

# Recognizing Relation Expression between Named Entities based on Inherent and Context-dependent Features of Relational words

Toru Hirano<sup>†</sup>, Hisako Asano<sup>‡</sup>, Yoshihiro Matsuo<sup>†</sup>, Genichiro Kikui<sup>†</sup>

<sup>†</sup>NTT Cyber Space Laboratories, NTT Corporation

<sup>‡</sup>Innovative IP Architecture Center, NTT Communications Corporation

hirano.tohru@lab.ntt.co.jp

hisako.asano@ntt.com

{matsuo.yoshihiro,kikui.genichiro}@lab.ntt.co.jp

## Abstract

This paper proposes a supervised learning method to recognize expressions that show a relation between two named entities, e.g., person, location, or organization. The method uses two novel features, 1) whether the candidate words inherently express relations and 2) how the candidate words are influenced by the past relations of two entities. These features together with conventional syntactic and contextual features are organized as a tree structure and are fed into a boosting-based classification algorithm. Experimental results show that the proposed method outperforms conventional methods.

## 1 Introduction

Much attention has recently been devoted to using enormous amount of web text covering an exceedingly wide range of domains as a huge knowledge resource with computers. To use web texts as knowledge resources, we need to extract information from texts that are merely sequences of words and convert them into a structured form. Although extracting information from texts as a structured form is difficult, relation extraction is a way that makes it possible to use web texts as knowledge resources.

The aim of relation extraction is to extract semantically related named entity pairs,  $X$  and  $Y$ , and their relation,  $R$ , from a text as a structured form  $[X, Y, R]$ . For example, the triple [Yukio Hatoyama, Japan, prime minister] would be extracted from the text “Yukio Hatoyama is the prime minister of Japan”. This extracted triple

provides important information used in information retrieval (Zhu et al., 2009) and building an ontology (Wong et al., 2010).

It is possible to say that all named entity pairs that co-occur within a text are semantically related in some way. However, we define that named entity pairs are semantically related if they satisfy either of the following rules:

- One entity is an attribute value of the other.
- Both entities are arguments of a predicate.

Following the above definition, explicit and implicit relations should be extracted. An explicit relation means that there is an expression that shows the relation between a named entity pair in a given text, while an implicit relation means that there is no such expression. For example, the triple [Yukio Hatoyama, Kunio Hatoyama, brother] extracted from the text “Yukio Hatoyama, the Democratic Party, is Kunio Hatoyama’s brother” is an explicit relation. In contrast, the triple [Yukio Hatoyama, the Democratic Party, member] extracted from the same text is an implicit relation because there is no expression showing the relation (e.g. member) between “Yukio Hatoyama” and “the Democratic Party” in the text.

Extracting triples  $[X, Y, R]$  from a text involves two tasks. One is detecting semantically related pairs from named entity pairs that co-occur in a text and the other is determining the relation between a detected pair. For the former task, various supervised learning methods (Culotta and Sorensen, 2004; Zelenko et al., 2003; Hirano et al., 2007) and bootstrapping methods (Brin, 1998; Pantel and Pennacchiotti, 2006) have been explored to date. In contrast, for the latter task,

only a few methods have been proposed so far (Hasegawa et al., 2004; Banko and Etzioni, 2008; Zhu et al., 2009). We therefore addressed the problem of how to determine relations between a given pair.

We used a three-step approach to address this problem. The first step is to recognize an expression that shows explicit relations between a given named entity pair in a text. If no such expression is recognized, the second step is to estimate the relationship that exists between a given named entity pair that has an implicit relation. The last step is to identify synonyms of the relations that are recognized or estimated in the above steps. In this paper, we focus on the first step. The task is selecting a phrase from the text that contains a relation expression linking a given entity pair and outputting the expression as one showing the relationship between the pair.

In our preliminary experiment, it was found that using only structural features of a text, such as syntactic or contextual features, is not good enough for a number of examples. For instance, the two Japanese sentences shown in Figure 1 have the same syntactic structure but (a) contains a relation expression and (b) does not. We therefore assume there are clues for recognizing relation expressions other than conventional syntactic and contextual information. In this paper, we propose a supervised learning method that includes two novel features of relational words as well as conventional syntactic and contextual features. The novel features of our method are:

**Inherent Feature:** Some words are able to express the relations between named entities and some are not. Thus, it would be useful to know the words that inherently express these relations.

