Detecting Semantic Relations between Named Entities in Text Using Contextual Features

Toru Hirano, Yoshihiro Matsuo, Genichiro Kikui
NTT Cyber Space Laboratories, NTT Corporation
1-1 Hikarinooka, Yokosuka-Shi, Kanagawa, 239-0847, Japan
{hirano.tohru, matsuo.yoshihiro, kikui.genichiro}@lab.ntt.co.jp

Abstract

This paper proposes a supervised learning method for detecting a semantic relation between a given pair of named entities, which may be located in different sentences. The method employs newly introduced contextual features based on centering theory as well as conventional syntactic and word-based features. These features are organized as a tree structure and are fed into a boosting-based classification algorithm. Experimental results show the proposed method outperformed prior methods, and increased precision and recall by 4.4% and 6.7%.

1 Introduction

Statistical and machine learning NLP techniques are now so advanced that named entity (NE) taggers are in practical use. Researchers are now focusing on extracting semantic relations between NEs, such as "George Bush (*person*)" is "president (*relation*)" of "the United States (*location*)", because they provide important information used in information retrieval, question answering, and summarization.

We represent a semantic relation between two NEs with a tuple [NE_1, NE_2, Relation Label]. Our final goal is to extract tuples from a text. For example, the tuple [George Bush (*person*), the U.S. (*location*), president (*Relation Label*)] would be extracted from the sentence "George Bush is the president of the U.S.". There are two tasks in extracting tuples from text. One is detecting whether or not a given pair of NEs are semantically related (*relation detection*), and the other is determining the relation label (*relation characterization*).

In this paper, we address the task of relation detection. So far, various supervised learning approaches have been explored in this field (Culotta and Sorensen, 2004; Zelenko et al., 2003). They use two kinds of features: syntactic ones and word-based ones, for example, the path of the given pair of NEs in the parse tree and the word n-gram between NEs (Kambhatla, 2004).

These methods have two problems which we consider in this paper. One is that they target only intra-sentential relation detection in which NE pairs are located in the same sentence, in spite of the fact that about 35% of NE pairs with semantic relations are inter-sentential (See Section 3.1). The other is that the methods can not detect semantic relations correctly when NE pairs located in a parallel sentence arise from a predication ellipsis. In the following Japanese example[1], the syntactic feature, which is the path of two NEs in the dependency structure, of the pair with a semantic relation ("Ken_{11}" and "Tokyo_{12}") is the same as the feature of the pair with no semantic relation ("Ken_{11}" and "New York_{14}").

(S-1) *Ken_{11}-wa $Tokyo_{12}$-de, Tom_{13}-wa New $York_{14}$-de $umareta_{15}$.*
(Ken_{11} was born_{15} in Tokyo_{12}, Tom_{13} in New York_{14}.)

To solve the above problems, we propose a supervised learning method using contextual features.

The rest of this paper is organized as follows. Section 2 describes the proposed method. We report the results of our experiments in Section 3 and conclude the paper in Section 4.

2 Relation Detection

The proposed method employs contextual features based on centering theory (Grosz et al., 1983) as well as conventional syntactic and word-based features. These features are organized as a tree structure and are fed into a boosting-based classification algorithm. The method consists of three parts: preprocessing (POS tagging, NE tagging, and parsing),

[1]The numbers show correspondences of words between Japanese and English.

Proceedings of the ACL 2007 Demo and Poster Sessions, pages 157–160,
Prague, June 2007. ©2007 Association for Computational Linguistics

feature extraction (contextual, syntactic, and word-based features), and classification.

In this section, we describe the underlying idea of contextual features and how contextual features are used for detecting semantic relations.

2.1 Contextual Features

When a pair of NEs with a semantic relation appears in different sentences, the antecedent NE must be contextually easily referred to in the sentence with the following NE. In the following Japanese example, the pair "Ken$_{22}$" and "amerika$_{32}$ (the U.S.)" have a semantic relation "wataru$_{33}$ (go)", because "Ken$_{22}$" is contextually referred to in the sentence with "amerika$_{32}$" (In fact, the zero pronoun ϕ_i refers to "Ken$_{22}$"). Meanwhile, the pair "Naomi$_{25}$" and "amerika$_{32}$" has no semantic relation, because the sentence with "amerika$_{32}$" does not refer to "Naomi$_{25}$".

