Proceedings of HLT-NAACL 2003
Main Papers , pp. 8-15
Edmonton, May-June 2003

Japanese Named Entity Extraction with Redundant Morphological Analysis

Masayuki Asahara and Yuji Matsumoto
Graduate School of Information Science
Nara Institute of Science and Technology, Japan
{masayu-a,matsu}@is.aist-nara.ac.jp

Abstract

Named Entity (NE) extraction is an important subtask of document processing such as information extraction and question answering. A typical method used for NE extraction of Japanese texts is a cascade of morphological analysis, POS tagging and chunking. However, there are some cases where segmentation granularity contradicts the results of morphological analysis and the building units of NEs, so that extraction of some NEs are inherently impossible in this setting. To cope with the unit problem, we propose a character-based chunking method. Firstly, the input sentence is analyzed redundantly by a statistical morphological analyzer to produce multiple (*n*-best) answers. Then, each character is annotated with its character types and its possible POS tags of the top *n*-best answers. Finally, a support vector machine-based chunker picks up some portions of the input sentence as NEs. This method introduces richer information to the chunker than previous methods that base on a single morphological analysis result. We apply our method to IREX NE extraction task. The cross validation result of the F-measure being 87.2 shows the superiority and effectiveness of the method.

1 Introduction

Named Entity (NE) extraction aims at identifying proper nouns and numerical expressions in a text, such as persons, locations, organizations, dates, and so on. This is an important subtask of document processing like information extraction and question answering.

A common standard data set for Japanese NE extraction is provided by IREX workshop (IREX Committee, editor, 1999). Generally, Japanese NE extraction is done in the following steps: Firstly, a Japanese text is segmented into words and is annotated with POS tags by a morphological analyzer. Then, a chunker brings together the words into NE chunks based on contextual information. However, such a straightforward method cannot extract NEs whose segmentation boundary contradicts that of morphological analysis outputs. For example, a sentence "小泉純一郎首相が９月に訪朝" is segmented as "小泉/純一郎/首相/が/９月/に/訪朝" by a morphological analyzer. "小泉純一郎" ("Koizumi Jun'ichiro" – family and first names) as a person name and "９月" ("September") as a date will be extracted by combining word units. On the other hand, "朝" (abbreviation of North Korea) cannot be extracted as a name of location because it is contained by the word unit "訪朝" (visiting North Korea). Figure 1 illustrates the example with English translation.

Some previous works try to cope with the word unit problem: Uchimoto (Uchimoto et al., 2000) introduces transformation rules to modify the word units given by a morphological analyzer. Isozaki (Isozaki and Kazawa, 2002) controls the parameters of a statistical morphological analyzer so as to produce more fine-grained output. These method are used as a preprocessing of chunking.

By contrast, we propose more straightforward method in which we perform the chunking process based on character units. Each character receives annotations with character type and multiple POS information of the words found by a morphological analyzer. We make use of redundant outputs of the morphological analysis as the base features for the chunker to introduce more information-rich features. We use a support vector machine (SVM)-based chunker *yamcha* (Kudo and Matsumoto, 2001) for the chunking process. Our method achieves better score than all the systems reported previously for IREX NE extraction task.

Section 2 presents the IREX NE extraction task. Section 3 describes our method in detail. In section 4, we show the results of experiments, and finally we give conclusions in section 5.

2 IREX NE extraction task

The task of NE extraction in the IREX workshop is to recognize eight NE types as shown in Table 1 (IREX Committee, editor, 1999). In their definitions, "ARTIFACT" contains book titles, laws, brand names and so on. The task can be defined as a chunking problem to iden-

Example Sentence:

小泉	純一郎	首相	が	９月	に	訪朝
Koizumi	Jun'ichiro	Prime-Minister	particle	September	particle	visiting-North-Korea

Prime Minister Koisumi Jun'ichiro will visit North Korea in September.

