NAIST-IS-DT0161001

Doctor's Thesis

Corpus-based Japanese morphological analysis

Masayuki Asahara

December 17, 2003

Department of Information Processing
Graduate School of Information Science
Nara Institute of Science and Technology

Doctor's Thesis
submitted to Graduate School of Information Science,
Nara Institute of Science and Technology
in partial fulfillment of the requirements for the degree of
DOCTOR of ENGINEERING

Masayuki Asahara

Thesis committee: Yuji Matsumoto, Professor
Shunsuke Uemura, Professor
Kiyohiro Shikano, Professor
Kentaro Inui, Associate Professor

Corpus-based Japanese morphological analysis*

Masayuki Asahara

Abstract

The goal of this study is to improve corpus-based *Japanese morphological analysis* which is composed by word segmentation and part-of-speech (below POS) tagging. We divide the problem of Japanese morphological analysis into three subproblems: models for known word, models for unknown word and corpus maintenance schema. Firstly, we discuss Markov model-based approaches for known word processing. We point phenomena which are difficult to be analyzed by a simple Markov model. Special transactions are necessary for these phenomena. Therefore, we introduce three extensions for Markov model: lexicalized POS, position-wise grouping and selective trigram. Secondly, we discuss unknown word processing. We newly propose an offline model for unknown word based on a pattern recognition approach. Unknown words are extracted from the text by chunking in advance. Next, the POSs for the extracted words are estimated by a word sense disambiguation-like approach. Thirdly, we discuss maintenance schema for word segmented and POS tagged corpus. The corpus maintenance is a crucial issue for corpus-based models. We propose a relational database usage to keep consistency in the corpora. The relational database enables us synchronous transaction between the lexicon and the corpora. Therefore, the risk of discrepancy in the corpus is reduced by the proposed method.

As side issues, we discuss Japanese named entity extraction and filler filtering. Japanese named entity extraction is an application in information extraction. We propose two extensions for the application. One is a character-based chunking method which solves a word boundary discrepancy problem. The other is use of point-wise *n*-best answers of Japanese morphological analyzer which makes the model robust. The proposed method achieves the best accuracy in the preceding works. Filler filtering is a preprocessing for Japanese morphological analysis. Many fillers and disfluencies appear in transcriptions of spoken language. These phenomena are factors of the errors in Japanese morphological analysis. We introduce a pattern recognition method for filler and disfluency filtering from the transcription.

Keywords:

Japanese morphological analysis, corpus-based approach, Japanese named entity extraction, unknown word processing, linguistic databases

*Doctor's Thesis, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-DT0161001, December 17, 2003.

Acknowledgements

English is the Lingua Franca of Computer Science
– Clyde P. Kruskal

感謝の気持は日本語で。

まず第一に指導教官である松本教授に感謝致します。「良い研究者を育てるには良い研究者を見せることだ」を実践していただき、多くの研究者と一緒に仕事をする機会を私に与えていただきました。５年９ヶ月の在学期間の中、３分の１は海外で研究させていただきました。また、奈良にいる間も、自ら真摯に研究に取り組む姿勢を見させていただき、そこから得るものは大変多かったと考えております。ありがとうございました。鹿野教授には、修士研究以来数々のコメントを頂きました。本論文中の５章 “Filler filtering” の話は鹿野教授の修士論文に対するコメントから産まれたものです。このアイデアに基づき、３，４章で示されているような日本語の情報抽出における問題点の解消へとつながりました。ありがとうございました。植村教授は、お忙しい中、丁寧に論文を読んでいただき、多くの修正点の指摘と、示唆に富むコメントを頂きました。ありがとうございました。

修士在学中、当時本学の助教授であった伝さん（現千葉大助教授）に大変お世話になりました。師弟の関係にありながら、年の近い友人のように接していただき、また異動後も私の研究に多くのコメント（要望？）を提供していただきました。乾さんは、情熱のある指導をしていただいた一方、研究に対して的確なコメントを提供していただきました。常に物事の本質を考える姿勢に、多くのことを学ばせていただきました。ありがとうございました。

宇津呂さん、宮田さん、Miyamoto さん、新保さんには、多くの手厳しいツッコミをいただきました。特に後者２人には、逃げ道をコトゴトく封じられたので、常に緊張感を持って研究に邁進することができました。ありがとうございました。研究生活の上で、秘書の登さん、大川さんに大変お世話になりました。ありがとうございました。また、在学期間には多くの有能な学生に囲まれたことも私に取って大変刺激になりました。ひとりひとりの名前はあげませんが、先輩、同輩、後輩含めて多種多様な才能を持った方々と一緒に仕事ができて、大変うれしく思っております。

最後に、ここまで自由勝手にさせてくれた両親に感謝の意を表します。

平成１５年１２月吉日
浅原 正幸

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Just play. Have fun. Enjoy the game.
– **Michael Jordan**(1963-)

1.1 Motivation

Sentences of Japanese are written without word boundaries. There is no agreed standard for word boundary in such languages. The definition of word boundary varies from the applications. Still, the process of word boundary determination is widely used as preprocessing for most of natural language processing applications: information retrieval, information extraction, parsing, machine translation and so on. Truly, word boundary analysis is an essential technique for these languages.

An ancient word boundary analysis is based on only dictionary looking up. If there are several ambiguous word boundary candidates, the boundary candidates for the longest word are selected. The technique is called *the longest match method*. For example, the longest match method will choose **(i)** in the following example: [1]

Input: ここではきものをぬいでください

(i) ここ／で／はきもの／を／ぬい／で／ください
please put off your boots here

(ii) ここ／で／は／きもの／を／ぬい／で／ください
please take off your clothes here

Such simple method like the longest match cannot disambiguate essential word boundary ambiguity in Japanese. Most of word boundary analyses do POS (Part of Speech) tagging for the word candidates at the same time. The POS information in the context is used as clue for both the word boundary analysis and the POS tagging. Moreover, Japanese language has conjugation phenomena. Then, such languages require conjugation processing for word boundary analysis and POS tagging. *Morphological analysis* means such an integrated processing.

The POS information is helpful for word boundary disambiguation. For example, consider the following sentence [2] [3]:

Input: 明日もしよう

[1] These examples are based on IPA POS tagset and its word unit definition.
[2] '∗' indicates an unacceptable output.
[3] [ ] means the POS of the word. The POS tag name is simplified translation for intuitive understanding.

(i) 明日 [Adv.] ／も [Particle] ／しよ [Verb] ／う [Auxil. Verb]
I will do it again tomorrow

(ii) ∗明日 [Adv.] ／もし [Adv.] ／よう [Nominal Suffix]
∗ *Tomorrow, if, manner*

Whereas the longest match will choose the wrong word boundaries in **(ii)**, a simple POS based rule – such as "Verb is likely to precede auxiliary verb adjacently" – will disambiguate and choose the correct boundaries as **(i)**. A Japanese morphological analyzer *JUMAN* [66] contains such hand-coded rules. The rules are annotated with costs and constrained by POSs for the preceding words.

Nowadays, several large scale word segmented and POS tagged corpora are available [65] [61] [36]. Then, most of practical morphological analyzers use statistical and/or machine learning method based on the POS tagged corpora. Still, there are some specific problems in Japanese. We will propose several extensions to the stasitical model to adopt for the specific phenomena.

One of related tasks for Japanese morphological analysis is named entity extraction which is a task to idenfity named entities and numeral expressions in texts. The method is used for information extraction and/or question answering systems. On the one hand, European languages are written with word boundaries. Then, the named entity extraction is defined as a chunking task. The example for English is as follows: [4]

Input: Wolff, currently a journalist in Argentina, played with Del Bosque in the final years of the seventies in Real Madrid.

→ ⟨PERSON⟩Wolff⟨/PERSON⟩ , currently a journalist in ⟨LOCATION⟩Argentina⟨/LOCATION⟩ , played with ⟨PERSON⟩Del Bosque⟨/PERSON⟩ in the final years of the seventies in ⟨ORGANIZATION⟩Real Madrid⟨/ORGANIZATION⟩ .

On the other hand, Japanese named entity extraction cannot be defined as a simple chunking task without word segmentation process. Most of methods are based on the output of Japanese morphological analyzer. The example of Japanese named entity extraction is as follows:

Input: 村山富市首相は年頭にあたり首相官邸で内閣記者会と二十八日会見し

→ 村山／富市／首相／は／年頭／にあたり／首相／官邸／で／内閣／記者／会／と／二／十／八／日／会見／し

→ ⟨PERSON⟩ 村山／富市 ⟨/PERSON⟩ 首相／は ⟨DATE⟩ 年頭 ⟨/DATE⟩ にあたり ⟨LOCATION⟩ 首相／官邸 ⟨/LOCATION⟩ で ⟨ORGANIZATION⟩ 内閣／記者／会 ⟨/ORGANIZATION⟩ と ⟨DATE⟩ 二／十／八／日 ⟨/DATE⟩ 会見／し
Prime minister Murayama Tomiichi had meeting with the press club at the prime minister's official residence for New Year annual speech on 28th.

However, the discrepancy may be appeared between the named entity boundary and the word boundary generated by a Japanese morphological analyzer. In the following example, "朝" (the abbreviation of North Korea) is embeded in the word "訪朝" (visit North Korea). Then, the word boundary problem should be solved in the named entity extraction task simultaneously.

