Japanese Dependency Structure Analysis Based on Maximum Entropy Models

Kiyotaka Uchimoto[†] **Satoshi Sekine**[‡] **Hitoshi Isahara**[†]

[†]Communications Research Laboratory
Ministry of Posts and Telecommunications
588-2, Iwaoka, Iwaoka-cho, Nishi-ku
Kobe, Hyogo, 651-2401, Japan
[uchimoto|isahara]@crl.go.jp

[‡]New York University
715 Broadway, 7th floor
New York, NY 10003, USA
sekine@cs.nyu.edu

Abstract

This paper describes a dependency structure analysis of Japanese sentences based on the maximum entropy models. Our model is created by learning the weights of some features from a training corpus to predict the dependency between bunsetsus or phrasal units. The dependency accuracy of our system is 87.2% using the Kyoto University corpus. We discuss the contribution of each feature set and the relationship between the number of training data and the accuracy.

1 Introduction

Dependency structure analysis is one of the basic techniques in Japanese sentence analysis. The Japanese dependency structure is usually represented by the relationship between phrasal units called 'bunsetsu.' The analysis has two conceptual steps. In the first step, a dependency matrix is prepared. Each element of the matrix represents how likely one bunsetsu is to depend on the other. In the second step, an optimal set of dependencies for the entire sentence is found. In this paper, we will mainly discuss the first step, a model for estimating dependency likelihood.

So far there have been two different approaches to estimating the dependency likelihood. One is the rule-based approach, in which the rules are created by experts and likelihoods are calculated by some means, including semiautomatic corpus-based methods but also by manual assignment of scores for rules. However, hand-crafted rules have the following problems.

- They have a problem with their coverage. Because there are many features to find correct dependencies, it is difficult to find them manually.
- They also have a problem with their consistency, since many of the features compete with each other and humans cannot create consistent rules or assign consistent scores.
- As syntactic characteristics differ across different domains, the rules have to be changed when the target domain changes. It is costly to create a new hand-made rule for each domain.

Another approach is a fully automatic corpus-based approach. This approach has the potential to overcome the problems of the rule-based approach. It automatically learns the likelihoods of dependencies from a tagged corpus and calculates the best dependencies for an input sentence. We take this approach. This approach is taken by some other systems (Collins, 1996; Fujio and Matsumoto, 1998; Haruno et al., 1998). The parser proposed by Ratnaparkhi (Ratnaparkhi, 1997) is considered to be one of the most accurate parsers in English. Its probability estimation is based on the maximum entropy models. We also use the maximum entropy model. This model learns the weights of given features from a training corpus. The weights are calculated based on the frequencies of the features in the training data. The set of features is defined by a human. In our model, we use features of bunsetsu, such as character strings, parts of speech, and inflection types of bunsetsu, as well as information between bunsetsus, such as the existence of punctuation, and the distance between bunsetsus. The probabilities of dependencies are estimated from the model by using those features in input sentences. We assume that the overall dependencies in a whole sentence can be determined as the product of the probabilities of all the dependencies in the sentence.

Now, we briefly describe the algorithm of dependency analysis. It is said that Japanese dependencies have the following characteristics.

(1) Dependencies are directed from left to right

(2) Dependencies do not cross

(3) A bunsetsu, except for the rightmost one, depends on only one bunsetsu

(4) In many cases, the left context is not necessary to determine a dependency[1]

The analysis method proposed in this paper is designed to utilize these features. Based on these properties, we detect the dependencies in a sentence by analyzing it backwards (from right to left). In the past, such a backward algorithm has been used with rule-based parsers (e.g., (Fujita, 1988)). We applied it to our statistically based approach. Because of the statistical property, we can incorporate a beam search, an effective way of limiting the search space in a backward analysis.

