A Hybrid Japanese Parser with Hand-crafted Grammar and Statistics

Hiroshi Kanayama†, **Kentaro Torisawa**‡∗,
Yutaka Mitsuishi‡ and **Jun'ichi Tsujii**‡⋆

† Tokyo Research Laboratory, IBM Japan, Ltd.
1623-14 Shimo-tsuruma, Yamato-shi, Kanagawa 242-8502, Japan

‡ Department of Information Science, Graduate School of Science, University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-0033, Japan

∗ Information and Human Behavior, PRESTO, Japan Science and Technology Corporation
Kawaguchi Hon-cho 4-1-8, Kawaguchi-shi, Saitama 332-0012, Japan

⋆ CCL, UMIST, U.K.

{kanayama, torisawa, mitsuisi, tsujii}@is.s.u-tokyo.ac.jp

Abstract

This paper describes a hybrid parsing method for Japanese which uses both a hand-crafted grammar and a statistical technique. The key feature of our system is that in order to estimate likelihood for a parse tree, the system uses information taken from alternative partial parse trees generated by the grammar. This utilization of alternative trees enables us to construct a new statistical model called *Triplet/Quadruplet Model*. We show that this model can capture a certain tendency in Japanese syntactic structures and this point contributes to improvement of parsing accuracy on a shallow level. We report that, with an underspecified HPSG-based grammar and a maximum entropy estimation, our parser achieved high accuracy: 88.6% accuracy in dependency analysis of the EDR annotated corpus, and that it outperformed other purely statistical parsing methods on the same corpus. This result suggests that proper treatment of hand-crafted grammars can contribute to parsing accuracy on a shallow level.

1 Introduction

There have been many attempts to combine hand-crafted high-level grammars, such as FB-LTAG, HPSG and LFG, and statistical disambiguation techniques to obtain precise linguistic structures (Schabes, 1992; Abney, 1996; Carroll et al., 1998). One evident advantage of this approach over purely statistical parsing techniques is that grammars can provide precise semantic representations. However, considering that remarkable parsing accuracy in a shallow level has been achieved by purely statistical techniques (e.g. Ratnaparkhi (1997)), it may be thought more reasonable to use high-level grammars just for postprocessing which maps results of shallow syntactical analyses onto deep analyses.

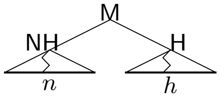

Figure 1: A tree M with a non-head daughter NH and a head daughter H.

In this work we propose that hand-crafted high-level grammars can be useful in shallow-level analyses and statistical models. In our framework, grammars are used to obtain precise features for probability estimation, which are difficult to obtain without a grammar, and we show that such features contribute to high parsing accuracy on a shallow level.

In this paper, the most preferable parse trees are chosen with a statistical model. In our method, the likelihood value $L(\mathsf{M})$ of a (partial) tree M in Figure 1 is defined as in (1):

$$L(\mathsf{M}) \overset{\text{def}}{=} L(\mathsf{NH}) \times L(\mathsf{H}) \times P(n \to h) \qquad (1)$$

where NH is M's non-head daughter (whose lexical head is n), H is the head-daughter (whose lexical head is h), and $P(n \to h)$ is the probability of n being related to h. For a single lexical item W, $L(\mathsf{W})$ is defined as 1.0.

In most models already proposed, the probability $P(n \to h)$ is calculated with the conditional probability (2):

$$P(n \to h) \overset{\text{def}}{=} P(\mathrm{T} \mid \Phi_n, \Psi_h, \Delta_{n,h}) \qquad (2)$$

where T indicates that the dependency is true; Φ_n and Ψ_h are attributes of n and h, respectively. And $\Delta_{n,h}$, the distance between the two words, is widely used, because this attribute is believed to strongly affect whether those two words are going to be related.

In contrast, in the statistical model proposed in this paper, $P(n \to h)$ depends not only on the attributes of the tree M, but also on alternative trees

This work was conducted while the first author was a graduate student at Univ. of Tokyo.

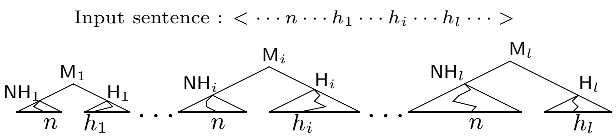

Figure 2: Partial trees whose non-head daughter's lexical head is n.