(S-2) *asu*$_{21}$*, Ken*$_{22}$*-wa Osaka*$_{23}$*-o otozure*$_{24}$ *Naomi*$_{25}$*-to au*$_{26}$*.*
(Ken$_{22}$ is going to visit$_{24}$ Osaka$_{23}$ to see$_{26}$ Naomi$_{25}$, tomorrow$_{21}$.)

(S-3) *sonogo*$_{31}$*, (*ϕ_i*-ga) amerika*$_{32}$*-ni watari*$_{33}$ *Tom*$_{34}$*-to ryoko*$_{35}$ *suru.*
(Then$_{31}$, (he$_i$) will go$_{33}$ to the U.S.$_{32}$ to travel$_{35}$ with Tom$_{34}$.)

Furthermore, when a pair of NEs with a semantic relation appears in a parallel sentence arise from predication ellipsis, the antecedent NE is contextually easily referred to in the phrase with the following NE. In the example of "(S-1)", the pair "Ken$_{11}$" and "Tokyo$_{12}$" have a semantic relation "umareta$_{15}$ (was born)". Meanwhile, the pair "Ken$_{11}$" and "New York$_{14}$" has no semantic relation.

Therefore, using whether the antecedent NE is referred to in the context with the following NE as features of a given pair of NEs would improve relation detection performance. In this paper, we use centering theory (Kameyama, 1986) to determine how easily a noun phrase can be referred to in the following context.

2.2 Centering Theory

Centering theory is an empirical sorting rule used to identify the antecedents of (zero) pronouns. When there is a (zero) pronoun in the text, noun phrases that are in the previous context of the pronoun are sorted in order of likelihood of being the antecedent. The sorting algorithm has two steps. First, from the beginning of the text until the pronoun appears, noun phrases are stacked depending on case markers such as particles. In the above example, noun phrases, "asu$_{21}$", "Ken$_{22}$", "Osaka$_{23}$" and "Naomi$_{25}$", which are in the previous context of the zero pronoun ϕ_i, are stacked and then the information shown in Figure 1 is acquired. Second, the stacked information is sorted by the following rules.

Priority

wa	Ken$_{22}$
ga	
ni	
o	Osaka$_{23}$
others	asu$_{21}$, Naomi$_{25}$

Figure 1: Information Stacked According to Centering Theory

1. The priority of case markers is as follows: "wa > ga > ni > o > others"
2. The priority of stack structure is as follows: last-in first-out, in the same case marker

For example, Figure 1 is sorted by the above rules and then the order, 1: "Ken$_{22}$", 2: "Osaka$_{23}$", 3: "Naomi$_{25}$", 4: "asu$_{21}$", is assigned. In this way, using centering theory would show that the antecedent of the zero pronoun ϕ_i is "Ken$_{22}$".

2.3 Applying Centering Theory

When detecting a semantic relation between a given pair of NEs, we use centering theory to determine how easily the antecedent NE can be referred to in the context with the following NE. Note that we do not explicitly execute anaphora resolutions here.

Applied centering theory to relation detection is as follows. First, from the beginning of the text until the following NE appears, noun phrases are stacked depending on case markers, and the stacked information is sorted by the above rules (Section 2.2). Then, if the top noun phrase in the sorted order is identical to the antecedent NE, the antecedent NE is "positive" when being referred to in the context with the following NE.

When the pair of NEs, "Ken$_{22}$" and "amerika$_{32}$", is given in the above example, the noun phrases, "asu$_{21}$", "Ken$_{22}$", "Osaka$_{23}$" and "Naomi$_{25}$", which are in the previous context of the following NE "amerika$_{32}$", are stacked (Figure 1). Then they are sorted by the above sorting rules and the order, 1: "Ken$_{22}$", 2: "Osaka$_{23}$", 3: "Naomi$_{25}$", 4: "asu$_{21}$", is acquired. Here, because the top noun phrase in the sorted order is identical to the antecedent NE, the antecedent NE "Ken$_{22}$" is "positive" when be-

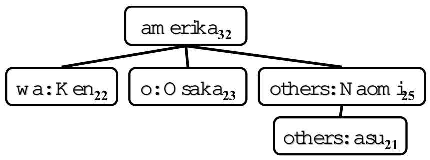

Figure 2: Centering Structure

ing referred to in the context with the following NE "amerika$_{32}$". Whether or not the antecedent NE is referred to in the context with the following NE is used as a feature. We call this feature Centering Top (CT).