Input: 小泉首相が９月に訪朝

→ 小泉／首相／が／９月／に／ 訪朝

→ ⟨PERSON⟩ 小泉 ⟨/PERSON⟩ 首相／が ⟨DATE⟩ ９月 ⟨/DATE⟩ に／ 訪 ⟨LOCATION⟩ 朝 ⟨/LOCATION⟩
Prime minister Koizumi will visit North Korea on September.

[4] Even the language is written with the word boundaries, a tokenizer is nesesary for special symbols such as punctuation marks.

Now, we would like to come back to Japanese morphological analysis. One of crucial problem in Japanese morphological analysis is the occurrence of unknown words. If one of word candidates in the input sentence doesn't appear in the POS tagged lexicon, the morphological analysis becomes difficult. For example, consider the following sentence:

Input: 海外でケータイを使いたい

→ 海外 [Noun] ／で [Particle] ／ケータイ [Unknown] ／を [Particle] ／使い [Verb] ／たい [Auxil. Verb]
I want to use the cellular phone in foreign countries

"ケータイ" (Cellular phone) is an unknown word – out of vocabulary word. Three character types are used in Japanese: *hiragana* ひらがな, *katakana* カタカナ, and *kanzi* 漢字. One of simple method to identify unknown words is based on character types. The point, at which the character type changes, tends to be the word boundary. Still, the POS information cannot be used to estimate the boundaries and POSs of preceding and succeedng words. The error may be appeared around the unknown words.

If an unknown word is embeded in known words, the problem becomes worse. For example, the "マイニング" (mining) is embedded in the known words "タマ" (Cat's name) and "イニング" (inning) in the following sentence:

Input: データマイニングの研究が盛ん

(i) ∗ デー [Noun] ／タマ [Noun] ／イニング [Numeral Suffix] ／の [Particle] ／研究 [Noun] ／が [Particle] ／盛ん [Adjective Noun]
∗ *the study on "day, Tama(Cat's Name), inning" is active*

(ii) データ [Noun] ／マイニング [Unknown] ／の [Particle] ／研究 [Noun] ／が [Particle] ／盛ん [Adjective Noun]
the study on data mining is active

The unknown word processing is necessary to analyze the example sentence correctly.

Morphological analyzer for spoken languages is in demand for developing basic data for text-to-speech and/or speech-to-text system. Still, the accuracy of the morphological analyzer for spoken language is not satisfiable. Matsumoto et al. [85] points out four problems in morphological analysis of spoken language – "Filler", "Disfluency", "Ungrammaticality" and "Peculiar representation in spoken language". They proposed a method based on a mixed statistical models of spoken and written language.

UniDic project [83] has been started for compiling an electronic dictionary usable for morphological analysis of both written and spoken languages. *UniDic* lexicon will cover "Peculiar representation in spoken language" such as contractions. Furthermore, the statistical model of *UniDic* is estimated from the "Corpus of Spontaneous Japanese" [38]. *UniDic* is expected to cope with two problems namely "Ungrammaticality" and "Peculiar representation in spoken language".

Nevertheless, the first two problems – "Filler" and "Disfluency" – cannot be solved by these methods. Statistical morphological analyzers based on Markov model make errors at the positions where fillers or disfluencies occur, since the contextual information is broken at such positions. Thus, automatic filler/disfluency identification methods are in demand. We focus on the problem of automatic filler/disfluency identification.

Most of morphological analyzers are based on corpus-based approaches. The accuracy of a morphological analyzer depends on not only the model but also the corpus. The consistency in the corpus is a requisite factor. The corpus should be maintained both POSs and word boundaries. Nevertheless, maintainining word boundaries is cumbersome. For example, the following table is an example of corpus:

大阪大学　名詞-固有名詞-組織
で　助詞-格助詞-一般
３　名詞-数
４　名詞-数
教授　名詞-一般
が　助詞-格助詞-一般
定年　名詞-一般
退職　名詞-サ変接続

34 professors in Osaka Univ. will retire.

Now, suppose that for some reason, we decide to change the word boundary of “大阪大学” into “大阪/大学” as follows:

大阪　名詞-固有名詞-地域-一般
大学　名詞-一般
で　助詞-格助詞-一般
３　名詞-数
４　名詞-数
教授　名詞-一般
が　助詞-格助詞-一般
定年　名詞-一般
退職　名詞-サ変接続

The word number in the sentence changes. Then, the number of rows also changes. Still, we should keep the word order. We also mention about the other occurences of “大阪大学”. If we change the word boundary of “大阪大学”, we should change all occurences of “大阪大学” in the corpus. Moreover, we should change the boundaries of all words with suffix “大学” because of the consistency in the corpus.

1.2 Problem setting

This thesis addresses five problems. First problem is about known word processing in Japanese morphological analysis. We split the problem of Japanese morphological analysis into known word processing and unknown word processing. We assume that all word candidates in the input sentence are registered in the POS tagged lexicon. We concentrate disambiguation of possible word boundaries and POS candidates. We mention about Japanese specific language phenomena which cause analyzer errors and propose several solutions for them. Second problem is Japanese named entity extraction. The task can be defined as chunking problem based on Japanese morphological analyzer outputs. We focus on the word boundary discrepancy problem in the task. Third problem is unknown word processing in Japanese morphological analysis. We present a method to identify unknown words, namely out-of-vocabulary words, in the texts. We also present POS guessing method for the identified words. Fourth problem is morphological analysis of spoken language. We focus on fillers and disfluencies in transcriptions of spoken language. Fifth problem is corpus maintenance schema for word segmented and POS tagged corpus. We propose a maintenance schema to keep consistency in the corpora.

1.3 Approach

Now, we present the strategy for each problem briefly.

1.3.1 Markov model for known words

Markov model [55][56] is utilized for Japanese morphological analysis. The method is the same as the well-known use of Markov model for POS tagging problem [45], except that the word boundaries

are also determined at the same time. We propose three sorts of extension for Markov model-based Japanese morphological analysis: *lexicalized POS*, *position-wise grouping*, and *selective trigrams*.

In English, preposition is one of important words in POS tagging. Similarly, postpositions and auxiliary verbs in Japanese are important words for morphological analysis. Moreover, in the case of Japanese, these function words are written in Hiragana orthography. Word segmentation of these words is one of the hurdles to analyze. To focus on these words, we use *lexicalized POS*, which means to assign one individual POS to one word. This technique is widely used for POS tagging. In our methods, this technique is used with following *position-wise grouping*.

We utilize a slightly modified version of the IPA POS tag set [61] with 66 distinct POS tags. The real tag set is even larger since some words are treated as distinct POS tags in the statistical model. The size of the tag set is too large to build trigram rules and even bigram rules which take all the tags as distinct. A usual technique for coping with such fine-grained tags is to reduce the size of the tag set by grouping the set of tags into smaller equivalence classes [20]. We introduce the concept of *position-wise grouping* where the tag set is partitioned into different equivalence classes at each position in the conditional probabilities in the Markov model. This feature is especially useful for Japanese morphological analysis since Japanese is a conjugated language, where conjugation forms have a great effect on their succeeding morphemes, but have little to do with their preceding morphemes. Moreover, in colloquial language, a number of contracted expressions are common, where two or more morphemes are contracted into a single word. Contracted words behave as though they belong to different POS by connecting to the preceding word or to the succeeding word. Position-wise grouping enables us to group such words differently according to the positions in which they appear.

Data sparseness is always a serious problem when we deal with a large tag set. It is unrealistic to adopt a simple POS trigram model to large tag set. Nevertheless, some language phenomena require trigram context rules to analyze. For this purpose, we base our model on a bigram model and augment it with *selective trigrams*. By selective trigram, we mean that only specific contexts are conditioned by a trigram model and are mixed with the ordinary bigram model. We also incorporate some smoothing techniques for coping with the data sparseness problem.

1.3.2 Character-based chunking for named entity extraction

A named entity tagged texts in Japanese [31] is available in public. Then, we introduce corpus-based approach as preceding works. We incorporate support vector machine-based chunking [72] for Japanese named entity extraction as Yamada's work [29].

A typical method used for Japanese named entity extraction is a cascade of morphological analysis and chunking. However, there are some cases where segmentation granularity contradicts the results of morphological analysis and the building units of named entities, so that extraction of some named entities 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 position-wize *n*-best answers. Then, each character is annotated with its character type 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 named entities. This method introduces richer information to the chunker than preceding works that based on a single morphological analysis result.

1.3.3 Pattern recognition method for unknown word processing

There are mainly two strategies for the unknown word identification in preceding works. One strategy is that frequent strings in text are identified as unknown words based on statistical methods with smoothing technique [67]. However, the method cannot identify unknown words with low frequency. The other strategy is a character type pattern-based method which enables to find out unknown words with low frequency. Nevertheless, it can retrieve only words which match the defined character type patterns [51].

We propose an alternative word identification method which can identify infrequent words in new character type patterns. The method is based on a pattern recognition approach. We define unknown word identification as a chunking task based on the redundant outputs of Japanese morphological analysis, which is proposed in named entity extraction. The procedure identifies the ambiguous parts in the morphological analysis which are presumably caused by occurrence of unknown words.

POS guessing is necessary for the extracted words. We incorporate word sense disambiguation method with support vector machines for POS guessing. We introduce the redundant outputs of Japanese morphological analysis as contextual features.

