Coordination Structure Analysis using Dual Decomposition

Atsushi Hanamoto [1]   **Takuya Matsuzaki** [1]   **Jun'ichi Tsujii** [2]

1. Department of Computer Science, University of Tokyo, Japan
2. Web Search & Mining Group, Microsoft Research Asia, China

{hanamoto, matuzaki}@is.s.u-tokyo.ac.jp
jtsujii@microsoft.com

Abstract

Coordination disambiguation remains a difficult sub-problem in parsing despite the frequency and importance of coordination structures. We propose a method for disambiguating coordination structures. In this method, dual decomposition is used as a framework to take advantage of both HPSG parsing and coordinate structure analysis with alignment-based local features. We evaluate the performance of the proposed method on the Genia corpus and the Wall Street Journal portion of the Penn Treebank. Results show it increases the percentage of sentences in which coordination structures are detected correctly, compared with each of the two algorithms alone.

1 Introduction

Coordination structures often give syntactic ambiguity in natural language. Although a wrong analysis of a coordination structure often leads to a totally garbled parsing result, coordination disambiguation remains a difficult sub-problem in parsing, even for state-of-the-art parsers.

One approach to solve this problem is a grammatical approach. This approach, however, often fails in noun and adjective coordinations because there are many possible structures in these coordinations that are grammatically correct. For example, a noun sequence of the form "n_0 n_1 *and n_2 n_3*" has as many as five possible structures (Resnik, 1999). Therefore, a grammatical approach is not sufficient to disambiguate coordination structures. In fact, the Stanford parser (Klein and Manning, 2003) and Enju (Miyao and Tsujii, 2004) fail to disambiguate a sentence *I am*

a freshman advertising and marketing major. Table 1 shows the output from them and the correct coordination structure.

The coordination structure above is obvious to humans because there is a symmetry of conjuncts (*-ing*) in the sentence. Coordination structures often have such structural and semantic symmetry of conjuncts. One approach is to capture local symmetry of conjuncts. However, this approach fails in VP and sentential coordinations, which can easily be detected by a grammatical approach. This is because conjuncts in these coordinations do not necessarily have local symmetry.

It is therefore natural to think that considering both the syntax and local symmetry of conjuncts would lead to a more accurate analysis. However, it is difficult to consider both of them in a dynamic programming algorithm, which has been often used for each of them, because it explodes the computational and implementational complexity. Thus, previous studies on coordination disambiguation often dealt only with a restricted form of coordination (e.g. noun phrases) or used a heuristic approach for simplicity.

In this paper, we present a statistical analysis model for coordination disambiguation that uses the dual decomposition as a framework. We consider both of the syntax, and structural and semantic symmetry of conjuncts so that it outperforms existing methods that consider only either of them. Moreover, it is still simple and requires only $O(n^4)$ time per iteration, where n is the number of words in a sentence. This is equal to that of coordination structure analysis with alignment-based local features. The overall system still has a quite simple structure because we need just slight modifications of existing models in this approach,

Stanford parser/Enju

I am a (freshman advertising) and (marketing major)

Correct coordination structure

I am a freshman ((advertising and marketing) major)

Table 1: Output from the Stanford parser, Enju and the correct coordination structure

so we can easily add other modules or features for future.

The structure of this paper is as follows. First, we describe three basic methods required in the technique we propose: 1) coordination structure analysis with alignment-based local features, 2) HPSG parsing, and 3) dual decomposition. Finally, we show experimental results that demonstrate the effectiveness of our approach. We compare three methods: coordination structure analysis with alignment-based local features, HPSG parsing, and the dual-decomposition-based approach that combines both.

2 Related Work

Many previous studies for coordination disambiguation have focused on a particular type of NP coordination (Hogan, 2007). Resnik (1999) disambiguated coordination structures by using semantic similarity of the conjuncts in a taxonomy. He dealt with two kinds of patterns, $[n_0\ n_1$ and $n_2\ n_3]$ and $[n_1$ and $n_2\ n_3]$, where n_i are all nouns. He detected coordination structures based on similarity of form, meaning and conceptual association between n_1 and n_2 and between n_1 and n_3. Nakov and Hearst (2005) used the Web as a training set and applied it to a task that is similar to Resnik's.

