

Coordination Disambiguation without Any Similarities

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

The use of similarities has been one of the main approaches to resolve the ambiguities of coordinate structures. In this paper, we present an alternative method for coordination disambiguation, which does not use similarities. Our hypothesis is that coordinate structures are supported by surrounding dependency relations, and that such dependency relations rather yield similarity between conjuncts, which humans feel. Based on this hypothesis, we built a Japanese fully-lexicalized generative parser that includes coordination disambiguation. Experimental results on web sentences indicated the effectiveness of our approach, and endorsed our hypothesis.

1 Introduction

The interpretation of coordinate structures directly affects the meaning of the text. Addressing coordination ambiguities is fundamental to natural language understanding. Previous studies on coordination disambiguation suggested that conjuncts in coordinate structures have syntactic or semantic similarities, and dealt with coordination ambiguities using (sub-)string matching, part-of-speech matching, semantic similarities, and so forth (Agarwal and Boggess, 1992). Semantic similarities are acquired from thesauri (Kurohashi and Nagao, 1994; Resnik, 1999) or distributional similarity (Chantree et al., 2005).

For instance, consider the following example:

- (1) eat Caesar salad and Italian pasta

The above methods detect the similarity between *salad* and *pasta* using a thesaurus or distributional similarity, and identify the coordinate structure that conjoins *salad* and *pasta*. They do not use the information of the word *eat*.

On the other hand, this coordinate structure can be analyzed by using selectional preference of *eat*. Since *eat* is likely to have *salad* and *pasta* as its objects, it is plausible that *salad* and *pasta* are coordinated. Such selectional preferences are thought to support the construction of coordinate structures and to yield similarity between conjuncts on the contrary.

We present a method of coordination disambiguation without using similarities. Coordinate structures are supported by their surrounding dependency relations that provide selectional preferences. These relations implicitly work as similarities, and thus it is not necessary to use similarities explicitly.

In this paper, we focus on Japanese. Coordination disambiguation is integrated in a fully-lexicalized generative dependency parser (Kawahara and Kurohashi, 2007). For the selectional preferences, we use lexical knowledge, such as case frames, which is extracted from a large raw corpus.

The remainder of this paper is organized as follows. Section 2 summarizes previous work related to coordination disambiguation and its integration into parsing. Section 3 briefly describes the background of this study. Section 4 overviews our idea, and section 5 describes our model in detail. Section 6 is devoted to our experiments. Finally, section 7 gives the conclusions.

© 2008. Licensed under the *Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported* license (<http://creativecommons.org/licenses/by-nc-sa/3.0/>). Some rights reserved.

2 Related Work

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

There are several methods that use similarities between the heads of conjuncts. Similarities are obtained from manually assigned semantic tags (Agarwal and Boggess, 1992), a thesaurus (Resnik, 1999) and a distributional thesaurus (Chantree et al., 2005). Other approaches used cooccurrence statistics. To determine the attachments of ambiguous coordinate noun phrases, Goldberg (1999) applied a cooccurrence-based probabilistic model, and Nakov and Hearst (2005) used web-based frequencies. The performance of these methods ranges from 50% to 80%.

Of the above approaches, Resnik (1999) and Nakov and Hearst (2005) considered the statistics of noun-noun modification. For example, the coordinate structure “((mail and securities) fraud)” is guided by the estimation that *mail fraud* is a salient compound nominal phrase. On the other hand, the coordinate structure “(corn and (peanut butter))” is led because *corn butter* is not a familiar concept. They did not use the selectional preferences of the predicates that the conjuncts depend on. Therefore, this idea is subsumed into ours.

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

Several techniques have considered the characteristics of coordinate structures in a generative or reranking parser. Dubey et al. (2006) proposed an unlexicalized PCFG parser that modified PCFG probabilities to condition the existence of syntactic parallelism. Hogan (2007) improved a generative lexicalized parser by considering the symmetry between words in each conjunct. As for a reranking parser, Charniak and Johnson (2005) incorporated some features of syntactic parallelism in coordinate structures into their MaxEnt reranking parser.

Nilsson et al. tried to transform the tree representation of a treebank into a more suitable representation for data-driven dependency parsers (Nilsson et al., 2006; Nilsson et al., 2007). One of their targets is the representation of coordinate structures. They succeeded in improving a deterministic parser, but failed for a globally optimized discriminative parser.

