Probabilistic Coordination Disambiguation in a Fully-lexicalized Japanese Parser

Daisuke Kawahara
National Institute of Information and Communications Technology,
3-5 Hikaridai Seika-cho, Soraku-gun,
Kyoto, 619-0289, Japan
dk@nict.go.jp

Sadao Kurohashi
Graduate School of Informatics,
Kyoto University,
Yoshida-Honmachi, Sakyo-ku,
Kyoto, 606-8501, Japan
kuro@i.kyoto-u.ac.jp

Abstract

This paper describes a probabilistic model for coordination disambiguation integrated into syntactic and case structure analysis. Our model probabilistically assesses the parallelism of a candidate coordinate structure using syntactic/semantic similarities and cooccurrence statistics. We integrate these probabilities into the framework of fully-lexicalized parsing based on large-scale case frames. This approach simultaneously addresses two tasks of coordination disambiguation: the detection of coordinate conjunctions and the scope disambiguation of coordinate structures. Experimental results on web sentences indicate the effectiveness of our approach.

1 Introduction

Coordinate structures are a potential source of syntactic ambiguity in natural language. Since their interpretation directly affects the meaning of the text, their disambiguation is important for natural language understanding.

Coordination disambiguation consists of the following two tasks:

- the detection of coordinate conjunctions,
- and finding the scope of coordinate structures.

In English, for example, coordinate structures are triggered by coordinate conjunctions, such as *and* and *or*. In a coordinate structure that consists of more than two conjuncts, commas, which have various usages, also function like coordinate conjunctions. Recognizing true coordinate conjunctions from such possible coordinate conjunctions is a task of coordination disambiguation (Kurohashi, 1995). The other is the task of identifying the range of coordinate phrases or clauses.

Previous work on coordination disambiguation has focused on the task of addressing the scope ambiguity (e.g., (Agarwal and Boggess, 1992; Goldberg, 1999; Resnik, 1999; Chantree et al., 2005)). Kurohashi and Nagao proposed a similarity-based method to resolve both of the two tasks for Japanese (Kurohashi and Nagao, 1994). Their method, however, heuristically detects coordinate conjunctions by considering only similarities between possible conjuncts, and thus cannot disambiguate the following cases[1]:

(1) a. *kanojo-to gakkou-ni itta*
she-cmi school-acc went

(ϕ went to school with her)

b. *kanojo-to watashi-ga goukaku-shita*
she-cnj I-nom passed an exam

(she and I passed an exam)

In sentence (1a), postposition "*to*" is used as a comitative case marker, but in sentence (1b), postposition "*to*" is used as a coordinate conjunction.

To resolve this ambiguity, predicative case frames are required. Case frames describe what kinds of

[1]In this paper, we use the following abbreviations: nom (nominative), acc (accusative), abl (ablative), cmi (comitative), cnj (conjunction) and TM (topic marker).

Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 306–314, Prague, June 2007.

Table 1: Case frame examples (Examples are written in English. Numbers following each example represent its frequency.).

	CS	Examples
yaku (1) (broil)	*ga*	I:18, person:15, craftsman:10, ···
	wo	bread:2484, meat:1521, cake:1283, ···
	de	oven:1630, frying pan:1311, ···
yaku (2) (have difficulty)	*ga*	teacher:3, government:3, person:3, ···
	wo	fingers:2950
	ni	attack:18, action:15, son:15, ···
yaku (3) (burn)	*ga*	maker:1, distributor:1
	wo	data:178, file:107, copy:9, ···
	ni	R:1583, CD:664, CDR:3, ···
⋮	⋮	⋮
oyogu (1) (swim)	*ga*	dolphin:142, student:50, fish:28, ···
	wo	sea:1188, underwater:281, ···
	de	crawl:86, breaststroke:49, stroke:24, ···
⋮	⋮	⋮
migaku (1) (brush)	*ga*	I:4, man:4, person:4, ···
	wo	tooth:5959, molar:27, foretooth:12
	de	brush:38, salt:13, powder:12, ···
⋮	⋮	⋮

nouns are related to each predicate. For example, a case frame of "*iku*" (go) has a "*to*" case slot filled with the examples such as "*kanojo*" (she) or human. On the other hand, "*goukaku-suru*" (pass an exam) does not have a "*to*" case slot but does have a "*ga*" case slot filled with "*kanojo*" (she) and "*watashi*" (I). These case frames provide the information for disambiguating the postpositions "*to*" in sentences (1a) and (1b): (1a) is not coordinate and (1b) is coordinate.

