Japanese Named Entity Recognition Using Structural Natural Language Processing

Ryohei Sasano*
Graduate School of Information Science and Technology, University of Tokyo
ryohei@nlp.kuee.kyoto-u.ac.jp

Sadao Kurohashi
Graduate School of Infomatics, Kyoto University
kuro@i.kyoto-u.ac.jp

Abstract

This paper presents an approach that uses structural information for Japanese named entity recognition (NER). Our NER system is based on Support Vector Machine (SVM), and utilizes four types of structural information: cache features, coreference relations, syntactic features and caseframe features, which are obtained from structural analyses. We evaluated our approach on CRL NE data and obtained a higher F-measure than existing approaches that do not use structural information. We also conducted experiments on IREX NE data and an NE-annotated web corpus and confirmed that structural information improves the performance of NER.

1 Introduction

Named entity recognition (NER) is the task of identifying and classifying phrases into certain classes of named entities (NEs), such as names of persons, organizations and locations.

Japanese texts, which we focus on, are written without using blank spaces. Therefore, Japanese NER has tight relation with morphological analysis, and thus it is often performed immediately after morphological analysis (Masayuki and Matsumoto, 2003; Yamada, 2007). However, such approaches rely only on local context. The Japanese NER system proposed in (Nakano and Hirai, 2004), which achieved the highest F-measure among conventional systems, introduced the *bunsetsu*[1] feature in order to consider wider context, but considers only adjacent *bunsetsus*.

On the other hand, as for English or Chinese, various NER systems have explored global information and reported their effectiveness. In (Malouf, 2002; Chieu and Ng, 2002), information about features assigned to other instances of the same token is utilized. (Ji and Grishman, 2005) uses the information obtained from coreference analysis for NER. (Mohit and Hwa, 2005) uses syntactic features in building a semi-supervised NE tagger.

In this paper, we present a Japanese NER system that uses global information obtained from several structural analyses. To be more specific, our system is based on SVM, recognizes NEs after syntactic, case and coreference analyses and uses information obtained from these analyses and the NER results for the previous context, integrally. At this point, it is true that NER results are useful for syntactic, case and coreference analyses, and thus these analyses and NER should be performed in a complementary way. However, since we focus on NER, we recognize NE after these structural analyses.

2 Japanese NER Task

A common standard definition for Japanese NER task is provided by IREX workshop (IREX Committee, 1999). IREX defined eight NE classes as shown in Table 1. Compared with the MUC-6 NE task definition (MUC, 1995), the NE class "ARTIFACT," which contains book titles, laws, brand names and so on, is added.

NER task can be defined as a chunking problem to identify token sequences that compose NEs. The chunking problem is solved by annotating chunk tags to tokens. Five chunk tag sets, IOB1, IOB2, IOE1, IOE2 and IOBES are commonly used. In this paper, we use the IOBES model, in which "S" denotes a chunk itself, and "B," "I" and "E" denote the

*Research Fellow of the Japan Society for the Promotion of Science (JSPS)

[1] *Bunsetsu* is a commonly used linguistic unit in Japanese, consisting of one or more adjacent content words and zero or more following functional words.

Table 1: Definition of NE in IREX.

NE class	Examples
ORGANIZATION	NHK Symphony Orchestra
PERSON	Kawasaki Kenjiro
LOCATION	Rome, Sinuiju
ARTIFACT	Nobel Prize
DATE	July 17, April this year
TIME	twelve o'clock noon
MONEY	sixty thousand dollars
PERCENT	20%, thirty percents

beginning, intermediate and end parts of a chunk. If a token does not belong to any named entity, it is tagged as "O." Since IREX defined eight NE classes, tokens are classified into 33 (= 8 × 4 + 1) NE tags. For example, NE tags are assigned as following:

(1) *Kotoshi* *4* *gatsu* *Roma* *ni itta.*
this year April Rome to went
B-DATE I-DATE E-DATE S-LOCATION O O
(ϕ went to Rome on April this year.)

3 Motivation for Our Approach

Our NER system utilizes structural information. In this section, we describe the motivation for our approach.