1.3.4 Filler and disfluency filtering for spoken language processing

Matsumoto [85] presented four problems in morphological analysis of spoken language – "Filler", "Disfluency", "Ungrammaticality" and "Peculiar representation in spoken language". We focus attention on fillers and disfluencies.

The occurence of filler and/or disfluency interrupts the contextual feature for the Markov model. Then, we propose filler and disfluency filtering as a preprocessing. The method for filler and disfluency filtering is same as the method in unknown word identification. We define filler and disfluency identification as a chunking problem.

1.3.5 Stand-off annotation for corpus maintenance schema

Keeping consintency in the word segmented and POS tagged corpora is one of crucial problem for Japanese morphological analysis. Especially, the word boundary definition should be consistent in the corpora. We present a maintenance schema which can keep the consistency of the word bounary. The schema is based on stand-off annotation, in which the surface form and the tag information are kept in separate. The tag information is defined with the pointers for the surface form table. We also introduce links between the tag information and the POS tagged lexicon. The links enable us to keep consistency of the word boundary definitions through corpus modification.

1.4 Organization of the thesis

This thesis is organized as follows. In chapter 2, we discuss the known word model for Japanese morphological analysis. We describe the proposed extensions for Markov model-based Japanese morphological analysis. In chapter 3, we discuss Japanese named entity extraction. We present a character-based chunking method for Japanese named entity extraction. In chapter 4, we discuss unknown word processing. We present how the pattern recognition method is introduced for unknown word identification problem. In chapter 5, we discuss filler and disfluency filtering. In chapter 6, we discuss maintenance schema for word segmented corpus. In chapter 7, we conclude this thesis with some remarks.

Chapter 2, 4 and 5 are directly related in terms of the model of morphological analysis. Chapter 3, 4 and 5 are related in terms of the pattern recognition approach. Figure 1.1 illustrates the organization of this thesis.

POS names are written in Japanese through this thesis. Appendix A shows translation of Japanese POS tagset. Appendix B presents how we make unknown word tagged corpora. We also discuss the definition of unknown word in the preceding works and this thesis.

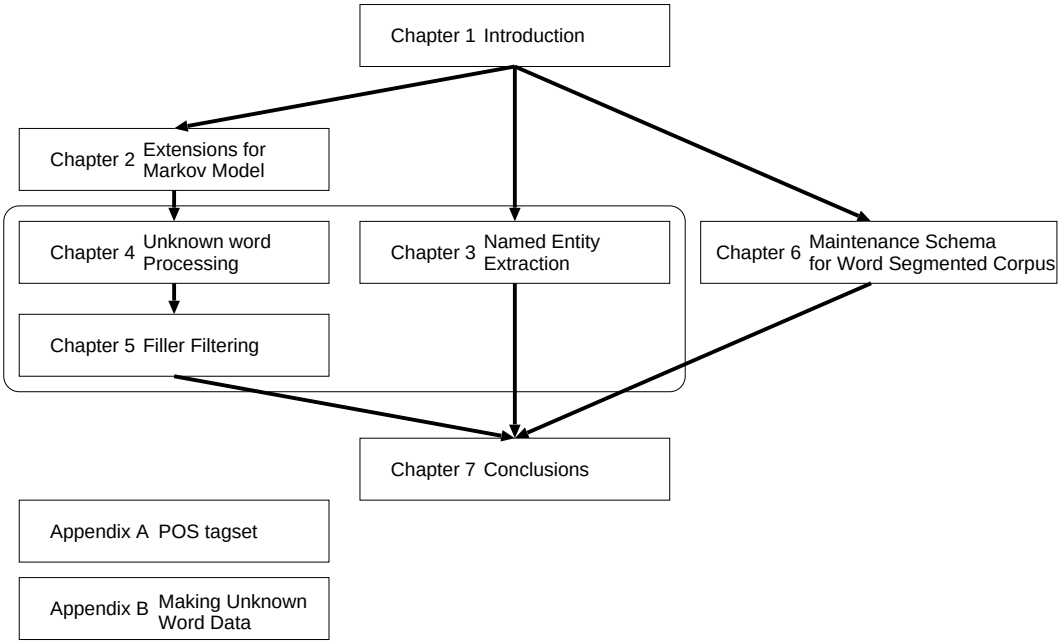

Figure 1.1. Organization of this thesis

Chapter 2

Extensions for Markov Model

If you label a person as having only so much ability,
you will never bring out her/his true potential
– **Masaru Ibuka**(1908-98)

In this chapter, we discuss the known word model based on Markov model. We introduce three sorts of extension for Markov model to solve problems in Japanese. The Markov model is one of well-known model for POS tagging and morphological analysis. Firstly, we argue about the basic Markov model for morphological analysis. Secondly, we discuss the problems in Japanese morphological analysis. Thirdly, we present the extensions for Markov model. Fourthly, we evaluate the proposed extensions.

2.1 Basic Model

2.1.1 Markov model for Japanese morphological analysis

The morphological analysis is to find the sequence of POS tags $T = t_1, \ldots, t_n$ for the word sequence $W = w_1, \ldots, w_n$ in the input string S. The target is to find W and T that maximizes the following probability:

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

Using the Bayes' rule of probability theory, $P(T|W)$ can be decomposed as the product of tag probability and word probability.

$$\begin{aligned} arg\max_{W,T} P(T|W) &= arg\max_{W,T} \frac{P(W|T)P(T)}{P(W)} \\ &= arg\max_{W,T} P(W|T)P(T). \end{aligned}$$

We approximate that the word probability is constrained only by its tag, and the tag probability is constrained only by its preceding tags either with bigram or trigram model. The word probability and the tag probability become as follows. To illustrate the formulae, $P_w(\cdot|\cdot)$ is used for the word probability, and $P_t(\cdot|\cdot)$ is used for the tag probability.

$$P(W|T) \quad = \quad \prod_{i=1}^{n} P_w(w_i|t_i),$$

$$P(T) \quad = \quad \prod_{i=1}^{n} P_t(t_i|t_{i-1}) \ \ (or\ P_t(t_i|t_{i-2}, t_{i-1})).$$

To simplify the formulae, we use following notations; $\langle w_i, t_i \rangle$ means the event of an occurrence of word w_i with t_i as its POS tag. $\langle \cdot, t_i \rangle$ means the event of an occurrence of words with t_i as its POS tag.

$F(E)$ means the frequency of an event E in the corpora. For example, $F(\langle w_i, t_i \rangle)$ means the frequency of word w_i with t_i as its POS tag. $F(\langle \cdot, t_i \rangle)$ means the frequency of POS tag t_i. $F(\langle \cdot, t_{i-1} \rangle, \langle \cdot, t_i \rangle)$ means the frequency of bigram context with POS tags t_{i-1} and t_i in this order. $F(\langle \cdot, t_{i-2} \rangle, \langle \cdot, t_{i-1} \rangle, \langle \cdot, t_i \rangle)$ means the frequency of trigram context with POS tags t_{i-2}, t_{i-1} and t_i in this order.

The values are estimated from the frequencies in tagged corpora using maximum likelihood estimation as following formulae:

$$\begin{aligned} P_w(w_i|t_i) &= \frac{F(\langle w_i, t_i \rangle)}{F(\langle \cdot, t_i \rangle)}, \\ P_t(t_i|t_{i-1}) &= \frac{F(\langle \cdot, t_{i-1} \rangle, \langle \cdot, t_i \rangle)}{F(\langle \cdot, t_{i-1} \rangle)}, \\ P_t(t_i|t_{i-2}, t_{i-1}) &= \frac{F(\langle \cdot, t_{i-2} \rangle, \langle \cdot, t_{i-1} \rangle, \langle \cdot, t_i \rangle)}{F(\langle \cdot, t_{i-2} \rangle, \langle \cdot, t_{i-1} \rangle)}. \end{aligned}$$

Using these parameters, the most probable tag sequence is determined by the Viterbi algorithm [20].

2.2 Target phenomena

In this section, we argue about the problems in Japanese morphological analysis. We present four sorts of target phenomena for the proposed method: function words, conjugation, contracted words and fine-grained POS tagset.

2.2.1 Function words

Function words are more important than contents words in Japanese morphological analysis. The function words tend to be crucial feature for word boundary and POS determination of other words. Then, the improvement of the accuracy for the functional words is directly related with the improvement of the accuracy for the whole sentence. Japanese language has two sorts of function words: particle and auxiliary verb. Particles do case-marking for the preceding word or relation-marking between the preceding and succeeding word or sentence-end-marking. Auxiliary verbs do mood-marking or tense-marking for the preceding word. Moreover, auxiliary verbs require conjugation form restriction for the preceding words.

Though the function words are crucial in Japanese morphological analysis, the word boundary and POS determination for the function words is quite difficult because of their character type – *hiragana*. Japanese language three character types: *kanzi*(漢字), *katakana*(カタカナ) and *hiragana*(ひらがな). *Kanzi* is used in contents words. Since *kanzi* has many characters, the word boundaries and POS possibilities for each *kanzi* character are not so complicated. *Katakana* is mainly used in nouns derived from foreign words and is not used in function words. Then, the word boundary and POS determination for *katakana* words is not so difficult except compounding phenomena. Still, *hiragana* is used for most of POS and has only about 80 characters. The word boundary and POS possibilities per one *hiragana* character are more complicated.

Furthermore, the word length of the functional words causes the other problem. The word length of the functional words is shorter than the word length of the content words. Then, the function words are embeded in other contentual word candidates (see the following example).