This paper proposes a method for integrating coordination disambiguation into probabilistic syntactic and case structure analysis. This method simultaneously addresses the two tasks of coordination disambiguation by utilizing syntactic/semantic parallelism in possible coordinate structures and lexical preferences in large-scale case frames. We use the case frames that were automatically constructed from the web (Table 1). In addition, cooccurrence statistics of coordinate conjuncts are incorporated into this model.

2 Related Work

Previous work on coordination disambiguation has focused mainly on finding the scope of coordinate structures.

Agarwal and Boggess proposed a method for identifying coordinate conjuncts (Agarwal and Boggess, 1992). Their method simply matches parts of speech and hand-crafted semantic tags of the head words of the coordinate conjuncts. They tested their method using the Merck Veterinary Manual and found their method had an accuracy of 81.6%.

Resnik described a similarity-based approach for coordination disambiguation of nominal compounds (Resnik, 1999). He proposed a similarity measure based on the notion of shared information content. He conducted several experiments using the Penn Treebank and reported an F-measure of approximately 70%.

Goldberg applied a cooccurrence-based probabilistic model to determine the attachments of ambiguous coordinate phrases with the form "n1 p n2 cc n3" (Goldberg, 1999). She collected approximately 120K unambiguous pairs of two coordinate words from a raw newspaper corpus for a one-year period and estimated parameters from these statistics. Her method achieved an accuracy of 72% using the Penn Treebank.

Chantree et al. presented a binary classifier for coordination ambiguity (Chantree et al., 2005). Their model is based on word distribution information obtained from the British National Corpus. They achieved an F-measure ($\beta = 0.25$) of 47.4% using their own test set.

The previously described methods focused on coordination disambiguation. Some research has been undertaken that integrated coordination disambiguation into parsing.

Kurohashi and Nagao proposed a Japanese parsing method that included coordinate structure detection (Kurohashi and Nagao, 1994). Their method first detects coordinate structures in a sentence, and then heuristically determines the dependency structure of the sentence under the constraints of the detected coordinate structures. Their method correctly analyzed 97 Japanese sentences out of 150.

Charniak and Johnson used some features of syntactic parallelism in coordinate structures for their MaxEnt reranking parser (Charniak and Johnson, 2005). The reranker achieved an F-measure of 91.0%, which is higher than that of their generative parser (89.7%). However, they used a numerous number of features, and the contribution of the

Table 2: Expressions that indicate coordinate structures.

(a) coordinate noun phrase:
,(comma) *to ya toka katsu oyobi ka aruiwa ...*
(b) coordinate predicative clause:
-*shi ga oyobi ka aruiwa matawa ...*
(c) incomplete coordinate structure:
,(comma) *oyobi narabini aruiwa ...*

parallelism features is unknown.

Dubey et al. proposed an unlexicalized PCFG parser that modified PCFG probabilities to condition the existence of syntactic parallelism (Dubey et al., 2006). They obtained an F-measure increase of 0.4% over their baseline parser (73.0%). Experiments with a lexicalized parser were not conducted in their work.

A number of machine learning-based approaches to Japanese parsing have been developed. Among them, the best parsers are the SVM-based dependency analyzers (Kudo and Matsumoto, 2002; Sassano, 2004). In particular, Sassano added some features to improve his parser by enabling it to detect coordinate structures (Sassano, 2004). However, the added features did not contribute to improving the parsing accuracy. This failure can be attributed to the inability to consider global parallelism.

3 Coordination Ambiguity in Japanese

In Japanese, the *bunsetsu* is a basic unit of dependency that consists of one or more content words and the following zero or more function words. A bunsetsu corresponds to a base phrase in English and "*eojeol*" in Korean.

Coordinate structures in Japanese are classified into three types. The first type is the *coordinate noun phrase*.

(2) *nagai enpitsu-**to** keshigomu-wo katta*
long pencil-cnj eraser-acc bought

(bought a long pencil and an eraser)

We can find these phrases by referring to the words listed in Table 2-a.

The second type is the *coordinate predicative clause*, in which two or more predicates form a coordinate structure.

