

Japanese Dependency Parsing Using Co-occurrence Information and a Combination of Case Elements

Takeshi Abekawa

Graduate School of Education
University of Tokyo
abekawa@p.u-tokyo.ac.jp

Manabu Okumura

Precision and Intelligence Laboratory
Tokyo Institute of Technology
oku@pi.titech.ac.jp

Abstract

In this paper, we present a method that improves Japanese dependency parsing by using large-scale statistical information. It takes into account two kinds of information not considered in previous statistical (machine learning based) parsing methods: information about dependency relations among the case elements of a verb, and information about co-occurrence relations between a verb and its case element. This information can be collected from the results of automatic dependency parsing of large-scale corpora. The results of an experiment in which our method was used to rerank the results obtained using an existing machine learning based parsing method showed that our method can improve the accuracy of the results obtained using the existing method.

1 Introduction

Dependency parsing is a basic technology for processing Japanese and has been the subject of much research. The Japanese dependency structure is usually represented by the relationship between phrasal units called *bunsetsu*, each of which consists of one or more content words that may be followed by any number of function words. The dependency between two *bunsetsus* is direct from a dependent to its head.

Manually written rules have usually been used to determine which *bunsetsu* another *bunsetsu* tends to modify, but this method poses problems in terms of the coverage and consistency of the rules. The recent availability of larger-scale corpora annotated with dependency information has thus resulted in more work on statistical dependency analysis technologies that use machine learning algorithms (Kudo and Matsumoto, 2002; Sassano,

2004; Uchimoto et al., 1999; Uchimoto et al., 2000).

Work on statistical Japanese dependency analysis has usually assumed that all the dependency relations in a sentence are independent of each other, and has considered the *bunsetsus* in a sentence independently when judging whether or not a pair of *bunsetsus* is in a dependency relation. In judging which *bunsetsu* a *bunsetsu* modifies, this type of work has used as features the information of two *bunsetsus*, such as the head words of the two *bunsetsus*, and the morphemes at the ends of the *bunsetsus* (Uchimoto et al., 1999). It is necessary, however, to also consider features for the contextual information of the two *bunsetsus*. One such feature is the constraint that two case elements with the same case do not modify a verb.

Statistical Japanese dependency analysis takes into account syntactic information but tends not to take into account lexical information, such as co-occurrence between a case element and a verb.

The recent availability of more corpora has enabled much information about dependency relations to be obtained by using a Japanese dependency analyzer such as KNP (Kurohashi and Nagao, 1994) or CaboCha (Kudo and Matsumoto, 2002). Although this information is less accurate than manually annotated information, these automatic analyzers provide a large amount of co-occurrence information as well as information about combinations of multiple cases that tend to modify a verb.

In this paper, we present a method for improving the accuracy of Japanese dependency analysis by representing the lexical information of co-occurrence and dependency relations of multiple cases as statistical models. We also show the results of experiments demonstrating the effectiveness of our method.

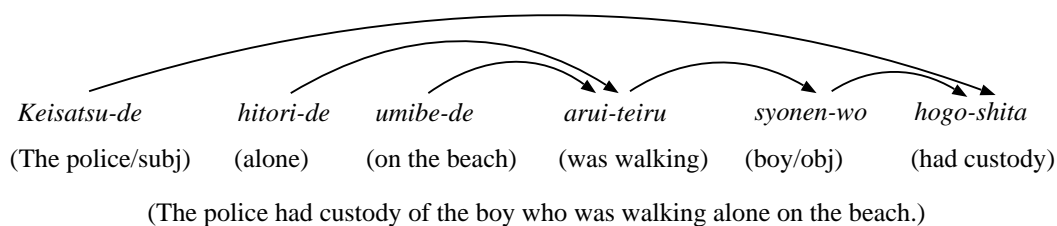


Figure 1: Example of a Japanese sentence, *bunsetsu* and dependencies

2 Parsing Japanese

The Japanese language is basically an SOV language, but word order is relatively free. In English the syntactic function of each word is represented by word order, while in Japanese it is represented by postpositions. For example, one or more postpositions following a noun play a role similar to the declension of nouns in German, which indicates grammatical case.

The syntax of a Japanese sentence is analyzed by using segments, called *bunsetsu*, that usually contain one or more content words like a noun, verb, or adjective, and zero or more function words like a particle (case marker) or verb/noun suffix. By defining a *bunsetsu* in this manner, we can analyze a sentence in a way similar to that used when analyzing the grammatical roles of words in inflected languages like German.