2 The Probability Model

Given a tokenization of a test corpus, the problem of dependency structure analysis in Japanese can be reduced to the problem of assigning one of two tags to each relationship which consists of two bunsetsus. A relationship could be tagged as "0" or "1" to indicate whether or not there is a dependency between the bunsetsus, respectively. The two tags form the space of "futures" for a maximum entropy formulation of our dependency problem between bunsetsus. A maximum entropy solution to this, or any other similar problem allows the computation of $P(f|h)$ for any f from the space of possible futures, F, for every h from the space of possible histories, H. A "history" in maximum entropy is all of the conditioning data which enables you to make a decision among the space of futures. In the dependency problem, we could reformulate this in terms of finding the probability of f associated with the relationship at index t in the test corpus as:

$$P(f|h_t) = P(f|\text{ Information derivable from the test corpus related to relationship } t)$$

The computation of $P(f|h)$ in M.E. is dependent on a set of "features" which, hopefully, are helpful in making a prediction about the future. Like most current M.E. modeling efforts in computational linguistics, we restrict ourselves to features which are binary functions of the history and future. For instance, one of our features is

$$g(h,f) = \begin{cases} 1 : \text{if has}(h,x) = \text{ture}, \\ \quad x = \text{"Posterior} - \text{Head} - \\ \quad \text{POS(Major)} : \text{動詞(verb)"} \\ \quad \& \ f = 1 \\ 0 : \text{otherwise.} \end{cases} \quad (1)$$

Here "has(h,x)" is a binary function which returns true if the history h has an attribute x. We focus on attributes on a bunsetsu itself and those between bunsetsus. Section 3 will mention these attributes.

Given a set of features and some training data, the maximum entropy estimation process produces a model in which every feature g_i has associated with it a parameter α_i. This allows us to compute the conditional probability as follows (Berger et al., 1996):

$$P(f|h) = \frac{\prod_i \alpha_i^{g_i(h,f)}}{Z_\lambda(h)} \quad (2)$$

$$Z_\lambda(h) = \sum_f \prod_i \alpha_i^{g_i(h,f)}. \quad (3)$$

The maximum entropy estimation technique guarantees that for every feature g_i, the expected value of g_i according to the M.E. model will equal the empirical expectation of g_i in the training corpus. In other words:

$$\sum_{h,f} \tilde{P}(h,f) \cdot g_i(h,f) = \sum_h \tilde{P}(h) \cdot \sum_f P_{ME}(f|h) \cdot g_i(h,f). \quad (4)$$

Here $\tilde{P}$ is an empirical probability and P_{ME} is the probability assigned by the M.E. model.

We assume that dependencies in a sentence are independent of each other and the overall dependencies in a sentence can be determined based on the product of probability of all dependencies in the sentence.

3 Experiments and Discussion

In our experiment, we used the Kyoto University text corpus (version 2) (Kurohashi and Nagao, 1997), a tagged corpus of the Mainichi newspaper. For training we used 7,958 sentences from newspaper articles appearing from January 1st to January 8th, and for testing we used 1,246 sentences from articles appearing on January 9th. The input sentences were morphologically analyzed and their bunsetsus were identified. We assumed that this preprocessing was done correctly before parsing input sentences. If we used automatic morphological analysis and bunsetsu identification, the parsing accuracy would not decrease so much because the rightmost element in a bunsetsu is usually a case marker, a verb ending, or a adjective ending, and each of these is easily recognized. The automatic preprocessing by using public domain

[1] Assumption (4) has not been discussed very much, but our investigation with humans showed that it is true in more than 90% of cases.

tools, for example, can achieve 97% for morphological analysis (Kitauchi et al., 1998) and 99% for bunsetsu identification (Murata et al., 1998).

We employed the Maximum Entropy tool made by Ristad (Ristad, 1998), which requires one to specify the number of iterations for learning. We set this number to 400 in all our experiments.

In the following sections, we show the features used in our experiments and the results. Then we describe some interesting statistics that we found in our experiments. Finally, we compare our work with some related systems.