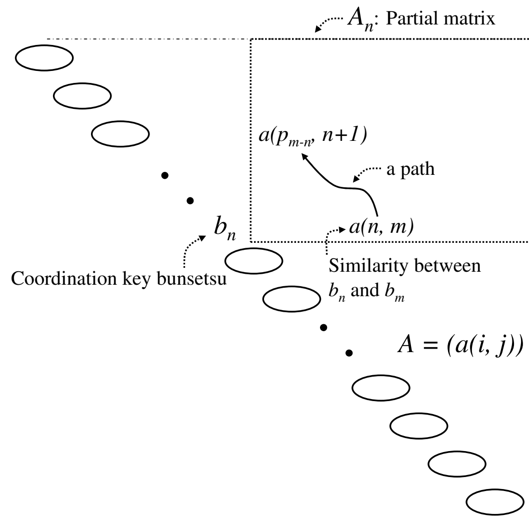


Figure 1: Method using triangular matrix.

(3) *kanojo-to kekkon-**shi** ie-wo katta*
she-cmi married house-acc bought

(married her and bought a house)

We can find these clauses by referring to the words and ending forms listed in Table 2-b.

The third type is the *incomplete coordinate structure*, in which some parts of coordinate predicative clauses are present.

(4) *Tom-wa inu-**wo,** Jim-wa neko-**wo** kau*
Tom-TM dog-acc Jim-TM cat-acc buys

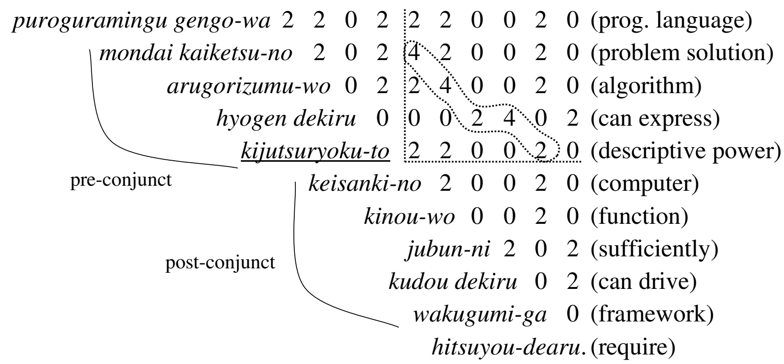
(Tom (buys) a dog, and Jim buys a cat)

We can find these structures by referring to the words listed in Table 2-c and also the correspondence of case-marking postpositions.

For all of these types, we can detect the possibility of a coordinate structure by looking for a *coordination key bunsetsu* that accompanies one of the words listed in Table 2 (in total, we have 52 coordination expressions). That is to say, the left and right sides of a coordination key bunsetsu constitute possible pre- and post-conjuncts, and the key bunsetsu is located at the end of the pre-conjunct. The size of the conjuncts corresponds to the scope of the coordination.

4 Calculating Similarity between Possible Coordinate Conjuncts

We assess the parallelism of potential coordinate structures in a probabilistic parsing model. In this



(Programming language requires descriptive power to express an algorithm for solving problems and a framework to sufficiently drive functions of a computer.)

Figure 2: Example of calculating path scores.

section, we describe a method for calculating similarities between potential coordinate conjuncts.

To measure the similarity between potential pre- and post-conjuncts, a lot of work on the coordination disambiguation used the similarity between conjoined heads. However, not only the conjoined heads but also other components in conjuncts have some similarity and furthermore structural parallelism. Therefore, we use a method to calculate the similarity between two whole coordinate conjuncts (Kurohashi and Nagao, 1994). The remainder of this section contains a brief description of this method.

To calculate similarity between two series of bunsetsus, a triangular matrix, A, is used (illustrated in Figure 1).

$$A = (a(i,j)) \;\; (0 \leq i \leq l; i \leq j \leq l) \tag{1}$$

where l is the number of bunsetsus in a sentence, diagonal element $a(i,j)$ is the i-th bunsetsu, and element $a(i,j)$ $(i < j)$ is the similarity value between bunsetsus b_i and b_j. A similarity value between two bunsetsus is calculated on the basis of POS matching, exact word matching, and their semantic closeness in a thesaurus tree (Kurohashi and Nagao, 1994). We use the *Bunruigoihyo* thesaurus, which contains 96,000 Japanese words (The National Institute for Japanese Language, 2004).