**Context-dependent Feature:** There are a number of typical relationships that change as time passes, such as “dating”  $\Rightarrow$  “engagement”  $\Rightarrow$  “marriage” between persons. Furthermore, present relations are influenced by the past relations of a given named entity pair. Thus, it would be useful to know the past relations between a given pair and how the relations change as time passes.

In the rest of this paper, Section 2 references related work, Section 3 outlines our method’s main features and related topics, Section 4 describes our experiments and experimental results, and Section 5 briefly summarizes key points and future work to be done.

## 2 Related Work

The “Message Understanding Conference” and “Automatic Content Extraction” programs have promoted relational extraction. The task was studied so as to extract predefined semantic relations of entity pairs in a text. Examples include the supervised learning method cited in (Kambhatla, 2004; Culotta and Sorensen, 2004; Zelenko et al., 2003) and the bootstrapping method cited in (Pantel and Pennacchiotti, 2006; Agichtein and Gravano, 2000). Recently, open information extraction (Open IE), a novel domain-independent extraction paradigm, has been suggested (Banko and Etzioni, 2008; Hasegawa et al., 2004). The task is to detect semantically related named entity pairs and to recognize the relation between them without using predefined relations.

Our work is a kind of open IE, but our approach differs from that of previous studies. Banko (2008) proposed a supervised learning method using conditional random fields to recognize the relation expressions from words located between a given pair. Hasegawa (2004) also proposed a rule-based method that selects all words located between a given pair as a relation expression if a given named entities appear within ten words. The point of these work is that they selected relation expressions only from the words located between

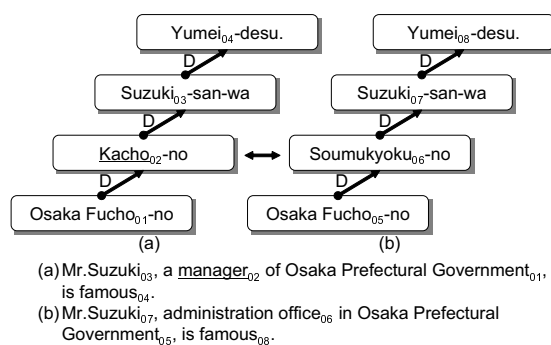


Figure 1: Same syntactic examples

given entities in the text, because as far as English texts are concerned, 86% of the relation expressions of named entity pairs appear between the pair (Banko and Etzioni, 2008). However, our target is Japanese texts, in which only 26% of entity pair relation expressions appear between the pair. Thus, it is hard to incorporate previous approaches into a Japanese text.

To solve the problem, our task was to select a phrase from the entire text that would include a relation expression for connecting a given pair.

### 3 Recognizing Relation Expressions between Named Entities

To recognize the relation expression for a given pair, we need to select a phrase that includes an expression that shows the relation between a given entity pair from among all noun and verb phrases in a text. Actually, there are two types of candidate phrases in this case. One is from a sentence that contains a given pair (intra-sentential), and the other is from a sentence that does not (inter-sentential). For example, the triple [Miyaji<sub>21</sub>, Ishii<sub>22</sub>, taiketsu<sub>12</sub>] extracted from the following text is inter-sentential.

(S-1) Chumokoku<sub>11</sub>-no taiketsu<sub>12</sub>-ga  
 mamonaku<sub>13</sub> hajimaru<sub>14</sub>.  
 (The showcase<sub>11</sub> match<sub>12</sub> will start<sub>14</sub> soon<sub>13</sub>.)

(S-2) Ano Miyaji<sub>21</sub>-to Ishii<sub>22</sub>-toiu  
 kanemochi<sub>23</sub>-niyoru yume<sub>24</sub>-no  
 kikaku<sub>25</sub>.  
 (The dream<sub>24</sub> event<sub>25</sub> between the rich mens<sub>23</sub>,  
 Miyaji<sub>21</sub> and Ishii<sub>22</sub>.)