In terms of integrating coordination disambiguation with an existing parsing model, our approach resembles the approach by Hogan (2007). She detected noun phrase coordinations by finding symmetry in conjunct structure and the dependency between the lexical heads of the conjuncts. They are used to rerank the n-best outputs of the Bikel parser (2004), whereas two models interact with each other in our method.

Shimbo and Hara (2007) proposed an alignment-based method for detecting and disambiguating non-nested coordination structures.

They disambiguated coordination structures based on the edit distance between two conjuncts. Hara et al. (2009) extended the method, dealing with nested coordinations as well. We used their method as one of the two sub-models.

3 Background

3.1 Coordination structure analysis with alignment-based local features

Coordination structure analysis with alignment-based local features (Hara et al., 2009) is a hybrid approach to coordination disambiguation that combines a simple grammar to ensure consistent global structure of coordinations in a sentence, and features based on sequence alignment to capture local symmetry of conjuncts. In this section, we describe the method briefly.

A sentence is denoted by $\boldsymbol{x} = x_1...x_k$, where x_i is the i-th word of $\boldsymbol{x}$. A *coordination boundaries set* is denoted by $\boldsymbol{y} = y_1...y_k$, where

$$
y_i = \begin{cases} (b_l, e_l, b_r, e_r) & \text{(if } x_i \text{ is a coordinating} \\ & \text{conjunction having left} \\ & \text{conjunct } x_{b_l}...x_{e_l} \text{ and} \\ & \text{right conjunct } x_{b_r}...x_{e_r}) \\ null & \text{(otherwise)} \end{cases}
$$

In other words, y_i has a non-null value only when it is a coordinating conjunction. For example, a sentence *I bought books and stationary* has a coordination boundaries set $(null, null, null, (3, 3, 5, 5), null)$.

The score of a coordination boundaries set is defined as the sum of score of all coordinating conjunctions in the sentence.

$$
\begin{aligned}
score(\boldsymbol{x}, \boldsymbol{y}) &= \sum_{m=1}^{k} score(\boldsymbol{x}, y_m) \\
&= \sum_{m=1}^{k} \boldsymbol{w} \cdot \boldsymbol{f}(\boldsymbol{x}, y_m) \qquad (1)
\end{aligned}
$$

where $\boldsymbol{f}(\boldsymbol{x}, y_m)$ is a real-valued feature vector of the coordination conjunct x_m. We used almost the same feature set as Hara et al. (2009): namely, the surface word, part-of-speech, suffix and prefix of the words, and their combinations. We used the averaged perceptron to tune the weight vector $\boldsymbol{w}$.

Hara et al. (2009) proposed to use a context-free grammar to find a properly nested coordination structure. That is, the scoring function Eq (1)

COORD	Coordination.
CJT	Conjunct.
N	Non-coordination.
CC	Coordinating conjunction like "and".
W	Any word.

Table 2: Non-terminals

Rules for coordinations:
$$COORD_{i,m} \rightarrow CJT_{i,j} CC_{j+1,k-1} CJT_{k,m}$$
Rules for conjuncts:
$$CJT_{i,j} \rightarrow (COORD|N)_{i,j}$$
Rules for non-coordinations:
$$N_{i,k} \rightarrow COORD_{i,j} N_{j+1,k}$$
$$N_{i,j} \rightarrow W_{i,i} (COORD|N)_{i+1,j}$$
$$N_{i,i} \rightarrow W_{i,i}$$
Rules for pre-terminals:
$$CC_{i,i} \rightarrow (and|or|but|,|;|+|+/-)_i$$
$$CC_{i,i+1} \rightarrow (,|;|)_i (and|or|but)_{i+1}$$
$$CC_{i,i+2} \rightarrow (as)_i (well)_{i+1} (as)_{i+2}$$
$$W_{i,i} \rightarrow *_i$$

Table 3: Production rules

Figure 1: subject-head schema (left) and head-complement schema (right); taken from Miyao et al. (2004).

is only defined on the coordination structures that are licensed by the grammar. We only slightly extended their grammar for convering more variety of coordinating conjunctions.