To detect a coordinate structure involving a key bunsetsu, b_n, we consider only a partial matrix (denoted A_n), that is, the upper right part of b_n (Figure 1).

$$A_n = (a(i,j)) \;\; (0 \leq i \leq n; n+1 \leq j \leq l) \tag{2}$$

To specify correspondences between bunsetsus in potential pre- and post-conjuncts, a path is defined as follows:

$$path ::= (a(p_1, m), \quad a(p_2, m-1), \; \ldots, \\ a(p_{m-n}, n+1)) \tag{3}$$

where $n+1 \leq m \leq l$, $a(p_1,m) \neq 0$, $p_1 = n$, $p_i \geq p_{i+1}$, $(1 \leq i \leq m-n-1)$.
That is, a path represents a series of elements from a non-zero element in the lowest row in A_n to an element in the leftmost column in A_n. The path has an only element in each column and extends toward the upper left. The series of bunsetsus on the left side of the path and the series under the path are potential conjuncts for key b_n. Figure 2 shows an example of a path.

A path score is defined based on the following criteria:

- the sum of each element's points on the path
- penalty points when the path extends non-diagonally (which causes conjuncts of unbalanced lengths)
- bonus points on expressions signaling the beginning or ending of a coordinate structure, such as "*kaku*" (each) and *nado*" (and so on)
- the total score of the above criteria is divided by the square root of the number of bunsetsus covered by the path for normalization

The score of each path is calculated using a dynamic programming method. We consider each path as a candidate of pre- and post-conjuncts.

5 Integrated Probabilistic Model for Syntactic, Coordinate and Case Structure Analysis

This section describes a method of integrating coordination disambiguation into a probabilistic parsing model. The integrated model is based on a fully-lexicalized probabilistic model for Japanese syntactic and case structure analysis (Kawahara and Kurohashi, 2006b).

5.1 Outline of the Model

This model gives a probability to each possible dependency structure, T, and case structure, L, of the input sentence, S, and outputs the syntactic, coordinate and case structure that have the highest probability. That is to say, the model selects the syntactic structure, T_{best}, and the case structure, L_{best}, that maximize the probability, $P(T, L|S)$:

$$\begin{aligned}(T_{best}, L_{best}) &= \mathrm{argmax}_{(T,L)} P(T, L|S) \\ &= \mathrm{argmax}_{(T,L)} \frac{P(T, L, S)}{P(S)} \\ &= \mathrm{argmax}_{(T,L)} P(T, L, S) \qquad (4)\end{aligned}$$

The last equation is derived because $P(S)$ is constant.

The model considers a clause as a generation unit and generates the input sentence from the end of the sentence in turn. The probability $P(T, L, S)$ is defined as the product of probabilities for generating clause C_i as follows:

$$P(T, L, S) = \prod_{i=1..n} P(C_i, rel_{ih_i}|C_{h_i}) \qquad (5)$$

where n is the number of clauses in S, C_{h_i} is C_i's modifying clause, and rel_{ih_i} is the dependency relation between C_i and C_{h_i}. The main clause, C_n, at the end of a sentence does not have a modifying head, but a virtual clause C_{h_n} = EOS (End Of Sentence) is inserted. Dependency relation rel_{ih_i} is first classified into two types C (coordinate) and D (normal dependency), and C is further divided into five classes according to the binned similarity (path score) of conjuncts. Therefore, rel_{ih_i} can be one of the following six classes.

$$rel_{ih_i} = \{D, C0, C1, C2, C3, C4\} \qquad (6)$$

For instance, $C0$ represents a coordinate relation with a similarity of less than 1, and $C4$ represents a coordinate relation with a similarity of 4 or more.

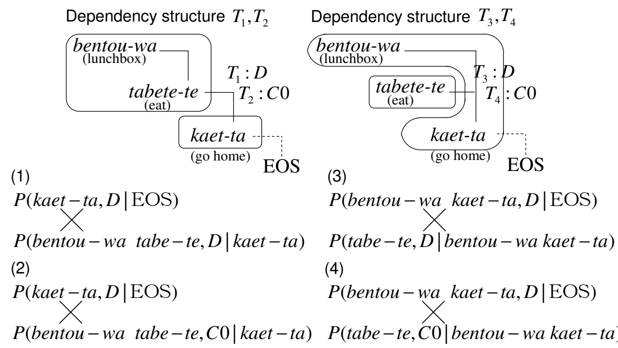


Figure 3: Example of probability calculation.