Japanese dependencies have the following characteristics:

- Each *bunsetsu* except the rightmost one has only one head.
- Each head *bunsetsu* is always placed to the right of (*i.e.* after) its modifier.
- Dependencies do not cross one another.

Statistical Japanese dependency analyzers (Kudo and Matsumoto, 2005; Kudo and Matsumoto, 2002; Sassano, 2004; Uchimoto et al., 1999; Uchimoto et al., 2000) automatically learn the likelihood of dependencies from a tagged corpus and calculate the best dependencies for an input sentence. These likelihoods are learned by considering the features of *bunsetsus* such as their character strings, parts of speech, and inflection types, as well as information between *bunsetsus* such as punctuation and the distance between *bunsetsus*. The weight of given features is learned

from a training corpus by calculating the weights from the frequencies of the features in the training data.

3 Japanese dependency analysis taking account of co-occurrence information and a combination of multiple cases

One constraint in Japanese is that multiple nouns of the same case do not modify a verb. Previous work on Japanese dependency analysis has assumed that all the dependency relations are independent of one another. It is therefore necessary to also consider such a constraint as a feature for contextual information. Uchimoto et al., for example, used as such a feature whether a particular type of *bunsetsu* is between two *bunsetsus* in a dependency relation (Uchimoto et al., 1999), and Sassano used information about what is just before and after the modifying *bunsetsu* and modifyee *bunsetsu* (Sassano, 2004).

In the artificial example shown in Figure 1, it is natural to consider that “*keisatsu-de*” will modify “*hogo-shita*”. Statistical Japanese dependency analyzers (Uchimoto et al., 2000; Kudo and Matsumoto, 2002), however, will output the result where “*keisatsu-de*” modifies “*arui-teiru*”. This is because in sentences without internal punctuation a noun tends to modify the nearest verb, and these analyzers do not take into account a combination of multiple cases.

Another kind of information useful in dependency analysis is the co-occurrence of a noun and a verb, which indicates to what degree the noun tends to modify the verb. In the above example, the possible modifyees of “*keisatsu-de*” are “*arui-teiru*” and “*hogo-shita*”. Taking into account information about the co-occurrence of “*keisatsu-de*” and “*arui-teiru*” and of “*keisatsu-de*” and “*hogo-shita*” makes it obvious that “*keisatsu-de*” is more likely to modify “*hogo-shita*”.

In summary, we think that statistical Japanese dependency analysis needs to take into account at least two more kinds of information: the dependency relation between multiple cases where multiple nouns of the same case do not modify a verb, and the co-occurrence of nouns and verbs. One way to use such information in statistical dependency analysis is to directly use it as features. However, Kehler et al. pointed out that this does not make the analysis more accurate (Kehler et al., 2004). This paper therefore presents a model that uses the co-occurrence information separately and reranks the analysis candidates generated by the existing machine learning model.

4 Our proposed model

We first introduce the notation for the explanation of the dependency structure T :

- $m(T)$: the number of verbs in T
- $v_i(T)$: the i -th verb in T
- $c_i(T)$: the number of case elements that modify the i -th verb in T
- $es_i(T)$: the set of case elements that modify the i -th verb in T
- $rs_i(T)$: the set of particles in the set of case elements that modify the i -th verb in T
- $ns_i(T)$: the set of nouns in the set of case elements that modify the i -th verb in T
- $r_{i,j}(T)$: the j -th particle that modifies the i -th verb in T
- $n_{i,j}(T)$: the j -th noun that modifies the i -th verb in T

We defined case element as a pair of a noun and following particles. For the dependency structure we assume the conditional probability $P(es_i(T)|v_i(T))$ that the set of case elements $es_i(T)$ depends on the $v_i(T)$, and assume the set of case elements $es_i(T)$ is composed of the set of noun $ns_i(T)$ and particles $rs_i(T)$.

$$P(es_i(T)|v_i(T)) \stackrel{def}{=} P(rs_i(T), ns_i(T)|v_i(T)) \quad (1)$$

$$= P(rs_i(T)|v_i(T)) \times P(ns_i(T)|rs_i(T), v_i(T)) \quad (2)$$

$$\simeq P(rs_i(T)|v_i(T)) \times \prod_{j=1}^{c_i(T)} P(n_{i,j}(T)|rs_i(T), v_i(T)) \quad (3)$$

$$\simeq P(rs_i(T)|v_i(T)) \times \prod_{j=1}^{c_i(T)} P(n_{i,j}(T)|r_{i,j}(T), v_i(T)) \quad (4)$$

In the transformation from Equation (2) to Equation (3), we assume that the set of noun $ns_i(T)$ is independent of the verb $v_i(T)$. And in the transformation from Equation (3) to Equation (4), we assume that the noun $n_{i,j}(T)$ is dependent on only its following particle $r_{i,j}(T)$.