Named Entities in the Sentence:

- 小泉純一郎/"Koizumi Jun'ichiro"/PERSON,
- ９月/"September"/DATE,
- 朝/"North Korea"/LOCATION

Figure 1: Example of word unit problem

	小	泉	首	相	は	日	朝	間	…
IOB1	I-PERSON	I-PERSON	O	O	O	I-LOCATION	B-LOCATION	O	
IOB2	B-PERSON	I-PERSON	O	O	O	B-LOCATION	B-LOCATION	O	
IOE1	I-PERSON	I-PERSON	O	O	O	E-LOCATION	I-LOCATION	O	
IOE2	I-PERSON	E-PERSON	O	O	O	E-LOCATION	E-LOCATION	O	
SE	B-PERSON	E-PERSON	O	O	O	S-LOCATION	S-LOCATION	O	

Prime Minister Koizumi does . . . between Japan and North Korea.

Figure 2: Examples of NE tag sets

Table 1: Examples of NEs in IREX

NE Type	Examples in English
ARTIFACT	Nobel Prize in Chemistry
DATE	May 5th
LOCATION	Republic of Korea
MONEY	2 million dollars
ORGANIZATION	Social Democratic Party
PERCENT	20 %, thirty percents
PERSON	Murayama Tomiichi
TIME	five in the morning

tify word sequences which compose NEs. The chunking problem is solved by annotation of chunk tags to tokens. Five chunk tag sets, IOB1, IOB2, IOE1, IOE2 (Ramshaw and Marcus, 1995) and SE (Uchimoto et al., 2000), are commonly used. In IOB1 and IOB2 models, three tags I, O and B are used, meaning inside, outside and beginning of a chunk. In IOB1, B is used only at the beginning of a chunk that immediately follows another chunk, while in IOB2, B is always used at the beginning of a chunk. IOE1 and IOE2 use E tag instead of B and are almost the same as IOB1 and IOB2 except that the end points of chunks are tagged with E. In SE model, S is tagged only to one-symbol chunks, and B, I and E denote exactly the beginning, intermediate and end points of a chunk. Generally, the words given by the single output of a morphological analyzer are used as the units for chunking. By contrast, we take characters as the units. We annotate a tag on each character.

Figure 2 shows examples of character-based NE annotations according to the five tag sets. "小泉"(PERSON), "日"(LOCATION) and "朝"(LOCATION) are NEs in the sentence and annotated as NEs. While the detailed explanation of the tags will be done later, note that an NE tag is a pair of an NE type and a chunk tag.

3 Method

In this section, we describe our method for Japanese NE extraction. The method is based on the following three steps:

1. A statistical morphological/POS analyzer is applied to the input sentence and produces POS tags of the *n*-best answers.
2. Each character in the sentences is annotated with the character type and multiple POS tag information according to the *n*-best answers.
3. Using annotated features, NEs are extracted by an SVM-based chunker.

Now, we illustrate each of these three steps in more detail.

3.1 Japanese Morphological Analysis

Our Japanese morphological/POS analysis is based on Markov model. Morphological/POS analysis can be de-

fined as the determination of POS tag sequence T once a segmentation into a word sequence W is given. The goal is to find the POS and word sequences T and W that maximize the following probability:

$$T = arg\max_{T} P(T|W).$$

Bayes' rule allows $P(T|W)$ to be decomposed as the product of tag and word probabilities.

$$arg\max_{T} P(T|W) = arg\max_{T} P(W|T)P(T).$$

We introduce approximations that the word probability is conditioned only on the tag of the word, and the tag probability is determined only by the immediately preceding tag. The probabilities are estimated from the frequencies in tagged corpora using Maximum Likelihood Estimation. Using these parameters, the most probable tag and word sequences are determined by the Viterbi algorithm.

In practice, we use log likelihood as cost. Maximizing probabilities means minimizing costs. In our method, redundant analysis output means the top *n*-best answers within a certain cost width. The *n*-best answers are picked up for each character in the order of the accumulated cost from the beginning of the sentence. Note that, if the difference between the costs of the best answer and *n*-th best answer exceeds a predefined cost width, we abandon the *n*-th best answer. The cost width is defined as the lowest probability in all events which occur in the training data.

