Japanese Unknown Word Identification by Character-based Chunking

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

Abstract

We introduce a character-based chunking for unknown word identification in Japanese text. A major advantage of our method is an ability to detect low frequency unknown words of unrestricted character type patterns. The method is built upon SVM-based chunking, by use of character *n*-gram and surrounding context of *n*-best word segmentation candidates from statistical morphological analysis as features. It is applied to newspapers and patent texts, achieving 95% precision and 55-70% recall for newspapers and more than 85% precision for patent texts.

1 Introduction

Japanese and Chinese sentences are written without spaces between words. A word segmentation process is a prerequisite for natural language processing (NLP) of non-segmented language family. Statistical morphological analyzers are often used for word segmentation in Japanese NLP, which achieve over 96% precision. However, unknown word processing still remains an issue to be addressed in those morphological analyzers. Unknown word processing in non-segmented languages are more challenging, as it first needs to identify boundaries of unknown words in texts, prior to assignment of correspoinding part-of-speech.

Unknown word processing in morphological analysis of non-segmented language can follow one of either approaches: modular or embedded. In the modular approach, a separate off-line module is used to extract unknown words from text (Mori 1996; Ikeya 2000). They are checked and added to the lexicon of morphological analyzers. In the embedded approach, an on-line module which statistically induces the likelihood of a particular string being a word is embedded in a morphological analyzer (Nagata, 1999; Uchimoto et al., 2001). A modular approach is generally preferable in practice, since it allows developers to maintain a high quality lexicon which is crucial for good performance. Previous work of the modular approach was either unable to detect low frequency unknown words (Mori 1996) or limited to predefined character patterns for low frequency unknown words (Ikeya 2000).

We propose a general-purpose unknown word identification based on character-based chunking in order to address these shortcomings. A cascade model of a morphological analyzer (trained with Markov Model) and a chunker (trained with Support Vector Machines) is applied. The morphological analyzer produces *n*-best word segmentation candidates, from which candidate segmentation boundaries, character *n*-gram and surrounding contexts are extracted as features for each character. The chunker determines the boundaries of unknown words based on the features.

The rest of this paper is as follows. We describe our method in Section 2, and present experimental results on newspaper articles and patent text in Section 3. Related work is provided in Section 4, and a summary and future directions are given in Section 5.

2 Method

We describe our method for unknown word identification. The method is based on the following three steps:

1. A statistical morphological analyzer is applied to the input sentence and produces *n*-best segmentation candidates with their correspoinding part-of-speech (POS).

2. Features for each character in the sentence are annotated as the character type and multiple POS tag information according to the *n*-best word candidates.

3. Unknown words are identified by a support vector machine (SVM)-based chunker based on annotated features.

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

2.1 Japanese Morphological Analysis

Japanese morphological analysis is based on Markov model. The goal is to find the word and POS tag sequences W and T that maximize the following probability:

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

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

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

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. Redundant analysis outputs in our method mean the top *n*-best word candidates within a certain cost width. The *n*-best word candidates are picked up for each character in the ascending order of the accumulated cost from the beginning of the sentence. Note that, if the difference between the costs of the best candidate and *n*-th best candidate exceeds a predefined cost width, we abandon the *n*-th best candidate. The cost width is defined as the lowest probability in all events which occur in the training data. We use *ChaSen* [1] as the morphological analyzer. *ChaSen* induces the *n*-best segmentation within a user-defined width.

2.2 Feature for Chunking

There are two general indicators of unknown words in Japanese texts. First, they have highly ambiguous boundaries. Thus, a morphological analyzer, which is trained only with known words, often produces a confused segmentation and POS assignment for an unknown word. If we inspect the lattice built during the analysis, subgraphs around unknown words are often dense with many equally plausible paths. We intend to reflect this observation as a feature and do this by use of *n*-best candidates from the morphological analyzer. As shown Figure 1, each character (Char.) in an input sentence is annotated with a feature encoded as a pair of segmentation tag and POS tag. For example, the best POS of the character "死" is "GeneralNoun-B". This renders as the POS is a common noun (General Noun) and its segmentation makes the character be the first one in a multi-character token. The POS tagset is based on IPADIC (Asahara and Matsumoto, 2002) and the segmentation tag is summarized in Table 1. The *3*-best candidates from the morphological analyzer is used. The second indicator of Japanese unknown words is the character type. Unknown words occur around long *Katakana* sequences and alphabetical characters. We use character type (Char. Type) as feature, as shown in Figure 1. Seven character types ar defined: *Space, Digit, Lowercase alphabet, Uppercase alphabet, Hiragana, Katakana, Other (Kanji).* The character type is directly or indirectly used in most of previous work and appears an important feature to characterize unknown words in Japanese texts.