High-performance Japanese NER systems are often based on supervised learning, and most of them use only local features, such as features obtained from the target token, two preceding tokens and two succeeding tokens. However, in some cases, NEs cannot be recognized by using only local features.

For example, while "*Kawasaki*" in the second sentence of (2) is the name of a person, "*Kawasaki*" in the second sentence of (3) is the name of a soccer team. However, the second sentences of (2) and (3) are exactly the same, and thus it is impossible to correctly distinguish these NE classes by only using information obtained from the second sentences.

(2) *Kachi-ha senpatsu-no* ***Kawasaki*** *Kenjiro.*
winner starter
Kawasaki-ha genzai 4 shou 3 pai.
now won lost
(The winning pitcher is the starter Kenjiro **Kawasaki**. Kawasaki has won 4 and lost 3.)

(3) *Dai 10 setsu-wa* ***Kawasaki*** *Frontale-to taisen.*
the round against
Kawasaki-ha genzai 4 shou 3 pai.
now won lost
(The 10th round is against **Kawasaki** Frontale. Kawasaki has won 4 and lost 3.)

In order to recognize these NE classes, it is essential to use the information obtained from the previous context. Therefore, we utilize information obtained from the NER for the previous context: **cache feature** and **coreference relation**.

For another example, "*Shingishu*" in (4) is the name of city in North Korea. The most important clue for recognizing "*Shingishu*" as "LOCATION" may be the information obtained from the head verb, "*wataru* (get across)."

(4) *Shingishu-kara Ouryokko-wo* ***wataru***.
Sinuiju from Amnokkang get across
(ϕ **gets across** the Amnokkang River from Sinuiju.)

However, when using only local features, the word "*wataru*" is not taken into consideration because there are more than two morphemes between "*shu*[2]" and "*wataru*." In order to deal with such problem, we use the information obtained from the head verb: **syntactic feature** and **caseframe feature**.

4 NER Using Structural Information

4.1 Outline of Our NER System

Our NER system performs the chunking process based on morpheme units because character-based methods do not outperform morpheme-based methods (Masayuki and Matsumoto, 2003) and are not suitable for considering wider context.

A wide variety of trainable models have been applied to Japanese NER task, including maximum entropy models (Utsuro et al., 2002), support vector machines (Nakano and Hirai, 2004; Yamada, 2007) and conditional random fields (Fukuoka, 2006). Our system applies SVMs because, for Japanese NER, SVM-based systems achieved higher F-measure than the other systems. (Isozaki and Kazawa, 2003) proposed an SVM-based NER system with Viterbi search, which outperforms an SVM-based NER system with sequential determination, and our system basically follows this system. Our NER system consists of the following four steps:

1. Morphological analysis
2. Syntactic, case and coreference analyses
3. Feature extraction for chunking
4. SVM and Viterbi search based chunking

The following sections describe each of these steps in detail.

[2]Since the dictionary for morphological analysis has no entry "*Shingishu*," "*Shingishu*" is analyzed as consisting of three morphemes: "*shin*," "*gi*" and "*shu*."

Input sentence:

Gai mu sho no shin Bei ha .
foreign affairs ministry in pro America group
(Pro-America group in the Ministry of Foreign Affairs.)

Output of JUMAN:

Gaimu	*sho*	*no*	*shin*	*Bei*	*ha*	.
noun	noun	particle	noun	noun	noun	

Output of ChaSen:

Gaimusho	*no*	*shin-Bei*	*ha*	.
noun	particle	noun	noun	

Figure 1: Example of morphological analyses.

4.2 Morphological Analysis

While most existing Japanese NER systems use ChaSen (Matsumoto et al., 2003) as a morphological analyzer, our NER system uses a Japanese morphological analyzer JUMAN (Kurohashi and Kawahara, 2005) because of the following two reasons.