According to our annotated data shown in Table 2, 53% of the semantically-related named entity pairs are intra-sentential and 12% are inter-sentential. Thus, we first select a phrase from those in a sentence that contains a given pair, and if no phrase is selected, select one from the rest of the sentences in a text.

We propose a supervised learning method that uses two novel features of relational words as well as conventional syntactic and contextual features. These features are organized as a tree structure and are fed into a boosting-based classification algorithm (Kudo and Matsumoto, 2004). The highest-scoring phrase is then selected if the score

exceeds a given threshold. Finally, the head of the selected phrase is output as the relation expression of a given entity pair.

The method consists of four parts: preprocessing (POS tagging and parsing), feature extraction, classification, and selection. In this section, we describe the idea behind using our two novel features and how they are implemented to recognize the relation expressions of given pairs. Before that, we will describe our proposed method’s conventional features.

#### 3.1 Conventional Features

##### Syntactic feature

To recognize the intra-sentential relation expressions for a given pair, we assume that there is a discriminative syntactic structure that consists of given entities and their relation expression. For example, there is a structure for which the common parent phrase of the given pair,  $X = \text{“Hatoyama Yukio}_{32}\text{”}$  and  $Y = \text{“Hatoyama Kunio}_{33}\text{”}$ , has the relation expression,  $R = \text{“ani}_{34}\text{”}$  in the Japanese sentence S-3. Figure 2 shows the dependency tree of sentence S-3.

(S-3) Minshuto<sub>31</sub>-no Hatoyama Yukio<sub>32</sub>-wa  
 Hatoyama Kunio<sub>33</sub>-no ani<sub>34</sub>-desu.  
 (Yukio Hatoyama<sub>32</sub>, the Democratic Party<sub>31</sub>,  
 is Kunio Hatoyama<sub>33</sub>’s brother<sub>34</sub>.)

To use a discriminative structure for each candidate, we make a minimum tree that consists of given entities and the candidate where each phrase is represented by a case marker “CM”, a dependency type “DT”, an entity class, and the string and POS of the candidate (See Figure 3).

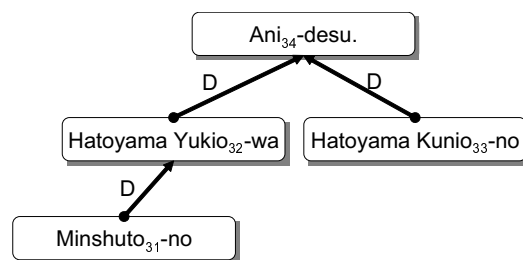


Figure 2: Dependency tree of sentence S-3

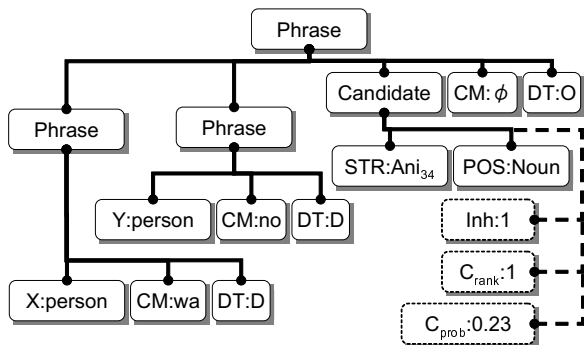


Figure 3: Intra-sentential feature tree

### Contextual Feature

To recognize the inter-sentential relation expressions for a given pair, we assume that there is a discriminative contextual structure that consists of given entities and their relation expression. Here, we use a Salient Referent List (SRL) to obtain contextual structure. The SRL is an empirical sorting rule proposed to identify the antecedent of (zero) pronouns (Nariyama, 2002), and Hirano (2007) proposed a way of applying SRL to relation detection. In this work, we use this way to apply SRL to recognize inter-sentential relation expressions.