For example, consider the sentence shown in Figure 3. There are four possible dependency structures in this figure, and the product of the probabilities for each structure indicated below the tree is calculated. Finally, the model chooses the structure with the highest probability (in this case T_1 is chosen).

Clause C_i is decomposed into its clause type, f_i, (including the predicate's inflection and function words) and its remaining content part $C_i{}'$. Clause C_{h_i} is also decomposed into its content part, $C_{h_i}{}'$, and its clause type, f_{h_i}.

$$\begin{aligned}&P(C_i, rel_{ih_i}|C_{h_i}) = P(C_i{}', f_i, rel_{ih_i}|C_{h_i}{}', f_{h_i}) \\ &= P(C_i{}', rel_{ih_i}|f_i, C_{h_i}{}', f_{h_i}) \times P(f_i|C_{h_i}{}', f_{h_i}) \\ &\approx P(C_i{}', rel_{ih_i}|f_i, C_{h_i}{}') \times P(f_i|f_{h_i}) \qquad (7)\end{aligned}$$

Equation (7) is derived because the content part, $C_i{}'$, is usually independent of its modifying head type, f_{h_i}, and in most cases, the type, f_i, is independent of the content part of its modifying head, C_{h_i}.

We call $P(C_i{}', rel_{ih_i}|f_i, C_{h_i}{}')$ *generative probability of a case and coordinate structure*, and $P(f_i|f_{h_i})$ *generative probability of a clause type*. The latter is the probability of generating function words including topic markers and punctuation marks, and is estimated using a syntactically annotated corpus in the same way as (Kawahara and Kurohashi, 2006b).

The generative probability of a case and coordinate structure can be rewritten as follows:

$$\begin{aligned}&P(C_i{}', rel_{ih_i}|f_i, C_{h_i}{}') \\ &= P(C_i{}'|rel_{ih_i}, f_i, C_{h_i}{}') \times P(rel_{ih_i}|f_i, C_{h_i}{}') \\ &\approx P(C_i{}'|rel_{ih_i}, f_i, C_{h_i}{}') \times P(rel_{ih_i}|f_i) \qquad (8)\end{aligned}$$

Equation (8) is derived because dependency relations (coordinate or not) heavily depend on modifier's types including coordination keys. We call $P(C_i'|rel_{ih_i}, f_i, C_{h_i}')$ *generative probability of a case structure*, and $P(rel_{ih_i}|f_i)$ *generative probability of a coordinate structure*. The following two subsections describe these probabilities.

5.2 Generative Probability of Coordinate Structure

The most important feature to decide whether two clauses are coordinate is coordination keys. Therefore, we consider a coordination key, k_i, as clause type f_i. The generative probability of a coordinate structure, $P(rel_{ih_i}|f_i)$, is defined as follows:

$$P(rel_{ih_i}|f_i) = P(rel_{ih_i}|k_i) \quad (9)$$

We classified coordination keys into 52 classes according to the classification proposed by (Kurohashi and Nagao, 1994). If type f_i does not contain a coordination key, the relation is always D (normal dependency), that is $P(rel_{ih_i}|f_i) = P(D|\phi) = 1$.

The generative probability of a coordinate structure was estimated from a syntactically annotated corpus using maximum likelihood. We used the Kyoto Text Corpus (Kurohashi and Nagao, 1998), which consists of 40K Japanese newspaper sentences.

5.3 Generative Probability of Case Structure

We consider that a case structure consists of a predicate, v_i, a case frame, CF_l, and a case assignment, CA_k. Case assignment CA_k represents correspondences between the input case components and the case slots shown in Figure 4. Thus, the generative probability of a case structure is decomposed as follows:

$$\begin{aligned}
&P(C_i'|rel_{ih_i}, f_i, C_{h_i}') \\
&= P(v_i, CF_l, CA_k|rel_{ih_i}, f_i, C_{h_i}') \\
&= P(v_i|rel_{ih_i}, f_i, C_{h_i}') \\
&\quad \times P(CF_l|rel_{ih_i}, f_i, C_{h_i}', v_i) \\
&\quad \times P(CA_k|rel_{ih_i}, f_i, C_{h_i}', v_i, CF_l) \\
&\approx P(v_i|rel_{ih_i}, f_i, w_{h_i}) \\
&\quad \times P(CF_l|v_i) \\
&\quad \times P(CA_k|CF_l, f_i) \qquad (10)
\end{aligned}$$

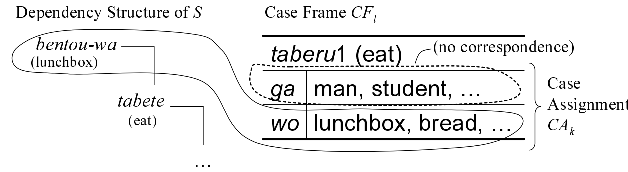


Figure 4: Example of case assignment.