Table 1: 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)

2.3 Support Vector Machine-based Chunking

We use the chunker *YamCha* (Kudo and Matsumoto, 2001), which is based on SVMs (Vapnik, 1998). 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 a decision function $f(\mathbf{x}) = sign(g(\mathbf{x}))$ for an input vector $\mathbf{x}$ where

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

$K(\mathbf{x}, \mathbf{z})$ is a kernel function which maps vectors into a higher dimensional space. We use a polynomial kernel of degree 2 given by $K(\mathbf{x}, \mathbf{z}) = (1 + \mathbf{x} \cdot \mathbf{z})^2$.

SVMs are binary classifiers. We extend binary classifiers to an *n*-class classifier in order to compose chunking rules. Two methods are often used

[1] http://chasen.naist.jp/

Char. id	Char.	Char. Type	POS(Best)	POS(2nd)	POS(3rd)	unknown word tag
$i-2$	焼	Other	PrefixNoun-S	GeneralNoun-S	SuffixNoun-S	B
$i-1$	死	Other	GeneralNoun-B	GeneralNoun-S	SuffixVerbalNoun-S	I
i	体	Other	GeneralNoun-E	SuffixNoun-S	GeneralNoun-S	I
$i+1$	で	Hiragana	CaseParticle-S	Auxil.Verb-S	ConjunctiveParticle-S	
$i+2$	発	Other	VerbalNoun-B	*	*	

Figure 1: An example of features for chunking

for the extension, the "One vs. Rest method" and the "Pairwise method". In the "One vs. Rest methods", we prepare n binary classifiers between one class and the remain classes. Whereas in the "Pairwise method", we prepare ${}_nC_2$ binary classifiers between all pairs of classes. We use "Pairwise method" since it is efficient to train than the "One vs. Rest method".

Chunking is performed by deterministically annotating a tag on each character. Table 2 shows the unknown word tags for chunking, which are known as the IOB2 model (Ramshaw and Marcus, 1995).

Table 2: Tags for unknown word chunking

Tag	Description
B	first character in an unknown word
I	character in an unknown word (except B)
O	character in a known word

We perform chunking either from the beginning or from the end of the sentence. Figure 1 illustrates a snapshot of chunking procedure. Two character contexts on both sides are referred to. Information of two preceding unknown word tags is also used since the chunker has already determined them and they are available. In the example, the chunker uses the features appearing within the solid box to infer the unknown word tag ("I") at the position i.

We perform chunking either from the beginning of a sentence (forward direction) or from the end of a sentence (backward direction).

3 Experiments and Evaluation

3.1 Experiments for measuring Recall

Firstly, we evaluate recall of our method. We use *RWCP text corpus* (Real World Computing Partnership, 1998) as the gold standard and *IPADIC* (Version 2.6.3) (Asahara and Matsumoto, 2002) as the base lexicon. We set up two data sets based on the hit number of a web search engine which is shown in Appendix A. Table 3 shows the two data sets. Words with lower hit number than the threshold are regarded as unknown. We evaluate how many unknown words in the corpus are identified.

Table 3: Two data for recall evaluation

data	threshold	# of word in the lexicon (rate)	# of unknown word in the corpus (rate)
A	1,000	108,471 (44.2%)	9,814 (1.06%)
B	10,000	52,069 (21.2%)	33,201 (3.60%)

Table 4: Results – Recall by Newspaper

	Token		Type	
Setting	Rec.	Prec.	Rec.	Prec.
A/for	55.9%	75.3%	55.8%	69.5%
A/back	53.5%	73.4%	53.8%	68.0%
B/for	74.5%	82.2%	74.2%	75.8%
B/back	72.0%	80.9%	72.0%	74.3%

We perform five fold cross validation and average the five results. We carefully separate the data into the training and test data. The training and test data do not share any unknown word. We evaluate recall and precision on both token and type as follows:

$$\text{Recall} = \frac{\text{\# of words correctly identified}}{\text{\# of words in Gold Std. Data}}$$

$$\text{Precision} = \frac{\text{\# of words correctly identified}}{\text{\# of words identified}}$$

The experiment is conducted only for recall, since it is difficult to make fair judgment of precision in this setting. The accuracy is estimated by the word segmentation defined in the corpus. Nevertheless, there are ambiguities of word segmentation in the corpus. For example, while "京都大学 (Kyoto University)" is defined as one word in a corpus, "大阪/大学 (Osaka University)" is defined as two words in the same corpus. Our analyzer identifies "大阪大学" as one word based on generalization of "京都大学". Then, it will be judged as false in this experiment. We make fairer precision evaluation in the next section. However, since several related works make evaluation in this setting, we also present precision for reference.