すももももももものうち

→ すもも [Noun] ／も [Particle] ／もも [Noun] ／も [Particle] ／もも [Noun] ／も [Particle] ／うち [Noun]
Peach includes Japanese plum, too.

Generally, the longer word candidates are tend to be selected in the framework of Markov model. The special care for the functional words is nessesary for Japanese morphological analysis.

2.2.2 Conjugation

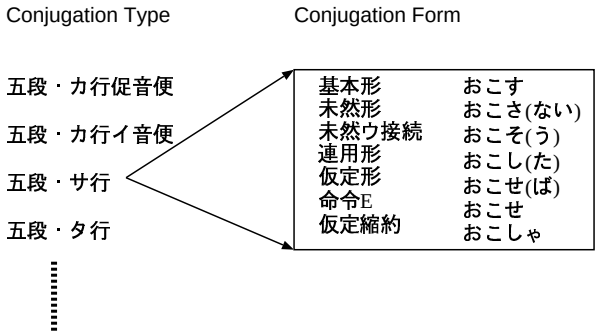

Figure 2.1. Example: conjugation types and conjugation forms

Japanese language has conjugation phenomena. We define some terms for conjugation phenomena to discuss it. *Conjugation form* means the connection characteristics of the form (e.g. 未然形, 連用形 and so on). *Conjugation type* means the type of the possible surface forms of the word (e.g. カ行・五段, 一段 and so on). Figure 2.1 shows examples of conjugation types and conjugation forms. Verb, adjective and auxiliary verb have conjugation in Japanese language. All possible conjugation forms are extracted as word candidates from the base form lexicon according to the conjugation type in practical Japanese morphological analyzers.

Whereas the conjugation form of a word is highly related to the POS of the adjacent succeeding word, it is not so related to the preceeding word. The conjugation type constraints the possible succeeding auxiliary verb. Then, the conjugation form and type are not negligible as the feature for Japanese morphological analysis.

2.2.3 Contracted words

Japanese morphological analysis is also used for making language model for speech synthesis and/or speech recognition. The adoptation for spoken language is in demand for Japanese morphological analysis. Contracted words occur frequently in spoken language. However, contracted words behave differently from words in written language.

Figure 2.2 shows an example of contracted word. The verb “ちゃう” is an contracted word, which is originally the particle “て” and the verb “しまう”. Whereas the verb “ちゃう” connects with the adjacent preceding words as the particle “て”, it connects with the succeeding words as the adjacent verb “しまう”. The model should be cover such phenomena.

The other problem of the contracted word is the corpus size of the spoken language. Most of corpora are based on newspaper. Then, the corpus size of spoken language is still small. The statistics of the contracted word cannot be acquired from the corpus adequately.

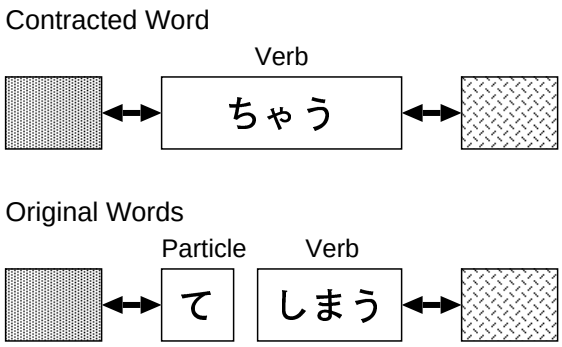

Figure 2.2. Example: contracted word

2.2.4 Fine-grained POS tagset

We handle IPA POS tagset [50] in this thesis[1]. The number of POSs is 66. We also regard conjugation types and conjugation forms as distinct tags. Then, the number of all tag is around 600. When we use such fine-grained POS tag set, we cannot make normal trigram model because of the data sparseness problem. On the other hand, a bigram model is too weak to cope with some phenomena in Japanese morphology. For example, consider the following sentences:

a) くだものはない

→ くだもの [Noun] ／は [Particle] ／ない [Adj.]
There is no fruit.

b) くだものではない

→ くだもの [Noun] ／で [Auxil. Verb] ／は [Particle] ／ない [Auxil. verb]
It is not fruit.

The word “ない” preceding the topic particle “は” can be adjective or auxiliary verb depending on the two preceding POS. If the two preceding word is noun, the POS of “ない” is adjective. If the two preceding word is auxiliary verb, the POS of “ない” is auxiliary verb. Therefore, trigram context is necessary to solve the problem.

2.3 Proposed extensions

Now, we present three sorts of extensions for Markov model in Japanese morphological analysis: *lexicalized POS*, *position-wise grouping* and *selective trigram*.

2.3.1 Lexicalized POS

Some words behave differently from the other words even in the same POS. To handle these words, we introduce *lexicalized POS*. It means that some words are regarded to belong to distinct POSs. In addition, we introduce the position-wise extension that we select different words for the current and preceding positions. The extension is useful for particles and auxiliary verbs, which is very important to determine the POS of the preceding or succeeding word. By lexicalized POSs, we can focus on ambiguous words like particles and auxiliary verbs.

[1] Section **??** includes all POS tagset with English translation.

We describe lexicalized POS in detail. When we introduce lexicalized POS to focus on exceptional words, the tag set becomes fine grained. The original tag set $\mathcal{T}$ extends to two new tag sets: $\mathcal{T}^c$ (for the current position) and $\mathcal{T}^p$ (for the preceding position). In these tag sets, we define some words as individual POSs. Modification to the probability formulae for the lexicalized POS is straightforward.

When the word of the preceding position w_{i-1} is extended, the word probability is not changed. The tag probability is changed as follows:

$$\begin{aligned} P_t(t_i|t_{i-1}) &= P(\langle \cdot, t_i \rangle | \langle w_{i-1}, t_{i-1} \rangle) \\ &= \frac{F(\langle w_{i-1}, t_{i-1} \rangle, \langle \cdot, t_i \rangle)}{F(\langle w_{i-1}, t_{i-1} \rangle)}. \end{aligned}$$

When the word of the current position w_i is extended, the word probability becomes as follows:

$$\begin{aligned} P_w(w_i|t_i) &= P(\langle w_i, t_i \rangle | \langle w_i, t_i \rangle) \\ &= 1. \end{aligned}$$

The tag probability is defined as follows:

$$\begin{aligned} P_t(t_i|t_{i-1}) &= P(\langle w_i, t_i \rangle | \langle \cdot, t_{i-1} \rangle) \\ &= \frac{F(\langle \cdot, t_{i-1} \rangle, \langle w_i, t_i \rangle)}{F(\langle \cdot, t_{i-1} \rangle)}. \end{aligned}$$

Note that, the statistics of the POS level should be modified, when some words in the POS are lexicalized.

Suppose that the POS tags A and B defined in $\mathcal{T}$. Some words $w_{a_1}, \ldots, w_{a_n} \in A$ are lexicalized in $\mathcal{T}^p$ (in the preceding position). We define the tag $A^p \in \mathcal{T}^p$ as follows:

$$A^p = A \setminus \{w_{a_1}, \ldots, w_{a_n}\}.$$

In the same way, some words $w_{b_1}, \ldots, w_{b_m} \in B$ are lexicalized in $\mathcal{T}^c$ (in the current position). We define the tag $B^c \in \mathcal{T}^c$ as follows:

$$B^c = B \setminus \{w_{b_1}, \ldots, w_{b_m}\}.$$

To estimate the probability for the connection (A, B), the frequency $F(\langle \cdot, A^p \rangle, \langle \cdot, B^c \rangle)$ is used instead of the total frequency $F(\langle \cdot, A \rangle, \langle \cdot, B \rangle)$. Figure 2.3 illustrates the statistics of this situation.

When we defined some words as lexicalized POSs, the occurrence frequencies of the words usually are so low that we may need to collect many examples to get enough size of corpus for analysis. As another method, to relax the data sparseness problem of the lexicalized POSs, we use the POS level statistics. We introduce a smoothing method between the word level statistics and the POS level statistics.

We define two smoothing rates: λ_{Lpre} is the smoothing rate for the preceding position, and λ_{Lcur} is the smoothing rate for the current position ($0 \leq \lambda_{Lpre} \leq 1, 0 \leq \lambda_{Lcur} \leq 1$).