The above approximation is given because it is natural to consider that the predicate v_i depends on its modifying head w_{h_i} instead of the whole modifying clause, that the case frame CF_l only depends on the predicate v_i, and that the case assignment CA_k depends on the case frame CF_l and the clause type f_i.

The generative probabilities of case frames and case assignments are estimated from case frames themselves in the same way as (Kawahara and Kurohashi, 2006b). The remainder of this section describes the generative probability of a predicate, $P(v_i|rel_{ih_i}, f_i, w_{h_i})$.

The generative probability of a predicate captures cooccurrences of coordinate or non-coordinate phrases. This kind of information is not handled in case frames, which aggregate only predicate-argument relations.

The generative probability of a predicate mainly depends on a coordination key in the clause type, f_i, as well as the generative probability of a coordinate structure. We define this probability as follows:

$$P(v_i|rel_{ih_i}, f_i, w_{h_i}) = P(v_i|rel_{ih_i}, k_i, w_{h_i})$$

If C_i' is a nominal clause and consists of a noun n_i, we consider the following probability in stead of equation (10):

$$P_n(C_i'|rel_{ih_i}, f_i, C_{h_i}') \approx P(n_i|rel_{ih_i}, f_i, w_{h_i})$$

This is because a noun does not have a case frame and any case components in the current framework.

To estimate these probabilities, we first applied a conventional parsing system with coordination disambiguation to a huge corpus, and collected coordinate bunsetsus from the parses. We used KNP[2] (Kurohashi and Nagao, 1994) as the parser and a web corpus consisting of 470M Japanese sentences (Kawahara and Kurohashi, 2006a). The generative probability of a predicate was estimated from the

[2]http://nlp.kuee.kyoto-u.ac.jp/nl-resource/knp-e.html

collected coordinate bunsetsus using maximum likelihood.

5.4 Practical Issue

The proposed model considers all the possible dependency structures including coordination ambiguities. To reduce this high computational cost, we introduced the CKY framework to the search.

Each parameter in the model is smoothed by using several back-off levels in the same way as (Collins, 1999). Smoothing parameters are optimized using a development corpus.

6 Experiments

We evaluated the coordinate structures and dependency structures that were outputted by our model. The case frames used in this paper were automatically constructed from 470M Japanese sentences obtained from the web. Some examples of the case frames are listed in Table 1 (Kawahara and Kurohashi, 2006a).

In this work, the parameters related to unlexical types are calculated from a small tagged corpus of newspaper articles, and lexical parameters are obtained from a huge web corpus. To evaluate the effectiveness of our fully-lexicalized model, our experiments are conducted using web sentences. As the test corpus, we prepared 759 web sentences [3]. The web sentences were manually annotated using the same criteria as the Kyoto Text Corpus. We also used the Kyoto Text Corpus as a development corpus to optimize the smoothing parameters. The system input was automatically tagged using the JUMAN morphological analyzer [4].

We used two baseline systems for comparative purposes: the rule-based dependency parser, KNP (Kurohashi and Nagao, 1994), and the probabilistic model of syntactic and case structure analysis (Kawahara and Kurohashi, 2006b), in which coordination disambiguation is the same as that of KNP.

6.1 Evaluation of Detection of Coordinate Structures

First, we evaluated detecting coordinate structures, namely whether a coordination key bunsetsu triggers a coordinate structure. Table 3 lists the experimental results. The F-measure of our method is slightly higher than that of the baseline method (KNP). In particular, our method achieved good precision.