Now we assume the dependency structure T of the whole sentence is composed of only the dependency relation between case elements and verbs, and propose the sentence probability defined by Equation (5).

$$P(T) = \prod_{i=1}^{m(T)} P(rs_i(T)|v_i(T)) \times \prod_{j=1}^{c_i(T)} P(n_{i,j}(T)|r_{i,j}(T), v_i(T)) \quad (5)$$

We call $P(rs_i(T)|v_i(T))$ the co-occurrence probability of the particle set and the verb, and we call $P(n_{i,j}(T)|r_{i,j}(T), v_i(T))$ the co-occurrence probability of the case element set and the verb.

In the actual dependency analysis, we try to select the dependency structure \hat{T} that maximizes the Equation (5) from the possible parses T for the inputted sentence:

$$\hat{T} = \underset{T}{\operatorname{argmax}} \prod_{i=1}^{m(T)} P(rs_i(T)|v_i(T)) \times \prod_{j=1}^{c_i(T)} P(n_{i,j}(T)|r_{i,j}(T), v_i(T)). \quad (6)$$

The proposed model is inspired by the semantic role labeling method (Gildea and Jurafsky, 2002), which uses the frame element group in place of the particle set.

It differs from the previous parsing models in that we take into account the dependency relations among particles in the set of case elements that modify a verb. This information can constrain the combination of particles (cases) among *bunsetsus* that modify a verb. Assuming the independence among particles, we can rewrite Equation (5) as

$$P(T) = \prod_{i=1}^{m(T)} \prod_{j=1}^{c_i(T)} P(n_{i,j}(T), r_{i,j}(T)|v_i(T)). \quad (7)$$

4.1 Syntactic property of a verb

In Japanese, the “*ha*” case that indicates a topic tends to modify the main verb in a sentence and tends not to modify a verb in a relative clause. The

	verb: ‘ <i>aru-ku</i> ’		verb: ‘ <i>hogo-suru</i> ’	
	case elements	particle set	case elements	particle set
a	<i>keisatsu-de umibe-de hitori-de</i>	{ <i>de,de,de</i> }		<i>syonen-wo</i> { <i>wo</i> }
b	<i>umibe-de hitori-de</i>	{ <i>de,de</i> }	<i>keisatsu-de</i>	<i>syonen-wo</i> { <i>de,wo</i> }
c	<i>hitori-de</i>	{ <i>de</i> }	<i>keisatsu-de umibe-de</i>	<i>syonen-wo</i> { <i>de,de,wo</i> }
d		{ <i>none</i> }	<i>keisatsu-de umibe-de hitori-de</i>	<i>syonen-wo</i> { <i>de,de,de,wo</i> }

Table 1: Analytical process of the example sentence

co-occurrence probability of the particle set therefore tends to be different for verbs with different syntactic properties.

Like (Shirai, 1998), to take into account the reliance of the co-occurrence probability of the particle set on the syntactic property of a verb, instead of using $P(rs_i(T)|v_i(T))$ in Equation (5), we use $P(rs_i(T)|syn_i(T), v_i(T))$, where $syn_i(T)$ is the syntactic property of the i -th verb in T and takes one of the following three values:

‘**verb**’ when v modifies another verb

‘**noun**’ when v modifies a noun

‘**main**’ when v modifies nothing (when it is at the end of the sentence, and is the main verb)

4.2 Illustration of model application

Here, we illustrate the process of applying our proposed model to the example sentence in Figure 1, for which there are four possible combinations of dependency relations. The *bunsetsu* combinations and corresponding sets of particles are listed in Table 1. In the analytical process, we calculate for all the combinations the co-occurrence probability of the case element set (*bunsetsu* set) and the co-occurrence probability of the particle set, and we select the \hat{T} that maximizes the probability.