When the word w_{i-1} in the preceding position is defined as lexicalized POS and w_{i-1} belongs to POS t_{i-1}, the tag probability P_t^{Lsmth} is defined as follows:

$$\begin{aligned} P_t^{Lsmth}(t_i|t_{i-1}) &= P_t^{Lsmth}(\langle \cdot, t_i \rangle | \langle w_{i-1}, t_{i-1} \rangle) \\ &= (1 - \lambda_{Lpre}) P_t(\langle \cdot, t_i \rangle | \langle \cdot, t_{i-1} \rangle) + \lambda_{Lpre} P_t(\langle \cdot, t_i \rangle | \langle w_{i-1}, t_{i-1} \rangle). \end{aligned}$$

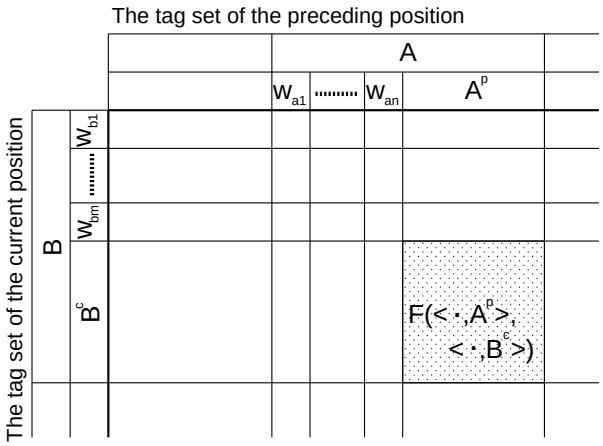

Figure 2.3. Statistics estimation for lexicalized POS

When the word w_i at the current position is defined as a lexicalized POS and w_i belongs to POS t_i, the tag probability P_t^{Lsmth} is defined as follows:

$$\begin{aligned} P_t^{Lsmth}(t_i|t_{i-1}) &= P_t^{Lsmth}(\langle w_i, t_i\rangle|\langle\cdot, t_{i-1}\rangle) \\ &= (1-\lambda_{Lcur})P_t(\langle\cdot, t_i\rangle|\langle\cdot, t_{i-1}\rangle) + \lambda_{Lcur}P_t(\langle w_i, t_i\rangle|\langle\cdot, t_{i-1}\rangle). \end{aligned}$$

2.3.2 Position-wise grouping

Since we use a fine-grained tag set, it is important to classify them into some equivalence classes to reduce the number of probabilistic parameters. Moreover, as is discussed in the preceding section, some words or POS behaves differently according to the positions they appear. In Japanese, for instance, the conjugation form plays an important role only to disambiguate the words at their succeeding position. In other words, the conjugation form should be taken into account only when they appear at the preceding position t_{i-1} in either bigram or trigram model. This means that when the statistics of verbs, adjectives and auxiliary verbs are taken, they should be grouped differently according to the positions.

Position-wise grouping for conjugation

First, the position-wise grouping is good for Japanese conjugation. Table 2.1, 2.2 shows the grouping rules for Japanese conjugation. *Group* means the grouping tag for the right column four tags; *lexicalization*, *POS*, *conjugation type* and *conjugation form*. We already presented that our tag includes three sorts of information; *POS*, *conjugation type* and *conjugation form*. In addition, we introduce a tag *lexicalization* for lexicalized POS. To simplify the notation, we use "`_all_`" which means all possible tags.

We describe the grouping rules for conjugation detailly. The difference between conjugation type "カ変・クル" and conjugation type "カ変・来ル" is only an ambiguity of orthography; *kanji* or *hiragana*. So we introduce the grouping to put these conjugation types in a group. For the preceding position, the information of conjugation form is much more important. We regard the all possible conjugation forms separately. For the current position, the information of conjugation form is not important so that we regard the all possible conjugation forms as one group tag. By these grouping, we can introduce the characteristics of Japanese conjugation for the model.

Table 2.1. Example: Grouping rules for conjugation (the preceding position)

Group	lexicalization	POS	conjugation type	conjugation form
カ変（基本形）	_all_	動詞-自立	カ変・クル	基本形
カ変（基本形）	_all_	動詞-自立	カ変・来ル	基本形
カ変（未然形）	_all_	動詞-自立	カ変・クル	未然形
カ変（未然形）	_all_	動詞-自立	カ変・来ル	未然形
カ変（連用形）	_all_	動詞-自立	カ変・クル	連用形
カ変（連用形）	_all_	動詞-自立	カ変・来ル	連用形
カ変（仮定形）	_all_	動詞-自立	カ変・クル	仮定形
カ変（仮定形）	_all_	動詞-自立	カ変・来ル	仮定形
カ変（命令形）	_all_	動詞-自立	カ変・クル	命令ｙｏ
カ変（命令形）	_all_	動詞-自立	カ変・来ル	命令ｙｏ
カ変（命令形）	_all_	動詞-自立	カ変・クル	命令ｉ
カ変（命令形）	_all_	動詞-自立	カ変・来ル	命令ｉ
...				

Table 2.2. Example: Grouping rules for conjugation (the current position)

Group	lexicalization	POS	conjugation type	conjugation form
カ変	_all_	動詞-自立	カ変・クル	_all_
カ変	_all_	動詞-自立	カ変・来ル	_all_

Position-wise grouping for contracted word

Second, the position-wise grouping is good for contracted word in colloquial expressions, too. Table 2.3 and 2.4 show the grouping rules for contracted forms “ちゃう”. To simplify the notation, we use “_each_” which means that the all possible tags are treated as independent tags.

The verb “ちゃう” is a contracted word consisting of two words “て” and “しまう” and behaves quite differently from the other words. One way to learn its statistical behavior is to collect various usages of the word and add the data to the training data after correctly annotating them. In contrast, the idea of point-wise grouping provides a nice alternative solution to this problem. By simply grouping the word into the same equivalence class of “て” for the current position t_i and grouping it into the same equivalent class of “しまう” for the preceding position t_{i-1} in $P(t_i|t_{i-1})$, it inherits the statistical behavior from these classes.

Table 2.3. Example: Grouping rules for contracted form (the preceding position)

Group	lexicalization	POS	conjugation type	conjugation form
しまう	“しまう”	動詞-非自立	五段・ワ行促音便	_each_
しまう	“ちゃう”	動詞-非自立	五段・ワ行促音便	_each_

Definition of position-wise grouping

Now, we describe the point-wise grouping in a precise way. We assume a bigram model for simplicity. However, we can easily extend the idea for a trigram model, too. Let $\mathcal{T}^p = \{A^p, B^p, C^p, \cdots\}$ be the original tag set at the preceding position and let $\mathcal{T}^c = \{A^c, B^c, C^c, \cdots\}$ be the original tag set at the current position. These tag sets can be used lexicalized POSs. We introduce two grouping tag sets: One is for the preceding position $\mathcal{G}^p = \{G_1^p = \{A^p, B^p\}, G_2^p = \{C^p\}, \cdots\}$. The other is for the current

Table 2.4. Example: Grouping rules for contracted form (the current position)

Group	lexicalization	POS	conjugation type	conjugation form
Particle(て)	“て”	助詞-接続助詞	N/A	N/A
Particle(て)	“ちゃう”	動詞-非自立	五段・ワ行促音便	_all_

position $\mathcal{G}^c = \{G_1^c = \{A^c\}, G_2^c = \{B^c, C^c\}, \cdots\}$. We define the equivalence mapping of the preceding position: $I^p(\mathcal{T}^p \to \mathcal{G}^p)$, and the mapping of the current position: $I^c(\mathcal{T}^c \to \mathcal{G}^c)$.

Suppose we express the equivalence class to which the tag t belongs as the grouping tag $[t]^p \in \mathcal{G}^p$ for the preceding position (mapped by I^p) and the grouping tag $[t]^c \in \mathcal{G}^c$ for the current position (mapped by I^c), then:

$$
\begin{aligned}
P_w(w_i|t_i) &= P(\langle w_i, [t_i]^c\rangle|\langle\cdot, [t_i]^c\rangle) \\
&= \frac{F(\langle w_i, [t_i]^c\rangle)}{F(\langle\cdot, [t_i]^c\rangle)} \\
&= \frac{F(\langle w_i, t_i\rangle)}{F(\langle\cdot, [t_i]^c\rangle)}, \\
P_t(t_i|t_{i-1}) &= P(\langle\cdot, [t_i]^c\rangle|\langle\cdot, [t_{i-1}]^p\rangle) \\
&= \frac{F(\langle\cdot, [t_{i-1}]^p\rangle, \langle\cdot, [t_i]^c\rangle)}{F(\langle\cdot, [t_{i-1}]^p\rangle)}.
\end{aligned}
$$

Position-wise grouping for other phenomena

By using the grouping, it becomes easy to handle a word that has several surface forms. For example, Japanese have three kinds of orthography; *kanji*, *katakana* and *hiragana*. Some words are written in two or more kinds of orthography. Grouping can identify these surface forms uniquely. As another example, Japanese words with conjugation have *okurigana*, which is *hiragana* suffix orthography for the preceding *kanji* orthography. *Okuriganas* have various length from one word. In a future work, we would like to introduce the grouping for all words which settles these ambiguities of the surface form.

2.3.3 Selective trigram

When we use a fine-grained POS tag set, we cannot make normal trigram model because of the data sparseness problem. On the other hand, a bigram model is too weak to cope with some phenomena. We introduce *selective trigram* to solve this problem. We can consider a trigram context as an exceptional one in this model.

As is discussed in the preceding section, the word “ない” has POS ambiguity between adjective and auxiliary verb. When the particle “は” appeared in the preceding position, succeeding “ない” is usually an adjective. An exception is, when auxiliary verb “で” appeared in the second preceding position, “ない” is an auxiliary verb. To analyze these phenomena correctly, we introduce the following trigram rules selectively. Figure 2.4 shows example rules. Note that, $*$ means all possible POSs except auxiliary verb “で”.

Now we present selective trigram detailly. Selective trigram is a mixture of trigram statistics with bigram ones. When a bigram context and a trigram context have some intersection, the trigram context is regarded as an exception within the bigram context. In this sense, all the contexts are mutually disjoint as well in our model. When a bigram context overlaps with a trigram context, the bigram statistics are taken by excluding the trigram statistics.