2.4 Using Stack Structure

The sorting algorithm using centering theory tends to rank highly thoes words that easily become subjects. However, for relation detection, it is necessary to consider both NEs that easily become subjects, such as person and organization, and NEs that do not easily become subjects, such as location and time.

We use the stack described in Section 2.3 as a structural feature for relation detection. We call this feature Centering Structure (CS). For example, the stacked information shown in Figure 1 is assumed to be structure information, as shown in Figure 2. The method of converting from a stack (Figure 1) into a structure (Figure 2) is described as follows. First, the following NE, "amerika$_{32}$", becomes the root node because Figure 1 is stacked information until the following NE appears. Then, the stacked information is converted to Figure 2 depending on the case markers. We use the path of the given pair of NEs in the structure as a feature. For example, "amerika$_{32}$ → wa:Ken$_{22}$"[2] is used as the feature of the given pair "Ken$_{22}$" and "amerika$_{32}$".

2.5 Classification Algorithm

There are several structure-based learning algorithms proposed so far (Collins and Duffy, 2001; 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 the BACT system.

In relation detection, given a set of training examples each of which represents contextual, syntactic, and word-based features of a pair of NEs as a tree labeled as either having semantic relations or not, the BACT system learns that a set of rules are effective in classifying. Then, given a test instance, which represents contextual, syntactic, and word-based features of a pair of NEs as a tree, the BACT system classifies using a set of learned rules.

[2]"A → B" means A has a dependency relation to B.

Type	% of pairs with semantic relations
(A) Intra-sentential	31.4% (3333 / 10626)
(B) Inter-sentential	0.8% (1777 / 225516)
(A)+(B) Total	2.2% (5110 / 236142)

Table 1: Percent of pairs with semantic relations in annotated text

3 Experiments

We experimented with texts from Japanese newspapers and weblogs to test the proposed method. The following four models were compared:

1. **WD** : Pairs of NEs within n words are detected as pairs with semantic relation.
2. **STR** : Supervised learning method using syntactic[3] and word-based features, the path of the pairs of NEs in the parse tree and the word n-gram between pairs of NEs (Kambhatla, 2004)
3. **STR-CT** : STR with the centering top feature explained in Section 2.3.
4. **STR-CS** : STR with the centering structure feature explained in Section 2.4.

3.1 Setting

We used 1451 texts from Japanese newspapers and weblogs, whose semantic relations between person and location had been annotated by humans for the experiments[4]. There were 5110 pairs with semantic relations out of 236,142 pairs in the annotated text. We conducted ten-fold cross-validation over 236,142 pairs of NEs so that sets of pairs from a single text were not divided into the training and test sets.

We also divided pairs of NEs into two types: (A) intra-sentential and (B) inter-sentential. The reason for dividing them is so that syntactic structure features would be effective in type (A) and contextual features would be effective in type (B). Another reason is that the percentage of pairs with semantic relations out of the total pairs in the annotated text differ significantly between types, as shown in Table 1.

In the experiments, all features were automatically acquired using a Japanese morphological and dependency structure analyzer.

[3]There is no syntactic feature in inter-sentential.

[4]We are planning to evaluate the other pairs of NEs.

	(A)+(B) Total		(A) Intra-sentential		(B) Inter-sentential	
	Precision	Recall	Precision	Recall	Precsion	Recall
WD10	43.0(2501/5819)	48.9(2501/5110)	48.1(2441/5075)	73.2(2441/3333)	8.0(60/744)	3.4(60/1777)
STR	69.3(2562/3696)	50.1(2562/5110)	75.6(2374/3141)	71.2(2374/3333)	33.9(188/555)	10.6(188/1777)
STR-CT	71.4(2764/3870)	54.1(2764/5110)	78.4(2519/3212)	75.6(2519/3333)	37.2(245/658)	13.8(245/1777)
STR-CS	73.7(2902/3935)	56.8(2902/5110)	80.1(2554/3187)	76.6(2554/3333)	46.5(348/748)	27.6(348/1777)

WD10: NE pairs that appear within 10 words are detected.