Table 4 shows the result of recall evaluation. For

Table 5: Results – Recall of each POS

POS	# of token	Recall
GeneralNoun	9,009	67.1
PN (First Name)	3,938	86.8
PN (Organization)	3,800	63.8
PN (Last Name)	3,717	90.4
Verb	3,446	73.4
VerbalNoun	2,895	87.5
PN (Location)	1,911	79.3
PN (Other)	1,864	58.3
AdjectiveNoun	624	83.2
PN (Country)	449	88.4

"PN" stands for "Proper Noun" Data Set B, forward direction.
Shown POSs are higher than the rank 11th by the token sizes.

example, an experimental setting "A/for" stands for the data set A with a forward direction chunking, while "A/Back" stands for the data set A with a backward direction chunking. Since there is no significant difference between token and type, our method can detect both high and low frequency words in the corpus. Table 5 shows the recall of each POS in the setting data set B and forward direction chunking. While the recall is slightly poor for the words which include compounds such as organization names and case particle collocations, it achieves high scores for the words which include no compounds such as person names. There are typical errors of conjugational words such as verbs and adjectives which are caused by ambiguities between conjugational suffixes and auxiliary verbs.

3.2 Experiments for measuring Precision

Secondly, we evaluate precision of our method manually. We perform unknown word identification on newspaper articles and patent texts.

3.2.1 Unknown Word Identification in Newspapers

Firstly, we examine unknown word identification experiment in newspaper articles. We use articles of *Mainichi Shinbun* in January 1999 (116,863 sentences). Note that, the model is made by *RWCP text corpus*, which consists of articles of *Mainichi Shinbun* in 1994 (about 35,000 sentences).

We evaluate the models by the number of identified words and precisions. The number of identified words are counted in both token and type. To estimate the precision, 1,000 samples are selected at random with the surrounding context and are showed in KWIC (KeyWord in Context) format. One human judge checks the samples. When the selected string can be used as a word, we regard it as a correct answer. The precision is the percentage of correct answers over extracted candidates.

Concerning with compound words, we reject the words which do not match any constituent of the dependency structure of the largest compound word. Figure 2 illustrates judgment for compound words. In this example, we permit "海外留学 (overseas study)". However, we reject "短期海外 (short-term overseas)" since it does not compose any constitutent in the compound word.

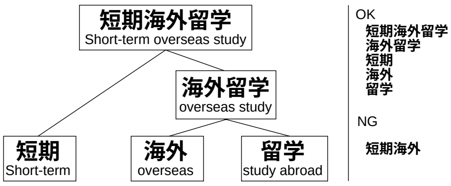

Figure 2: Judgement for compound words

We make two models: Model A is composed by data set A in Table 3 and model B is composed by data set B. We make two settings for the direction of chunking, forward (from BOS to EOS) and backward (from EOS to BOS).

Table 6 shows the precision for newspaper articles. It shows that our method achieves around 95% precision in both models. There is almost no difference in the several settings of the direction and the contextual feature.

Table 6: Results – Precision by Newspaper

	# of identified words		Precision
Setting	Token	Type	
A/For	58,708	19,880	94.6%
A/Back	59,029	19,658	94.0%
B/For	142,591	41,068	95.3%
B/Back	142,696	41,035	95.5%

3.2.2 Unknown Word Identification from Patent Texts

We also examine word identification experiment with patent texts. We use patent texts (25,084 sentences), which are OCR recognized. We evaluate models by the number of extracted words and precisions as in the preceding experiment. In this experiments, the extracted tokens may contain errors of the OCR reader. Thus, we define three categories for the judgment: Correct, Wrong and OCR Error. We use the rate of three categories for evaluation. Note that, our method does not categorize the outputs into Correct and OCR Error.

Table 7 shows the precision for patent texts. The backward direction of chunking gets better score than the forward one. Since suffixes are critical clues for the long word identification, the backward direction is effective for this task.