Some of the co-occurrence probabilities of the particle sets for the verbs “*aru-ku*” and “*hogo-suru*” in the sentence are listed in Table 2. How to estimate these probabilities is described in section 5.3. Basically, the larger the number of particles, the lower the probability is. As you can see in the comparison between {*de, wo*} and {*de, de*}, the probability becomes lower when multiple same cases are included. Therefore, the probability can reflect the constraint that multiple case elements of the same particle tend not to modify a verb.

5 Experiments

We evaluated the effectiveness of our model experimentally. Since our model treats only the de-

rs_i	$P(rs_i noun, v_1)$ $v_1 = \text{“aru-ku”}$	$P(rs_i main, v_2)$ $v_2 = \text{“hogo-suru”}$
{ <i>none</i> }	0.29	0.35
{ <i>wo</i> }	0.30	0.24
{ <i>ga</i> }	0.056	0.072
{ <i>ni</i> }	0.040	0.041
{ <i>de</i> }	0.032	0.033
{ <i>ha</i> }	0.035	0.041
{ <i>de, wo</i> }	0.022	0.018
{ <i>de, de</i> }	0.00038	0.00038
{ <i>de, de, wo</i> }	0.00022	0.00018
{ <i>de, de, de</i> }	0.0000019	0.0000018
{ <i>de, de, de, wo</i> }	0.00000085	0.00000070

Table 2: Example of the co-occurrence probabilities of particle sets

pendency relations between a noun and a verb, we cannot determine all the dependency relations in a sentence. We therefore use one of the currently available dependency analyzers to generate an ordered list of n -best possible parses for the sentence and then use our proposed model to rerank them and select the best parse.

5.1 Dependency analyzer for outputting n -best parses

We generated the n -best parses by using the “posterior context model” (Uchimoto et al., 2000). The features we used were those in (Uchimoto et al., 1999) and their combinations. We also added our original features and their combinations, with reference to (Sassano, 2004; Kudo and Matsumoto, 2002), but we removed the features that had a frequency of less than 30 in our training data. The total number of features is thus 105,608.

5.2 Reranking method

Because our model considers only the dependency relations between a noun and a verb, and thus cannot determine all the dependency relations in a sentence, we restricted the possible parses for

reranking as illustrated in Figure 2. The possible parses for reranking were the first-ranked parse and those of the next-best parses in which the verb to modify was different from that in the first-ranked one. For example, parses 1 and 3 in Figure 2 are the only candidates for reranking. In our experiments, n is set to 50.

The score we used for reranking the parses was the product of the probability of the posterior context model and the probability of our proposed model:

$$score = P_{context}(T)^\alpha \times P(T), \quad (8)$$

where $P_{context}(T)$ is the probability of the posterior context model. The α here is a parameter with which we can adjust the balance of the two probabilities, and is fixed to the best value by considering development data (different from the training data)¹.

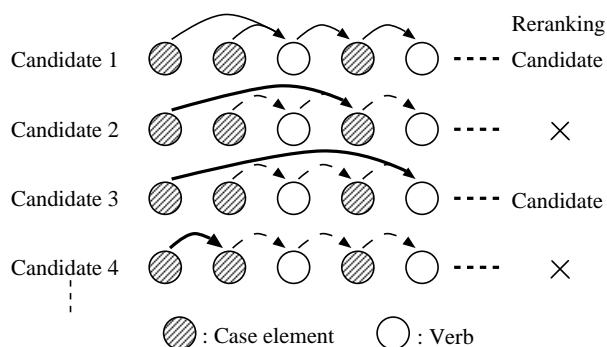


Figure 2: Selection of possible parses for reranking

Many methods for reranking the parsing of English sentences have been proposed (Charniak and Johnson, 2005; Collins and Koo, 2005; Henderson and Titov, 2005), all of which are discriminative methods which learn the difference between the best parse and next-best parses. While our reranking model using generation probability is quite simple, we can easily verify our hypothesis that the two proposed probabilities have an effect on improving the parsing accuracy. We can also verify that the parsing accuracy improves by using imprecise information obtained from an automatically parsed corpus.

Klein and Manning proposed a generative model in which syntactic (PCFG) and semantic (lexical dependency) structures are scored with separate models (Klein and Manning, 2002), but

¹In our experiments, α is set to 2.0 using development data.

they do not take into account the combination of dependencies. Shirai et al. also proposed a statistical model of Japanese language which integrates lexical association statistics with syntactic preference (Shirai et al., 1998). Our proposed model differs from their method in that it explicitly uses the combination of multiple cases.