First, JUMAN tends to segment a sentence into smaller morphemes than ChaSen, and this is a good tendency for morpheme-based NER systems because the boundary contradictions between morphological analysis and NEs are considered to be reduced. Figure 1 shows an example of the outputs of JUMAN and ChaSen. Although both analyses are reasonable, JUMAN divided "*Gaimusho*" and "*shin-Bei*" into two morphemes, while ChaSen left them as a single morpheme. Second, JUMAN adds categories to some morphemes, which can be utilized for NER. In JUMAN, about thirty categories are defined and tagged to about one fifth of morphemes. For example, "*ringo* (apple)," "*inu* (dog)" and "*byoin* (hospital)" are tagged as "FOOD," "ANIMAL" and "FACILITY," respectively.

4.3 Syntactic, Case and Coreference Analyses

syntactic analysis Syntactic analysis is performed by using the Japanese parser KNP (Kurohashi and Nagao, 1994). KNP employs some heuristic rules to determine the head of a modifier.

case analysis Case analysis is performed by using the system proposed in (Kawahara and Kurohashi, 2002). This system uses Japanese case frames that are automatically constructed from a large corpus. To utilize case analysis for NER, we constructed case frames that include NE labels in advance. We explain details in Section 4.4.2. The case analysis is applied to each predicate in an input sentence. For details see (Kawahara and Kurohashi, 2002).

coreference analysis Coreference analysis is performed by using the coreference analyzer proposed by (Sasano et al., 2007). As will be mentioned in Section 4.4.2, our NER system uses coreference relations only when coreferential expressions do not share same morphemes. Basically, such coreference relations are recognized by using automatically acquired synonym knowledge.

4.4 Feature Extraction

4.4.1 Basic Features

As basic features for chunking, our NER system uses the morpheme itself, character type, POS tag and category if it exists.

As character types, we defined seven types: "*kanji*," "*hiragana*," "*katakana*," "*kanji* with *hiragana*," "punctuation mark," "alphabet" and "digit." As for POS tag, more than one POS feature are extracted if the target morpheme has POS ambiguity. In addition, besides POS tag obtained by JUMAN, our system also uses POS tag obtained from Japanese morphological analyzer MeCab[3] that uses IPADIC as a word dictionary (Asahara and Matsumoto, 2002). The JUMAN dictionary has few named entity entries; thus our system supplements the lack of lexical knowledge by using MeCab.

4.4.2 Structural Features

Our NER system uses three types of global features: cache features, syntactic features and case-frame features, and a rule that reflects coreference relations. Although the coreference relations are not used as features, we describe how to use them in this section.

cache feature If the same morpheme appears multiple times in a single document, in most cases the NE tags of these morphemes have some relation to each other, and the NER results for previous parts of the document can be a clue for the analysis for following parts.

We consider the examples (2) and (3) again. Although the second sentences of (2) and (3) are exactly the same, we can recognize "*Kawasaki*" in the second sentence of (2) is "S-PERSON" and "*Kawasaki*" in the second sentence of (3) is "S-ORGANIZATION" by reading the first sentences.

To utilize the information obtained from previous parts of the document, our system uses the NER results for previous parts of the document as features, called cache features. When analyzing (2), our system uses the outputs of NE recognizer for

[3] http://mecab.sourceforge.jp/

"*Kawasaki*" in the first sentence as a feature for "*Kawasaki*" in the second sentence. For simplicity, our system uses correct NE tags when training. That is, as a feature for "*Kawasaki*" in the second sentence of (2), the correct feature "B-PERSON" is always added when training, not always added when analyzing.

coreference rule Coreference relation can be a clue for NER. This clue is considered by using cache features to a certain extent. However, if the same morpheme is not used, cache features cannot work.

For example, "*NHK kokyo gakudan*" and "*N-kyo*" in (5) have coreference relation, but they do not share the same morpheme.

(5) ***NHK kokyo gakudan****-no ongaku kantoku-ni*
symphony orchestra musical director
shuunin. *N-kyo-to kyoen-shite* *irai ... .*
became perform together since
(He became musical director of the **NHK Symphony Orchestra**. Since performing together with N-kyo ... .)

In this case, "*NHK kokyo gakudan*" can easily be recognized as "ORGANIZATION," because it ends with "*kokyo gakudan* (symphony orchestra)." Meanwhile, "*N-kyo*," the abbreviation of "*NHK kokyo gakudan*," cannot easily be recognized as "ORGANIZATION."