Table 2: Results for Relation Detection

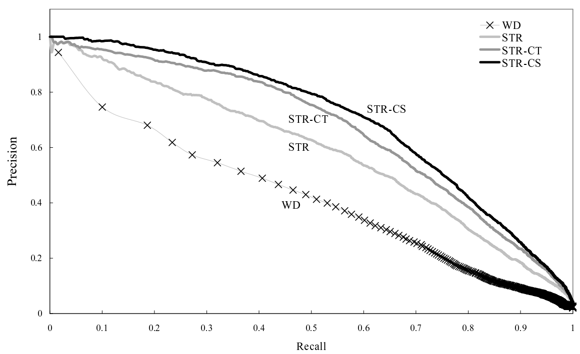

Figure 3: Recall-precision Curves: (A)+(B) total

3.2 Results

To improve relation detection performance, we investigated the effect of the proposed method using contextual features. Table 2 shows results for Type (A), Type (B), and (A)+(B). We also plotted recall-precision curves[5], altering threshold parameters, as shown in Figure 3.

The comparison between STR and STR-CT and between STR and STR-CS in Figure 3 indicates that the proposed method effectively contributed to relation detection. In addition, the results for Type (A): intra-sentential, and (B): inter-sentential, in Table 2 indicate that the proposed method contributed to both Type (A), improving precision by about 4.5% and recall by about 5.4% and Type (B), improving precision by about 12.6% and recall by about 17.0%.

3.3 Error Analysis

Over 70% of the errors are covered by two major problems left in relation detection.

Parallel sentence: The proposed method solves problems, which result from when a parallel sentence arises from predication ellipsis. However, there are several types of parallel sentence that differ from the one we explained. (For example, Ken and Tom was born in Osaka and New York, respectively.)

Definite anaphora: Definite noun phrase, such as "Shusho (the Prime Minister)" and "Shacho (the President)", can be anaphors. We should consider them in centering theory, but it is difficult to find them in Japanese .

4 Conclusion

In this paper, we propose a supervised learning method using words, syntactic structures, and contextual features based on centering theory, to improve both inter-sentential and inter-sentential relation detection. The experiments demonstrated that the proposed method increased precision by 4.4%, up to 73.7%, and increased recall by 6.7%, up to 56.8%, and thus contributed to relation detection.

In future work, we plan to solve the problems relating to parallel sentence and definite anaphora, and address the task of relation characterization.

References

M. Collins and N. Duffy. 2001. Convolution Kernels for Natural Language. *Proceedings of the Neural Information Processing Systems*, pages 625–632.

A. Culotta and J. Sorensen. 2004. Dependency Tree Kernels for Relation Extraction. *Annual Meeting of Association of Computational Linguistics*, pages 423–429.

B. J. Grosz, A. K. Joshi, and S. Weistein. 1983. Providing a unified account of definite nounphrases in discourse. *Annual Meeting of Association of Computational Linguistics*, pages 44–50.

N. Kambhatla. 2004. Combining Lexical, Syntactic, and Semantic Features with Maximum Entropy Models for Information Extraction. *Annual Meeting of Association of Computational Linguistics*, pages 178–181.

M. Kameyama. 1986. A property-sharing constraint in centering. *Annual Meeting of Association of Computational Linguistics*, pages 200–206.

T. Kudo and Y. Matsumoto. 2004. A boosting algorithm for classification of semi-structured text. *In Proceedings of the 2004 EMNLP*, pages 301–308.

J. Suzuki, T. Hirao, Y. Sasaki, and E. Maeda. 2003. Hierarchical directed acyclic graph kernel : Methods for structured natural language data. *Annual Meeting of Association of Computational Linguistics*, pages 32–39.

D. Zelenko, C. Aone, and A. Richardella. 2003. Kernel Methods for Relation Extraction. *Journal of Machine Learning Research*, pages 3:1083–1106.

[5]Precision = # of correctly detected pairs / # of detected pairs
Recall = # of correctly detected pairs / # of pairs with semantic relations