Table 2 and Table 3 show the non-terminals and production rules used in the model. The only objective of the grammar is to ensure the consistency of two or more coordinations in a sentence, which means for any two coordinations they must be either non-overlapping or nested coordinations. We use a bottom-up chart parsing algorithm to output the coordination boundaries with the highest score. Note that these production rules don't need to be isomorphic to those of HPSG parsing and actually they aren't. This is because the two methods interact only through dual decomposition and the search spaces defined by the methods are considered separately.

This method requires $O(n^4)$ time, where n is the number of words. This is because there are $O(n^2)$ possible coordination structures in a sentence, and the method requires $O(n^2)$ time to get a feature vector of each coordination structure.

3.2 HPSG parsing

HPSG (Pollard and Sag, 1994) is one of the linguistic theories based on lexicalized grammar
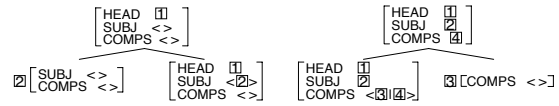
formalism. In a lexicalized grammar, quite a small numbers of schemata are used to explain general grammatical constraints, compared with other theories. On the other hand, rich word-specific characteristics are embedded in lexical entries. Both of schemata and lexical entries are represented by typed feature structures, and constraints in parsing are checked by *unification* among them. Figure 1 shows examples of HPSG schema.

Figure 2 shows an HPSG parse tree of the sentence "*Spring has come.*" First, the lexical entries of "*has*" and "*come*" are joined by head-complement schema. Unification gives the HPSG sign of mother. After applying schemata to HPSG signs repeatedly, the HPSG sign of the whole sentence is output.

We use Enju for an English HPSG parser (Miyao et al., 2004). Figure 3 shows how a co-ordination structure is built in the Enju grammar. First, a coordinating conjunction and the right conjunct are joined by coord_right_schema. Afterwards, the parent and the left conjunct are joined by coord_left_schema.

The Enju parser is equipped with a disam-biguation model trained by the maximum entropy method (Miyao and Tsujii, 2008). Since we do not need the probability of each parse tree, we treat the model just as a linear model that defines the score of a parse tree as the sum of feature weights. The features of the model are defined on local subtrees of a parse tree.

The Enju parser takes $O(n^3)$ time since it uses the CKY algorithm, and each cell in the CKY parse table has at most a constant number of edges because we use beam search algorithm. Thus, we can regard the parser as a decoder for a weighted CFG.

3.3 Dual decomposition

Dual decomposition is a classical method to solve complex optimization problems that can be de-
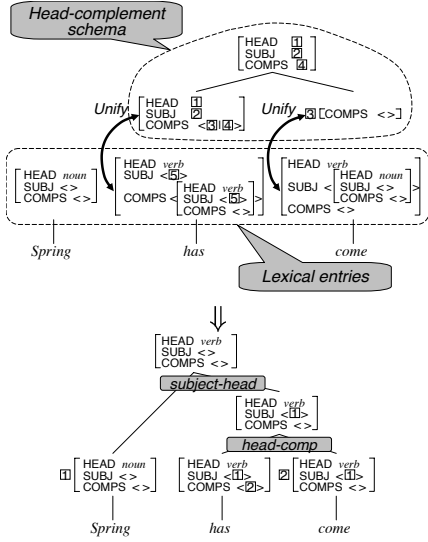
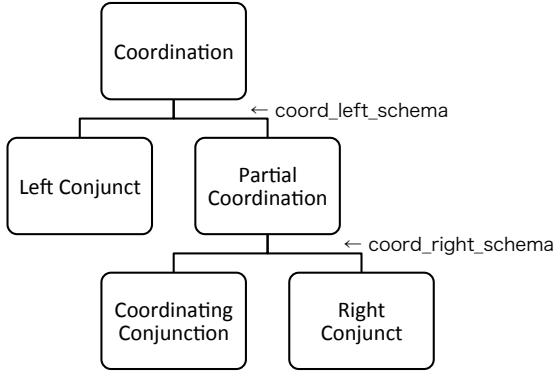Figure 2: HPSG parsing; taken from Miyao et al. (2004).

Figure 3: Construction of coordination in Enju

composed into efficiently solvable sub-problems. It is becoming popular in the NLP community and has been shown to work effectively on several NLP tasks (Rush et al., 2010).

