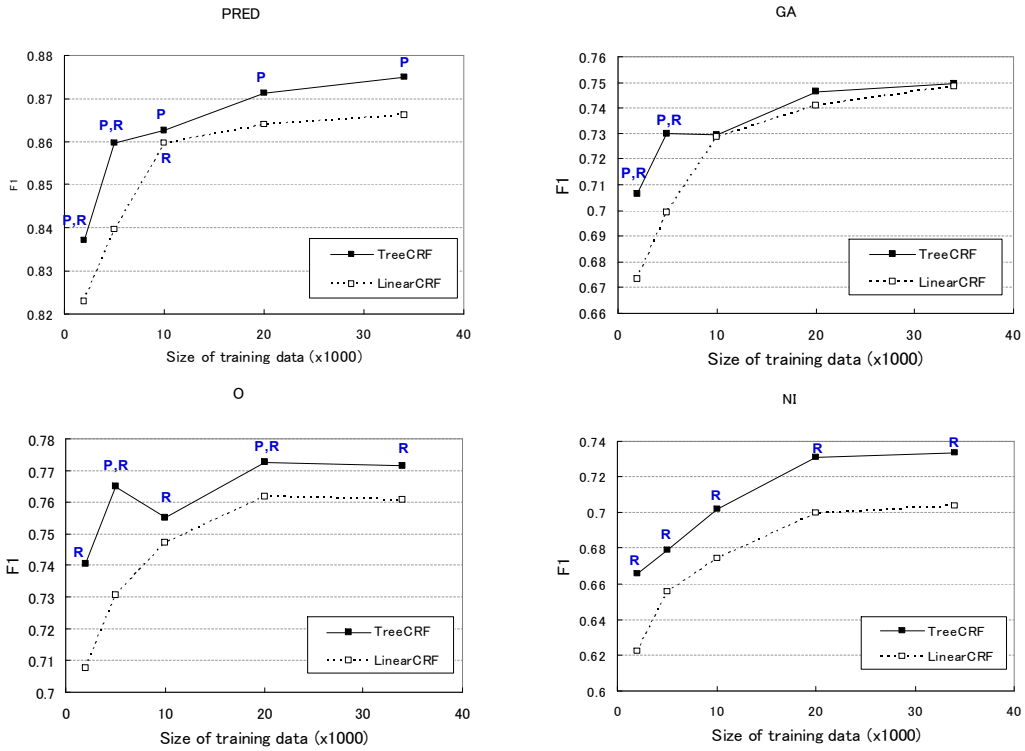

Figure 2: Comparison of Tree-CRF & Linear Chain on 100 iterations

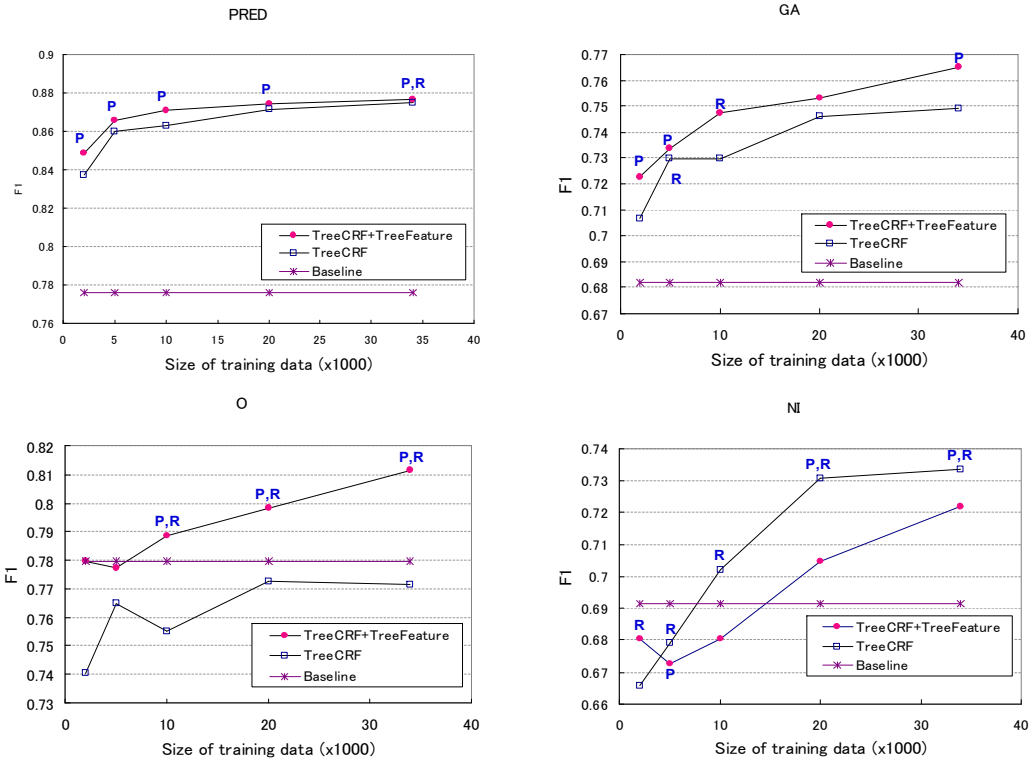

Figure 3: Comparison of Tree-CRF with and without tree features and baseline system (100 iterations)

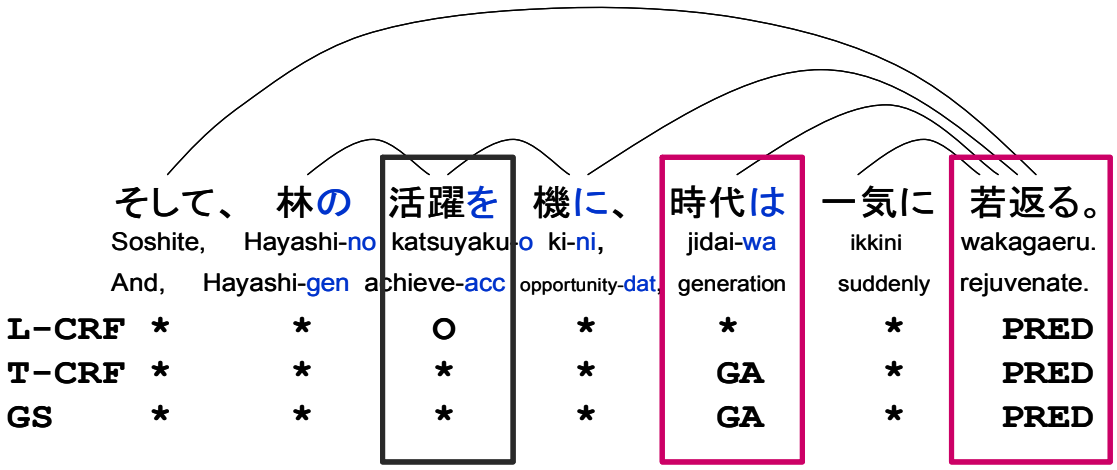

Figure 4. Example of a sentence with where case markers do not map to semantic roles. Same sentence as Figure 1.

7. Conclusion

In this paper, we proposed a new approach to the Semantic Role Labeling task, where we predict all the arguments and predicates in the sentence at the same time. We also revisited the problem of using Tree-CRF for the Semantic Role Labeling task. The earlier work on using Tree-CRF for English SRL didn't do that well and in this paper we try to investigate this and compare it with linear-chain CRF. Tree-CRF does better than Linear-chain CRF even with very little features. Tree-CRF can outperform Linear-chain CRF due to the inherent tree structure which captures the semantic and syntactic dependencies better. The difference in performance of Tree & Chain CRF is more with lesser training data which indicates Tree-CRF learns the target concept faster. And the improvement is statistically significant as shown by the Bootstrapping resampling significance test. Dependency tree is a better structure to model Tree-CRF as compared to constituency parse tree which is deeper and hence more complex and computationally expensive. Tree features help in improving the performance. To our knowledge, this is one of the first works in Semantic Role Labeling for Japanese language.