3.2 Feature Extraction for Chunking

From the output of redundant analysis, each character receives a number of features. POS tag information is subcategorized so as to encode relative positions of characters within a word. For encoding the position we employ SE tag model. Then, a character is tagged with a pair of POS tag and the position tag within a word as one feature. For example, the character at the initial, intermediate and final positions of a common noun (Noun-General) are represented as "Noun-General-B", "Noun-General-I" and "Noun-General-E", respectively. The list of tags for positions in a word is illustrated in Table 2. Note that O tag is not necessary since every character is a part of a certain word.

Character types are also used for features. We define seven character types as listed in Table 3.

Figure 3 shows an example of the features used for chunking process.

Table 2: Tags for positions in a word

Tag	Description
S	one-character word
B	first character in a multi-character word
E	last character in a multi-character word
I	intermediate character in a multi-character word (only for words longer than 2 chars)

Table 3: Tags for character types

Tag	Description
ZSPACE	Space
ZDIGIT	Digit
ZLLET	Lowercase alphabetical letter
ZULET	Uppercase alphabetical letter
HIRAG	Hiragana
KATAK	Katakana
OTHER	Others (Kanji etc.)

3.3 Support Vector Machine-based Chunking

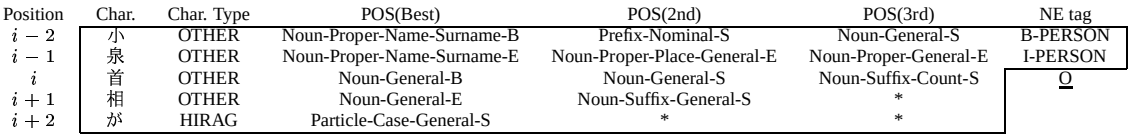
We used the chunker *yamcha* (Kudo and Matsumoto, 2001), which is based on support vector machines (Vapnik, 1998). Below we present support vector machine-based chunking briefly.

Suppose we have a set of training data for a binary class problem: $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_N, y_N)$, where $\mathbf{x}_i \in R^n$ is a feature vector of the *i*-th sample in the training data and $y_i \in \{+1, -1\}$ is the label of the sample. The goal is to find a decision function which accurately predicts y for an unseen $\mathbf{x}$. An support vector machine classifier gives the decision function $f(\mathbf{x}) = sign(g(\mathbf{x}))$ for an input vector $\mathbf{x}$ where

$$g(\mathbf{x}) = \sum_{\mathbf{z}_i \in SV} \alpha_i y_i K(\mathbf{x}, \mathbf{z}_i) + b.$$

$f(\mathbf{x}) = +1$ means that $\mathbf{x}$ is a positive member, $f(\mathbf{x}) = -1$ means that $\mathbf{x}$ is a negative member. The vectors $\mathbf{z}_i$ are called support vectors. Support vectors and other constants are determined by solving a quadratic programming problem. $K(\mathbf{x}, \mathbf{z})$ is a kernel function which maps vectors into a higher dimensional space. We use the polynomial kernel of degree 2 given by $K(\mathbf{x}, \mathbf{z}) = (1+\mathbf{x}\cdot\mathbf{z})^2$.

To facilitate chunking tasks by SVMs, we have to extend binary classifiers to *n*-class classifiers. There are two well-known methods used for the extension, "One-vs-Rest method" and "Pairwise method". In "One-vs-Rest method", we prepare n binary classifiers, one between a class and the rest of the classes. In "Pairwise method", we prepare ${}_nC_2$ binary classifiers between all pairs of classes.