Table 3: Experimental results of detection of coordinate structures.

	baseline	proposed
precision	366/460 (79.6%)	361/435 (83.0%)
recall	366/447 (81.9%)	361/447 (80.8%)
F-measure	– (80.7%)	– (81.9%)

6.2 Evaluation of Dependency Parsing

Secondly, we evaluated the dependency structures analyzed by the proposed model. Evaluating the scope ambiguity of coordinate structures is subsumed within this dependency evaluation. The dependency structures obtained were evaluated with regard to dependency accuracy — the proportion of correct dependencies out of all dependencies except for the last dependency in the sentence end [5]. Table 4 lists the dependency accuracy. In this table, "syn" represents the rule-based dependency parser, KNP, "syn+case" represents the probabilistic parser of syntactic and case structure (Kawahara and Kurohashi, 2006b), and "syn+case+coord" represents our proposed model. The proposed model significantly outperformed both of the baseline systems (McNemar's test; $p < 0.01$).

In the table, the dependency accuracies are classified into four types on the basis of the bunsetsu classes (PB: predicate bunsetsu and NB: noun bunsetsu) of a dependent and its head. "syn+case" outperformed "syn". In particular, the accuracy of predicate-argument relations ("NB→PB") was improved, but the accuracies of "NB→NB" and "PB→PB" decreased. "syn+case+coord" outperformed the two baselines for all of the types. Not only the accuracy of predicate-argument relations ("NB→PB") but also the accuracies of coordinate noun/predicate bunsetsus (related to "NB→NB" and "PB→PB") were improved. These improvements are conduced by the integration of coordination disambiguation and syntactic/case structure analysis.

[3] The test set was not used to construct case frames and estimate probabilities.

[4] http://nlp.kuee.kyoto-u.ac.jp/nl-resource/juman-e.html

[5] Since Japanese is head-final, the second last bunsetsu unambiguously depends on the last bunsetsu, and the last bunsetsu has no dependency.

Table 4: Experimental results of dependency parsing.

	syn	syn+case	syn+case+coord
all	3,833/4,436 (86.4%)	3,852/4,436 (86.8%)	3,893/4,436 (87.8%)
NB→PB	1,637/1,926 (85.0%)	1,664/1,926 (86.4%)	1,684/1,926 (87.4%)
NB→NB	1,032/1,136 (90.8%)	1,029/1,136 (90.6%)	1,037/1,136 (91.3%)
PB→PB	654/817 (80.0%)	647/817 (79.2%)	659/817 (80.7%)
PB→NB	510/557 (91.6%)	512/557 (91.9%)	513/557 (92.1%)

To compare our results with a state-of-the-art discriminative dependency parser, we input the same test corpus into an SVM-based Japanese dependency parser, CaboCha[6](Kudo and Matsumoto, 2002). Its dependency accuracy was 86.3% (3,829/4,436), which is equivalent to that of "syn" (KNP). This low accuracy is attributed to the out-of-domain training corpus. That is, the parser is trained on a newspaper corpus, whereas the test corpus is obtained from the web, because of the non-availability of a tagged web corpus that is large enough to train a supervised parser.

6.3 Discussion

Figure 5 shows some analysis results, where the dotted lines represent the analysis by the baseline, "syn+case", and the solid lines represent the analysis by the proposed method, "syn+case+coord". These sentences are incorrectly analyzed by the baseline but correctly analyzed by the proposed method. For instance, in sentence (1), the noun phrase coordination of "*apurikeesyon*" (application) and "*doraiba*" (driver) can be correctly analyzed. This is because the case frame of "*insutooru-sareru*" (installed) is likely to generate "*doraiba*", and "*apurikeesyon*" and "*doraiba*" are likely to be coordinated.

One of the causes of errors in dependency parsing is the mismatch between analysis results and annotation criteria. As per the annotation criteria, each bunsetsu has only one modifying head. Therefore, in some cases, even if analysis results are semantically correct, they are judged as incorrect from the viewpoint of the annotation. For example, in sentence (4) in Figure 6, the baseline method, "syn", correctly recognized the head of "*iin-wa*" (commissioner-TM) as "*hirakimasu*" (open). However, the proposed method incorrectly judged it as "*oujite-imasuga*" (offer). Both analysis results can be considered to be semantically correct, but from the viewpoint of our annotation criteria, the latter is not a syntactic relation (i.e., incorrect), but an ellipsis relation. This kind of error is caused by the strong lexical preference considered in our method.

To address this problem, it is necessary to simultaneously evaluate not only syntactic relations but also indirect relations, such as ellipses and anaphora. This kind of mismatch also occurred for the detection of coordinate structures.