5.3 Estimation of co-occurrence probability

We estimated the co-occurrence probability of the particle set and the co-occurrence probability of the case element set used in our model by analyzing a large-scale corpus. We collected a 30-year newspaper corpus², applied the morphological analyzer JUMAN (Kurohashi and Nagao, 1998b), and then applied the dependency analyzer with a posterior context model³. To ensure that we collected reliable co-occurrence information, we removed the information for the *bunsetsus* with punctuation⁴.

Like (Torisawa, 2001), we estimated the co-occurrence probability $P(\langle n, r, v \rangle)$ of the case element set (noun n , particle r , and verb v) by using probabilistic latent semantic indexing (PLSI) (Hofmann, 1999)⁵. If $\langle n, r, v \rangle$ is the co-occurrence of n and $\langle r, v \rangle$, we can calculate $P(\langle n, r, v \rangle)$ by using the following equation:

$$P(\langle n, r, v \rangle) = \sum_{z \in Z} P(n|z)P(\langle r, v \rangle|z)P(z), \quad (9)$$

where z indicates a latent semantic class of co-occurrence (hidden class). Probabilistic parameters $P(n|z)$, $P(\langle r, v \rangle|z)$, and $P(z)$ in Equation (9) can be estimated by using the EM algorithm. In our experiments, the dimension of the hidden class z was set to 300. As a result, the collected $\langle n, r, v \rangle$ total 102,581,924 pairs. The number of n and v is 57,315 and 15,098, respectively.

The particles for which the co-occurrence probability was estimated were the set of case particles, the “*ha*” case particle, and a class of “*fukujoshi*”

²13 years’ worth of articles from the Mainichi Shimbun, 14 years’ worth from the Yomiuri Shimbun, and 3 years’ worth from the Asahi Shimbun.

³We used the following package for calculation of Maximum Entropy:

http://homepages.inf.ed.ac.uk/s0450736/maxent_toolkit.html.

⁴The result of dependency analysis with a posterior context model for the Kyodai Corpus showed that the accuracy for the *bunsetsu* without punctuation is 90.6%, while the accuracy is only 76.4% for those with punctuation.

⁵We used the following package for calculation of PLSI: <http://chasen.org/~taku/software/plsi/>.

		<i>Bunsetsu</i> accuracy	Sentence accuracy
Whole data	Context model	90.95% (73,390/80,695)	54.40% (5,052/9,287)
	Our model	91.21% (73,603/80,695)	55.17% (5,124/9,287)
Only for reranked sentences	Context model	90.72% (68,971/76,026)	48.33% (3,813/7,889)
	Our model	91.00% (69,184/76,026)	49.25% (3,885/7,889)
Only for case elements	Context model	91.80% (28,849/31,427)	–
	Our model	92.47% (29,062/31,427)	–

Table 3: Accuracy before/after reranking

particles. Therefore, the total number of particles was 10.

We also estimated the co-occurrence probability of the particle set $P(rs|syn, v)$ by using PLSI. We regarded the triple $\langle rs, syn, v \rangle$ (the co-occurrence of particle set rs , verb v , and the syntactic property syn) as the co-occurrence of rs and $\langle syn, v \rangle$. The dimension of the hidden class was 100. The total number of $\langle rs, syn, v \rangle$ pairs was 1,016,508, v was 18,423, and rs was 1,490. The particle set should be treated not as a non-ordered set but as an occurrence ordered set. However, we think correct probability estimation using an occurrence ordered set is difficult, because it gives rise to an explosion in the number of combination,

5.4 Experimental environment

The evaluation data we used was Kyodai Corpus 3.0, a corpus manually annotated with dependency relations (Kurohashi and Nagao, 1998a). The statistics of the data are as follows:

- Training data: 24,263 sentences, 234,474 *bunsetsus*
- Development data: 4,833 sentences, 47,580 *bunsetsus*
- Test data: 9,287 sentences, 89,982 *bunsetsus*

The test data contained 31,427 case elements, and 28,801 verbs.

The evaluation measures we used were *bunsetsu* accuracy (the percentage of *bunsetsu* for which the correct modifyee was identified) and sentence accuracy (the percentage of sentences for which the correct dependency structure was identified).