We consider an optimization problem

$$\arg\max_x(f(x) + g(x)) \qquad (2)$$

which is difficult to solve (e.g. NP-hard), while $\arg\max_x f(x)$ and $\arg\max_x g(x)$ are effectively solvable. In dual decomposition, we solve

$$\min_u \max_{x,y}(f(x) + g(y) + u(x - y))$$

instead of the original problem.

To find the minimum value, we can use a subgradient method (Rush et al., 2010). The subgradient method is given in Table 4. As the algorithm

$u^{(1)} \leftarrow 0$
for $k = 1$ to K **do**
  $x^{(k)} \leftarrow \arg\max_x(f(x) + u^{(k)}x)$
  $y^{(k)} \leftarrow \arg\max_y(g(y) - u^{(k)}y)$
  if $x = y$ **then**
    return $u^{(k)}$
  end if
  $u^{(k+1)} \leftarrow u^k - a_k(x^{(k)} - y^{(k)})$
end for
return $u^{(K)}$

Table 4: The subgradient method

shows, you can use existing algorithms and don't need to have an exact algorithm for the optimization problem, which are features of dual decomposition.

If $x^{(k)} = y^{(k)}$ occurs during the algorithm, then we simply take $x^{(k)}$ as the primal solution, which is the exact answer. If not, we simply take $x^{(K)}$, the answer of coordination structure analysis with alignment-based features, as an approximate answer to the primal solution. The answer does not always solve the original problem Eq (2), but previous works (e.g., (Rush et al., 2010)) has shown that it is effective in practice. We use it in this paper.

4 Proposed method

In this section, we describe how we apply dual decomposition to the two models.

4.1 Notation

We define some notations here. First we describe weighted CFG parsing, which is used for both coordination structure analysis with alignment-based features and HPSG parsing. We follows the formulation by Rush et al., (2010). We assume a context-free grammar in Chomsky normal form, with a set of non-terminals N. All rules of the grammar are either the form $A \rightarrow BC$ or $A \rightarrow w$ where $A, B, C \in N$ and $w \in V$. For rules of the form $A \rightarrow w$ we refer to A as the pre-terminal for w.

Given a sentence with n words, $w_1 w_2 ... w_n$, a parse tree is a set of rule productions of the form $\langle A \rightarrow BC, i, k, j \rangle$ where $A, B, C \in N$, and $1 \leq i \leq k \leq j \leq n$. Each rule production represents the use of CFG rule $A \rightarrow BC$ where non-terminal A spans words $w_i ... w_j$, non-terminal B

spans word $w_i...w_k$, and non-terminal C spans word $w_{k+1}...w_j$ if $k < j$, and the use of CFG rule $A \rightarrow w_i$ if $i = k = j$.

We now define the *index set* for the coordination structure analysis as

$$\mathcal{I}_{csa} = \{\langle A \rightarrow BC, i, k, j \rangle : A, B, C \in N,$$
$$1 \leq i \leq k \leq j \leq n\}$$

Each parse tree is a vector $y = \{y_r : r \in \mathcal{I}_{csa}\}$, with $y_r = 1$ if rule r is in the parse tree, and $y_r = 0$ otherwise. Therefore, each parse tree is represented as a vector in $\{0,1\}^m$, where $m = |\mathcal{I}_{csa}|$. We use $\mathcal{Y}$ to denote the set of all valid parse-tree vectors. The set $\mathcal{Y}$ is a subset of $\{0,1\}^m$.

In addition, we assume a vector $\theta^{csa} = \{\theta_r^{csa} : r \in \mathcal{I}_{csa}\}$ that specifies a score for each rule production. Each θ_r^{csa} can take any real value. The optimal parse tree is $y^* = \arg\max_{y \in \mathcal{Y}} y \cdot \theta^{csa}$ where $y \cdot \theta^{csa} = \sum_r y_r \cdot \theta_r^{csa}$ is the inner product between y and θ^{csa}.

We use similar notation for HPSG parsing. We define $\mathcal{I}_{hpsg}$, $\mathcal{Z}$ and θ^{hpsg} as the index set for HPSG parsing, the set of all valid parse-tree vectors and the weight vector for HPSG parsing respectively.