Therefore, our system uses a heuristic rule that if a morpheme sequence is analyzed to be coreferential to a previous morpheme sequence that is recognized as an NE class, the latter morpheme sequence is recognized as the same NE class. Since this heuristic rule is introduced in order to utilize the coreference relation that is not reflected by cache features, our system applies this rule only when coreferential expressions do not have any morphemes in common.

syntactic feature As mentioned in Section 3, our system utilizes the information obtained from the head verb. As syntactic features, our system uses the head verb itself and the surface case of the *bunsetsu* that includes the target morpheme.

For the morpheme "*shin*" in example (4), the head verb "*wataru* (get across)" and the surface case "*kara* (from)" are added as syntactic features.

caseframe feature Syntactic features cannot work if the head verb does not appear in the training data. To overcome this data sparseness problem, caseframe features are introduced.

Table 2: Case frame of "*haken* (dispatch)."

case	examples
ga (nominative)	Japan:23,party:13,country:12,government:7, company6,ward:6,corps:5,UN:4,US:4,Korea:4, team:4,. . . (ORGANIZATION,LOCATION)
wo (objective)	party:1249,him:1017,soldier:932,official:906, company6:214,instructor:823,expert:799, helper:694,staff:398,army:347,. . .
ni (locative)	Iraq:700,on-the-scene:576,abroad:335, home:172,Japan:171,Indirect Ocean:142, scene:141,China:125,. . . (LOCATION)

For example, although the head verb "*haken* (dispatch)" can be a clue for recognizing "*ICAO*" in (6) as "ORGANIZATION," syntactic features cannot work if "*haken* (dispatch)" did not appear in the training data.

(6) *ICAO-ha genchi-ni senmonka-wo* ***haken****-shita.*
scene to expert dispatched
(*ICAO* **dispatched** experts to the scene)

However, this clue can be utilized if there is knowledge that the "*ga* (nominative)" case of "*haken* (dispatch)" is often assigned by "ORGANIZATION."

Therefore, we construct case frames that include NE labels in advance. Case frames describe what kinds of cases each verb has and what kinds of nouns can fill a case slot. We construct them from about five hundred million sentences. We first recognize NEs appearing in the sentences by using a primitive NER system that uses only local features, and then construct the case frames from the NE-recognized sentences. To be more specific, if one tenth of the examples of a case are classified as a certain NE class, the corresponding label is attached to the case. Table 2 shows the constructed case frame of "*haken* (dispatch)." In the "*ga* (nominative)" case, the NE labels, "ORGANIZATION" and "LOCATION" are attached.

We then explain how to utilize these case frames. Our system first performs case analysis, and uses as caseframe features the NE labels attached in the case to which the target morpheme is assigned. For instance, by the case analyzer, the postpositional particle "*-ha*" in (6) is recognized as meaning nominative and "*ICAO*" is assigned to the "*ga* (nominative)" case of the case frame of "*haken* (dispatch)."Therefore, the caseframe features, "ORGANIZATION" and "LOCATION" are added to the features for the morpheme "*ICAO*."

4.5 SVM and Viterbi Search Based Chunking

To utilize cache features obtained from the previous parts of the same sentence, our system determines

Table 3: Experimental results (F-measure).

	CRL	IREX	WEB
baseline	88.63	85.47	68.98
+ cache	88.81 +0.18*	85.94 +0.47	69.67 +0.69*
+ coreference	88.68 +0.05	86.52 +1.05***	69.17 +0.19
+ syntactic	88.80 +0.17*	85.77 +0.30	70.25 +1.27**
+ caseframe	88.57 −0.06	85.51 +0.04	70.12 +1.14*
+ thesaurus	88.77 +0.14	86.36 +0.89*	68.63 −0.35
use all	**89.40** +0.77***	**87.72** +2.25***	**71.03** +2.05***

significant at the .1 level:*, .01 level:**, .001 level:***

NE tags clause by clause. The features extracted from two preceding morphemes and two succeeding morphemes are also used for chunking a target morpheme. Since SVM can solve only a two-class problem, we have to extend a binary classifier SVM to n-class classifier. Here, we employ the one versus rest method, in which we prepared n binary classifiers and each classifier is trained to distinguish a class from the rest of the classes.