Another errors were caused by an inherent characteristic of generative models. Generative models have some advantages, such as their application to language models. However, it is difficult to incorporate various features that seem to be useful for addressing syntactic and coordinate ambiguity. We plan to apply discriminative reranking to the n-best parses produced by our generative model in the same way as (Charniak and Johnson, 2005).

7 Conclusion

This paper has described an integrated probabilistic model for coordination disambiguation and syntactic/case structure analysis. This model takes advantage of lexical preference of a huge raw corpus and large-scale case frames and performs coordination disambiguation and syntactic/case analysis simultaneously. The experiments indicated the effectiveness of our model. Our future work involves incorporating ellipsis resolution to develop an integrated model for syntactic, case, and ellipsis analysis.

Acknowledgment

This research is partially supported by special coordination funds for promoting science and technology.

References

Rajeev Agarwal and Lois Boggess. 1992. A simple but useful approach to conjunct identification. In *Proceedings of ACL1992*, pages 15–21.

[6]http://chasen.org/˜taku/software/cabocha/

(1)	insutooru-sareteiru	apurikeesyon-oyobi	doraiba-tono	kyougou-niyori	dousa-shinai	baai-ga	arimasu.
	installed	application	driver	conflict	not work	case-nom	exist

(due to the conflict between installed application and driver, there is a case that (it) does not work.)

(2)	...	kuroji-wa	41oku-doru-to,	zennen-yori	10oku-doru	gensyou-shita.
		surplus-TM	4.1 billion dollars	preceding year-abl	1 billion dollars	reduced

(... surplus was 4.1 billion dollars and was reduced by 1 billion dollars from the preceding year.)

(3)	...	gurupu-wa	sugu	ugokidasu-node	wakaru-nodaga,	ugokidasa-nai	gurupu-mo	aru.
		group-TM	soon	start to work	see	not start to work	group also	be

(... can see the groups that start to work soon, but there are groups that do not start to work.)

Figure 5: Examples of correct analysis results. The dotted lines represent the analysis by the baseline, "syn+case", and the solid lines represent the analysis by the proposed method, "syn+case+coord".

(4)	iin-wa,	jitaku-de	minasan-karano	gosoudan-ni	oujite-imasuga, ...	soudansyo-wo	hirakimasu
	commissioner-TM	at home	all of you	consultation-acc	offer	window	open

(the commissioner offers consultation to all of you at home, but opens a window ...)

Figure 6: An example of incorrect analysis results caused by the mismatch between analysis results and annotation criteria.

Francis Chantree, Adam Kilgarriff, Anne de Roeck, and Alistair Wills. 2005. Disambiguating coordinations using word distribution information. In *Proceedings of RANLP2005*.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of ACL2005*, pages 173–180.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Amit Dubey, Frank Keller, and Patrick Sturt. 2006. Integrating syntactic priming into an incremental probabilistic parser, with an application to psycholinguistic modeling. In *Proceedings of COLING-ACL2006*, pages 417–424.

Miriam Goldberg. 1999. An unsupervised model for statistically determining coordinate phrase attachment. In *Proceedings of ACL1999*, pages 610–614.

Daisuke Kawahara and Sadao Kurohashi. 2006a. Case frame compilation from the web using high-performance computing. In *Proceedings of LREC2006*.

Daisuke Kawahara and Sadao Kurohashi. 2006b. A fully-lexicalized probabilistic model for Japanese syntactic and case structure analysis. In *Proceedings of HLT-NAACL2006*, pages 176–183.

Taku Kudo and Yuji Matsumoto. 2002. Japanese dependency analysis using cascaded chunking. In *Proceedings of CoNLL2002*, pages 29–35.

Sadao Kurohashi and Makoto Nagao. 1994. A syntactic analysis method of long Japanese sentences based on the detection of conjunctive structures. *Computational Linguistics*, 20(4):507–534.

Sadao Kurohashi and Makoto Nagao. 1998. Building a Japanese parsed corpus while improving the parsing system. In *Proceedings of LREC1998*, pages 719–724.

Sadao Kurohashi. 1995. Analyzing coordinate structures including punctuation in English. In *Proceedings of IWPT1995*, pages 136–147.

Philip Resnik. 1999. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130.

Manabu Sassano. 2004. Linear-time dependency analysis for Japanese. In *Proceedings of COLING2004*, pages 8–14.

The National Institute for Japanese Language. 2004. *Bunruigoihyo*. Dainippon Tosho, (In Japanese).