We extend the index sets for both the coordination structure analysis with alignment-based features and HPSG parsing to make a constraint between the two sub-problems. For the coordination structure analysis with alignment-based features we define the extended index set to be $\mathcal{I}'_{csa} = \mathcal{I}_{csa} \bigcup \mathcal{I}_{uni}$ where

$$\mathcal{I}_{uni} = \{(a, b, c) : a, b, c \in \{1...n\}\}$$

Here each triple (a, b, c) represents that word w_c is recognized as the last word of the right conjunct and the scope of the left conjunct or the coordinating conjunction is $w_a...w_b$[1]. Thus each parse-tree vector y will have additional components $y_{a,b,c}$. Note that this representation is over-complete, since a parse tree is enough to determine unique coordination structures for a sentence: more explicitly, the value of $y_{a,b,c}$ is

---

[1]This definition is derived from the structure of a coordination in Enju (Figure 3). The triples show where the coordinating conjunction and right conjunct are in coord_right_schema, and the left conjunct and partial coordination are in coord_left_schema. Thus they alone enable not only the coordination structure analysis with alignment-based features but Enju to uniquely determine the structure of a coordination.

1 if rule $\mathsf{COORD}_{a,c} \rightarrow \mathsf{CJT}_{a,b}\mathsf{CC}_{\_,\_}\mathsf{CJT}_{\_,c}$ or $\mathsf{COORD}_{\_,c} \rightarrow \mathsf{CJT}_{\_,\_}\mathsf{CC}_{a,b}\mathsf{CJT}_{\_,c}$ is in the parse tree; otherwise it is 0.

We apply the same extension to the HPSG index set, also giving an over-complete representation. We define $z_{a,b,c}$ analogously to $y_{a,b,c}$.

4.2 Proposed method

We now describe the dual decomposition approach for coordination disambiguation. First, we define the set $\mathcal{Q}$ as follows:

$$\mathcal{Q} = \{(y, z) : y \in \mathcal{Y}, z \in \mathcal{Z}, y_{a,b,c} = z_{a,b,c}$$
$$\text{for all } (a, b, c) \in \mathcal{I}_{uni}\}$$

Therefore, $\mathcal{Q}$ is the set of all (y, z) pairs that agree on their coordination structures. The coordination structure analysis with alignment-based features and HPSG parsing problem is then to solve

$$\max_{(y,z) \in \mathcal{Q}} (y \cdot \theta^{csa} + \gamma z \cdot \theta^{hpsg}) \qquad (3)$$

where $\gamma > 0$ is a parameter dictating the relative weight of the two models and is chosen to optimize performance on the development test set.

This problem is equivalent to

$$\max_{z \in \mathcal{Z}} (g(z) \cdot \theta^{csa} + \gamma z \cdot \theta^{hpsg}) \qquad (4)$$

where $g : \mathcal{Z} \rightarrow \mathcal{Y}$ is a function that maps a HPSG tree z to its set of coordination structures $z = g(y)$.

We solve this optimization problem by using dual decomposition. Figure 4 shows the resulting algorithm. The algorithm tries to optimize the combined objective by separately solving the sub-problems again and again. After each iteration, the algorithm updates the weights $u(a, b, c)$. These updates modify the objective functions for the two sub-problems, encouraging them to agree on the same coordination structures. If $y^{(k)} = z^{(k)}$ occurs during the iterations, then the algorithm simply returns $y^{(k)}$ as the exact answer. If not, the algorithm returns the answer of coordination analysis with alignment features as a heuristic answer.

It is needed to modify original sub-problems for calculating (1) and (2) in Table 4. We modified the sub-problems to regard the score of $u(a, b, c)$ as a bonus/penalty of the coordination. The modified coordination structure analysis with alignment features adds $u^{(k)}(i, j, m)$ and $u^{(k)}(j+1, l-$

$$u^{(1)}(a, b, c) \leftarrow 0 \text{ for all } (a, b, c) \in \mathcal{I}_{uni}$$
for $k = 1$ to K **do**
    $y^{(k)} \leftarrow \arg\max_{y \in \mathcal{Y}}(y \cdot \theta^{csa} - \sum_{(a,b,c) \in \mathcal{I}_{uni}} u^{(k)}(a, b, c) y_{a,b,c}) \ldots (1)$
    $z^{(k)} \leftarrow \arg\max_{z \in \mathcal{Z}}(z \cdot \theta^{hpsg} + \sum_{(a,b,c) \in \mathcal{I}_{uni}} u^{(k)}(a, b, c) z_{a,b,c}) \ldots (2)$
    if $y^{(k)}(a, b, c) = z^{(k)}(a, b, c)$ for all $(a, b, c) \in \mathcal{I}_{uni}$ **then**
        return $y^{(k)}$
    end if
    for all $(a, b, c) \in \mathcal{I}_{uni}$ **do**
        $u^{(k+1)}(a, b, c) \leftarrow u^{(k)}(a, b, c) - a_k(y^{(k)}(a, b, c) - z^{(k)}(a, b, c))$
    end for
end for
return $y^{(K)}$