3.1 Results of Experiments

The features used in our experiments are listed in Tables 1 and 2. Each row in Table 1 contains a feature type, feature values, and an experimental result that will be explained later. Each feature consists of a type and a value. The features are basically some attributes of a bunsetsu itself or those between bunsetsus. We call them 'basic features.' The list is expanded from Haruno's list (Haruno et al., 1998). The features in the list are classified into five categories that are related to the "Head" part of the anterior bunsetsu (category "a"), the "Type" part of the anterior bunsetsu (category "b"), the "Head" part of the posterior bunsetsu (category "c"), the "Type" part of the posterior bunsetsu (category "d"), and the features between bunsetsus (category "e") respectively. The term "Head" basically means a rightmost content word in a bunsetsu, and the term "Type" basically means a function word following a "Head" word or an inflection type of a "Head" word. The terms are defined in the following paragraph. The features in Table 2 are combinations of basic features ('combined features'). They are represented by the corresponding category name of basic features, and each feature set is represented by the feature numbers of the corresponding basic features. They are classified into nine categories we constructed manually. For example, twin features are combinations of the features related to the categories "b" and "c." Triplet, quadruplet and quintuplet features basically consist of the twin features plus the features of the remainder categories "a," "d" and "e." The total number of features is about 600,000. Among them, 40,893 were observed in the training corpus, and we used them in our experiment.

The terms used in the table are the following:

Anterior: left bunsetsu of the dependency

Posterior: right bunsetsu of the dependency

Head: the rightmost word in a bunsetsu other than those whose major part-of-speech[2] category is "特殊 (special marks)," "助詞 (post-positional particles)," or "接尾辞 (suffix)"

Head-Lex: the fundamental form (uninflected form) of the head word. Only words with a frequency of three or more are used.

Head-Inf: the inflection type of a head

Type: the rightmost word other than those whose major part-of-speech category is "特殊 (special marks)." If the major category of the word is neither "助詞 (post-positional particles)" nor "接尾辞 (suffix)," and the word is inflectable[3], then the type is represented by the inflection type.

JOSHI1: the rightmost post-positional particle in the bunsetsu

JOSHI2: the second rightmost post-positional particle in the bunsetsu if there are two or more post-positional particles in the bunsetsu

TOUTEN, WA: TOUTEN means if a comma (Touten) exists in the bunsetsu. WA means if the word WA (a topic marker) exists in the bunsetsu

BW: BW means "between bunsetsus"

BW-Distance: the distance between the bunsetsus

BW-TOUTEN: if TOUTEN exists between bunsetsus

BW-IDto-Anterior-Type: BW-IDto-Anterior-Type means if there is a bunsetsu whose type is identical to that of the anterior bunsetsu between bunsetsus

BW-IDto-Anterior-Type-Head-POS: the part-of-speech category of the head word of the bunsetsu of "BW-IDto-Anterior-Type"

BW-IDto-Posterior-Head: if there is between bunsetsus a bunsetsu whose head is identical to that of the posterior bunsetsu

BW-IDto-Posterior-Head-Type(String): the lexical information of the bunsetsu "BW-IDto-Posterior-Head"

The results of our experiment are listed in Table 3. The dependency accuracy means the percentage of correct dependencies out of all dependencies. The sentence accuracy means the percentage of sentences in which all dependencies were analyzed correctly. We used input sentences that had already been morphologically analyzed and for which bunsetsus had been identified. The first line in Table 3 (deterministic) shows the accuracy achieved when the test sentences were analyzed deterministically (beam width $k = 1$). The second line in Table 3 (best beam search) shows the best accuracy among the experiments when changing the beam breadth k from 1 to 20. The best accuracy was achieved when $k = 11$, although the variation in accuracy was very small. This result supports assumption (4) in Chapter 1 because

[2]Part-of-speech categories follow those of JUMAN(Kurohashi and Nagao, 1998).

[3]The inflection types follow those of JUMAN.