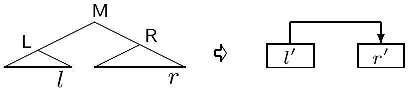

Figure 3: Transformation from a tree to a dependency. l' and r' denote the *bunsetsu*s l and r belong to, respectively.

in the parse forest generated by the grammar. More precisely, when $P(n \rightarrow h)$ is calculated, we consider partial trees whose non-head daughter's lexical head is n, as displayed in Figure 2. Here alternative possible h_k ($k = 1, \cdots, l$) are taken into consideration, and ordered according to their distance to n. We call such set of h_k *modification candidates*, and all modification candidates are placed together in the conditional part of the probability as in (3). Now assume $h = h_i$.

$$P(n \rightarrow h_i) \overset{\text{def}}{=} P(i \mid \Phi_n, \Psi_{h_1}, \Psi_{h_2}, \cdots, \Psi_{h_i}, \cdots, \Psi_{h_l}) \quad (3)$$

where "i" indicates the ith candidate among the modification candidates. Equation (3) shows two important properties of our model. One point lies in *the new distance metric*. (3) is the probability that n chooses the ith candidate as the modifiee among the modification candidates which are ordered according to their distance to n. Thus, we no longer require the distance metric $\Delta_{n,h}$, instead we use the *relative position* among the modification candidates, which works as an attribute of the modification. The other point is the *use of the attributes of the alternative parse trees*, that is, attributes of the modifier and all its modification candidates are considered *simultaneously*. We show that these techniques sophisticate our model, by providing linguistic examples in Section 3.2.

In practice, however, treating all candidates is not feasible because of data-sparseness. We therefore apply a strategy of restricting the modification candidates to at most three. The strategy and its justification are discussed in Section 3.1.

Applying the strategy to the equation (3), we obtain equations (4) and (5):

$$P(n \rightarrow h_i) \overset{\text{def}}{=} P(i \mid \Phi_n, \Psi_{h_1}, \Psi_{h_2}) \qquad (i = 1, 2) \quad (4)$$

$$P(n \rightarrow h_i) \overset{\text{def}}{=} P(i \mid \Phi_n, \Psi_{h_1}, \Psi_{h_2}, \Psi_{h_l}) \ (i = 1, 2, l) (5)$$

When there are only two candidates, equation (4) is used; otherwise, equation (5) is used. Our statistical model is called the *Triplet/Quadruplet Model*, which was named after the number of constituents in the conditional parts of the equations.

We report that our parsing framework achieved high accuracy (88.6%) in dependency analysis of Japanese with a combination of an underspecified HPSG-based Japanese grammar, SLUNG (Mitsuishi et al., 1998) and the maximum entropy method (Berger et al., 1996). Moreover, the resulting parse trees generated by our hybrid parser are legitimate trees in terms of given hand-crafted grammars, and we are expecting that we can enjoy advantages provided by high-level grammar formalisms, such as construction of semantic structures.

In the above explanation, we used the notion of lexical heads for the estimation of probabilities of trees for the sake of simplicity. But, in the present implementation, we use *bunsetsu*s instead of lexical heads, and a relation on a tree is converted to a *bunsetsu*-dependency as shown in Figure 3. A *bunsetsu* is a basic syntactic unit in Japanese. It consists of a content word and some functional morphemes such as a particle.

In Section 2, we describe some existing statistical parsers, and the Japanese grammar which we adopted. Section 3 describes our statistical method and its advantages in detail. We report experimental results in Section 4.

2 Background

In this section, we describe several models for Japanese dependency analysis and works on statistical approaches with grammars. Next, we introduce SLUNG, the HPSG-based Japanese grammar which is used in our hybrid parser.

2.1 Previous Dependency Analysis Models of Japanese

Several statistical models for Japanese dependency analysis which do not utilize a hand-crafted grammar have been proposed. We evaluate the accuracy of *bunsetsu*-dependencies as they do, thus here we introduce them for comparison. All models introduced below are based on the likelihood value of the dependency between two *bunsetsu*s. But they differ from each other in the attributes or outputs which are considered when a likelihood value is calculated.