Figure 4: Proposed algorithm

$1, m)$, as well as adding $\boldsymbol{w} \cdot \boldsymbol{f}(\boldsymbol{x}, (i, j, l, m))$ to the score of the subtree, when the rule production COORD$_{i,m} \rightarrow$ CJT$_{i,j}$CC$_{j+1,l-1}$CJT$_{l,m}$ is applied.

The modified Enju adds $u^{(k)}(a, b, c)$ when coord_right_schema is applied, where word $w_a...w_b$ is recognized as a coordinating conjunction and the last word of the right conjunct is w_c, or coord_left_schema is applied, where word $w_a...w_b$ is recognized as the left conjunct and the last word of the right conjunct is w_c.

COORD	WSJ	Genia
NP	63.7	66.3
VP	13.8	11.4
ADJP	6.8	9.6
S	11.4	6.0
PP	2.4	5.1
Others	1.9	1.5

Table 6: The percentage of each conjunct type (%) of each test set

5 Experiments

5.1 Test/Training data

We trained the alignment-based coordination analysis model on both the Genia corpus (Kim et al., 2003) and the Wall Street Journal portion of the Penn Treebank (Marcus et al., 1993), and evaluated the performance of our method on (i) the Genia corpus and (ii) the Wall Street Journal portion of the Penn Treebank. More precisely, we used HPSG treebank converted from the Penn Treebank and Genia, and further extracted the training/test data for coordination structure analysis with alignment-based features using the annotation in the Treebank. Table 5 shows the corpus used in the experiments.

The Wall Street Journal portion of the Penn Treebank in the test set has 2317 sentences from WSJ articles, and there are 1356 coordinations in the sentences, while the Genia corpus in the test set has 1764 sentences from MEDLINE abstracts, and there are 1848 coordinations in the sentences. Coordinations are further subcatego-

rized into phrase types such as a NP coordination or PP coordination. Table 6 shows the percentage of each phrase type in all coordianitons. It indicates the Wall Street Journal portion of the Penn Treebank has more VP coordinations and S coordianitons, while the Genia corpus has more NP coordianitons and ADJP coordiations.

5.2 Implementation of sub-problems

We used Enju (Miyao and Tsujii, 2004) for the implementation of HPSG parsing, which has a wide-coverage probabilistic HPSG grammar and an efficient parsing algorithm, while we re-implemented Hara et al., (2009)'s algorithm with slight modifications.

5.2.1 Step size

We used the following step size in our algorithm (Figure 4). First, we initialized a_0, which is chosen to optimize performance on the development set. Then we defined $a_k = a_0 \cdot 2^{-\eta_k}$, where η_k is the number of times that $L(u^{(k')}) > L(u^{(k'-1)})$ for $k' \leq k$.

435

	Task (i)	Task (ii)
Training	WSJ (sec. 2–21) + Genia (No. 1–1600)	WSJ (sec. 2–21)
Development	Genia (No. 1601–1800)	WSJ (sec. 22)
Test	Genia (No. 1801–1999)	WSJ (sec. 23)

Table 5: The corpus used in the experiments

	Proposed	Enju	CSA
Precision	**72.4**	66.3	65.3
Recall	**67.8**	65.5	60.5
F1	**70.0**	65.9	62.8

Table 7: Results of Task (i) on the test set. The precision, recall, and F1 (%) for the proposed method, Enju, and Coordination structure analysis with alignment-based features (CSA)
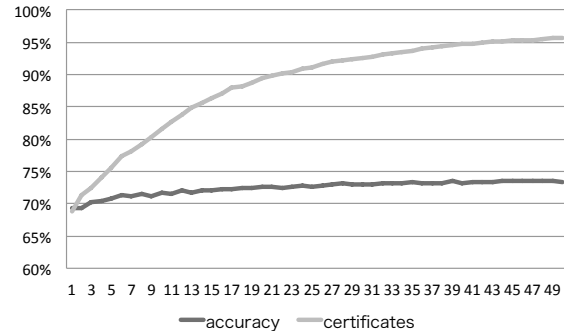

Figure 5: Performance of the approach as a function of K of Task (i) on the development set. accuracy (%): the percentage of sentences that are correctly parsed. certificates (%): the percentage of sentences for which a certificate of optimality is obtained.