Table 1: Features (basic features)

Basic features (5 categories, 43 types)				
Category	Feature number	Feature type	Feature values ... (Number of values)	Accuracy without each feature
a	1	Anterior-Head-Lex	(2204)	86.98% (−0.16%)
	2	Anterior-Head-POS(Major)	動詞 (verb), 形容詞 (adjective), 名詞 (noun), ... (11)	86.43% (−0.71%)
	3	Anterior-Head-POS(Minor)	普通名詞 (common noun), 数詞 (quantifier), ... (24)	
	4	Anterior-Head-Inf(Major)	母音動詞 (vowel verb), ... (30)	87.14% (±0%)
	5	Anterior-Head-Inf(Minor)	語幹 (stem), 基本形 (fundamental form), ... (60)	
b	6	Anterior-Type(String)	こそ, こと, そして, だけ, と, に, も, ... (73)	69.73% (−17.41%)
	7	Anterior-Type(Major)	助詞 (post-positional particle), ... (43)	
	8	Anterior-Type(Minor)	格助詞 (case marker), 命令形 (imperative form) ... (102)	
	9	Anterior-JOSHI1(String)	から, まで, のみ, へ, ねえ, ... (63)	87.11% (−0.03%)
	10	Anterior-JOSHI1(Minor)	[nil], 格助詞 (case marker), ... (5)	
	11	Anterior-JOSHI2(String)	けど, まま, や, よ, か, ... (63)	87.08% (−0.06%)
	12	Anterior-JOSHI2(Minor)	格助詞 (case marker), ... (4)	
	13	Anterior-punctuation	[nil], comma, period (3)	85.47% (−1.67%)
	14	Anterior-bracket-open	[nil], 「, 『, (, “, 〔, ＜, ≪, ... (14)	87.12% (−0.02%)
	15	Anterior-bracket-close	[nil], 」, 』,), ”, 〕, ＞, ≫, ... (14)	87.10% (−0.04%)
c	16	Posterior-Head-Lex	The same values as those of feature number 1.	86.31% (−0.83%)
	17	Posterior-Head-POS(Major)	The same values as those of feature number 2.	76.15% (−10.99%)
	18	Posterior-Head-POS(Minor)	The same values as those of feature number 3.	
	19	Posterior-Head-Inf(Major)	The same values as those of feature number 4.	87.14% (±−0%)
	20	Posterior-Head-Inf(Minor)	The same values as those of feature number 5.	
d	21	Posterior-Type(String)	The same values as those of feature number 6.	86.06% (−1.08%)
	22	Posterior-Type(Major)	The same values as those of feature number 7.	
	23	Posterior-Type(Minor)	The same values as those of feature number 8.	
	24	Posterior-JOSHI1(String)	The same values as those of feature number 9.	87.16% (+0.02%)
	25	Posterior-JOSHI1(Minor)	The same values as those of feature number 10.	
	26	Posterior-JOSHI2(String)	The same values as those of feature number 11.	87.11% (−0.03%)
	27	Posterior-JOSHI2(Minor)	The same values as those of feature number 12.	
	28	Posterior-punctuation	The same values as those of feature number 13.	84.62% (−2.52%)
	29	Posterior-bracket-open	The same values as those of feature number 14.	86.87% (−0.27%)
	30	Posterior-bracket-close	The same values as those of feature number 15.	86.85% (−0.29%)
e	31	BW-Distance	A(1), B(2 〜 5), C(6 or more) (3)	84.64% (−2.50%)
	32	BW-TOUTEN	[nil], [exist] (2)	86.81% (−0.33%)
	33	BW-WA	[nil], [exist] (2)	86.96% (−0.18%)
	34	BW-brackets	[nil], close, open, open-close (4)	86.08% (−1.06%)
	35	BW-IDto-Anterior-Type	[nil], [exist] (2)	86.99% (−0.15%)
	36	BW-IDto-Anterior-Type-Head-POS(Major)	The same values as those of feature number 2.	
	37	BW-IDto-Anterior-Type-Head-POS(Minor)	The same values as those of feature number 3.	
	38	BW-IDto-Anterior-Type-Head-Inf(Major)	The same values as those of feature number 4.	
	39	BW-IDto-Anterior-Type-Head-Inf(Minor)	The same values as those of feature number 5.	
	40	BW-IDto-Posterior-Head	[nil], [exist] (2)	86.75% (−0.39%)
	41	BW-IDto-Posterior-Head-Type(String)	The same values as those of feature number 6.	
	42	BW-IDto-Posterior-Head-Type(Major)	The same values as those of feature number 7.	
	43	BW-IDto-Posterior-Head-Type(Minor)	The same values as those of feature number 8.	