There are some models which calculate the likelihood values of a dependency between *bunsetsu* i and j as in (6), such as a decision tree model (Haruno et al., 1998), a maximum entropy model (Uchimoto et al., 1999), a model based on distance and lexical information (Fujio and Matsumoto, 1998). Attributes Φ_i and Ψ_j consist of a part-of-speech (POS), a lexical item, presence of a comma, and so on. And $\Delta_{i,j}$

is the number of intervening *bunsetsu*s between i and j.

$$P(i \rightarrow j) \overset{\text{def}}{=} P(\mathrm{T} \mid \Phi_i, \Psi_j, \Delta_{i,j}) \quad (6)$$

However, these models fail to reflect contextual information because attributes of the surrounding *bunsetsu*s are not considered.

Uchimoto et al. (2000) proposed a model using *posterior context*. The model utilizes not only attributes about *bunsetsu*s i, j but also attributes about all *bunsetsu*s (including j) which follow *bunsetsu* i. That is, instead of learning two output values "T(true)" or "F(false)" for the dependency between two *bunsetsu*s, three output values are used for learning: the *bunsetsu* i is "bynd (dependent on a *bunsetsu* beyond j)", "dpnd (dependent on the *bunsetsu* j)" or "btwn (dependent on a *bunsetsu* between i and j)". The probability is calculated by multiplying probabilities for all *bunsetsu*s which follow *bunsetsu* i as in (7). They report that this kind of contextual information improves accuracy. However, the model has to assume the independency of all the random variables, which may cause some errors.

$$P(i \rightarrow j) \overset{\text{def}}{=} \prod_{i<k<j} P(\mathrm{bynd} \mid \Phi_i, \Psi_k, \Delta_{i,k}) \times P(\mathrm{dpnd} \mid \Phi_i, \Psi_j, \Delta_{i,j}) \times \prod_{k>j} P(\mathrm{btwn} \mid \Phi_i, \Psi_k, \Delta_{i,k}) \quad (7)$$

The difference between our model and these previous models are discussed in Section 3.

2.2 Statistical Approaches with a grammar

There have been many proposals for statistical frameworks particularly designed for parsers with hand-crafted grammars (Schabes, 1992; Briscoe and Carroll, 1993; Abney, 1996; Inui et al., 1997). The main issue in this type of research is how to assign likelihoods to a single linguistic structure generated by a grammar. Some of them (Briscoe and Carroll, 1993; Inui et al., 1997) treat information on contexts, but the contextual information is derived only from a structure to which the parser is trying to assign a likelihood value. Then, the major difference between their method and ours is that we consider the attributes of alternative linguistic structures generated by the grammar in order to determine the likelihood for linguistic structures.

2.3 SLUNG : Japanese Grammar

The Japanese grammar which we adopted, SLUNG (Mitsuishi et al., 1998), is an HPSG-based underspecified grammar. It consists of 8 rule schemata, 48 lexical templates for POSs and 105 lexical entries for functional words. As can be seen from these figures, the grammar does not contain detailed lexical information that needs intensive labor for development. However, it is precise in the sense that it achieves 83.7% dependency accuracy with a simple heuristics[2] for the EDR annotated corpus, and it can produce at least one parse tree for 98.4% sentences in the EDR annotated corpus. We use the grammar for generating parse tree forests, and our Triplet/Quadruplet Model is used for picking up a single tree from a forest.

3 The Hybrid Parsing Method

This section describes the procedure of parsing with the Triplet/Quadruplet Model. Our hybrid parsing method proceeds as follows:

- At the beginning, dependency structures are obtained from trees generated by SLUNG. For each *bunsetsu*, modification candidates are enumerated, and if there are four or more candidates, they are restricted to three. The heuristic used in this process is described in Section 3.1.
- Then, with the *Triplet/Quadruplet Model* and maximum entropy estimation, probabilities of the dependencies are calculated. Section 3.2 discusses the characteristics and advantages of the model.
- Finally, the most preferable trees for the whole sentence are selected.

3.1 Restriction of Modification Candidates

Kanayama et al. (1999) report that when modification candidates are enumerated according to SLUNG, 98.6% of the correct modifiees are in one of the following three positions among the candidates: the nearest one from the modifier, the second nearest one, and the farthest one.

As a consequence, we can simplify the problem by considering only these three candidates and discarding the other candidates, with only 1.4% potential errors. We therefore assume that the number of modification candidates is always three or less.

This idea is similar to that of Sekine (2000)'s study, which restricts the candidates to five, but in his case, without a grammar.