Kurohashi and Nagao proposed a Japanese pars-

ing method that included coordinate structure detection (Kurohashi and Nagao, 1994). Their method first detects coordinate structures in a sentence, and then determines the dependency structure of the sentence under the constraints of the detected coordinate structures. Their method correctly analyzed 97 out of 150 Japanese sentences. Kawahara and Kurohashi (2007) integrated this method into a generative parsing model. Shimbo and Hara (2007) considered many features for coordination disambiguation and automatically optimized their weights, which were heuristically determined in Kurohashi and Nagao (1994), using a discriminative learning model.

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. Tamura et al. (2007) learned not only standard modifier-head relations but also ancestor-descendant relations. With this treatment, their method can indirectly improve the handling of coordinate structures in limited cases.

3 Background

3.1 Japanese Grammar

Let us first briefly introduce Japanese grammar. The structure of a Japanese sentence can be described well by the dependency relation between *bunsetsus*. A *bunsetsu* is a basic unit of dependency, consisting 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. The Japanese language is head-final, that is, a *bunsetsu* depends on another *bunsetsu* to its right (but not necessarily the adjacent *bunsetsu*).

For example, consider the following sentence¹:

(2) *ane-to gakkou-ni itta*
sister-CMI school-ALL went

(went to school with (my) sister)

¹In this paper, we use the following abbreviations: NOM (nominative), ACC (accusative), ABL (ablative), ALL (allative), CMI (comitative), CNJ (conjunction) and TM (topic marker).

This sentence consists of three *bunsetsus*. The final *bunsetsu*, *itta*, is a predicate, and the other *bunsetsus*, *ane-to* and *gakkou-ni*, are its arguments. Their endings, *to* and *ni*, are postpositions that function as case markers.

3.2 Treebank

To evaluate our method, we use a web corpus that is manually annotated using the criteria of the Kyoto Text Corpus (Kurohashi and Nagao, 1998). The Kyoto Text Corpus is syntactically annotated in dependency formalism, and consists of 40K Japanese newspaper sentences. The web corpus, which is used in our evaluation, consists of 759 sentences extracted from the web.

Under the annotation criteria of the Kyoto Text Corpus, the last *bunsetsu* in a pre-conjunct depends on the last *bunsetsu* in a post-conjunct, as shown in the dependency trees of Figure 1.

4 Our Idea of Addressing Coordination Ambiguities

The target of our approach is nominal coordinate structures. Consider, for example, the following sentence, which contains a nominal coordinate structure.

- (3) *jinkou-no* *zouka-to* *taiki-no*
 population-GEN increase-CNJ air-GEN

 osen-ga *sokushin-sareta*
 pollution-NOM stimulated

(increase of population and pollution of air were stimulated)

In this sentence, the postposition *to* is a coordinate conjunction². In Japanese, a coordinate conjunction is attached to a verb or noun, forming a *bunsetsu*, like case-marking postpositions. We call a *bunsetsu* that contains a coordinate conjunction *coordination key bunsetsu*.

The coordinate structure in example (3) has four possible scopes as depicted in Figure 1. In this figure, our parser generates the constituent words according to the arrows in the reverse direction. Note that the words that have “1/2” marks are generated from multiple words, because they depend

²Note that the postposition *to* can be used as a coordinate conjunction and also a comitative case marker as in example (2). The detection of coordinate conjunctions is a task of coordination disambiguation as well as the identification of coordination scopes. Both of these tasks are simultaneously handled in our method.

on a coordinate structure. In this case, their generative probabilities, which are described later, are averaged.

The scope patterns in Figure 1 can be written in English as follows:

- a. (population increase) and (air pollution)
- b. population (increase and (air pollution))
- c. ((population increase) and air) pollution
- d. population (increase and air) pollution

In (a) and (b), two arguments, *zouka* (increase) and *osen* (pollution), are generated from the verb *sokushin-sareta* (stimulated), and are eligible for the *ga* (NOM) words of the verb *sokushin-sareta* (stimulated). However, (b) is not appropriate, because we cannot say the nominal compound “*jinkou-no osen*” (pollution of population). In (c) and (d), the heads of conjuncts, *zouka* (increase) and *taiki* (air), are generated from *osen* (pollution). These cases are also inappropriate, because we cannot say the nominal compound “*zouka-no osen*” (pollution of increase). Accordingly, in this case, the correct scope, (a), is derived based on the selectional preferences of predicates and nouns.