Table 2: Features (combined features)

Combined features (9 categories, 134 types)			
Combination type	Combinations		Accuracy without the feature
	Category	Feature set	
Twin features: related to the "Type" part of the anterior bunsetsu and the "Head" part of the posterior bunsetsu.	(b, c)	b = {6, 7, 8}, c = {16, 17, 18}	86.99% (−0.15%)
Triplet features: basically consist of the twin features plus the features between bunsetsus.	(b_1, b_2, c) (b, c, e) (d_1, d_2, e)	(b_1, b_2) = {(9, 11), (10, 12)}, c = {17, 18} b = {6, 7, 8}, c = {17, 18}, e = {31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43} (d_1, d_1, e) = (29, 30, 34)	86.47% (−0.67%)
Quadruplet features: basically consist of the twin features plus the features related to the "Head" part of the anterior bunsetsu, and the "Type" part of the posterior bunsetsu.	(b_1, b_2, c, d) (b, c, e_1, e_2) (a, b, c, d)	b_1 = {6, 7, 8}, c = {17, 18}, (b_2, d) = (13, 28) b = {6, 7, 8}, c = {17, 18}, (e_1, e_2) = (35, 40) (a, c) = {(1, 16), (2, 17), (3, 18)}, (b, d) = {(6, 21), (7, 22), (8, 23)}	85.65% (−1.49%)
Quintuplet features: basically consist of the quadruplet features plus the features between bunsetsus.	(a, b_1, b_2, c, d) (a, b, c, d, e)	(a, c) = {(2, 17), (3, 18)}, (b_1, b_2) = {(9, 11), (10, 12)}, d = {21,22,23} (a, c) = {(1, 16), (2, 17), (3, 18)}, (b, d) = {(6, 21), (7, 22), (8, 23}, e = 31	86.96% (−0.18%)

Table 3: Results of dependency analysis

	Dependency accuracy	Sentence accuracy
Deterministic ($k = 1$)	87.14% (9814/11263)	40.60% (503/1239)
Best beam search($k = 11$)	87.21% (9822/11263)	40.60% (503/1239)
Baseline	64.09% (7219/11263)	6.38% (79/1239)

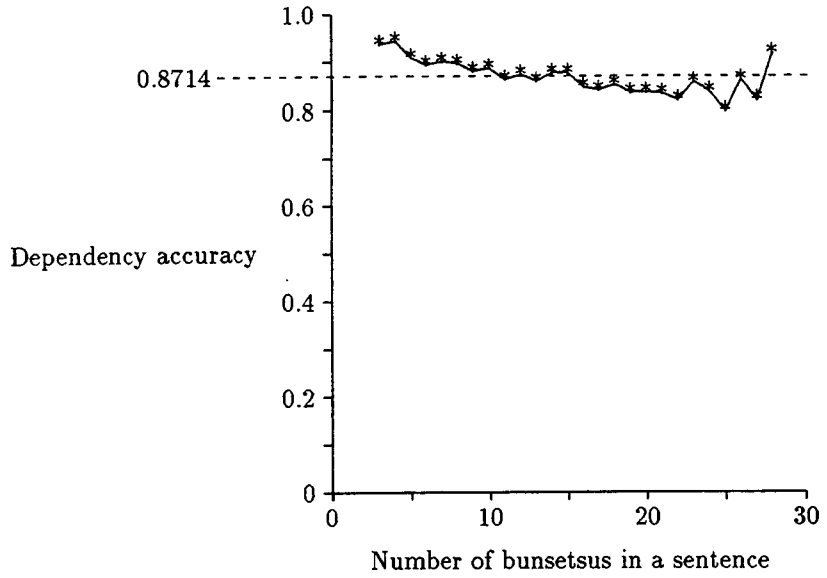

Figure 1: Relationship between the number of bunsetsus in a sentence and dependency accuracy.

it shows that the previous context has almost no effect on the accuracy. The last line in Table 3 represents the accuracy when we assumed that every bunsetsu depended on the next one (baseline).