5.3 Evaluation metric

We evaluated the performance of the tested methods by the accuracy of coordination-level bracketing (Shimbo and Hara, 2007); i.e., we count each of the coordination scopes as one output of the system, and the system output is regarded as correct if both of the beginning of the first output conjunct and the end of the last conjunct match annotations in the Treebank (Hara et al., 2009).

5.4 Experimental results of Task (i)

We ran the dual decomposition algorithm with a limit of $K = 50$ iterations. We found the two sub-problems return the same answer during the algorithm in over 95% of sentences.

We compare the accuracy of the dual decomposition approach to two baselines: Enju and coordination structure analysis with alignment-based features. Table 7 shows all three results. The dual decomposition method gives a statistically significant gain in precision and recall over the two methods[2].

Table 8 shows the recall of coordinations of each type. It indicates our re-implementation of CSA and Hara et al. (2009) have a roughly similar performance, although their experimental settings are different. It also shows the proposed method took advantage of Enju and CSA in NP coordination, while it is likely just to take the answer of Enju in VP and sentential coordinations. This means we might well use dual decomposi-

tion only on NP coordinations to have a better result.

Figure 5 shows performance of the approach as a function of K, the maximum number of iterations of dual decomposition. The graphs show that values of K much less than 50 produce almost identical performance to $K = 50$ (with $K = 50$, the accuracy of the method is 73.4%, with $K = 20$ it is 72.6%, and with $K = 1$ it is 69.3%). This means you can use smaller K in practical use for speed.

5.5 Experimental results of Task (ii)

We also ran the dual decomposition algorithm with a limit of $K = 50$ iterations on Task (ii). Table 9 and 10 show the results of task (ii). They show the proposed method outperformed the two methods statistically in precision and recall[3].

Figure 6 shows performance of the approach as a function of K, the maximum number of iterations of dual decomposition. The convergence speed for WSJ was faster than that for Genia. This is because a sentence of WSJ often have a simpler coordination structure, compared with that of Genia.

---

[2] $p < 0.01$ (by chi-square test)

[3] $p < 0.01$ (by chi-square test)

COORD	#	Proposed	Enju	CSA		#	Hara et al. (2009)
Overall	1848	**67.7**	63.3	61.9		3598	61.5
NP	1213	**67.5**	61.4	64.1		2317	64.2
VP	208	**79.8**	78.8	66.3		456	54.2
ADJP	193	58.5	**59.1**	54.4		312	80.4
S	111	51.4	**52.3**	34.2		188	22.9
PP	110	**64.5**	59.1	57.3		167	59.9
Others	13	**78.3**	73.9	65.2		140	49.3

Table 8: The number of coordinations of each type (#), and the recall (%) for the proposed method, Enju, coordination structure analysis with alignment-based features (CSA) , and Hara et al. (2009) of Task (i) on the development set. Note that Hara et al. (2009) uses a different test set and different annotation rules, although its test data is also taken from the Genia corpus. Thus we cannot compare them directly.

	Proposed	Enju	CSA
Precision	**76.3**	70.7	66.0
Recall	**70.6**	69.0	60.1
F1	**73.3**	69.9	62.9

Table 9: Results of Task (ii) on the test set. The precision, recall, and F1 (%) for the proposed method, Enju, and Coordination structure analysis with alignment-based features (CSA)

COORD	#	Proposed	Enju	CSA
Overall	1017	**71.6**	68.1	60.7
NP	573	**76.1**	71.0	67.7
VP	187	62.0	**62.6**	47.6
ADJP	73	**82.2**	75.3	79.5
S	141	**64.5**	62.4	42.6
PP	19	**52.6**	47.4	47.4
Others	24	62.5	**70.8**	54.2

Table 10: The number of coordinations of each type (#), and the recall (%) for the proposed method, Enju, and coordination structure analysis with alignment-based features (CSA) of Task (ii) on the development set.