When we include the POS trigram context (A, B, C) in the model, we use the following tag probability:

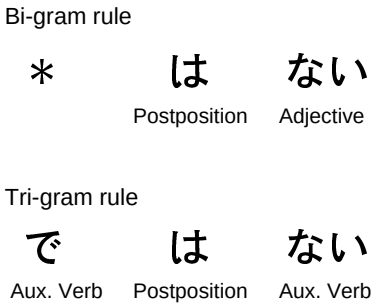

Figure 2.4. Example: selective trigram

$$\begin{aligned} P_t(t_i|t_{i-2},t_{i-1}) &= P_t(\langle\cdot,C\rangle|\langle\cdot,A\rangle,\langle\cdot,B\rangle) \\ &= \frac{F(\langle\cdot,A\rangle,\langle\cdot,B\rangle,\langle\cdot,C\rangle)}{F(\langle\cdot,A\rangle,\langle\cdot,B\rangle)}. \end{aligned}$$

To make the context mutually disjoint, the statistics of the bigram context (B,C) is taken as follows (F stands for true frequency in training corpora, while F' stands for estimated frequency to be used for probability estimation):

$$\begin{aligned} F'(\langle\cdot,B\rangle,\langle\cdot,C\rangle) &= F(\langle\cdot,B\rangle,\langle\cdot,C\rangle) - F(\langle\cdot,A\rangle,\langle\cdot,B\rangle,\langle\cdot,C\rangle), \\ F'(\langle\cdot,B\rangle) &= F(\langle\cdot,B\rangle) - F(\langle\cdot,A\rangle,\langle\cdot,B\rangle), \\ P_t(t_i|t_{i-1}) &= P_t(\langle\cdot,C\rangle|\langle\cdot,B\rangle) \\ &= \frac{F'(\langle\cdot,B\rangle,\langle\cdot,C\rangle)}{F'(\langle\cdot,B\rangle)}. \end{aligned}$$

Figure 2.5 illustrates statistics estimation for selective trigram.

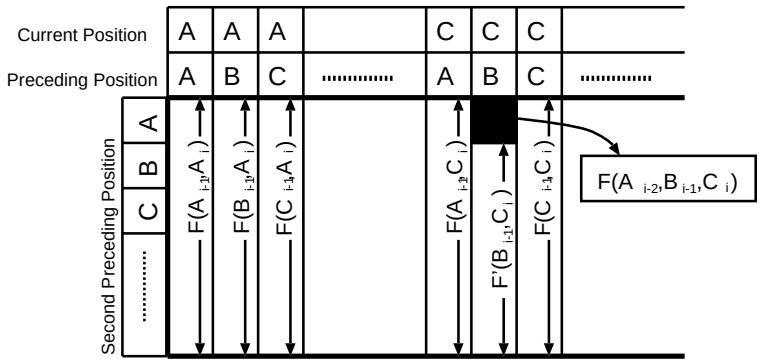

Figure 2.5. Statistics estimation for selective trigram

We permit lexicalized POSs and position-wise grouping over both bigram and trigram tag probabilities. When we do not introduce grouping over some selective trigram contexts, we introduce a smoothing technique on the context. We define smoothing rates λ_{con} between bigram and trigram contexts ($0 \leq \lambda_{con} \leq 1$). When we introduce the smoothing technique on the trigram context (A,B,C), the tag probability P_t^{Csmth} is defined as follows:

$$P_t^{Csmth}(t_i|t_{i-2},t_{i-1}) \quad = \quad (1-\lambda_{con})P_t(\langle\cdot,C\rangle|\langle\cdot,B\rangle) + \lambda_{con}P_t(\langle\cdot,C\rangle|\langle\cdot,A\rangle,\langle\cdot,B\rangle).$$

2.4 Experimental results

We manually determine the features for the model with the error information. The determination is based on the linguistical knowledge for the errors. These experiments are conducted to confirm the effect of each extensions. First, we present the methods of experiments. Second, we show results and give discussions.

To evaluate how the proposed extension improves the normal bigram model, we conducted several experiments. We group verbs according to the conjugation forms at the preceding position, take lexicalized POSs for all particles, auxiliary verbs and symbols, each of which is smoothed with the immediately higher POS level. Selective trigram contexts are defined for discriminating a few notoriously ambiguous particle "no" and auxiliary verbs "nai" and "aru." This is a very simple extension but suffices to evaluate the effect of the learning tools.

We use 5-fold cross evaluation over the RWCP tagged corpus [61]. The corpus data size is 37490 sentences (958678 words). The errors of the corpus are manually corrected. The annotated corpus is divided into the training data set (29992 sentences, 80%) and the test data set (7498 sentences, 20%). Experiments are repeated 5 times, and the results are averaged.

The evaluation is done at the following 3 levels:

- level 1: only word segmentation (tokenization)
- level 2: word segmentation and the top level POS
- level 3: all POS information

Table 2.5. Models for Japanese morphological analysis

Model	Description
Baseline	normal bigram model
+Lex	introduced lexicalized POSs
+Lex+Smth	introduced lexicalized POSs and smoothing for lexicalization
+Lex+Grp	introduced lexicalized POSs and grouping
+Lex+Smth+Grp	introduced lexicalized POSs, smoothing for lexicalization and grouping
+Lex+Smth+Grp+SelTri	introduced lexicalized POSs, smoothing for lexicalization, grouping and selective trigram

We make 6 models for the evaluation. We describe features of the models in Table 2.5. The smoothing rate for lexicalized POSs is fixed to 0.9 for each word. For selective trigram, we introduce smoothing technique between bigram and trigram. The smoothing rate for context length is fixed to 0.9 for each context.

To evaluate the results, we use the F-measure defined by the following formulae:

$$\text{Recall} \quad = \quad \frac{\text{the number of correct words}}{\text{the number of words in corpus}},$$

$$\text{Precision} \quad = \quad \frac{\text{the number of correct words}}{\text{the number of words by system output}},$$

$$F_\beta = \frac{(\beta^2+1)\cdot \text{Recall}\cdot \text{Precision}}{\beta^2\cdot(\text{Precision}+\text{Recall})}.$$

We evaluate the F-measure (with $\beta = 1$) for the test data set. Table 2.6 shows the result for each model and each level.

Table 2.6. Results: Japanese morphological analysis

Model	level 1	level 2	level 3
Baseline	99.006	98.440	97.356
+Lex	99.082	98.576	97.682
+Lex+Smth	99.082	98.578	97.682
+Lex+Grp	99.122	98.598	97.706
+Lex+Smth+Grp	99.126	98.610	97.710
+Lex+Smth+Grp+SelTri	99.128	98.704	97.812

Lexicalized POS is an extension for particles and auxiliary verbs. Japanese particles and auxiliary verbs are mostly written in *hiragana* so that the word segmentation of these words has much ambiguity. Table 2.6 indicates that the lexicalized POS disambiguates word segmentation of these words. Smoothing for lexicalized POSs improved the model slightly. In the experiments, we fix the smoothing rate as 0.9. We would like to determine more effective rate in our future work.

Position-wise grouping covers behavior of words with conjugation and behavior of contracted forms in colloquial expressions. By this extension, we can introduce some linguistical information for the model flexibly. In fact, some conjugation forms and some contracted forms do not appear in the corpora. The extension enables to cover such unseen phenomena.

Basic model cannot introduce trigram rules for a large scale tag set. Selective trigram makes it possible to introduce some trigram contexts partially. In this experiment, the selected trigram rules are limited only 30. Nevertheless, the results show that selective trigram improved the statistical model.

2.5 Related works

2.5.1 Related works for the extensions

Kim [32] uses the lexicalized POS for POS tagging of English. Kim's method has to select the same set of words for lexicalization at both the current and preceding position. In our method, we can select different words for lexcalization at each position.

Cutting [9] introduces a grouping method of words into equivalence classes based on the set of possible tags to reduce the number of the parameters. Schmid [28] introduces another way to define the equivalence classes. Their classes define not a partition of POS tags, but mixtures of some POS tags. These methods are based on the assumption that, when words have the same possible POS tags, they behave similarly. Meanwhile, in our method, the grouping is introduced with the feature of Japanese conjugation or contracted forms. The grouping is defined over the hierarchical structure. Moreover different groupings are defined at each position in conditional probabilities of Markov model.

Ron [12] presents a Markov model with variable memory length. Schütze [26] applies the idea to a model for POS tagging of English. Markov model with variable memory length is mixture of n-gram models with a series of values for n. In their model, the contexts represented by various length must be mutually disjoint so that the finite automaton constructed by the model to be deterministic. Meanwhile we introduce a different augmentation We regard the trigram contexts as exceptional contexts. When there is an intersection between bigram contexts and trigram contexts, we regard the trigram contexts as the exceptional contexts. In our idea, all contexts are mutually disjoint in its nature. Then our model can be converted to Ron's formulae.

We also these model for English POS tagging and Chinese morphological analysis. We incorporate an error-driven method for selecting lexicalized POS and selective trigrams (see [48]).

2.5.2 Other models for Japanese morphological analysis

Several extensions are proposed on Markov model. Kitauchi [2] introduces an error-driven method for feature selection for Markov model-based morphological analysis (RWCP Corpus). Nagata [53] introduces a statistical language model and a probabilisitic model for unknown word for Markov model(EDR Corpus).

Other machine learning methods are also introduced for Japanese morphological analysis in the preceding works. Uchimoto [44] [41] [42] introduces maximum entropy markov model based method (Kyoto Corpus/Corpus of Spontaneous Japanese). Nakagawa [75] introduces support vector machine-based revision learning method over the lattice of Markov model (Kyoto Corpus).