Figure 1 shows the relationship between the sentence length (the number of bunsetsus) and the dependency accuracy. The data for sentences longer than 28 segments are not shown, because there was at most one sentence of each length. Figure 1 shows that the accuracy degradation due to increasing sentence length is not significant. For the entire test corpus the average running time on a SUN Sparc Station 20 was 0.08 seconds per sentence.

3.2 Features and Accuracy

This section describes how much each feature set contributes to improve the accuracy.

The rightmost column in Tables 1 and 2 shows the performance of the analysis without each feature set. In parenthesis, the percentage of improvement or degradation to the formal experiment is shown. In the experiments, when a basic feature was deleted, the combined features that included the basic feature were also deleted.

We also conducted some experiments in which several types of features were deleted together. The results are shown in Table 4. All of the results in the experiments were carried out deterministically (beam width $k = 1$).

The results shown in Table 1 were very close to our expectation. The most useful features are the type of the anterior bunsetsu and the part-of-speech tag of the head word on the posterior bunsetsu. Next important features are the distance between bunsetsus, the existence of punctuation in the bunsetsu, and the existence of brackets. These results indicate preferential rules with respect to the features.

The accuracy obtained with the lexical features of the head word was better than that without them. In the experiment with the features, we found many idiomatic expressions, for example, "応じて (*oujite*, according to)— 決める (*kimeru*, decide)" and "形で (*katachi_de*, in the form of)— 行われる (*okonawareru*, be held)." We would expect to collect more of such expressions if we use more training data.

The experiments without some combined features are reported in Tables 2 and 4. As can be seen from the results, the combined features are very useful to improve the accuracy. We used these combined features in addition to the basic features because we thought that the basic features were actually related to each other. Without the combined features, the features are independent of each other in the maximum entropy framework.

We manually selected combined features, which are shown in Table 2. If we had used all combi-

Table 4: Accuracy without several types of features

Features	Accuracy
Without features 1 and 16 (lexical information about the head word)	86.30% (−0.84%)
Without features 35 to 43	86.83% (−0.31%)
Without quadruplet and quintuplet features	84.27% (−2.87%)
Without triplet, quadruplet, and quintuplet features	81.28% (−5.86%)
Without all combinations	68.83% (−18.31%)

nations, the number of combined features would have been very large, and the training would not have been completed on the available machine. Furthermore, we found that the accuracy decreased when several new features were added in our preliminary experiments. So, we should not use all combinations of the basic features. We selected the combined features based on our intuition.

In our future work, we believe some methods for automatic feature selection should be studied. One of the simplest ways of selecting features is to select features according to their frequencies in the training corpus. But using this method in our current experiments, the accuracy decreased in all of the experiments. Other methods that have been proposed are one based on using the gain (Berger et al., 1996) and an approximate method for selecting informative features (Shirai et al., 1998a), and several criteria for feature selection were proposed and compared with other criteria (Berger and Printz, 1998). We would like to try these methods.

Investigating the sentences which could not be analyzed correctly, we found that many of those sentences included coordinate structures. We believe that coordinate structures can be detected to a certain extent by considering new features which take a wide range of information into account.

3.3 Number of Training Data and Accuracy

Figure 2 shows the relationship between the number of training data (the number of sentences) and the accuracy. This figure shows dependency accuracies for the training corpus and the test corpus. Accuracy of 81.84% was achieved even with a very small training set (250 sentences). We believe that this is due to the strong characteristic of the maximum entropy framework to the data sparseness problem. From the learning curve, we can expect a certain amount of improvement if we have more training data.