Table 7: Results – Precision by Patent Texts

	# of identified words		Accuracy		
Setting	Token	Type	Correct	Wrong	OCR Error
A/For	56,008	12,263	83.9%	15.4%	0.7%
A/Back	56,004	10,505	89.2%	10.0%	0.8%
B/For	97,296	16,526	85.6%	13.7%	0.7%
B/Back	98,826	15,895	87.0%	11.8%	1.2%

3.3 Word Segmentation Accuracy

Thirdly, we evaluate how our method improves word segmentation accuracy. In the preceding experiments, we do chunking with tags in Table 2. We can do word segmentation with unknown word processing by annotating B and I tags to known words and rejecting O tag. *RWCP text corpus* and *IPADIC* are used for the experiment. We define single occurrence words as unknown words in the corpus. 50% of the corpus (unknown words/all words= 8,274/461,137) is reserved for Markov Model estimation. 40% of the corpus (7,485/368,587) is used for chunking model estimation. 10% of the corpus (1,637/92,222) is used for evaluation. As the baseline model for comparison, we make simple Markov Model using 50% and 90% of the corpus. The results of Table 8 show that the unknown word processing improves word segmentation accuracy.

Table 8: Results – Word Segmentation

	Rec.	Prec.	F-Measure
Baseline (50%)	97.7%	96.5%	97.1
Baseline (90%)	97.8%	96.6%	97.2
Our Method	98.5%	98.1%	98.3

4 Related Work

Mori (1996) presents a statistical method based on *n*-gram model for unknown word identification. The method estimates how likely the input string is to be a word. The method cannot cover low frequency unknown words. Their method achieves 87.4% precision and 73.2% recall by token, 57.1% precision and 69.1% recall by type[2] on EDR corpus. Ikeya (2000) presents a method to find unknown word boundaries for strings composed by only *kanji* characters. The method also uses the likelihood based on *n*-gram model. Their method achieves 62.8 (F-Measure) for two *kanji* character words and 18.2 (F-Measure) for three *kanji* character words in newspapers domain.

Nagata (1999) classifies unknown word types based on the character type combination in an unknown word. They define likelihood for each combination. The context POS information is also used. The method achieves 42.0% recall and 66.4% precision on *EDR corpus*[3].

Uchimoto (2001) presents Maximum Entropy based methods. They extract all strings less than six characters as the word candidates. Then, they do morphological analysis based on words in lexicon and extracted strings. They use *Kyoto University text corpus* (Version 2) (Kurohashi and Nagao, 1997) as the text and *JUMAN dictionary* (Version 3.61) (Kurohashi and Nagao, 1999) as the base lexicon[4]. The recall of Uchimoto's method is 82.4% (1,138/1,381) with major POS estimation. We also perform nearly same experiment[5]. The result of our method is 48.8% precision and 36.2% recall (293/809) with the same training data (newspaper articles from Jan. 1 to Jan. 8, 1995) and test data (articles on Jan. 9, 1995). When we use all of the corpus excluding the test data, the result is 53.7% precision and 42.7% recall (345/809).

Uchimoto (2003) also adopts their method for *CSJ Corpus* (Maekawa et al. 2000)[6]. They present that the recall for *short words* on the corpus is 55.7% (928/1,667) (without POS information). We try to perform the same experiment. However, we cannot get same version of the corpus. Then, we use *CSJ Corpus – Monitor Edition (2002)*. It only contains *short word* by the definition of *the National Institute of Japanese Language*. 80 % of the corpus is used for training and the rest 20 % is for test. The result is 68.4% precision and 61.1% recall (810/1,326)[7].

[2] The evaluation of their method depends on the threshold of the confidence F_{min} in their definition. We refer the precision and recall at $F_{min} = 0.25$.

[3] They do not assume any base lexicon. Base lexicon size 45,027 words (composed by only the words in the corpus), training corpus size 100,000 sentences, test corpus size 100,000 sentences. Unknown words are defined by single occurrence words in the corpus.

[4] Base lexicon size 180,000 words, training corpus size 7,958 sentences, test corpus size 1,246 sentences OOV (out-of-vocabulary) rate 17.7%. Unknown words are defined by single occurrence words in the corpus.

[5] The difference is the definition of unknown words. Whereas they define unknown words by the possible word form frequency, we define ones by the stem form frequency.

[6] Training corpus size 744,244 tokens, test corpus size 63,037 tokens, OOV rate 1.66%.