3.2 The Triplet/Quadruplet Model

The Triplet/Quadruplet Model calculates the likelihood of the dependency between *bunsetsu* i and *bunsetsu* c_n; $P(i \rightarrow c_n)$ with the formulas (8) and (9), where c_n denotes the nth candidate among *bunsetsu* i's candidates; Φ_i denotes some attributes of i; and Ψ_{c_n} denotes attributes of c_n (including attributes between i and c_n).

$$P(i \rightarrow c_n) \overset{\text{def}}{=} P(n \mid \Phi_i, \Psi_{c_1}, \Psi_{c_2}) \qquad (n = 1, 2) \quad (8)$$

$$P(i \rightarrow c_n) \overset{\text{def}}{=} P(n \mid \Phi_i, \Psi_{c_1}, \Psi_{c_2}, \Psi_{c_l}) \ (n = 1, 2, l) (9)$$

[2] This heuristics is a Japanese version of a left-association rule: see (Mitsuishi et al., 1998) for detail.

As (8) and (9) suggest, the model considers attributes of the modifier *bunsetsu* and attributes of all modification candidates *simultaneously* in the conditional parts of the probabilities. Moreover, what is calculated is not the probability of "whether the dependency is correct (T, see Formula(6))", but the probability of "which of the given candidates is chosen as the modifiee (n =1, 2, or l)". These characteristics imply the following two advantages.

Advantage 1 *A new distance metric.* The correct modifiee can be chosen by considering relative position among grammatically licensed candidates, instead of the absolute distance between *bunsetsu*s.

Advantage 2 *Treating alternative trees.* The candidates are taken into consideration simultaneously. But because the modification candidates are restricted to at most three, we considerably avoid data-sparseness problems.

Below we discuss these advantages in order. These advantages clarify the differences from previous models described in Section 2.1, and are empirically confirmed through the experiments in Section 4.

3.2.1 Advantage 1 : A new distance metric

As discussed in Section 2.1, the distance metric $\Delta_{i,j}$ used in previous statistical methods was obtained simply by counting intervening words or *bunsetsu*s between i and j. On the other hand, we use the relative position among the modification candidates as the distance metric. The following examples illustrate a difference between those two types of metric. The correct modifiee of *kare-ga* is *hashiru-no-wo* in both (10a) and (10b).

(10) a. *kare-ga* *hashiru-no-wo* *mita* *koto*
he-SUBJ run see fact
(the fact that I saw him run)

b. *kare-ga* *yukkuri* *hashiru-no-wo* *mita* *koto*
he-SUBJ slowly run see fact
(the fact that I saw him run slowly)

In previous models, (10a) and (10b) would yield,

$$P_a(\textit{kare-ga} \rightarrow \textit{hashiru-no-wo}) = P(\mathrm{T}|\textit{kare-ga},\ \textit{hashiru-no-wo}, \Delta_1)$$
$$P_b(\textit{kare-ga} \rightarrow \textit{hashiru-no-wo}) = P(\mathrm{T}|\textit{kare-ga},\ \textit{hashiru-no-wo}, \Delta_2)$$

respectively, where $\Delta_1 = 1$ and $\Delta_2 = 2$. Then, the two probabilities above do not have the same value in general.

Our grammar does not allow the dependency "*kare-ga* →*yukkuri*" for (10b). The modification candidates of *kare-ga* are *hashiru-no-wo* and *mita*, hence (8) gives the probabilities between *kare-ga* and *hashiru-no-wo* as follows, in both examples.

$$\begin{aligned} & P_a(\textit{kare-ga} \rightarrow \textit{hashiru-no-wo}) \\ = {} & P_b(\textit{kare-ga} \rightarrow \textit{hashiru-no-wo}) \\ & \qquad = P(1|\textit{kare-ga},\ \textit{hashiru-no-wo},\ \textit{mita}) \end{aligned}$$

Thus, $P(\textit{kare-ga} \rightarrow \textit{hashiru-no-wo})$ has the same value for both examples. Our interpretation of this difference is summarized as follows. The word *yukkuri* is an adverb modifying the verb *hashiru*. Our linguistic intuition tells us that the presence of such adverb should not affect the strength for the dependency between *kare-ga* and *hashiru-no-wo*. According to this intuition, the existence of the adverb should be considered as a noise. Our model allows us to ignore such a noise in learning from annotated corpus, while previous models are affected by such noisy elements.