In this framework, we require selectional preferences. We use case frames for predicates (Kawahara and Kurohashi, 2006) and occurrences of noun-noun modifications for nouns. Both of them are extracted from a large amount of raw text.

5 Our Model of Coordination Disambiguation

This section describes an integrated model of coordination disambiguation in a generative parsing framework. First, we describe resources for selectional preferences, and then illustrate our model of coordination disambiguation.

5.1 Resources for Selectional Preferences

As the resources of selectional preferences to support coordinate structures, we use automatically constructed case frames and cooccurrences of noun-noun modifications.

5.1.1 Automatically Constructed Case Frames

We employ automatically constructed case frames (Kawahara and Kurohashi, 2006). This section outlines the method for constructing the case frames.

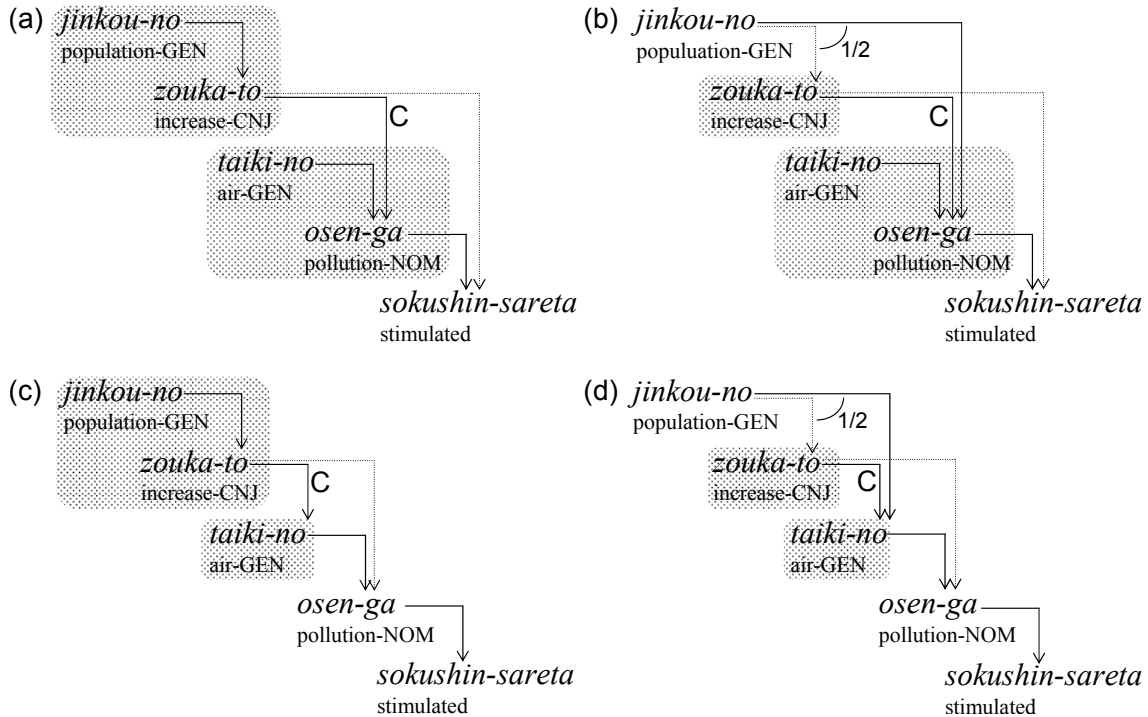


Figure 1: Four possible coordination scopes for example (3). Rounded rectangles represent conjuncts. The solid arrows represent dependency trees. The dotted arrows represent the additional processes of generation for coordinate structures. Note that the arrows with coordinate relation (“C” mark) do not participate in generation instead.

Table 1: Acquired case frames of *yaku*. Example words are expressed only in English due to space limitation. The number following each word denotes its frequency.