We applied SRL to each candidate as follows. First, from among given entities and the candidate, we choose the one appearing last in the text as the root of the tree. We then append noun phrases, from the chosen one to the beginning of the text, to the tree depending on case markers, “wa” (topicalised subject), “ga” (subject), “ni” (indirect object), “wo” (object), and “others”, with the following rules. If there are nodes of the same case marker already in the tree, the noun phrase is appended as a child of the leaf node of them. In other cases, the noun phrase is appended as a child of the root node. For example, we get the SRL tree shown in Figure 4 with the given entity pair,  $X = \text{“Miyaji}_{21}\text{”}$  and  $Y = \text{“Ishii}_{22}\text{”}$ , and the candidate, “taiketsu<sub>12</sub>”, with the text (S-1, S-2).

To use a discriminative SRL structure, we make a minimum tree that consists of given entities and the candidate where each phrase is represented by an entity class, and the string and POS of the candidate (See Figure 5). In this way, there is a problem when the candidate is a verb phrase, because

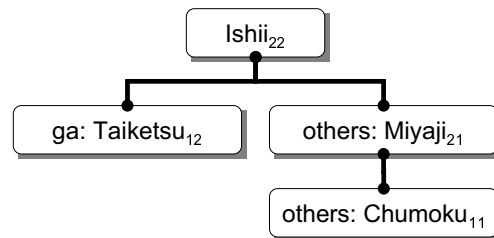


Figure 4: Salient referent list tree

only noun phrases are appended to the SRL tree. If the candidate is a verb phrase, we cannot make a minimum tree that consists of given entities and the candidate.

To solve this problem, a candidate verb phrase is appended to the feature tree using a syntactic structure. In a dependency tree, almost all verb phrases have some parent or child noun phrases that are in the SRL tree. Thus, candidate verb phrases are appended as offspring of these noun phrases represented syntactically as “parent” or “child”. For example, when given the entity pair,  $X = \text{“Miyaji}_{21}\text{”}$  and  $Y = \text{“Ishii}_{22}\text{”}$ , and the candidate, “hajimaru<sub>14</sub>” from the text (S-1, S-2), a feature tree cannot be made because the candidate is not in an SRL tree. By extending the way the syntactic structure is used, “hajimaru<sub>14</sub>” has a child node “taiketsu<sub>12</sub>”, which is in an SRL tree, and this makes it possible to make the feature tree shown in Figure 6.

### 3.2 Proposed Features

To recognize intra-sentential or inter-sentential relation expressions for given pairs, we assume there are clues other than syntactic and contextual information. Thus, we propose inherent and

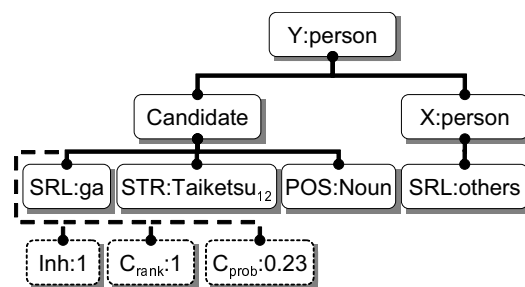


Figure 5: Inter-sentential feature tree

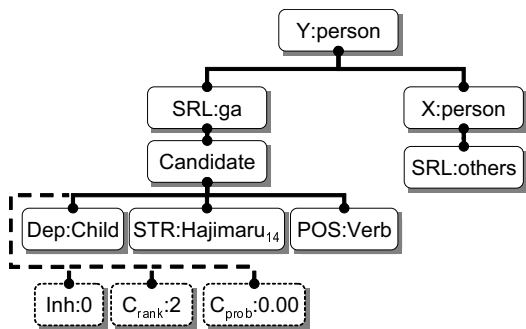


Figure 6: Extended inter-sentential feature tree

context-dependent features of relational words.

### Inherent Feature of Relational words

Some words are able to express the relations between named entities and some are not. For example, the word “mother” can express a relation, but the word “car” cannot. If there were a list of words that could express relations between named entities, it would be useful to recognize the relation expression of a given pair. As far as we know, however, no such list exists in Japanese. Thus, we estimate which words are able to express relations between entities. Here, we assume that almost all verbs are able to express relations, and accordingly we focus on nouns.