5.5 Experimental results

5.5.1 Evaluation of our model

Our first experiment evaluated the effectiveness of reranking with our proposed model. *Bunsetsu*

		Our reranking model	
		correct	incorrect
Context model	correct	73,119	271
	incorrect	484	6,821

Table 4: 2×2 contingency table of the number of correct *bunsetsu* (posterior context model \times our model)

and sentence accuracies before and after reranking, for the entire set of test data as well as for only those sentences whose parse was actually reranked, are listed in Table 3.

The results showed that the accuracy could be improved by using our proposed model to rerank the results obtained with the posterior context model. McNemar testing showed that the null hypothesis that there is no difference between the accuracy of the results obtained with the posterior context model and those obtained with our model could be rejected with a p value < 0.01 . The difference in accuracy is therefore significant.

5.5.2 Comparing variant models

We next experimentally compare the following variations of the proposed model:

- one in which the case element set is assumed to be independent [Equation (7)]
- one using the co-occurrence probability of the particle set, $P(rs|syn, v)$, in our model
- one using only the co-occurrence probability of the case element, $P(n|r, v)$, in our model
- one not taking into account the syntactic property of a verb (*i.e.* a model in which the co-occurrence probability is defined as $P(r|v)$, without the syntactic property syn)
- one in which the co-occurrence probability of the case element, $P(n|r, v)$, is simply added

	<i>Bunsetsu</i> accuracy	Sentence accuracy
Context model	90.95%	54.40%
Our model	91.21%	55.17%
model (a)	91.12%	54.90%
model (b)	91.10%	54.69%
model (c)	91.11%	54.91%
model (d)	91.15%	54.82%
model (e)	90.96%	54.33%
model (f)	89.50%	48.33%
Kudo et al 2005	91.37%	56.00%

Table 5: Comparison of various models

to a feature set used in the posterior context model

- (f) one using only our proposed probabilities without the probability of the posterior context model

The accuracies obtained with each of these models are listed in Table 5, from which we can conclude that it is effective to take into account the dependency between case elements because model (a) is less accurate than our model.

Since the accuracy of model (d) is comparable to that of our model, we can conclude that the consideration of the syntactic property of a verb does not necessarily improve dependency analysis.

The accuracy of model (e), which uses the co-occurrence probability of the case element set as features in the posterior context model, is comparable to that of the posterior context model. This result is similar to the one obtained by (Kehler et al., 2004), where the task was anaphora resolution. Although we think the co-occurrence probability is useful information for dependency analysis, this result shows that simply adding it as a feature does not improve the accuracy.

5.5.3 Changing the amount of training data

Changing the size of the training data set, we investigated whether the degree of accuracy improvement due to reranking depends on the accuracy of the existing dependency analyzer.

Figure 3 shows that the accuracy improvement is constant even if the accuracy of the dependency analyzer is varied.

5.6 Discussion

The score used in reranking is the product of the probability of the posterior context model and the

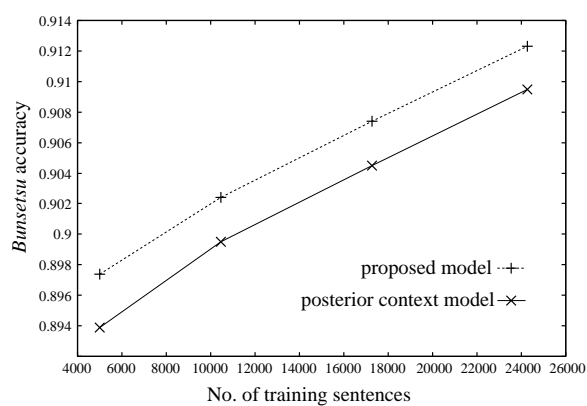


Figure 3: *Bunsetsu* accuracy when the size of the training data is changed

probability of our proposed model. The results in Table 5 show that the parsing accuracy of model (f), which uses only the probabilities obtained with our proposed model, is quite low. We think the reason for this is that our two co-occurrence probabilities cannot take account of syntactic properties, such as punctuation and the distance between two *bunsetsus*, which improve dependency analysis.

Furthermore, when the sentence has multiple verbs and case elements, the constraint of our proposed model tends to distribute case elements to each verb equally. To investigate such bias, we calculated the variance of the number of case elements per verb.

Table 6 shows that the variance for our proposed model (Equation [5]) is the lowest, and this model distributes case elements to each verb equally. The variance of the posterior context model is higher than that of the test data, probably because the syntactic constraint in this model affects parsing too much. Therefore the variance of the reranking model (Equation [8]), which is the combination of our proposed model and the posterior context model, is close to that of the test data.