Position	Char.	Char. Type	POS(Best)	POS(2nd)	POS(3rd)	NE tag
$i-2$	小	OTHER	Noun-Proper-Name-Surname-B	Prefix-Nominal-S	Noun-General-S	B-PERSON
$i-1$	泉	OTHER	Noun-Proper-Name-Surname-E	Noun-Proper-Place-General-E	Noun-Proper-General-E	I-PERSON
i	首	OTHER	Noun-General-B	Noun-General-S	Noun-Suffix-Count-S	O
$i+1$	相	OTHER	Noun-General-E	Noun-Suffix-General-S	*	
$i+2$	が	HIRAG	Particle-Case-General-S	*	*	

Figure 3: An example of features for chunking

Chunking is done deterministically either from the beginning or the end of sentence. Figure 3 illustrates a snapshot of chunking procedure. Two character contexts on both sides are referred to. Information of two preceding NE tags is also used since the chunker has already determined them and they are available. In the example, to infer the NE tag ("O") at the position i, the chunker uses the features appearing within the solid box.

3.4 The effect of n-best answer

The model copes with the problem of word segmentation by character-based chunking. Furthermore, we introduce n-best answers as features for chunking to capture the following behavior of the morphological analysis. The ambiguity of word segmentation occurs in compound words. When both longer and shorter unit words are included in the lexicon, the longer unit words are more likely to be output by the morphological analyzer. Then, the shorter units tend to be hidden behind the longer unit words. However, introducing the shorter unit words is more necessary to named entity extraction to generalize the model, because the shorter units are shared by many compound words. Figure 4 shows the example in which the shorter units are effective for NE extraction. In this example "日本" (Japan) is extracted as a location by second best answer, namely "Noun-Proper-Place-Country".

Unknown word problem is also solved by the n-best answers. Contextual information in Markov Model is lost at the position unknown word occurs. Then, preceding or succeeding words of an unknown word tend to be mistaken in POS tagging. However, correct POS tags occurring in n-best answer may help to extract named entity. Figure 5 shows such an example. In this example, the begining of the person name is captured by the best answer at the position 1 and the end of the person name is captured by the second best answer at the position 5.

4 Evaluation

4.1 Data

We use CRL NE data (IREX Committee, editor, 1999) for evaluation of our method. CRL NE data includes 1,174 newspaper articles and 19,262 NEs. We perform five-fold cross-validation on several settings to investigate the length of contextual feature, the size of redundant morphological analysis, feature selection and the degree of polynomial Kernel functions. For the chunk tag scheme we use IOB2 model since it gave the best result in a pilot study. F-Measure ($\beta = 1$) is used for evaluation.

4.2 The length of contextual feature

Firstly, we compare the extraction accuracies of the models by changing the length of contextual features and the direction of chunking. Table 4 shows the result in accuracy for each of NEs as well as the total accuracy of all NEs. For example, "L2R2" denotes the model that uses the features of two preceding and two succeeding characters. "For" and "Back" mean the chunking direction: "For" specifies the chunking direction from left to right, and "Back" specifies that from right to left.

Concerning NE types except for "TIME", "Back" direction gives better accuracy for all NE types than "For" direction. It is because suffixes are crucial feature for NE extraction. "For" direction gives better accuracy for "TIME", since "TIME" often contains prefixes such as "午前"(a.m.) and "午後"(p.m.).

"L2R2" gives the best accurary for most of NE types. For "ORGANIZATION", the model needs longer contextual length of features. The reason will be that the key prefixes and suffixes are longer in this NE type such as "株式会社"(company limited) and "研究所"(research institute).

4.3 The depth of redundant morphological analysis

Table 5 shows the results when we change the depth (the value *n* of the *n*-best answers) of redundant morphological analysis.

Redundant outputs of morphological analysis slightly improve the accuracy of NE extraction except for numeral expressions. The best answer seems enough to extract numeral experssions except for "MONEY". It is because numeral expressions do not cause much errors in morphological analysis. To extract "MONEY", the model needs more redundant output of morphological analysis. A typical occurs at "カナダドル" (Canadian dollars = MONEY) which is not including training data and is analyzed as "カナダ" (Canada = LOCATION). The similar error occurs at "香港ドル" (Hong Kong dollars) and so on.