To consider consistency of NE tags in a clause, our system uses Viterbi search with some constraints such as a "B-DATE" must be followed by "I-DATE" or "E-DATE." Since SVMs do not output probabilities, our system uses the SVM+sigmoid method (Platt et al., 2000). That is, a sigmoid function $s(x) = 1/(1+exp(-\beta x))$ is applied to map the output of SVM to a probability-like value. Our system determines NE tags by using these probability-like values. Our system is trained by TinySVM-0.09[4] with C = 0.1 and uses a fixed value β = 10. This process is almost the same as the process proposed by Isozaki and Kazawa and for details see (Isozaki and Kazawa, 2003).

5 Experiments

5.1 Data

For training, we use CRL NE data, which was prepared for IREX. CRL NE data has 18,677 NEs on 1,174 articles in Mainichi Newspaper.

For evaluation, we use three data: CRL NE data, IREX's formal test data called GENERAL and WEB NE data. When using CRL NE data for evaluation, we perform five-fold cross-validation. IREX test data has 1,510 NEs in 71 articles from Mainichi Newspaper. Although both CRL NE data and IREX test data use Mainichi Newspaper, these formats are not the same. For example, CRL NE data removes parenthesis expressions, but IREX test data does not. WEB NE data, which we annotated NEs on corpus collected from the Web, has 1,686 NEs in 354 articles. Although the domain of the web corpus differs from that of CRL NE data, the format of the web corpus is the same as CRL NE data format.

[4]http://chasen.org/ taku/software/TinySVM/

5.2 Experiments and Discussion

To confirm the effect of each feature, we conducted experiments on seven conditions as follows:

1. Use only basic features (baseline)
2. Add cache features to baseline
3. Add the coreference rule to baseline
4. Add parent features to baseline
5. Add caseframe features to baseline
6. Add thesaurus features to baseline
7. Use all structural information and thesaurus

Since (Masayuki and Matsumoto, 2003; Nakano and Hirai, 2004) reported the performance of NER system was improved by using a thesaurus, we also conducted experiment in which semantic classes obtained from a Japanese thesaurus "*Bunrui Goi Hyo*" (NLRI, 1993) were added to the SVM features. Table 3 shows the experimental results.

To judge the statistical significance of the differences between the performance of the baseline system and that of the others, we conducted a McNemar-like test. First, we extract the outputs that differ between the baseline method and the target method. Then, we count the number of the outputs that only baseline method is correct and that only target method is correct. Here, we assume that these outputs have the binomial distribution and apply binomial test. As significance level, we use .1 level, .01 level and .001 level. The results of the significance tests are also shown in Table 3.

When comparing the performance between data sets, we can say that the performance for WEB NE data is much worse than the others. This may be because the domain of the WEB corpus differs from that of CRL NE data.

As for the differences in the same data set, cache features and syntactic features improve the performance not dramatically but consistently and independently from the data set. The coreference rule also improves the performance for all data sets, but especially for IREX test data. This may be because IREX test data does not remove parenthesis expressions, and thus there are a many coreferential expressions in the data. Caseframe features improve the performance for WEB NE data, but do not contribute to the performance for CRL NE data and

Table 4: Comparison with previous work.

	CRL cross validation	IREX test data	Learning Method	Analysis Units	Features
(Isozaki and Kazawa, 2003)	86.77	85.10	SVM + Viterbi	morpheme	basic features
(Masayuki and Matsumoto, 2003)	87.21		SVM	character	+thesaurus
(Fukuoka, 2006)	87.71		Semi-Markov CRF	character	basic features
(Yamada, 2007)	88.33		SVM + Shift-Reduce	morpheme	+bunsetsu features
(Nakano and Hirai, 2004)	89.03		SVM	character	+bunsetsu features & thesaurus
Our system	**89.40**	**87.72**	SVM + Viterbi	morpheme	+structural information & thesaurus

IREX test data. This result shows that caseframe features are very generalized features and effective for data of different domain. On the other hand, thesaurus features improve the performance for CRL NE data and IREX test data, but worsen the performance for WEB NE data. The main cause for this may be overfitting to the domain of the training data.