3.2.2 Advantage 2 : Treating alternative trees or contextual information

Consider the following examples.

(11) a. *Taro-no* *kawaii* *musume*
NP Adj NP
Taro-POSS pretty daughter
(Taro's pretty daughter)

b. *Taro-no* *yuujin-no* *musume*
NP NP NP
Taro-POSS friend-POSS daughter
(Taro's friend's daughter)

Contrary to the previous examples, *Taro-no* in (11) modifies different modification candidates. In example (11a), "*Taro-no* →*musume*" is the correct dependency while "*Taro-no* →*musume*" is not correct in (11b). This difference is caused by the *bunsetsu* between *Taro-no* and *musume*, *kawaii* (Adj) in (11a) and *yuujin-no* (NP) in (11b). Actually, the grammar allows *Taro-no* to depend on either of these types of words. Thus, in our model,

$$\begin{aligned} & P_a(\textit{Taro-no} \rightarrow \textit{musume}) \\ & \qquad = P(2|\textit{Taro-no},\ \textit{kawaii},\ \textit{musume}) \\ & P_b(\textit{Taro-no} \rightarrow \textit{musume}) \\ & \qquad = P(2|\textit{Taro-no},\ \textit{yuujin-no},\ \textit{musume}) \end{aligned}$$

Then, $P(\textit{Taro-no} \rightarrow \textit{musume})$ has different values for the two examples. In the annotated corpus, $P(2|\textit{Taro-no},\ \textit{kawaii},\ \textit{musume})$ tends to have a high value since *kawaii* is an adjective. However, since *yuujin-no* is an NP, $P(2|\textit{Taro-no},\ \textit{yuujin-no},\ \textit{musume})$ tends to have a low value.

Now consider previous models.

$$P_a(\textit{Taro-no} \rightarrow \textit{musume}) = P(\mathrm{T}|\textit{Taro-no},\ \textit{musume},\ 2)$$
$$P_b(\textit{Taro-no} \rightarrow \textit{musume}) = P(\mathrm{T}|\textit{Taro-no},\ \textit{musume},\ 2)$$

Then, contrary to our model, $P(\textit{Taro-no} \rightarrow \textit{musume})$ has exactly the same value for both examples. The outcome is determined by

$$P_a(\textit{Taro-no} \rightarrow \textit{kawaii}) = P(\mathrm{T}|\textit{Taro-no},\ \textit{kawaii},\ 1)$$

In text corpora, $P(\mathrm{T}|\textit{Taro-no},\ \textit{yuujin-no},\ 1)$ tends to be high, and consequently, $P(\mathrm{T}|\textit{Taro-no},\ \textit{musume},\ 2)$ is very small. These values will make the correct prediction for (11b) as *yuujin-no* will be favored over *musume*. However, for (11a), these models are likely to incorrectly favor *kawaii* over *musume*. This is because $P(\mathrm{T}|\,\textit{Taro-no},\ \textit{musume},\ 2)$, being very small, is likely to be smaller than $P(\mathrm{T}|\,\textit{Taro-no},\ \textit{kawaii},\ 1)$.

4 Experiments and Discussion

This section reports a series of parsing experiments with our model, and gives some discussion.

4.1 Environments

We used the EDR Japanese Corpus (EDR, 1996) for training and evaluation of parsing accuracy. The EDR Corpus is a Japanese treebank which consists of 208,157 sentences from newspapers and magazines. We used 192,778 sentences for training, 6,744 for pre-analysis (as reported in Section 3.1), and 3,372 for testing[3].

With triplets constituted of a modifiee and two modification candidates extracted from the learning corpus the Triplet Model is constructed. With the quadruplets constituted of a modifiee and three candidates, the Quadruplet Model is constructed. These models are estimated by the ChoiceMaker Maximum Entropy Estimator (Borthwick, 1999).

The features for the estimation are listed in Table 1. The values partially follow other researches e.g. Uchimoto et al. (1999), and JUMAN's outputs are used for POS classification. Mainly the **head** of the *bunsetsu* (the rightmost morpheme in a *bunsetsu* except for whose major POS is "peculiar", "auxiliary verb", "particle", "suffix" or "copula") and **type** of the *bunsetsu* (the rightmost morpheme in a *bunsetsu* except for whose major POS is "peculiar") are used as the attributes. We show the meaning of some features below.