[7] Training corpus size 678,649 tokens, 83,819 utterances, test corpus size 185,573 tokens, 20,955 utterances OOV rate 0.71%. Single occurence word by the stem form is defined as the unknown word.

Note, the version of the corpus and the definition of unknown word are different between Uchimoto's one and ours.

The difference of the result may come from the word unit definition. The word unit in *Kyoto University Corpus* is longer than the word unit in *RWCP text Corpus* and the *short word* of *CSJ Corpus*. Though our method is good at shorter unknown words, the method is poor at longer words including compounds.

For Chinese language, Chen (2002) introduces a method using statistical methods and human-aided rules. Their method achieves 89% precision and 68% recall on CKIP lexicon. Zhang (2002) shows a method with role (position) tagging on characters in sentences. Their tagging method is based on Markov model. The role tagging resembles our method in that it is a character-based tagging. Their method achieves 69.88% presicion and 91.65% recall for the Chinese person names recognition in *the People's Daily*. Goh (2003) also uses a character-based position tagging method by support vector machines. Their method achieves 63.8% precision and 58.4% recall for the Chinese general unknown words in *the People's Daily*. Our method is one variation of the Goh's method with redundant outputs of a morphological analysis.

5 Summary and Future Direction

We introduce a character-based chunking method for general unknown word identification in Japanese texts. Our method is based on cascading a morphological analyzer and a chunker. The method can identify unknown words regardless of their occurence frequencies.

Our research need to include POS guessing for the identified words. One would argue that, once the word boundaries are identified, the POS guessing method in European language can be applied (Brants 2000; Nakagawa 2001). In our preliminary experiments of POS guessing, both SVM and Maximum Entropy with contextual information achieves 93% with a coarse-grained POS set evaluation, but reaches only around 65% with a fine-grained POS set evaluation.

The poor result may be due to the "possibility-based POS tagset". The tagset is not necessarily friendly for statistical morphological analyzer development, but is widely used in Japanese corpus annotation. In the scheme, the fine-grained POS *Verbal Noun* in Japanese means that the word can be used both as *Verbal Noun* with verb and *General Noun* without verb. It is difficult to estimate the POS *Verbal Noun*, if the word appear in the context without verb. We are currently pursuing the research to better estimate fine-grained POS for the possibility-based POS tagset.

A Unknown Word Definition by Search Engine Hits

Unknown words mean out-of-vocabulary (hereafter OOV) words. The definition of the unknown words depends on the base lexicon. We investigate the relationship between the base lexicon size and the number of OOV words. We examine how the reduction of lexicon size affects the OOV rate in a corpus.

When we reduce the size of lexicon, we reject the words in increasing order of frequency in a corpus. Then, we use hits on a web search engine as substitutes for frequencies. We use *goo*[8] as the search engine and *IPADIC* (Asahara and Matsumoto, 2002) as the base lexicon. Figure 3 shows the distribution of the hit numbers. The x-axis is the number of hits in the search engine. The y-axis is the number of words which get the number of hits. The curve on the graph is distorted at 100 at which round-off begins.

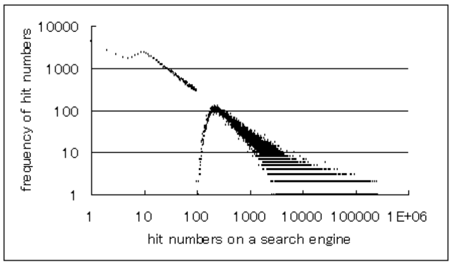

Figure 3: The hits of the words in *IPADIC*

We reduce the size of lexicon according to the number of hits. The rate of known words in a corpus is also reduced along the size of lexicon. Figure 4 shows the rate of known words in *RWCP text corpus* (Real World Computing Partnership, 1998). The x-axis is the threshold of the hit number. When the hit number of a word is less than the threshold, we regard the word as an unknown word. The left y-axis is the number of known words in the corpus. The right y-axis is the rate of known words in the corpus. Note, when the hit number of a word is 0, we do not remove the word from the corpus, because the word may be a stop word of the web search engine.