The best parse which uses this data set is (Kudo and Matsumoto, 2005), and their parsing accuracy is 91.37%. The features and the parsing method used by their model are almost equal to the posterior context model, but they use a different method of probability estimation. If their model could generate n -best parsing and attach some kind of score to each parse tree, we would combine their model in place of the posterior context model.

At the stage of incorporating the proposed approach to a parser, the consistency with other pos-

	context model	test data	Equation [8]	Equation [5]
variance (σ^2)	0.724	0.702	0.696	0.666

*The average number of elements per verb is 1.078.

Table 6: The variance of the number of elements per verb

sible methods that deal with other relations should be taken into account. This will be one of our future tasks.

6 Conclusion

We presented a method of improving Japanese dependency parsing by using large-scale statistical information. Our method takes into account two types of information, not considered in previous statistical (machine learning based) parsing methods. One is information about the dependency relations among the case elements of a verb, and the other is information about co-occurrence relations between a verb and its case element. Experimental results showed that our method can improve the accuracy of the existing method.

References

- Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n -best parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 173–180.
- Michael Collins and Terry Koo. 2005. Discriminative reranking for natural language parsing. *Computational Linguistics*, 31(1):25–69.
- Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(3):245–288.
- James Henderson and Ivan Titov. 2005. Data-defined kernels for parse reranking derived from probabilistic models. In *Proceedings of the 43rd Annual Meeting of the ACL*, pages 181–188.
- Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22nd Annual International SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57.
- Andrew Kehler, Douglas Appelt, Lara Taylor, and Aleksandr Simma. 2004. The (non)utility of predicate-argument frequencies for pronoun interpretation. In *Proceedings of the HLT/NAACL 2004*, pages 289–296.
- Dan Klein and Christopher D. Manning. 2002. Fast exact inference with a factored model for natural language parsing. In *Advances in Neural Information Processing Systems 15 (NIPS 2002)*, pages 3–10.
- Taku Kudo and Yuji Matsumoto. 2002. Japanese dependency analysis using cascaded chunking. In *CoNLL 2002: Proceedings of the 6th Conference on Natural Language Learning 2002 (COLING 2002 Post-Conference Workshops)*, pages 63–69.
- Taku Kudo and Yuji Matsumoto. 2005. Japanese dependency parsing using relative preference of dependency. *Transactions of Information Processing Society of Japan*, 46(4):1082–1092. (in Japanese).
- Sadao Kurohashi and Makoto Nagao. 1994. Kn parser: Japanese dependency/case structure analyzer. In *Proceedings of the Workshop on Sharable Natural Language Resources*, pages 48–55.
- Sadao Kurohashi and Makoto Nagao. 1998a. Building a Japanese parsed corpus while improving the parsing system. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, pages 719–724.
- Sadao Kurohashi and Makoto Nagao. 1998b. *Japanese Morphological Analysis System JUMAN version 3.5*. Department of Informatics, Kyoto University. (in Japanese).
- Manabu Sassano. 2004. Linear-time dependency analysis for Japanese. In *Proceedings of the COLING 2004*, pages 8–14.
- Kiyoaki Shirai, Kentaro Inui, Takenobu Tokunaga, and Hozumi Tanaka. 1998. An empirical evaluation on statistical parsing of Japanese sentences using lexical association statistics. In *Proceedings of the 3rd Conference on EMNLP*, pages 80–87.
- Kiyoaki Shirai. 1998. The integrated natural language processing using statistical information. Technical Report TR98-0004, Department of Computer Science, Tokyo Institute of Technology. (in Japanese).
- Kentaro Torisawa. 2001. An unsupervised method for canonicalization of Japanese postpositions. In *Proceedings of the 6th Natural Language Processing Pacific Rim Symposium (NLPRS)*, pages 211–218.
- Kiyotaka Uchimoto, Satoshi Sekine, and Hitoshi Isahara. 1999. Japanese dependency structure analysis based on maximum entropy models. *Transactions of Information Processing Society of Japan*, 40(9):3397–3407. (in Japanese).
- Kiyotaka Uchimoto, Masaki Murata, Satoshi Sekine, and Hitoshi Isahara. 2000. Dependency model using posterior context. In *Proceedings of the Sixth International Workshop on Parsing Technology (IWPT2000)*, pages 321–322.