4.4 Feature selection

We use *POS tags*, *characters*, *character types* and *NE tags* as features for chunking. To evaluate how they are

Position	Char.	POS(Best)	POS(2nd)	NE
1	日	Noun-General	Noun-Proper-Place-Country	LOCATION
2	本			
3	人		Noun-Suffix-General	

Figure 4: Effect of n-best answers (1)

Position	Char.	POS(Best)	POS(2nd)	NE
1	池	Noun-Proper-Name-Surname	Noun-General	PERSON
2	坊			
3	専	Unknown Word	*	
4	永	Noun-Proper-Name-Surname	Adjective	
5	家	Noun-General	Noun-Suffix-General	

Figure 5: Effect of n-best answers (2)

Table 4: The length of contextual feature and the extraction accuracy

	Pair Wise Method						One vs Rest Method					
Context Length	L1R1		L2R2		L3R3		L1R1		L2R2		L3R3	
Direction	For	Back	For	Back	For	Back	For	Back	For	Back	For	Back
ARTIFACT	29.74	46.36	42.17	**48.30**	43.90	46.36	29.79	45.59	39.84	**49.58**	42.35	47.82
DATE	84.98	90.33	91.16	**94.14**	92.47	93.72	85.15	90.22	91.21	**93.97**	92.42	93.41
LOCATION	80.16	86.17	84.07	**87.62**	85.75	87.18	80.22	86.62	84.31	**87.75**	86.06	87.61
MONEY	43.46	94.00	59.88	**95.82**	72.53	94.34	43.43	93.30	61.85	**93.85**	75.01	93.60
ORGANIZATION	66.06	74.73	72.63	78.79	75.55	**79.48**	65.69	74.80	72.74	78.33	75.95	**79.95**
PERCENT	67.66	**96.37**	83.77	96.31	85.26	94.14	69.12	95.96	85.66	**96.06**	88.56	94.16
PERSON	83.44	85.60	85.35	**87.31**	86.31	87.24	83.63	84.98	85.51	87.19	86.57	**87.65**
TIME	88.21	87.55	**89.82**	87.47	89.54	87.49	88.42	87.54	**90.38**	88.33	89.85	88.08
ALL	76.60	83.72	81.91	**86.19**	83.82	86.02	76.65	83.71	82.12	86.11	84.16	**86.33**

3-best answers of redundant morphological analysis, Feature(*POS*, *Character*, *Character Type* and *NE tag*), Polynomial kernel of degree 2.

effective we test four settings, that is, "using all features (ALL)", "excluding *characters* (– *Char.*)", "excluding *character types* (– *Char. Type*)" and "excluding *subcategory of POS tags* (– *POS* subcat.)". Table 6 shows the results for these settings.

"Excluding Characters" gives the worst accuracy, implying that *characters* are indispensable for NE extraction. "Excluding *POS subcat.*" results in worse accuracy. Some *subcategories of POS* include semantic information for proper nouns such that name, organization and location, and they are useful for NE extraction.

For numeral expressions, "excluding *Char Type*" gives better accuracy. The reason is that numbers in Kanji are not defined in our character type definition.

4.5 The degree of polynomial Kernel functions

We alter degrees of kernel functions and check how the combination of features affects the results. As shown in Table 7, degree 2 gives the best accuracy for most of NE types. The result shows that the combination of two features is effective for extract NE extraction. However, the tendency is not so significant in numeral expressions.

4.6 The effect of thesaurus

Table 8: The thesaurus and the extraction accuracy

	without thesaurus		with thesaurus	
Direction	For	Back	For	Back
ARTIFACT	41.12	**50.06**	43.28	49.15
DATE	91.19	94.18	91.78	**94.80**
LOCATION	84.67	87.61	85.78	**88.59**
MONEY	61.62	93.67	64.58	**95.34**
ORGANIZATION	73.70	79.27	75.69	**80.37**
PERCENT	86.23	96.02	86.64	**96.11**
PERSON	86.03	87.40	86.21	**87.73**
TIME	**90.54**	88.07	90.19	88.92
ALL	82.58	86.35	83.58	**87.12**

"L2R2" contextual feature, *2*-best answers of redundant morphological analysis, One vs Rest method with Features: *POS*, *Characters*, *Character Types* and *NE tags*.