	CS	examples
<i>yaku</i> (1) (bake)	<i>ga</i> <i>wo</i> <i>de</i>	I:18, person:15, craftsman:10, ... bread:2484, meat:1521, cake:1283, ... oven:1630, frying pan:1311, ...
<i>yaku</i> (2) (have difficulty)	<i>ga</i> <i>wo</i> <i>ni</i>	teacher:3, government:3, person:3, ... fingers:2950 attack:18, action:15, son:15, ...
<i>yaku</i> (3) (burn)	<i>ga</i> <i>wo</i> <i>ni</i>	maker:1, distributor:1 data:178, file:107, copy:9, ... R:1583, CD:664, CDR:3, ...
⋮	⋮	⋮

A large corpus is automatically parsed, and case frames are constructed from modifier-head examples in the resulting parses. The problems of automatic case frame construction are syntactic and semantic ambiguities. That is to say, the parsing results inevitably contain errors, and verb senses are intrinsically ambiguous. To cope with these problems, case frames are gradually constructed from reliable modifier-head examples.

First, modifier-head examples that have no syntactic ambiguity are extracted, and they are disam-

biguated by a pair consisting of a verb and its closest case component. Such pairs are explicitly expressed on the surface of text, and are thought to play an important role in sentence meanings. For instance, examples are distinguished not by verbs (e.g., “*yaku*” (bake/broil/have difficulty)), but by pairs (e.g., “*pan-wo yaku*” (bake bread), “*niku-wo yaku*” (broil meat), and “*te-wo yaku*” (have difficulty)). Modifier-head examples are aggregated in this way, and yield basic case frames.

Thereafter, the basic case frames are clustered to merge similar case frames. For example, since “*pan-wo yaku*” (bake bread) and “*niku-wo yaku*” (broil meat) are similar, they are clustered. The similarity is measured using a thesaurus (The National Institute for Japanese Language, 2004).

Using this gradual procedure, we constructed case frames from a web corpus (Kawahara and Kurohashi, 2006). The case frames were obtained from approximately 500M sentences extracted from the web corpus. They consisted of 90,000 verbs, and the average number of case frames for a verb was 34.3.

In Table 1, some examples of the resulting case frames of the verb *yaku* are listed. In this table, ‘CS’ indicates a case slot.

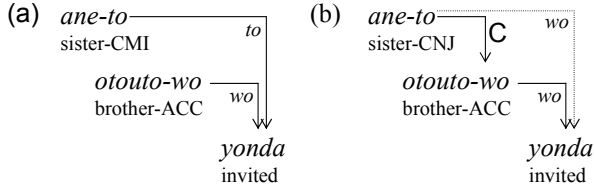


Figure 2: Dependency trees and generation processes for example (4). This example sentence has two possible dependency structures according to the interpretation of *to*: comitative in (a) and coordinate conjunction in (b).

5.1.2 Cooccurrences of Noun-noun Modifications

Adnominal nouns have selectional preferences to nouns, and thus this characteristic is useful for coordination disambiguation (Resnik, 1999). We collect dependency relations between nouns from automatic parses of the web corpus. As a result, 10.7M unique dependency relations were obtained.

5.2 Our Model

We employ a probabilistic generative dependency parser (Kawahara and Kurohashi, 2007) as a base model. This base model measures similarities between conjuncts in the same way as (Kurohashi and Nagao, 1994), and calculates probabilities of generating these similarities. Our proposed model, however, does not do both of them. Our model purely depends on selectional preferences provided by automatically acquired lexical knowledge.

Our model gives probabilities to all the possible dependency structures for an input sentence, and selects the structure that has the highest probability. For example, consider the following sentence:

- (4) *ane-to otouto-wo yonda*
 sister-CNJ brother-ACC invited
 (invited (my) sister and brother)

For this sentence, our model assesses the two dependency structures (a) and (b) in Figure 2. In our model, both of the pre-conjunct and post-conjunct are generated from the predicate. That is, in (b), both *ane* (sister) and *otouto* (brother) with *wo* (ACC) are generated from *yonda* (invited). To identify the correct structure, (b), it is essential that both *ane* (sister) and *otouto* (brother) are eligible for the accusative words of *yonda* (invited).

Therefore, selectional preferences play an important role in coordination disambiguation. On the other hand, in (a), *ane* (sister) with *to* (CMI) is generated from *yonda* (invited), and also *otouto* (brother) with *wo* (ACC) is generated from *yonda*. However, *yonda* is not likely to have the *to* case slot, so the probability of (a) is lower than that of (b). Our model can finally select the correct structure, (b), which has the highest probability. This kind of assessment is also performed to resolve the scope ambiguities of coordinate structures as shown in Figure 1.