When we reject the words less than 1,000 hits from the lexicon, the lexicon size becomes 1/3 and

[8] http://www.goo.ne.jp/

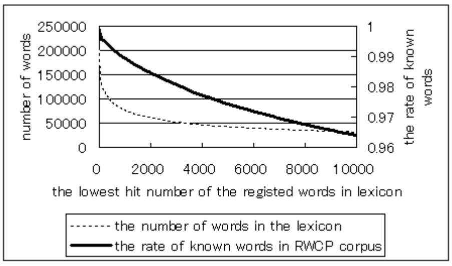

Figure 4: The rate of the known words

the OOV rate is 1%. When we reject the words less than 10,000 hits from the lexicon, the lexicon size becomes 1/6 and the OOV rate is 3.5%. We use these two data set, namely the lexicons and the definition of out-of-vocabulary words, for evaluation in section 3.1 and 3.2.

References

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

Masayuki Asahara and Yuji Matsumoto. 2003. Japanese Named Entity Extraction with Redundant Morphological Analysis. In *Proc. of HLT-NAACL 2003,* pages 8–15.

Thorsten Brants. 2000. TnT – A Statistical Part-of-Speech Tagger In *Proc. of ANLP-2000,*

Keh-Jiann Chen and Wei-Yun Ma. 2002. Unknown Word Extraction for Chinese Documents. In *Proc. of COLING-2002,* pages 169–175.

Chooi-Ling Goh, Masayuki Asahara and Yuji Matsumoto. 2003. Chinese Unknown Word Identification Using Position Tagging and Chunking. In *Proc. of ACL-2003 Interactive Poster/Demo Sessions, Companion volume,* pages 197–200.

Masanori Ikeya and Hiroyuki Shinnou. 2000. Extraction of Unknown Words by the Probability to Accept the Kanji Character Sequence as One Word (in Japanese). In *IPSJ SIG Notes NL-135,* pages 49–54.

Taku Kudo and Yuji Matsumoto. 2001. Chunking with Support Vector Machines. In *Proc. of NAACL 2001,* pages 192–199.

Sadao Kurohashi and Makoto Nagao. 1997. Building a Japanese Parsed Corpus while Improving the Parsing System. In *Proc. of NLPRS-97,* pages 451–456.

Sadao Kurohashi and Makoto Nagao. 1999. *Japanese Morphological Analysis System JUMAN Version 3.61.* Department of Informatics, Kyoto University, Japan.

Kikuo Maekawa, Hanae Koiso, Sasaoki Furui and Hiroshi Isahara. 2000. Spontaneous Speech Corpus of Japanese. In *Proc. of LREC-2000,* pages 947–952.

Shinsuke Mori and Makoto Nagao. 1996. Word Extraction from Corpora and Its Part-of-Speech Estimation Using Distributional Analysis. In *Proc. of COLING-96,* pages 1119–1122.

Masaaki Nagata. 1999. A Part of Speech Estimation Method for Japanese Unknown Words using a Statistical Model of Morphology and Context. In *Proc. of ACL-99,* pages 277–284.

Tetsuji Nakagawa, Taku Kudoh and Yuji Matsumoto. 2001 Unknown Word Guessing and Part-of-Speech Tagging Using Support Vector Machines. In *Proc. of NLPRS-2001,* pages 325–331.

Lance Ramshaw and Mitchell Marcus. 1995. Text Chunking using Transformation-based Learning. In *Proc. of the 3rd Workshop on Very Large Corpora,* pages 83–94.

Real World Computing Partnership. 1998. *RWC Text Database*.

Kiyotaka Uchimoto, Satoshi Sekine and Hitoshi Isahara. 2001. The Unknown Word Problem: a Morphological Analysis of Japanese Using Maximum Entropy Aided by a Dictionary. In *Proc. of EMNLP-2001,* pages 91–99.

Kiyotaka Uchimoto, Chikashi Nobata, Atsushi Yamada, Satoshi Sekine and Hiroshi Isahara. 2002. Morphological Analysis of the Spontaneous Speech Corpus. In *Proc. of COLING-2002,* pages 1298–1302.

Kiyotaka Uchimoto, Chikashi Nobata, Atsushi Yamada, Satoshi Sekine and Hiroshi Isahara. 2003. Morphological Analysis of a Large Spontaneous Speech Corpus in Japanese. In *Proc. of ACL-2003,* pages 479–488.

Vladimir Naumovich Vapnik. 1998. *Statistical Learning Theory.* A Wiley-Interscience Publication.

Kevin Zhang, Qun Liu, Hao Zhang and Xue-Qi Cheng. 2002. Automatic Recognition of Chinese Unknown Words Based on Roles Tagging. In *Proc. of 1st SIGHAN Workshop on Chinese Language Processing,* pages 71–77.