In the experimentation above, we follow the features used in the preceding work (Yamada et al., 2002). Isozaki (Isozaki and Kazawa, 2002) introduces the thesaurus – NTT Goi Taikei (Ikehara et al., 1999) – to augment the

Table 5: The depth of redundant analysis and the extraction accuracy

Pair Wise Method								
Depth of morph. analysis	only best ans.		*2*-best ans.		*3*-best ans.		*4*-best ans.	
Direction	For	Back	For	Back	For	Back	For	Back
ARTIFACT	44.37	**49.76**	43.57	48.84	42.17	48.30	42.10	49.04
DATE	90.53	93.81	91.22	**94.23**	91.16	94.14	91.00	93.71
LOCATION	84.35	**87.67**	84.20	**87.67**	84.07	87.62	83.92	87.60
MONEY	59.45	93.89	60.36	94.28	59.88	95.82	60.94	**95.96**
ORGANIZATION	73.83	79.12	73.71	**79.34**	72.63	78.79	72.46	78.39
PERCENT	84.44	**97.20**	84.87	96.76	83.77	96.31	83.51	96.81
PERSON	86.23	87.32	85.65	87.13	85.35	87.31	85.22	**87.46**
TIME	**90.22**	88.22	89.45	87.72	89.32	87.47	89.86	87.77
ALL	82.37	86.25	82.31	**86.30**	81.91	86.19	81.74	86.08
One vs Rest Method								
Depth of morph. analysis	only best ans.		*2*-best ans.		*3*-best ans.		*4*-best ans.	
Direction	For	Back	For	Back	For	Back	For	Back
ARTIFACT	43.11	48.96	41.12	**50.06**	39.84	49.58	38.65	48.45
DATE	90.79	**94.18**	91.19	**94.18**	91.21	93.97	90.96	93.83
LOCATION	84.72	87.65	84.67	87.61	84.31	87.75	84.15	**87.77**
MONEY	63.46	93.79	61.62	93.67	61.85	93.85	62.13	**95.47**
ORGANIZATION	74.37	78.96	73.70	**79.27**	72.74	78.33	72.73	78.12
PERCENT	86.07	**97.09**	86.23	96.02	85.66	96.06	85.51	96.28
PERSON	85.92	**87.69**	86.03	87.40	85.51	87.19	85.41	87.16
TIME	**90.98**	89.04	90.54	88.07	90.38	88.33	89.90	88.32
ALL	82.72	**86.40**	82.58	86.35	82.12	86.11	81.95	86.07

"L2R2" contextual features, Feature(*POS*, *Character*, *Character Type* and *NE tag*), Polynomial kernel of degree 2.

feature set. Table 8 shows the result when the class names in the thesaurus is used as features. Note that we introduced the leaf node tag for each morpheme. The thesaurus information is effective for NEs except for "ARTIFACT" and "TIME". Since "ARTIFACT" includes many unseen expressions, even if we introduce the information of the thesaurus, we cannot improve this model. Concerning "TIME", the words and characters in this NE type are limited. The information of thesaurus may not be necessary for "TIME" expression extraction. In this paper, we did not encode the tree structure of the thesaurus. Introducing hierarchical relationships in the thesaurus is one of our future works.

4.7 Discussion

Table 9: The best model and the extraction accuracy

NE	F-measure
ARTIFACT	50.16
DATE	94.80
LOCATION	88.57
MONEY	95.47
ORGANIZATION	80.44
PERCENT	97.09
PERSON	87.81
TIME	90.98
ALL	87.21

While we must have a fixed feature set among all NE types in Pairwise method, it is possible to select different feature sets and models when applying One-vs-Rest method. The best combined model achieves F-measure 87.21 (Table 9). The model uses one-vs-rest method with the best model for each type shown in Table 4-8. Table 10 shows comparison with related works. Our method attains the best result in the previously reported systems.