3.4 Comparison with Related Works

This section compares our work with related statistical dependency structure analyses in Japanese.

Comparison with
Shirai's work (Shirai et al., 1998b)

Shirai proposed a framework of statistical language modeling using several corpora: the EDR corpus, RWC corpus, and Kyoto University corpus. He combines a parser based on a hand-made CFG and a probabilistic dependency model. He also used the maximum entropy model to estimate the dependency probabilities between two or three post-positional particles and a verb. Accuracy of 84.34% was achieved using 500 test sentences of length 7 to 9 bunsetsus. In both his and our experiments, the input sentences were morphologically analyzed and their bunsetsus were identified. The comparison of the results cannot strictly be done because the conditions were different. However, it should be noted that the accuracy achieved by our model using sentences of the same length was about 3% higher than that of Shirai's model, although we used a much smaller set of training data. We believe that it is because his approach is based on a hand-made CFG.

Comparison with Ehara's work (Ehara, 1998)

Ehara also used the Maximum Entropy model, and a set of similar kinds of features to ours. However, there is a big difference in the number of features between Ehara's model and ours. Besides the difference in the number of basic features, Ehara uses only the combination of two features, but we also use triplet, quadruplet, and quintuplet features. As shown in Section 3.2, the accuracy increased more than 5% using triplet or larger combinations. We believe that the difference in the combination features between Ehara's model and ours may have led to the difference in the accuracy. The accuracy of his system was about 10% lower than ours. Note that Ehara used TV news articles for training and testing, which are different from our corpus. The average sentence length in those articles was 17.8, much longer than that (average: 10.0) in the Kyoto University text corpus.

Comparison with
Fujio's work (Fujio and Matsumoto, 1998)
and Haruno's work (Haruno et al., 1998)

Fujio used the Maximum Likelihood model with similar features to our model in his parser. Haruno proposed a parser that uses decision tree

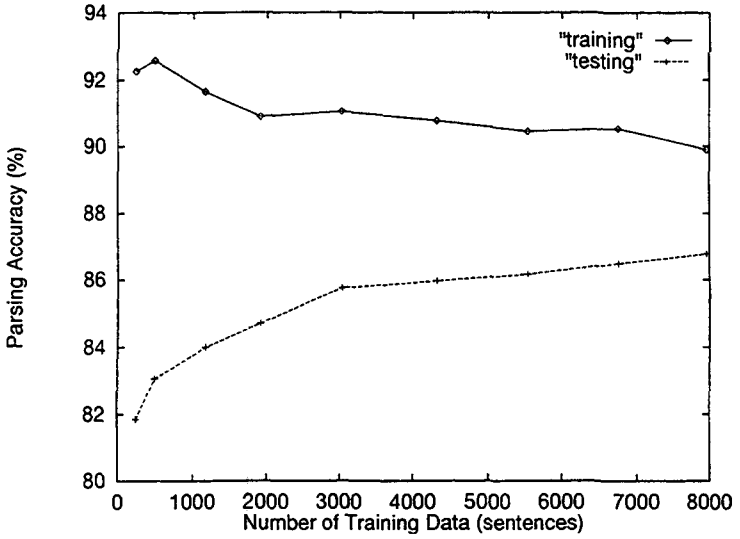

Figure 2: Relationship between the number of training data and the parsing accuracy. (beam breadth $k = 1$)

models and a boosting method. It is difficult to directly compare these models with ours because they use a different corpus, the EDR corpus which is ten times as large as our corpus, for training and testing, and the way of collecting test data is also different. But they reported an accuracy of around 85%, which is slightly worse than our model.

We carried out two experiments using almost the same attributes as those used in their experiments. The results are shown in Table 5, where the lines "Feature set(1)" and "Feature set(2)" show the accuracies achieved by using Fujio's attributes and Haruno's attributes respectively. Considering that both results are around 85% to 86%, which is about the same as ours. From these experiments, we believe that the important factor in the statistical approaches is not the model, i.e. Maximum Entropy, Maximum Likelihood, or Decision Tree, but the feature selection. However, it may be interesting to compare these models in terms of the number of training data, as we can imagine that some models are better at coping with the data sparseness problem than others. This is our future work.