As a future work, we would like to compare our algorithm with the other commonly used algorithms for Semantic Role labeling such as Maximum Entropy and SVM etc. In this work, we predict all the arguments and predicates in the sentence at once. Incase of multiple predicates in the sentence, we would need to link the arguments with the predicate it is associated with. We would like to implement this as a second stage of the Semantic Role Labeling task. Also, we would like to apply Tree-CRF to Semantic Role Labeling in other languages such as English & Chinese.

Acknowledgments

We are thankful to William Cohen and Vitor R. Carvalho, whose precious guidance helped us in proceeding with our ideas to their actual implementation. We would also like to thank the JAVELIN Project Team (Eric Nyberg and Teruko Mitamura) for helping us in getting the corpus and other Japanese language processing tools.

References

Xavier Carreras and Llu´ıs M`arquez. 2005. *Introduction to the CoNLL-2005 Shared Task: Semantic Role Labeling.* In Proceedings of the CoNLL-2005

Trevor Cohn and Philip Blunsom. 2005. *Semantic role labeling with tree conditional random fields*. In Proceedings of CoNLL-2005

Kadri Hacioglu, 2004. *Semantic role labeling using dependency trees.* In Proceedings of the 20th international conference on Computational Linguistics, August 23-27, 2004, Geneva, Switzerland.

Wassily Hoeffding, 1952. *The Large-Sample Power of Tests Based on Permutations of Observations*. Annals of Mathematical Statistics, 23, 169-192.

Taku Kudo and Yuji Matsumoto. *Japanese Dependency. Analysis using Cascaded Chunking.* In Proceedings. of the 6th Conference on Natural Language Learning. (CoNLL). pp.63-69, 2002.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data.* In Proceedings of the 18th International Conference on Machine Learning, pages 282–289.

Andrew McCallum and Wei Li. 2003. *Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons.* In Proceedings of the 7th Conference on Natural Language Learning, pages 188–191.

Srini Narayanan and Sanda Harabagiu. 2005. *Question Answering Based on Semantic Structures.* In Proceedings of the 20th international conference on Computational Linguistics, August 23-27, 2004, Geneva, Switzerland.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. *The proposition bank: An annotated corpus of semantic roles.* Computational Linguistics, 31(1).

S. Pradhan, W. Ward, K. Hacioglu, J. Martin & D. Jurafsky, 2004. *Shallow semantic parsing using support vector machines.* In Proceedings of HLT/NAACL-2004.

Fei Sha and Fernando Pereira. 2003. *Shallow parsing with conditional random fields.* In Proceedings of the Human Language Technology Conference and North American Chapter of the Association for Computational Linguistics, pages 213–220.

Charles Sutton. 2006. GRMM: A Graphical Models Toolkit. http://mallet.cs.umass.edu.

Charles Sutton and Andrew McCallum. ICML workshop on Statistical Relational Learning, 2004.

Hisami Suzuki and Kristina Toutanova,. 2006. *Learning to predict case markers in Japanese.* In Proceedings of the 21st international Conference on Computational Linguistics and the 44th Annual Meeting of the ACL (Sydney, Australia, July 17 - 18, 2006). Annual Meeting of the ACL. Association for Computational Linguistics, Morristown, NJ, 1049-1056.

Svetlana Stenchikova, Dilek Hakkani-Tür, Gokhan Tur,2005. *QASR: Spoken Question Answering Using Semantic Role Labeling.* At ASRU-2005, 9th biannual IEEE workshop on Automatic Speech Recognition and Understanding, Cancun, Mexico, December, 2005.

Kristina Toutanova and Hisami Suzuki*, 2007. Generating* Case Markers in Machine Translation. In Proceedings of the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT 2007).

Nianwen Xue and Martha Palmer. 2005. *Automatic Semantic Role Labeling for Chinese Verbs.* In Proceedings of the 19th International Joint Conference on Artificial Intelligence. Edinburgh, Scotland