Previous works report that POS information in preceding and succeeding two-word window is the most effective for Japanese NE extraction. Our current work disproves the widespread belief about the contextual feature. In our experiments, the preceding and succeeding two or three character window is the best effective.

Our method employs exactly same chunker with the work by Yamada et. al. (2002). To see the influence of boundary contradiction between morphological analysis and NEs, they experimented with an ideal setting in which morphological analysis provides the perfect results for the NE chunker. Their result shows F-measure 85.1 in the same data set as ours. Those results show that our method solves more than the word unit problem compared with their results.

Table 6: The feature set and the extraction accuracy

Pair Wise Method								
Feature set	All		− *Char.*		− *Char. Type*		− *POS* subcat.	
Direction	For	Back	For	Back	For	Back	For	Back
ARTIFACT	42.17	**48.30**	23.64	25.04	41.36	46.31	41.45	45.77
DATE	91.16	**94.14**	76.26	80.41	91.08	94.04	90.07	93.33
LOCATION	84.07	**87.62**	77.29	79.15	83.87	87.27	76.37	70.99
MONEY	59.88	**95.82**	47.09	87.48	58.44	95.81	57.84	90.91
ORGANIZATION	72.63	**78.79**	60.81	62.06	72.15	78.62	66.10	73.41
PERCENT	83.77	**96.31**	68.78	83.05	84.10	95.98	82.59	94.58
PERSON	85.35	**87.31**	81.46	83.05	84.59	86.29	73.55	78.42
TIME	**89.82**	87.47	83.33	81.56	89.53	87.57	89.68	86.26
全体	81.91	**86.19**	72.14	75.13	81.54	85.78	75.58	77.94
One vs Rest Method								
Feature set	All		− *Char.*		− *Char. Type*		− *POS* subcat.	
Direction	For	Back	For	Back	For	Back	For	Back
ARTIFACT	39.84	**49.58**	22.97	23.94	39.98	47.82	39.69	47.42
DATE	91.21	93.97	75.80	80.57	91.25	**94.09**	90.17	93.34
LOCATION	84.31	**87.75**	75.87	79.38	84.50	87.63	76.99	82.68
MONEY	61.35	93.85	45.19	85.19	60.33	**94.86**	59.62	89.89
ORGANIZATION	72.74	**78.33**	58.85	61.95	72.77	78.31	66.60	73.64
PERCENT	85.66	96.06	66.86	79.61	86.21	**96.09**	83.76	94.81
PERSON	85.51	**87.19**	80.43	82.33	84.87	86.59	73.92	79.07
TIME	**90.38**	88.33	80.44	77.31	90.36	88.27	88.96	86.59
全体	82.12	**86.11**	70.73	74.92	82.07	85.96	76.02	81.72

"L2R2" contextual features, *3*-best answers of redundant morphological analysis, Polynomial kernel of degree 2.

5 Conclusions

The proposed NE extraction method achieves F-measure 87.21 on CRL NE data. This is the best result in the previously reported systems. We made use of character level information with redundant outputs of a statistical morphological analyzer in an SVM-based chunker. It copes with the word unit problem in NE extraction. Furthermore, the method is robust for both errors of the morphological analyzer and occurences of unknown words, because character level prefixes and suffixes of NEs are clues for finding them. Fragments of possible words are used as features by the redundant morphological analysis. Though we tested this method only with Japanese, the method is applicable to any other languages that have word unit problem in NE extraction.

Acknowledgment

We thank Dr. Hiroyasu Yamada for his detailed discussion on the task of NE extraction. We also thank Mr. Taku Kudo for letting us use his chunking tools *yamcha*. This research was partially funded by JSPS Research Fellowships for Young Scientists.

References

Satoru Ikehara, Masahiro Miyazaki, Satoshi Shirai, Akio Yokoo, Hiromi Nakaiwa, Kentaro Ogura, Yoshifumi Ooyama, and Yoshihiki Hayashi. 1999. *Goi-Taikei – A Japanese Lexicon CDROM*. Iwanami Shoten, Tokyo.

IREX Committee, editor. 1999. Proceedings of the IREX workshop.