When the relation expression,  $R$ , of an entity pair,  $X$  and  $Y$ , is a noun, it is possible to say “ $Y$  is  $R$  of  $X$ ” or “ $Y$  is  $X$ ’s  $R$ ”. Here, we can say noun  $R$  takes an argument  $X$ . In linguistics, this kind of noun is called a relational noun. Grammatically speaking, a relational noun is a simple noun, but because its meaning describes a “relation” rather than a “thing”, it is used to describe relations just as prepositions do. To estimate which nouns are able to express the relations between named entities, we use the characteristics of relational nouns. In linguistics, many researchers describe the relationship between possessives and relational nouns (Chris, 2008). Thus, we use the knowledge that in the patterns “ $B$  of  $A$ ” or “ $A$ ’s  $B$ ”, if word  $B$  is a relational noun, the corresponding word  $A$  belongs to a certain semantic category. In contrast, if word  $B$  is not a relational noun, the corresponding word  $A$  belongs to many semantic categories (Tanaka et al., 1999). Figure 7 shows scattering of the semantic categories of “mother” and “car”

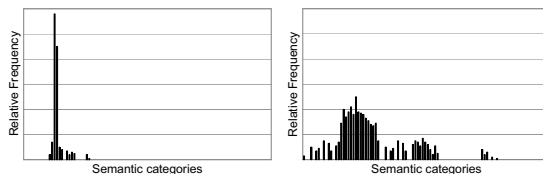


Figure 7: Scattering of semantic category of “mother” (left) and “car” (right).

acquired by the following way.

First, we acquired  $A$  and  $B$  using the patterns “ $A$  no  $B$ ”<sup>1</sup> from a large Japanese corpus, then mapped words  $A$  into semantic categories  $C = \{c_1, c_2, \dots, c_m\}$  using a Japanese lexicon (Ikehara et al., 1999). Next, for each word  $B$ , we calculated a scattering score  $\mathcal{H}c(B)$  using the semantic category of corresponding words  $A$ . Finally, we estimated whether a word is a relational noun by using  $k$ -NN estimation with positive and negative examples. As estimated results, “Inh:1” shows that it is a relational noun and “Inh:0” shows that it is not. In both cases, the result is appended to the feature tree as a child of the candidate node (See Figure 3, 5, or 6).

$$\mathcal{H}c(B) = - \sum_{c \in C} P(c|B) \log_m P(c|B)$$

$$P(c|B) = \frac{\text{freq}(c, B)}{\text{freq}(B)}$$

In our experiments, we acquired 55,412,811 pairs of  $A$  and  $B$  from 1,698,798 newspaper articles and 10,499,468 weblog texts. As training data, we used the words of relation expressions as positive examples and other words as negative examples.

### Context-dependent Feature of Relational words

There are a number of typical relationships that change as time passes, such as “dating”  $\Rightarrow$  “engagement”  $\Rightarrow$  “marriage” between persons. Furthermore, present relations are affected by the past relations of a given named entity pair. For instance, if the past relations of a given pair are “dating” and “engagement” and one of the candidates is “marriage”, “marriage” would be selected as the relation expression of the given pair. Therefore, if

<sup>1</sup>“ $B$  of  $A$ ” or “ $A$ ’s  $B$ ” in English.

Pair of entity class	$r_m$	$r_n$	$P_T(r_n r_m)$	$Count(r_m, r_n)$
$\langle \text{person, person} \rangle$	dating	dating	0.050	102
		marriage	0.050	101
		engagement	0.040	82
$\langle \text{person, person} \rangle$	engagement	marriage	0.157	786
		engagement	0.065	325
		wedding	0.055	276
$\langle \text{person, organization} \rangle$	vice president	president	0.337	17,081
		vice president	0.316	16,056
		CEO	0.095	4,798
$\langle \text{person, organization} \rangle$	researcher	fellow	0.526	61
		manager	0.103	12
		member	0.078	9
$\langle \text{organization, organization} \rangle$	alliance	alliance	0.058	8,358
		accommodated	0.027	3,958
		acquisition	0.027	3,863
$\langle \text{location, location} \rangle$	neighbour	mutual consultation	0.022	2,670
		support	0.015	1,792
		visit	0.012	1,492
$\langle \text{location, location} \rangle$	war	war	0.077	78,170
		mutual consultation	0.015	15,337
		support	0.010	10,226

Table 1: Examples of calculated relation trigger model between entity classes defined by IREX

we know the past relations of the given pair and the typical relational change that occurs as time passes, it would be useful to recognize the relation expression of a given pair.