POS JUMAN's minor POS (for both "head" and "type").

particle, adverb Frequent words: 26 particles and 69 adverbs.

head lex 294 lexical forms regardless of their POS.

type lex 70 suffixes or auxiliary verbs.

inflection 6 types of inflection : "normal", "adverbial", "adnominal", "*te*-form", "*ta*-form", and "others".

The column "variation" in Table 1 denotes the number of possible values for the feature. "Valid features" indicates the number of features which appeared three times or more in the training corpus.

4.2 Results

With our model and the features described above, the accuracy shown in Table 2 is achieved. We evaluate the following two types of accuracy:

ID	Feature type	Variation	Valid features	
			Trip.	Quad.
1	Head POS of modifier	24	42	64
2	Type POS of modifier	34	66	99
3	Particle of modifier	27	47	73
4	Adverb of modifier	70	131	193
5	Type lex of modifier	71	110	225
6	Inflection of modifier	6	12	18
7	Whether modifier has a comma	2	4	6
8	Head POS of modifiee	24	70	158
9	Type POS of modifiee	34	96	231
10	Head lex of modifiee	295	1164	2597
11	Particle of modifiee	27	92	204
12	Type lex of modifiee	71	216	454
13	Inflection of modifiee	6	24	53
14	Whether modifiee has a comma	2	8	18
15	Whether modifiee has "*wa*"	2	8	18
16	Whether modifiee has "*to*"	2	6	17
17	# of commas between two *bunsetsu*s	4	16	36
18	# of "*wa*" between two *bunsetsu*s	3	12	27
19	2 × 8	816	1187	2727
20	2 × 7 × 14	136	380	870
21	3 × 10	7965	6465	13463
22	2 × 9	1156	1213	3108
23	3 × 11	729	618	1637
24	2 × 11	918	1025	2494
25	2 × 12	2414	1483	3514
26	2 × 3 × 7 × 8	132192	1331	3058
27	1 × 2 × 6 × 8 × 13	705024	6605	14700
	Total	-	22433	50063

Table 1: Used features : Features from 8 to 27 are related to the modifiee, thus they are considered for each candidate. Features from 19 to 27 are combination features.

In-coverage sentences	*Bunsetsu* accuracy	**88.55**%(23078/26062)
	Sentence accuracy	46.90% (1560/3326)
All sentences	*Bunsetsu* accuracy	88.33% (23350/26436)
	Sentence accuracy	46.35% (1563/3372)

Table 2: Results of parsing with the Triplet/Quadruplet Model.

Bunsetsu accuracy The percentage of *bunsetsu*s whose modifiee is correctly identified. The denominator includes all *bunsetsu*s except for the last *bunsetsu* of a sentence.

Sentence accuracy The percentage of sentences whose dependencies are perfectly correct.

"In-coverage sentences" is the accuracy for the sentences for which SLUNG could generate parse trees. We give the accuracy for "All sentences" too, by partially parsing sentences which SLUNG fail to parse. The coverage of SLUNG is about 99%, thus high accuracy is achieved even for "All sentences".

Moreover, we conducted a series of experiments in order to evaluate the contribution of each characteristic in our parsing model. The parsing schemes used are the four in Figure 3. Major differences among them are (I) whether a grammar is used, (II) whether modification candidates are restricted to three, and (III) whether a previous pair model with Formula (6) or the Triplet/Quadruplet Model with Formula (8),(9) was used.

W/O Grammar Model This model does not use a grammar. Likelihood values for dependen-

[3] 5,263 sentences were removed because the order of the words in the annotation differed from that in the original sentences.

	G	R	F	*Bunsetsu* accuracy
W/O Grammar	−	−	P	86.70%(22594/26062)
W/O Restriction	+	−	P	87.37%(22770/26062)
Pair	+	+	P	87.67%(22849/26062)
Triplet/Quadruplet	+	+	T	**88.55**%(23078/26062)

Table 3: *Bunsetsu* accuracies for four models. Column "G" indicates whether the grammar is used, "R" indicates whether the modification candidates are restricted to three, and "F" denotes the formula; "P" is the pair formula (6), and "T" is the Triplet/Quadruplet formula (8), (9).

cies are calculated for all *bunsetsu*s that follow a modifier *bunsetsu*. Formula (6) is used, and as a distance metric $\Delta_{i,j}$, the number of *bunsetsu*s between the modifier and the modifiee[4] are combined with all features. In general lines, this model corresponds to models such as (Fujio and Matsumoto, 1998; Haruno et al., 1998; Uchimoto et al., 1999).