Hideki Isozaki and Hideto Kazawa. 2002. Efficient Support Vector Classifiers for Named Entity Recognition. In *Proceedings of COLING-2002*, pages 390–396.

Taku Kudo and Yuji Matsumoto. 2001. Chunking with Support Vector Machines. In *Proceedings of NAACL 2001*.

L. A. Ramshaw and M. P. Marcus. 1995. Text chunking using transformation-bases learning. In *Proceedings of the 3rd Workshop on Very Large Corpora*, pages 83–94.

Yoshikazu Takemoto, Toshikazu Fukushima, and Hiroshi Yamada. 2001. A Japanese Named Entity Extraction System Based on Building a Large-scale and High-quality Dictionary and Pattern-maching Rules. *IPSJ Journal*, 42(6):1580–1591.

Kiyotaka Uchimoto, Qing Ma, Masaki Murata, Hiromi Ozaku, Masao Utiyama, and Hitoshi Isahara. 2000. Named entity extraction based on a maximum entropy model and transformation rules (in Japanese). *Journal of Natural Language Processing*, 7(2):63–90.

Takehito Utsuro, Manabu Sassano, and Kiyotaka Uchimoto. 2002. Combining Outputs of Multiple Japanese

Table 7: The degree of polynomial kernel function and the extraction accuracy

	Pair Wise Method						One vs Rest Method					
Degree of p. ker.	1		2		3		1		2		3	
Direction	For	Back	For	Back	For	Back	For	Back	For	Back	For	Back
ARTIFACT	36.81	47.87	42.17	**48.30**	38.86	43.93	32.62	45.26	39.84	**49.58**	38.82	44.25
DATE	90.21	92.78	91.16	**94.14**	91.25	93.70	90.11	93.02	91.21	**93.97**	91.45	93.63
LOCATION	83.79	85.55	84.07	**87.62**	83.74	86.73	83.57	85.88	84.31	**87.75**	84.36	87.26
MONEY	55.22	95.42	59.88	**95.82**	59.63	93.88	55.36	**94.21**	61.85	93.85	63.55	93.91
ORGANIZATION	71.62	75.25	72.63	**78.79**	72.60	78.22	71.22	75.61	72.74	**78.33**	72.76	78.13
PERCENT	84.13	**97.04**	83.77	96.31	80.14	93.47	81.86	95.35	85.66	**96.06**	83.10	94.18
PERSON	83.25	85.15	85.35	**87.31**	85.13	86.48	82.71	85.05	85.51	**87.19**	85.54	86.90
TIME	89.09	88.42	89.82	87.47	**89.99**	85.80	85.26	88.06	**90.38**	88.33	89.86	87.25
ALL	80.66	84.10	81.91	**86.19**	81.66	85.36	80.36	84.23	82.12	**86.11**	82.17	85.65

"L2R2" contextual feature, *3*-best answers of redundant morphological analysis,
Features: *POS*, *Characters*, *Character Types* and *NE tags*.

Table 10: Comparison with related works

	CRL DATA	IREX GENERAL	Chunking Model	for the word unit problem
(Uchimoto et al., 2000)		80.17	ME	Transformation rules
(Yamada et al., 2002)	83.7		SVM	Examples in training data are segmented
(Takemoto et al., 2001)		83.86	Lexicon and Rules	Compound lexicon
(Utsuro et al., 2002)		84.07	Stacking (ME and Decision List)	
(Isozaki and Kazawa, 2002)	86.77	85.77	SVM with sigmoid curve	Parameter control for a statistical morphological analyzer
Our Method	87.21		SVM	Chunking by Character

Named Entity Chunkers by Stacking. In *Proceedings of EMNLP 2002*, pages 281–288.

V.N. Vapnik. 1998. *Statistical Learning Theory*. A Wiley-Interscience Publication.

Hiroyasu Yamada, Taku Kudoh, and Yuji Matsumoto. 2002. Japanese Named Entity Extraction Using Support Vector Machine (in Japanese). *IPSJ Journal*, 43(1):44–53.