By using all structural information, the performance is significantly improved for all data sets, and thus we can say that the structural information improves the performance of NER.

5.3 Comparison with Previous Work

Table 4 shows the comparison with previous work for CRL NE data and IREX test data. Our system outperforms all other systems, and thus we can confirm the effectiveness of our approach.

6 Conclusion

In this paper, we presented an approach that uses structural information for Japanese NER. We introduced four types of structural information to an SVM-based NER system: cache features, coreference relations, syntactic features and caseframe features, and conducted NER experiments on three data. As a consequence, the performance of NER was improved by using structural information and our approach achieved a higher F-measure than existing approaches.

References

Masayuki Asahara and Yuji Matsumoto, 2002. *IPADIC User Manual*. Nara Institute of Science and Technology, Japan.

Hai Leong Chieu and Hwee Tou Ng. 2002. Named entity recognition: A maximum entropy approach using global information. In *Proc. of COLING 2002*, pages 1–7.

Kenta Fukuoka. 2006. Named entity extraction with semi-markov conditional random fields (in Japanese). Master's thesis, Nara Institute of Science and Technology.

IREX Committee, editor. 1999. *Proc. of the IREX Workshop*.

Hideki Isozaki and Hideto Kazawa. 2003. Speeding up support vector machines for named entity recognition (in japanese). *Trans. of Information Processing Society of Japan*, 44(3):970–979.

Heng Ji and Ralph Grishman. 2005. Improving name tagging by reference resolution and relation detection. In *Proc. of ACL-2005*, pages 411–418.

Daisuke Kawahara and Sadao Kurohashi. 2002. Fertilization of Case Frame Dictionary for Robust Japanese Case Analysis. In *Proc. of COLING-2002*, pages 425–431.

Sadao Kurohashi and Daisuke Kawahara. 2005. Japanese morphological analysis system JUMAN version 5.1 manual.

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

R. Malouf. 2002. Markov models for language-independent named entity recognition. In *Proc. of CoNLL-2002*, pages 187–190.

Asahara Masayuki and Yuji Matsumoto. 2003. Japanese named entity extraction with redundant morphological analysis. In *Proc. of HLT-NAACL 2003*, pages 8–15.

Yuji Matsumoto, Akira Kitauchi, Tatsuo Yamashita, Yoshitaka Hirano, Hiroshi Matsuda, Kazuma Takaoka, and Masayuki Asahara. 2003. Morphological analysis System chasen 2.3.3 users manual.

Behrang Mohit and Rebecca Hwa. 2005. Syntax-based semi-supervised named entity tagging. In *Proc. of ACL Interactive Poster and Demonstration Sessoins*, pages 57–60.

MUC-6. 1995. *Proc. of the Sixth Message Understanding Conference*. Morgan Kaufmann Publishers, INC.

Keigo Nakano and Yuzo Hirai. 2004. Japanese named entity extraction with bunsetsu features (in Japanese). *Trans. of Information Processing Society of Japan*, 45(3):934–941.

The National Language Institute for Japanese Language, NLRI, editor. 1993. *Bunrui Goi Hyo (in Japanese)*. Shuuei Publishing.

John C. Platt, Nello Cristiani, and John ShaweTaylor. 2000. Lage margin DAGs for multiclas classification. In *Advances in Neural Information Processing System 12*.

Ryohei Sasano, Daisuke Kawahara, and Sadao Kurohashi. 2007. Improving coreference resolution using bridging reference resolution and automatically acquired synonyms. In *Proc. of DAARC-2007*.

Takehito Utsuro, Manabu Sassano, and Kiyotaka Uchimoto. 2002. Combing outputs of multiple named entity chunkers by stacking. In *Proc. of EMNLP-2002*.

Hiroyasu Yamada. 2007. Shift reduce chunking for Japanese named entity extraction (in Japanese). In *IPSJ SIG Notes NL-179-3*, pages 13–18.