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

$$\begin{aligned} (T_{best}, L_{best}) &= \operatorname{argmax}_{(T,L)} P(T, L|S) \\ &= \operatorname{argmax}_{(T,L)} \frac{P(T, L, S)}{P(S)} \\ &= \operatorname{argmax}_{(T,L)} P(T, L, S) \quad (1) \end{aligned}$$

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:

$$P(T, L, S) = \prod_{C_i \in S} P(C_i, rel_{ih_i} | C_{h_i}) \quad (2)$$

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 $C_{h_n} = \text{EOS}$ (End Of Sentence) is added. Dependency relation rel_{ih_i} is classified into two types, C (coordinate) and D (normal dependency).

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} .

$$\begin{aligned} P(C_i, rel_{ih_i} | C_{h_i}) &= P(C_i', f_i, rel_{ih_i} | C_{h_i}', f_{h_i}) \\ &\approx P(C_i', rel_{ih_i} | f_i, C_{h_i}') \times P(f_i | f_{h_i}) \\ &\approx P(C_i' | rel_{ih_i}, f_i, C_{h_i}') \times P(rel_{ih_i} | f_i) \\ &\quad \times P(f_i | f_{h_i}) \quad (3) \end{aligned}$$

Equation (3) is derived using appropriate approximations described in Kawahara and Kurohashi (2007).

We call $P(C_i' | rel_{ih_i}, f_i, C_{h_i}')$ *generative probability of a content part*, and $P(rel_{ih_i} | f_i)$ *generative probability of a dependency relation*. The following two subsections describe these probabilities.

5.2.1 Generative Probability of Dependency Relation

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

$$P(rel_{ih_i} | f_i) = P(rel_{ih_i} | k_i) \quad (4)$$

We classified coordination keys into 52 classes according to the classification described in (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 dependency relation was estimated from the Kyoto Text Corpus using maximum likelihood.

5.2.2 Generative Probability of Content Part

The generative probability of a content part changes according to the class of a content part, C_i' . We classify C_i' into two classes: predicate clause and nominal phrase.

If C_i' is a predicate clause, C_i' represents a 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 3. Thus, the generative probability of a content part is decomposed as follows:

$$\begin{aligned} & P_v(C_i' | rel_{ih_i}, f_i, C_{h_i}') \\ &= P(v_i, CF_l, CA_k | rel_{ih_i}, f_i, C_{h_i}') \\ &\approx P(v_i | rel_{ih_i}, f_i, w_{h_i}) \\ &\quad \times P(CF_l | v_i) \\ &\quad \times P(CA_k | CF_l, f_i) \end{aligned} \quad (5)$$

These generative probabilities are estimated from case frames themselves and parsing results of a large web corpus.

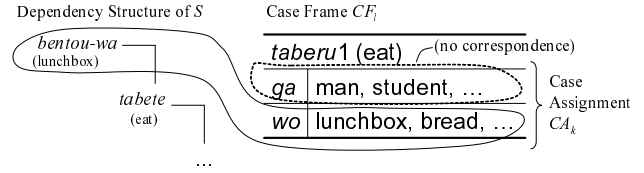


Figure 3: Example of case assignment.

If C_i' is a nominal phrase and consists of a noun n_i , we consider the following probability instead of equation (5):

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

This is because a noun does not have a case frame or any case components in the current framework. Since we do not use cooccurrences of coordinate phrases as used in the base model, rel_{ih_i} is always D (normal dependency). This probability is estimated from the cooccurrences of noun-noun modifications using maximum likelihood.

6 Experiments

We evaluated the dependency structures that were output by our model. The case frames used in this paper were automatically constructed from 500M Japanese sentences obtained from the web.

In this work, the parameters related to unlexical types were calculated from the Kyoto Text Corpus, which is a small tagged corpus of newspaper articles, and lexical parameters were obtained from a huge web corpus. To evaluate the effectiveness of our model, our experiments were conducted using web sentences. As the test corpus, we used 759 web sentences³, which are described in section 3.2. 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 analyzer⁴.

We used two baseline systems for comparative purposes: a rule-based dependency parser (Kurohashi and Nagao, 1994) and the probabilistic generative model of dependency, coordinate and case structure analysis (Kawahara and Kurohashi, 2007)⁵.

6.1 Evaluation of Dependency Structures

We evaluated the dependency structures that were analyzed by the proposed model. Evaluating the

³The test set was not used to construct case frames or estimate probabilities.