W/O Restriction Model Modification candidates are restricted by SLUNG. The remaining is the same as the W/O Grammar Model.

Pair Model Modification candidates are restricted to three, in the way described in Section 3.1. The remaining is the same as W/O Grammar Model.

Triplet/Quadruplet Model This is the model proposed in the paper. Modification candidates are restricted to three, and Formula (8) or (9) are used.

From the result shown in Table 3, we can say our method contributes to the improvement of our parser, because of the following reasons:

- The Triplet/Quadruplet Model outperforms the Pair Model by 0.9%. Both of them restricts modification candidates to three, but the accuracy got higher when all candidates are considered simultaneously. It is because of the two advantages described in Section 3.2.
- The Pair Model outperforms the W/O Restriction Model by 0.3%. Thus the restriction of modification candidates does not reduce the accuracy.
- The W/O Restriction Model outperforms the W/O Grammar Model by 0.7%. This means that the use of a grammar as a preprocessor works well to pick up possible modifiee.

We found that many structures similar to the ones described in Section 3.2 appeared in the EDR corpus. Our Triplet/Quadruplet model could treat these structures precisely as we intended. This is the main factor that contributed to the improvement of the overall parsing accuracy.

In-coverage sentences	*Bunsetsu* accuracy	87.08%	(8299/9530)
	Sentence accuracy	44.70%	(493/1103)

Table 4: Accuracy for Kyoto University Corpus

Based on the above experiments, we can say that our approach to use the grammar as a preprocessor before the calculating of the probability is appropriate for the improvement of parsing accuracy.

4.3 Comparison to other models

4.3.1 Models using the EDR corpus

There are several works which use the EDR corpus for evaluation. The decision tree model (Haruno et al., 1998) achieves around 85%, the integrated model of lexical/syntactic information (Shirai et al., 1998) achieves around 86%, and the lexicalized statistical model (Fujio and Matsumoto, 1999) achieves 86.8% in *bunsetsu* accuracy. Our model outperforms all of them by 2 or 3%.

4.3.2 Models using the Kyoto corpus

Shirai et al. (1998) used the Kyoto University text corpus (Kurohashi and Nagao, 1997) for evaluation and achieved around 86%. Uchimoto et al. (2000) also used the Kyoto corpus, and their accuracy was 87.9%. For comparison, we applied our method to the same 1,246 sentences that Uchimoto et al. (2000) used. The result is shown in Table 4.

Our result is worse than theirs. The reason is thought to be as follows:

- We use the EDR corpus for training. Although we used around 24 times the amount of training data that Uchimoto et al. used, our training data lead to errors in the analysis of the Kyoto Corpus, because of differences in the annotation schemes adopted.
- Uchimoto et al. used the correct morphological analyses, but we used JUMAN. Sometimes this may cause errors.
- The grammar SLUNG was designed for the EDR corpus, and some types of structures in the Kyoto Corpus are not allowed.

Clearly, our parser should be improved to overcome these problems and compared with other works directly.

4.4 Discussion and Future Work

The following are some observations about the speed of our parser. Existing statistical parsers are quite efficient compared to grammar-based systems. Particularly, our system used an HPSG-based grammar,

[4] Three values: "1", "from 2 to 5", "6 or more" are distinguished.

whose speed is said to be slow. However, recent advances in HPSG parsing (Torisawa et al., 2000) enabled us to obtain a unique parse tree with our system in 0.5 sec. in average for sentences in the EDR corpus.

Future work shall extend SLUNG so that semantic representations are produced. Carroll et al. (1998) discussed the precision of argument structures. We believe that the focus of our study will shift from a shallow level to such a deeper level for our final aim, realization of intelligent natural language processing systems.