2.5.3 Other models for POS tagging in other languages

Most of techniques for Japanese morphological analysis are highly related to the techniques in POS tagging in other languages. We review several methods in other languages briefly.

Brill [15] [16] proposes a rule based POS tagger. The rules are generated by transformation-based learning. Schmid [27] [28] proposed decision tree based tagger. Daelemans [82] proposes memory based learning method for POS tagging. Ratnaparkhi[5] [6] proposes maximum entropy based method. Roth [13] proposes a method based on a network of linear separators. Brants [71] proposes a trigram tagger with several smoothing methods. Kazama [33] proposes a method which combined a maximum entropy tagger and hidden markov models. Lafferty [34] introduces conditional random fields for POS tagging.

2.6 Summary

In this chapter, we propose three extensions of the statistical model for POS tagging and morphological analysis; lexicalized POS, position-wise grouping and selective trigram. These extensions make the model more flexible to cover exceptional phenomena.

By the lexicalized POS, the model can cope with exceptional phenomena related to particular lexical items. In Japanese, postpositions and auxiliary verb are written in Hiragana orthography. These words are difficult to do word segmentation. To focus on these words, we regard the words as lexicalized POSs. By the lexicalization, the accuracy of these words is improved. Moreover, these words have generally important features to determine the POS of the preceding or succeeding position. When the accuracy of these words improved, the number of errors in the entire sentence is also reduced.

Unfortunately, when we introduce lexicalized POSs, the tag set will be overly fine-grained. Data sparseness problem becomes more serious. Position-wise grouping relaxes this problem, since the number of statistically independent tags is reduced. Moreover, position-wise grouping covers some linguistical phenomena as well as data sparseness problem. One such phenomenon is Japanese conjugation. When a word with conjugation occurs the preceding position, the conjugation form of the word is one of important features for the succeeding word. However, when the word with conjugation occurs in the succeeding position, the conjugation form of the word is not important. A similar phenomenon can be observed in contracted form in spoken languages. Contracted words are originally composed by two or more words. These words may have different connection behavior whether the word is at the preceding position or at the succeeding position. It is because the original words, which compose the contracted word, may have different tags. Position-wise grouping can reflect these phenomena for the statistical model.

Data sparseness problem arises again when we make simple trigram model. When we make normal trigram model with lexicalized POSs, the number of rules will drastically increase, though some trigram

context phenomena do not occur in the corpora. Nevertheless, there are some phenomena which require trigram context statistics to analyze. One of the method to overcome this problem is a smoothing technique with bigram statistics. As another solution, we introduce selective trigram model. It is based on the bigram model and only meaningful trigram contexts are incorporated. By this extension, we can make the model with trigram contexts from adequate sized corpora.

Chapter 3

Named Entity Extraction

> *... to divide each problem I examined into as many parts as was feasible, and as was requisite for its better solution.*
>
> – **René Descartes**(1596-1650)

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 [31]. 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 "小泉首相が９月に訪朝" is segmented as "小泉/首相/が/９月/に/訪朝" by a morphological analyzer. "小泉" ("Koizumi" – family name) 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 3.1 illustrates an example with English translation.

Input sentence:

小泉首相が９月に訪朝

Morphologically analyzed sentence:

小泉	首相	が	９月	に	訪朝
Koizumi	Prime-Minister	particle	September	particle	visiting-North-Korea

Prime Minister Koisumi will visit North Korea in September.

Named entities in the sentence:

- 小泉/"Koizumi"/PERSON,
- ９月/"September"/DATE,
- 朝/"North Korea"/LOCATION

Figure 3.1. Example: word unit problem

Some preceding works try to cope with the word unit problem. Uchimoto [43] introduces transformation rules to modify the word units given by a morphological analyzer. Isozaki [23] controls the parameters of a statistical morphological analyzer so as to change the word unit. 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* [72] for the chunking process. Our method achieves better score than all the systems reported previously for IREX NE extraction task.

3.1 Task description

Table 3.1. Example: named entities in IREX

NE Type	Examples	English translation
ARTIFACT	ノーベル化学賞	Nobel Prize in Chemistry
DATE	５月５日	May 5th
LOCATION	大韓民国	Republic of Korea
MONEY	２００万ドル	2 million dollars
ORGANIZATION	社会民主党	Social Democratic Party
PERCENT	２０％, 三割	20%, thirty percents
PERSON	村山富市	Murayama Tomiichi
TIME	午前５時	a.m. 5:00

The task of named entity extraction in the IREX workshop is to recognize eight named entity types shown in table 3.1 [31]. In their definitions, “ARTIFACT” contains book titles, laws, brand names and so on. The task can be defined as a chunking problem to identify 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 [47] and SE [43], 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 3.2 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. Note that an NE tag is a pair of an NE type and a chunk tag.

3.2 Proposed method

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

1. A statistical morphological analyzer is applied to the input sentence and produces POS tags of the point-wise *n*-best answers.

	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.

List of named entities:

- 小泉 *Koizumi* – Person name
- 日 *Japan* – Location name
- 朝 *North Korea* – Location name

Figure 3.2. Example: chunk tag sets for Japanese named entities

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.

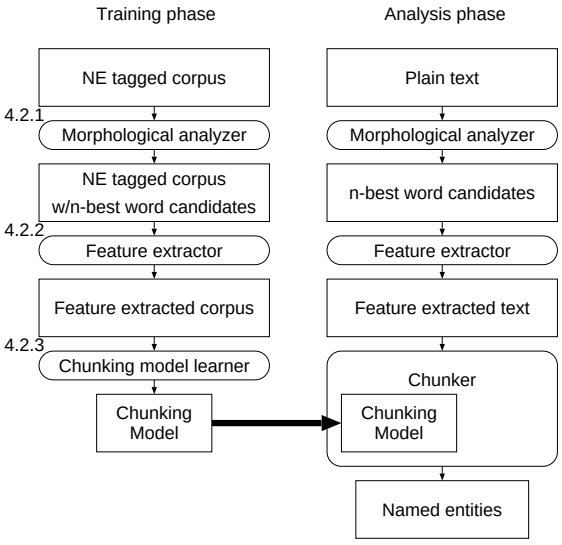

Figure 3.3. Overview of proposed method for Japanese named entity extraction

Figure 3.3 shows the method. The numbers in the figure denote section number in this thesis. Now, we illustrate each of these three steps in more detail.

3.2.1 Japanese morphological analysis

Japanese morphological analysis is based on Markov model. as we presented in section 2.1.1. We review the formula(T is a POS tag sequence and W is a word sequence):

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

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.

We use log likelihood as cost in practice. Maximizing probabilities means minimizing costs. In this method, redundant analysis output means the point-wise 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.

3.2.2 Feature extraction for chunking

From the output of redundant morphological analysis, each character receives a number of features. POS tag information is subcategorized so as to encode relative positions of characters within a word. We employ SE tag model for encoding the position. 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 (名詞-一般) are represented as "B-名詞-一般", "I-名詞-一般" and "E-名詞-一般", respectively. The list of tags for positions in a word is illustrated in Table 3.2. Note that O tag is not necessary since every character is a part of a certain word.

Table 3.2. Tags for positions in a word (SE tagset)

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)

Character types are also used for features. We define seven character types as listed in Table 3.3.

Table 3.3. 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.)

Figure 3.4 shows an example of the features used for chunking process.

Position	Char.	Char. Type	POS(Best)	POS(2nd)	POS(3rd)	NE tag
$i-2$	小	OTHER	名詞-固有名詞-人名-姓-B	接頭詞-名詞接続-S	名詞-一般-S	B-PERSON
$i-1$	泉	OTHER	名詞-固有名詞-人名-姓-E	名詞-固有名詞-地域-一般-E	名詞-固有名詞-一般-E	I-PERSON
i	首	OTHER	名詞-一般-B	名詞-一般-S	名詞-接尾-助数詞-S	O
$i+1$	相	OTHER	名詞-一般-E	名詞-接尾-一般-S	*	
$i+2$	が	HIRAG	助詞-格助詞-一般-S	*	*	

Figure 3.4. Example: features for chunking

3.2.3 Support vector machine-based chunking

We use *yamcha* [72], which is support vector machine [80]-based chunker. 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 classe. In "Pairwise method", we prepare ${}_nC_2$ binary classifiers between all pairs of the classes.

Chunking is done deterministically either from the beginning or the end of sentence. Figure 3.4 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.

3.2.4 The effect of *n*-best answers

Position	Char.	POS(Best)	POS(2nd)	NE
1	日	名詞-一般	名詞-固有名詞-地域-国	LOCATION
2	本			
3	人		名詞-接尾-一般	

Figure 3.5. Effect of *n*-best answers (1)

Position	Char.	POS(Best)	POS(2nd)	NE
1	池	名詞-固有名詞-人名-姓	名詞-一般	PERSON
2	坊			
3	専	未知語	*	
4	永	名詞-固有名詞-人名-姓	形容詞-自立	
5	家	名詞-一般	名詞-接尾-一般	

Figure 3.6. Effect of *n*-best answers (2)

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 3.5 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 "名詞-固有名詞-地名-国" (Noun, Proper Noun, Name of a place, Name of a 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 3.6 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.