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

⁵<http://nlp.kuee.kyoto-u.ac.jp/nl-resource/knp-e.html>

Table 2: Experimental results of dependency structures. “all” represents the accuracy of all the dependencies, and “coordination key” represents the accuracy of only the coordination key *bunsetsus*.

	rule-coord-w/sim	prob-coord-w/sim	prob-coord-wo/sim
all	3,821/4,389 (87.1%)	3,852/4,389 (87.8%)	3,877/4,389 (88.3%)
coordination key	878/1,106 (79.4%)	881/1,106 (79.7%)	897/1,106 (81.1%)

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 one in the sentence end⁶. Table 2 lists the dependency accuracy. In this table, “rule-coord-w/sim” represents a rule-based dependency parser; “prob-coord-w/sim” represents the probabilistic parser of dependency, coordinate and case structure (Kawahara and Kurohashi, 2007); and “prob-coord-wo/sim” represents our proposed model. “all” represents the overall accuracy, and “coordination key” represents the accuracy of only the coordination key *bunsetsus*. The proposed model, “prob-coord-wo/sim”, significantly outperformed both “rule-coord-w/sim” and “prob-coord-w/sim” (McNemar’s test; $p < 0.05$) for “all”.

Figure 4 shows some analyses that are correctly analyzed by the proposed method. For example, in sentence (1), our model can recognize the correct coordinate structure that conjoins “*densya-no hassyaaizu*” (departure signals of trains) and “*keitaidenwa-no tyakushinon*” (ring tones of cell phones). This is because the case frame of “*ongaku-ni naru*” (become music) is likely to generate “*hassyaaizu*” (departure signal) and “*tyakushinon*” (ring tone).

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, CaboCha⁷ (Kudo and Matsumoto, 2002). Its dependency accuracy was 86.7% (3,807/4,389), which is close to that of “rule-coord-w/sim”. This low accuracy is attributed to the lack of the consideration of coordinate structures. Though dependency structures are closely related to coordinate structures, the CaboCha parser failed to incorporate coordination features. Another cause of the low accuracy is 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.

6.2 Discussion

We presented a method for coordination disambiguation without using similarities, and this method achieved better performance than the conventional approaches based on similarities. Though we do not use similarities, we implicitly consider similarities between conjuncts. This is because the heads of pre- and post-conjuncts share a case marker and a predicate, and thus they are essentially similar. Our idea is related to the notion of distributional similarity. Chantree et al. (2005) applied the distributional similarity proposed by Lin (1998) to coordination disambiguation. Lin extracted from a corpus dependency triples of two words and the grammatical relationship between them, and considered that similar words are likely to have similar dependency relations. The difference between Chantree et al. (2005) and ours is that their method does not use the information of verbs in the sentence under consideration, but use only the cooccurrence information extracted from a corpus.

On the other hand, the disadvantage of our model is that it cannot consider the parallelism of conjuncts, which still seems to exist in especially strong coordinate structures. Handling of such parallelism is an open question of our model.

The generation process adopted in this work is similar to the design of dependency structure described in Hudson (1990), which lets the conjuncts have a dependency relation to the predicate. Nilsson et al. (2006) mentioned this notion, but did not consider this idea in their experiments of tree transformations for data-driven dependency parsers. In addition, it is not necessary for our method to transform dependency trees in pre- and post-processes, because we just changed the process of generation in the generative parser.

7 Conclusion

In this paper, we first came up with a hypothesis that coordinate structures are supported by

⁶Since Japanese is head-final, the second to last *bunsetsu* unambiguously depends on the last *bunsetsu*, and the last *bunsetsu* has no dependency.