In this paper, we represent typical relational changes that occur as time passes by a simple relation trigger model  $P_T(r_n|r_m)$ . Note that  $r_m$  is a past relation and  $r_n$  is a relation affected by  $r_m$ . This model disregards the span between  $r_n$  and  $r_m$ . To make the trigger model, we automatically extract triples  $[X, Y, R]$  from newspaper articles and weblog texts, which have time stamps of the document creation. Using these triples with time stamps for each entity pair, we sort relations in order of time and count pairs of present and previous relations. For example, if we extract “dating” occurring for an entity pair on January 10, 1998, “engagement” occurring on February 15, 2001, and “marriage” occurring on December 24, 2001, the pairs  $\langle \text{dating, engagement} \rangle$ ,  $\langle \text{dating, marriage} \rangle$ , and  $\langle \text{engagement, marriage} \rangle$  are counted. The counted score is then summed

up by the pair of entity class and the trigger model is calculated by the following formula.

$$P_T(r_n|r_m) = \frac{Count(r_m, r_n)}{\sum_{r_n} Count(r_m, r_n)}$$

For the evaluation, we extracted triples by named entity recognition (Suzuki et al., 2006), relation detection (Hirano et al., 2007), and the proposed method using the inherent features of relational words described in Section 3.2. A total of 10,463,232 triples were extracted from 8,320,042 newspaper articles and weblog texts with time stamps made between January 1, 1991 and June 30, 2006. As examples of the calculated relation trigger model, Table 1 shows the top three probability relations  $r_n$  of several relations  $r_m$  between Japanese standard named entity classes defined in the IREX workshop<sup>2</sup>. For instance, the relation “fellow” has the highest probability of being changed from the relation “researcher” between person and organization as time passes.

<sup>2</sup><http://nlp.cs.nyu.edu/irex/>

To obtain the past relations of a given pair in the input text, we again used the triples with time stamps extracted as above. The only relations we use as past relations,  $R_m = \{r_{m_1}, r_{m_2}, \dots, r_{m_k}\}$ , are those of a given pair whose time stamps are older than the input text. Finally, we calculated probabilities with the following formula using the past relations  $R_m$  and the trigger model  $P_T(r_n|r_m)$ .

$$P_T(r_n|R_m) = \max\{P_T(r_n|r_{m_1}), \\ P_T(r_n|r_{m_2}), \dots, P_T(r_n|r_{m_k})\}$$

Using this calculated probability, we ranked candidates and appended the rank “ $C_{rank}$ ” and the probability score “ $C_{prob}$ ” to the feature tree as a child of the candidate node (See Figure 3, 5, or 6). For example, if the past relations  $R_m$  were “dating” and “engagement” and candidates were “marriage”, “meeting”, “eating”, or “drinking”, the candidates probabilities were calculated and ranked as “marriage” ( $C_{prob}:0.15$ ,  $C_{rank}:1$ ), “meeting” ( $C_{prob}:0.08$ ,  $C_{rank}:2$ ), etc.

### 3.3 Classification Algorithms

Several structure-based learning algorithms have been proposed so far (Collins and Duffy, 2002; Suzuki et al., 2003; Kudo and Matsumoto, 2004). The experiments tested Kudo and Matsumoto’s boosting-based algorithm using sub-trees as features, which is implemented as a BACT system.

Given a set of training examples each of which is represented as a tree labeling whether the candidate is the relation expression of a given pair or not, the BACT system learns that a set of rules is effective in classifying. Then, given a test instance, the BACT system classifies using a set of learned rules.

## 4 Experiments

We conducted experiments using texts from Japanese newspaper articles and weblog texts to test the proposed method for both intra- and inter-sentential tasks. In the experiments, we compared the following methods:

**Conventional Features:** trained by conventional syntactic features for intra-sentential tasks as

Relation Types		#
Explicit	Intra-sentential	9,178
	Inter-sentential	2,058
Implicit		5,992
Total		17,228

Table 2: Details of the annotated data

described in Section 3.1, and contextual features for inter-sentential tasks as described in Section 3.1.