5 Conclusion

We presented a hybrid parsing scheme that uses a hand-crafted grammar and a statistical technique. As other hybrid parsing methods, the statistical technique is used for picking up the most preferable parse tree from the parse forest generated by the grammar. The difference from other works is that the precise contextual information needed to estimate the likelihood of a parse tree is obtained from alternative parse trees generated by the grammar, and that such contextual information from alternative trees enables us to construct our new statistical model called the Triplet/Quadruplet model. We have shown that these points contributed to substantial improvement of parsing accuracy in Japanese dependency analysis, through a series of experiments using an HPSG-based Japanese grammar SLUNG and the maximum entropy method.

References

Steven Abney. 1996. Stochastic attribute-value grammars. The Computation and Language E-Print Archive, October.

Adam L. Berger, Stephen A. Della Pietra, and Vincent. J. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Andrew Borthwick. 1999. Choicemaker maximum entropy estimator. ChoiceMaker Tech., Inc. Email borthwic@cs.nyu.edu for information.

Ted Briscoe and John Carroll. 1993. Generalized probabilistic LR parsing of natural language (corpora) with unification-based grammars. *Computational Linguistics*, 19(1):25–50.

John Carroll, Guido Minnen, and Ted Briscoe. 1998. Can subcategorisation probabilities help a statistical parser? In *Proc. of the 6th ACL/SIGDAT Workshop on Very Large Corpora*, pages 118–126.

EDR. 1996. EDR (Japan Electronic Dictionary Research Institute, Ltd.) dictionary version 1.5 technical guide. Second edition is available via http://www.iijnet.or.jp/edr/E_TG.html.

Masakazu Fujio and Yuji Matsumoto. 1998. Japanese dependency structure analysis based on lexicalized statistics. In *Proc. of the 3rd Conference on Empirical Methods in Natural Language Processing*, pages 88–96.

Masakazu Fujio and Yuuji Matsumoto. 1999. Statistical syntactic analysis based on co-occurrence probability of words. In *Proc. of 5th workshop of Natural Language Processing*, pages 71–78. (in Japanese).

Masahiko Haruno, Satoshi Shirai, and Yoshifumi Ooyama. 1998. Using decision trees to construct a practical parser. In *Proc. COLING–ACL '98*, pages 505–511.

Kentaro Inui, Virach Sornlertlamvanich, Hozumi Tanaka, and Takenobu Tokunaga. 1997. A new probabilistic LR language model for statistical parsing. Technical Report TR97-0005, Dept. of Computer Science, Tokyo Institute of Technology.

Hiroshi Kanayama, Kentaro Torisawa, Yutaka Mitsuishi, and Jun'ichi Tsujii. 1999. Statistical dependency analysis with an HPSG-based Japanese grammar. In *Proc. 5th NLPRS*, pages 138–143.

Sadao Kurohashi and Makoto Nagao. 1997. Kyoto University text corpus project. In *Proc. of 3rd Annual Meeting of Natural Language Processing*, pages 115–118. (in Japanese).

Yutaka Mitsuishi, Kentaro Torisawa, and Jun'ichi Tsujii. 1998. HPSG-style underspecified Japanese grammar with wide coverage. In *Proc. COLING–ACL '98*, pages 876–880, August.

Adwait Ratnaparkhi. 1997. A linear observed time statistical parser based on maximum entropy models. In *Proc. the Empirical Methods in Natural Language Processing Conference.*

Yves Schabes. 1992. Stochastic lexicalized tree-adjoining grammars. In *Proc. 14th COLING*, pages 426–432.

Satoshi Sekine. 2000. Japanese dependency analysis using a deterministic finite state transducer. In *Proc. COLING 2000.* (this proceedings).

Kiyoaki Shirai, Kentaro Inui, Takenobu Tokunaga, and Hozumi Tanaka. 1998. A framework of integrating syntactic and lexical statistics in statistical parsing. *Journal of Natural Language Processing*, 5(3). (in Japanese).

Kentaro Torisawa, Kenji Nishida, Yusuke Miyao, and Jun'ichi Tsujii. 2000. An HPSG parser with CFG filtering. *Journal of Natural Language Engineering.* (to appear).

Kiyotaka Uchimoto, Satoshi Sekine, and Hitoshi Isahara. 1999. Japanese dependency structure analysis based on maximum entropy models. In *Proc. 13th EACL*, pages 196–203.

Kiyotaka Uchimoto, Masaki Murata, Satoshi Sekine, and Hitoshi Isahara. 2000. Dependency model using posterior context. In *Proc. of Sixth International Workshop on Parsing Technologies.*