⁷<http://chasen.org/~taku/software/cabocha/>

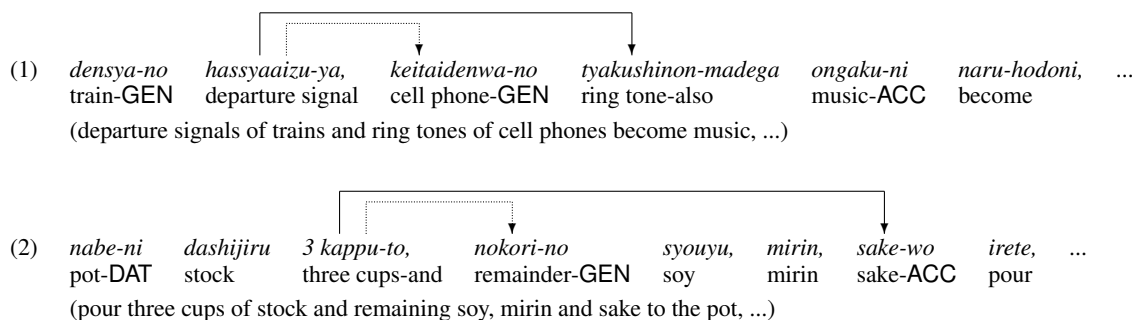


Figure 4: Examples of correct analyses. The dotted lines represent the analysis by the baseline, “prob-coord-w/sim”, and the solid lines represent the analysis by the proposed method, “prob-coord-wo/sim”.

surrounding dependency relations. Based on this hypothesis, we built an integrated probabilistic model for coordination disambiguation and dependency/case structure analysis. This model does not make use of similarities to analyze coordinate structures, but takes advantage of selectional preferences from a huge raw corpus and large-scale case frames. The experimental results indicate the effectiveness of our model, and thus support our hypothesis. Our future work involves incorporating ellipsis resolution to develop an integrated model for syntactic, case, and ellipsis analysis.

References

- Agarwal, Rajeev and Lois Boggles. 1992. A simple but useful approach to conjunct identification. In *Proceedings of ACL1992*, pages 15–21.
- Chantree, Francis, Adam Kilgarriff, Anne de Roeck, and Alistair Wills. 2005. Disambiguating coordinations using word distribution information. In *Proceedings of RANLP2005*.
- Charniak, Eugene and Mark Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of ACL2005*, pages 173–180.
- Dubey, Amit, 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.
- Goldberg, Miriam. 1999. An unsupervised model for statistically determining coordinate phrase attachment. In *Proceedings of ACL1999*, pages 610–614.
- Hogan, Deirdre. 2007. Coordinate noun phrase disambiguation in a generative parsing model. In *Proceedings of ACL2007*, pages 680–687.
- Hudson, Richard. 1990. *English Word Grammar*. Blackwell.
- Kawahara, Daisuke and Sadao Kurohashi. 2006. Case frame compilation from the web using high-performance computing. In *Proceedings of LREC2006*.
- Kawahara, Daisuke and Sadao Kurohashi. 2007. Probabilistic coordination disambiguation in a fully-lexicalized Japanese parser. In *Proceedings of EMNLP-CoNLL2007*, pages 306–314.
- Kudo, Taku and Yuji Matsumoto. 2002. Japanese dependency analysis using cascaded chunking. In *Proceedings of CoNLL2002*, pages 29–35.
- Kurohashi, Sadao 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.
- Kurohashi, Sadao and Makoto Nagao. 1998. Building a Japanese parsed corpus while improving the parsing system. In *Proceedings of LREC1998*, pages 719–724.
- Lin, Dekang. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of COLING-ACL98*, pages 768–774.
- Nakov, Preslav and Marti Hearst. 2005. Using the web as an implicit training set: Application to structural ambiguity resolution. In *Proceedings of HLT-EMNLP2005*, pages 835–842.
- Nilsson, Jens, Joakim Nivre, and Johan Hall. 2006. Graph transformations in data-driven dependency parsing. In *Proceedings of COLING-ACL2006*, pages 257–264.
- Nilsson, Jens, Joakim Nivre, and Johan Hall. 2007. Generalizing tree transformations for inductive dependency parsing. In *Proceedings of ACL2007*, pages 968–975.
- Resnik, Philip. 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.
- Sassano, Manabu. 2004. Linear-time dependency analysis for Japanese. In *Proceedings of COLING2004*, pages 8–14.
- Shimbo, Masashi and Kazuo Hara. 2007. A discriminative learning model for coordinate conjunctions. In *Proceedings of EMNLP-CoNLL2007*, pages 610–619.
- Tamura, Akihiro, Hiroya Takamura, and Manabu Okumura. 2007. Japanese dependency analysis using the ancestor-descendant relation. In *Proceedings of EMNLP-CoNLL2007*, pages 600–609.
- The National Institute for Japanese Language. 2004. *Bunruigoihyo*. Dainippon Tosho, (In Japanese).