4 Conclusion

This paper described a Japanese dependency structure analysis based on the maximum entropy model. Our model is created by learning the weights of some features from a training corpus to predict the dependency between bunsetsus or phrasal units. The probabilities of dependencies between bunsetsus are estimated by this model. The dependency accuracy of our system was 87.2% using the Kyoto University corpus.

In our experiments without the feature sets shown in Tables 1 and 2, we found that some basic and combined features strongly contribute to improve the accuracy. Investigating the relationship between the number of training data and the accuracy, we found that good accuracy can be achieved even with a very small set of training data. We believe that the maximum entropy framework has suitable characteristics for overcoming the data sparseness problem.

There are several future directions. In particular, we are interested in how to deal with coordinate structures, since that seems to be the largest problem at the moment.

References

Adam Berger and Harry Printz. 1998. A comparison of criteria for maximum entropy / minimum divergence feature selection. *Proceedings of Third Conference on Empirical Methods in Natural Language Processing*, pages 97–106.

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.

Michael Collins. 1996. A new statistical parser based on bigram lexical dependencies. *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 184–191.

Terumasa Ehara. 1998. Japanese bunsetsu dependency estimation using maximum entropy method. *Proceedings of The Fourth Annual*

Table 5: Simulation of Fujio's and Haruno's experiments

Feature set	Accuracy
Feature set (1) (Without features 4, 5, 9—12, 14, 15, 19, 20, 24—27, 29, 30, 34—43.)	85.71% (−1.43%)
Feature set (2) (Without features 4, 5, 9—12, 19, 20, 24—27, 34–43.)	86.47% (−0.67%)

Meeting of The Association for Natural Language Processing, pages 382–385. (in Japanese).

Masakazu Fujio and Yuuji Matsumoto. 1998. Japanese dependency structure analysis based on lexicalized statistics. *Proceedings of Third Conference on Empirical Methods in Natural Language Processing*, pages 87–96.

Katsuhiko Fujita. 1988. A deterministic parser based on karari-uke grammar. pages 399–402.

Masahiko Haruno, Satoshi Shirai, and Yoshifumi Ooyama. 1998. Using decision trees to construct a practical parser. *Proceedings of the COLING-ACL '98.*

Akira Kitauchi, Takehito Utsuro, and Yuji Matsumoto. 1998. Error-driven model learning of Japanese morphological analysis. *IPSJ-WGNL*, NL124-6:41–48. (in Japanese).

Sadao Kurohashi and Makoto Nagao. 1997. Kyoto university text corpus project. pages 115–118. (in Japanese).

Sadao Kurohashi and Makoto Nagao, 1998. *Japanese Morphological Analysis System JUMAN version 3.5.* Department of Informatics, Kyoto University.

Masaki Murata, Kiyotaka Uchimoto, Qing Ma, and Hitoshi Isahara. 1998. Machine learning approach to bunsetsu identification — comparison of decision tree, maximum entropy model, example-based approach, and a new method using category-exclusive rules —. *IPSJ-WGNL*, NL128-4:23–30. (in Japanese).

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

Eric Sven Ristad. 1998. Maximum entropy modeling toolkit, release 1.6 beta. http://www.mnemonic.com/software/memt.

Kiyoaki Shirai, Kentaro Inui, Takenobu Tokunaga, and Hozumi Tanaka. 1998a. Learning dependencies between case frames using maximum entropy method. pages 356–359. (in Japanese).

Kiyoaki Shirai, Kentaro Inui, Takenobu Tokunaga, and Hozumi Tanaka. 1998b. A framework of integrating syntactic and lexical statistics in statistical parsing. *Journal of Natural Language Processing*, 5(3):85–106. (in Japanese).