Figure 6: Performance of the approach as a function of K of Task (ii) on the development set. accuracy (%): the percentage of sentences that are correctly parsed. certificates (%): the percentage of sentences for which a certificate of optimality is provided.

6 Conclusion and Future Work

In this paper, we presented an efficient method for detecting and disambiguating coordinate structures. Our basic idea was to consider both grammar and symmetries of conjuncts by using dual decomposition. Experiments on the Genia corpus and the Wall Street Journal portion of the Penn Treebank showed that we could obtain statistically significant improvement in accuracy when using dual decomposition.

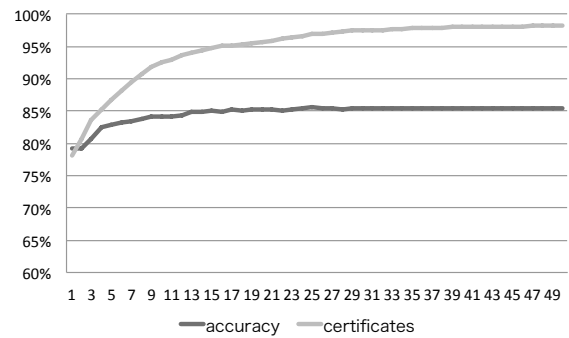We would need a further study in the following points of view: First, we should evaluate our method with corpus in different domains. Because characteristics of coordination structures differs from corpus to corpus, experiments on other corpus would lead to a different result. Second, we would want to add some features to coordination structure analysis with alignment-based local features such as ontology. Finally, we can add other methods (e.g. dependency parsing) as sub-problems to our method by using the extension of dual decomposition, which can deal with more than two sub-problems.

References

Kazuo Hara, Masashi Shimbo, Hideharu Okuma, and Yuji Matsumoto. 2009. Coordinate structure analysis with global structural constraints and alignment-based local features. In *Proceedings of the 47th Annual Meeting of the ACL and the 4th IJCNLP of the AFNLP*, pages 967–975, Aug.

Deirdre Hogan. 2007. Coordinate noun phrase disambiguation in a generative parsing model. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics (ACL 2007)*, pages 680–687.

Jun-Dong Kim, Tomoko Ohta, and Jun'ich Tsujii. 2003. Genia corpus - a semantically annotated corpus for bio-textmining. *Bioinformatics*, 19.

Dan Klein and Christopher D. Manning. 2003. Fast exact inference with a factored model for natural language parsing. *Advances in Neural Information Processing Systems*, 15:3–10.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19:313–330.

Yusuke Miyao and Jun'ich Tsujii. 2004. Deep linguistic analysis for the accurate identification of predicate-argument relations. In *Proceeding of COLING 2004*, pages 1392–1397.

Yusuke Miyao and Jun'ich Tsujii. 2008. Feature forest models for probabilistic hpsg parsing. *MIT Press*, 1(34):35–80.

Yusuke Miyao, Takashi Ninomiya, and Jun'ichi Tsujii. 2004. Corpus-oriented grammar development for acquiring a head-driven phrase structure grammar from the penn treebank. In *Proceedings of the First International Joint Conference on Natural Language Processing (IJCNLP 2004)*.

Preslav Nakov and Marti Hearst. 2005. Using the web as an implicit training set: Application to structural ambiguity resolution. In *Proceedings of the Human Language Technology Conference and Conference on Empirical Methods in Natural Language (HLT-EMNLP 2005)*, pages 835–842.

Carl Pollard and Ivan A. Sag. 1994. Head-driven phrase structure grammar. *University of Chicago Press*.

Philip Resnik. 1999. Semantic similarity in a takonomy. *Journal of Artificial Intelligence Research*, 11:95–130.

Alexander M. Rush, David Sontag, Michael Collins, and Tommi Jaakkola. 2010. On dual decomposition and linear programming relaxations for natural language processing. In *Proceeding of the conference on Empirical Methods in Natural Language Processing*.

Masashi Shimbo and Kazuo Hara. 2007. A discriminative learning model for coordinate conjunctions. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 610–619, Jun.