**+Inherent Features:** trained by conventional features plus inherent features of relational words described in Section 3.2.

**++Context-dependent Features<sub>TM</sub>:** trained by conventional and inherent features plus context-dependent features of relational words with the **trigger model** described in Section 3.2.

**++Context-dependent Features<sub>CM</sub>:** trained by conventional and inherent features plus context-dependent features of relational words with a **cache model**. We evaluated this method to compare it with Context-dependent Features<sub>TM</sub> to show the effectiveness of the proposed trigger model. The cache model is a simple way to use past relations in which the probability  $P_C(r_{cand})$  calculated by the following formula and the rank based on the probability is appended to every candidate feature tree.

$$P_C(r_{cand}) = \frac{|r_{cand} \text{ in past relations}|}{|\text{past relations}|}$$

### 4.1 Settings

We used 6,200 texts from Japanese newspapers and weblogs dated from January 1, 2004 to June 30, 2006, manually annotating the semantic relations between named entities for experiment purposes. There were 17,228 semantically-related entity pairs as shown in Table 2. In an intra-sentential experiment, 17,228 entity pairs were given, but only 9,178 of them had relation expressions. In contrast, in an inter-sentential experiment, 8,050 entity pairs excepted intra-sentential

	<i>Precision</i>	<i>Recall</i>	<i>F</i>
Conventional Features	63.5 % (3,436/5,411)	37.4 % (3,436/9,178)	0.471
+Inherent Features	67.2 % (4,036/6,001)	43.9 % (4,036/9,178)	0.531
++Context-dependent Features <sub>TM</sub>	70.7 % (4,460/6,312)	48.6 % (4,460/9,178)	<b>0.576</b>
++Context-dependent Features <sub>CM</sub>	67.5 % (4,042/5,987)	44.0 % (4,042/9,178)	0.533

Table 3: Experimental results of intra-sentential

	<i>Precision</i>	<i>Recall</i>	<i>F</i>
Conventional Features	70.1 % (579/825)	28.1 % (579/2,058)	0.401
+Inherent Features	77.1 % (719/932)	34.9 % (719/2,058)	0.480
++Context-dependent Features <sub>TM</sub>	75.2 % (794/1,055)	38.5 % (794/2,058)	<b>0.510</b>
++Context-dependent Features <sub>CM</sub>	74.3 % (732/985)	35.5 % (732/2,058)	0.481

Table 4: Experimental result of inter-sentential

were given, but only 2,058 of them had relation expressions.

We conducted five-fold cross-validation over 17,228 entity pairs so that sets of pairs from a single text were not divided into the training and test sets. In the experiments, all features were automatically acquired using a Japanese POS tagger (Fuchi and Takagi, 1998) and dependency parser (Imamura et al., 2007).

## 4.2 Results

Tables 3 and 4 show the performance of several methods for intra-sentential and inter-sentential. *Precision* is defined as the percentage of correct relation expressions out of recognized ones. *Recall* is the percentage of correct relation expressions from among the manually annotated ones. The *F* measure is the harmonic mean of precision and recall.

A comparison with the Conventional Features and Inherent Features method for intra-/inter-sentential tasks indicates that the proposed method using inherent features of relational words improved intra-sentential tasks *F* by 0.06 points and inter-sentential tasks *F* by 0.08 points. Using a statistical test (McNemar Test) demonstrably showed the proposed method’s effectiveness.

A comparison with the Inherent Features and Context-dependent Features<sub>TM</sub> method showed that the proposed method using context-dependent features of relational words improved intra-/inter-sentential task performance by 0.045 and 0.03

points, respectively. McNemar test results also showed the method’s effectiveness.

To further compare the usage of context-dependent features, trigger models, and cache models, we also used Context-dependent Features<sub>CM</sub> method for comparison. Tables 3 and 4 show that our proposed trigger model performed better than the cache model, and McNemar test results showed that there was a significant difference between the models. The reason the trigger model performed better than the cache model is that the trigger model correctly recognized the relation expressions that did not appear in the past relations of a given pair. Thus, we can conclude that using typical relationships that change as time passes helps to recognize relation expressions between named entities.

## 5 Conclusion

We proposed a supervised learning method that employs inherent and context-dependent features of relational words and uses conventional syntactic or contextual features to improve both intra- and inter-sentential relation expression recognition. Our experiments demonstrated that the method improves the *F* measure and thus helps to recognize relation expressions between named entities.

In future work, we plan to estimate implicit relations between named entities and to identify relational synonyms.



## References

- Agichtein, Eugene and Luis Gravano. 2000. Snowball: Extracting relations from large plain-text collections. In *Proceedings of the 5th ACM conference on Digital libraries*, pages 85–94.
- Banko, Michele and Oren Etzioni. 2008. The tradeoffs between open and traditional relation extraction. In *Proceedings of the 46th Annual Meeting on Association for Computational Linguistics: Human Language Technologies*, pages 28–36.
- Brin, Sergey. 1998. Extracting patterns and relations from the world wide web. In *WebDB Workshop at 6th International Conference on Extending Database Technology*, pages 172–183.
- Chris, Barker, 2008. *Semantics: An international handbook of natural language meaning*, chapter Possessives and relational nouns. Walter De Gruyter Inc.
- Collins, Michael and Nigel Duffy. 2002. Convolution kernels for natural language. *Advances in Neural Information Processing Systems*, 14:625–632.
- Culotta, Aron and Jeffrey Sorensen. 2004. Dependency tree kernels for relation extraction. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 423–429.
- Fuchi, Takeshi and Shinichiro Takagi. 1998. Japanese morphological analyzer using word co-occurrence - jtag. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, volume 1, pages 409–413.
- Hasegawa, Takaaki, Satoshi Sekine, and Ralph Grishman. 2004. Discovering relations among named entities from large corpora. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 415–422.
- Hirano, Toru, Yoshihiro Matsuo, and Genichiro Kikui. 2007. Detecting semantic relations between named entities in text using contextual features. In *Proceedings of the 45th Annual Meeting on Association for Computational Linguistics*, pages 157–160.
- Ikehara, Satoru, Masahiro Miyazaki, Satoru Shirai, Akio Yoko, Hiromi Nakaiwa, Kentaro Ogura, Masafumi Oyama, and Yoshihiko Hayashi. 1999. *Nihongo Goi Taikei (in Japanese)*. Iwanami Shoten.
- Imamura, Kenji, Genichiro Kikui, and Norihito Yasuda. 2007. Japanese dependency parsing using sequential labeling for semi-spoken language. In *Proceedings of the 45th Annual Meeting on Association for Computational Linguistics*, pages 225–228.
- Kambhatla, Nanda. 2004. Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, pages 178–181.
- Kudo, Taku and Yuji Matsumoto. 2004. A boosting algorithm for classification of semi-structured text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 301–308.
- Nariyama, Shigeiko. 2002. Grammar for ellipsis resolution in Japanese. In *Proceedings of the 9th International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 135–145.
- Pantel, Patrick and Marco Pennacchiotti. 2006. Espresso: Leveraging generic patterns for automatically harvesting semantic relations. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 113–120.
- Suzuki, Jun, Tsutomu Hirao, Yutaka Sasaki, and Eisaku Maeda. 2003. Hierarchical directed acyclic graph kernel: Methods for structured natural language data. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics*, pages 32–39.
- Suzuki, Jun, Erik McDermott, and Hideki Isozaki. 2006. Training conditional random fields with multivariate evaluation measures. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*.
- Tanaka, Shosaku, Yoichi Tomiura, and Toru Hitaka. 1999. Classification of syntactic categories of nouns by the scattering of semantic categories (in Japanese). *Transactions of Information Processing Society of Japan*, 40(9):3387–3396.
- Wong, Wilson, Wei Liu, and Mohammed Bennamoun. 2010. Acquiring semantic relations using the web for constructing lightweight ontologies. In *Proceedings of the 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining*.
- Zelenko, Dmitry, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of Machine Learning Research*, 3:1083–1106.
- Zhu, Jun, Zaiqing Nie, Xiaojing Liu, Bo Zhang, and Ji-Rong Wen. 2009. Statsnowball: a statistical approach to extracting entity relationships. In *Proceedings of the 18th international conference on World Wide Web*, pages 101–110.