3.3 Experimental results

3.3.1 Data

We use CRL NE data [31] for evaluation experiments. CRL NE data includes 1,174 newspaper articles and 19,262 named entities. We perform five-fold cross-validation on several parameter settings to investigate the length of contextual feature, the size of redundant morphological analysis, feature selection and the degree of polynomial Kernel functions. We use IOB2 model for the chunk tag scheme since it gave the best result in a pilot study. F-measure ($\beta = 1$) is used for evaluation.

$$\begin{aligned} \text{Recall} &= \frac{\text{the number of correct NEs}}{\text{the number of NEs in corpus}}, \\ \text{Precision} &= \frac{\text{the number of correct NEs}}{\text{the number of NEs by system output}}, \\ F_\beta &= \frac{(\beta^2+1)\cdot\text{Recall}\cdot\text{Precision}}{\beta^2\cdot(\text{Precision}+\text{Recall})}. \end{aligned}$$

3.3.2 Results by parameter setting

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 3.4 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 (Left) and two succeeding (Right) 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 "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).

Table 3.4. The length of contextual feature and the extraction accuracy

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

The depth of redundant morphological analysis

Table 3.5 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 situation 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.

Feature selection

We use *POS tags*, *characters*, *character types* and *NE tags* as features for chunking. To evaluate how they are effective we test four settings, that is, "using all features (ALL)", "except *characters* (– *Char.*)", "except *character types* (– *Char. Type*)" and "except *subcategory of POS tags* (– *POS* subcat.)". Table 3.6 shows the results for these settings.

"Except *characters*" gives the worst accuracy, implying that *characters* are indispensable for NE extraction. "Except *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, "except *Char Type*" gives better accuracy. The reason is that numbers in Kanji are not defined in our character type definition.

Table 3.5. 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.

Table 3.6. 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
ALL	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
ALL	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.

The degree of polynomial Kernel functions

Table 3.7. The degree of polynomial kernel function and the extraction accuracy

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

We alter degrees of polynomial kernel function and check how the combination of features affects the results. As shown in Table 3.7, 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.

The effect of thesaurus

Table 3.8. 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 [29]. Isozaki [23] introduces the thesaurus – NTT Goi Taikei [64] – to augment the feature set. Table 3.8 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 "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 thesis, we did not encode the tree structure of the thesaurus. Introducing hierarchical relationships in the thesaurus is one of our future works.

3.3.3 Comparison with the related works

Table 3.9. The best model and the extraction accuracy

Named entity	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

Table 3.10. Comparison with related works

	CRL DATA	IREX GENERAL	Chunking Model	for the word unit problem
Uchimoto [43]		80.17	ME	Transformation rules
Yamada [29]	83.7		SVM	Examples in training data are segmented
Takemoto [86]		83.86	Lexicon and Rules	Compound lexicon
Utsuro [76]		84.07	ME and Decision List	
Isozaki [23]	86.77	85.77	SVM + sigmoid	Parameter control for a statistical morphological analyzer
Proposed method	87.21		SVM	Chunking by Character
Nakano [39]	89.03		SVM	Chunking by Character

A competition for Japanese named entity recognition was held in IREX workshop[31]. Our model is also based on the data published in IREX workshop. Sekine [69] shows the results of IREX workshop.

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 we apply "One-vs-Rest method". The best combined model achieves F-measure 87.21 (table 3.9). The model uses one-vs-rest method with the best model for each type shown in table 3.4, 3.5, 3.6, 3.7 and 3.8. Table 3.10 shows comparison with related works. Our method attains the best result in the previously reported systems.

Preceding 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.

The proposed method employs exactly same chunker with the Yamada's work [29]. 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.

Nakano [39] introduces *Bunsetsu* (Japanese phrase) features for our method. Their method covers a weakpoint in our method. Our method cannot cover much longer NEs. The *Bunsetsu* feature covers the longer NEs in their method. Therefore, the method achieves the best accuracy in the reported works.

3.4 Named entity extraction for other languages

There are several competitions for English named entity recognition (MUC6 [7][10] and MUC7 [54]). The competions are extended for other languages in CoNLL2002 shared task [17] and CoNLL2003 shared task [18]. The methods use a position tagging method [47] based on machine learning such as transformation-based learning [47], hidden Markov model [11], maximum entropy approach [1] [58] [11], memory based learning [81][19] [30], network of linear separators [52], boosting [14], regularized winnow [78][79], risk minimization based method[77], conditional random fields [21] [3] and support vector machines [40] [35].

Nowadays, the state-of-the-art systems introduce hand-maintained gazetteers to cover the phenomena which do not appear in named entity tagged corpora. Some researchers also introduce unlabeled corpora, either for extracting training instances [58] [30] or obtaining extra named entities for gazetteers [19] [3]. Combination of different learning system is one of interesting issues. Voting[59] [14], bagging[59], and stacking [11][35][14] are used for learner combinations.

3.5 Summary

The proposed named entity extraction method achieves F-measure 87.21 on *CRL Named Entity 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 named entity 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 named entities 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 named entity extraction.

Chapter 4

Unknown Word Processing

Impossible is a word only to be found in the dictionary of fools.
– **Napoleon Bonaparte** (1769-1821)

We discuss unknown word processing in this chapter. We introduce a offline strategy for unknown word processing. We split the unknown word extraction problem into the following two subproblems: unknown word identification and POS guessing for the identified words. We introduce pattern recognition method for the unknown word identification. Unknown words are extracted from large-scale unlabeled texts in advance. We introduce word sense disambiguation-like method for unknown word's POS guessing. Then, the extracted words are registed into POS tagged lexicon in a morphological analyzer. Figure 4.1 illustrates the offline strategy for unknown word processing.

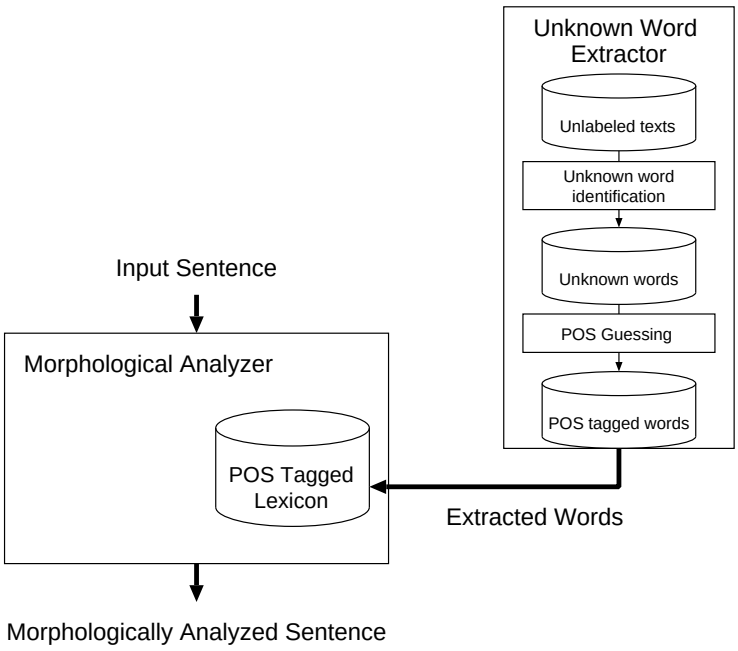

Figure 4.1. Offline strategy for unknown word processing

Section 4.1 presents how the pattern recognition method is introduced for unknown word identification. Section 4.2 presents how the word sense disambiguation method is introduced for unknown word's

POS guessing。 Section 4.3 presents evaluation experiments.

4.1 Pattern recognition method for unknown word identification

We introduce a pattern recognition method for unknown word identification. Unknown word occurences are identified by chunking. When we incorporate the method into unknown word identification, the unknown word tagged corpora are necessary. We define low frequent words as the unknown words in word segmented and POS tagged corpora. We introduce not only the actual frequency in the corpus but also the hit number on the web search engine as the criteria for the unknown words. We make 4 sorts of unknown word tagged corpora depending on criteria, use of lexicon and threshold. Table 4.1 shows the 4 corpora. Appendix B presents the definition of each corpus in detail.

Table 4.1. Unknown word tagged corpora

unknown word tagged corpus	criteria	with lexicon	threshold
freq. 1 w/o lex	frequency in the corpus	no	1
freq. 1 w/ lex	frequency in the corpus	yes	1
goo 1000	hit number on the search engine	yes	1000
goo 10000	hit number on the search engine	yes	10000

The pattern recognition method is same as the method in section 3.2. We review the method which is modified 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 POS tags of the point wise *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, unknown words are identified by an support vector machine-based chunker.

Firstly, we do morphological analysis for input sentences with redundant outputs. When an out-of-vocabulary word in a sentence, the statistical morphological analyzer faces difficulty in analyzing it. Because the analyzed output have a number of possible analyses at the position where the out-of-vocabulary word occurred. The ambiguities are extracted as features for the chunker at each character position. Finally, a support vector machine-based chunker retrieves the out-of-vocabulary words that cannot be identified by the morphological analyzer.

We use IOB2 model (figure 3.2 in the preceding chapter) for unknown word tagging. Table 4.2 shows the IOB2 tags for unknown word chunking.

Table 4.2. Tags for unknown word chunking

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

Chunking is done deterministically either from the beginning or from the end of the sentence. Figure 4.2 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, to infer the unknown word tag ("O") at the position i, the chunker uses the features appearing within the solid box.

Position	Char.	Char. Type	POS(Best)	POS(2nd)	POS(3rd)	unknown word tag
$i-2$	皆	OTHER	接頭詞-名詞接続-S	名詞-代名詞-一般-S	名詞-一般-S	B
$i-1$	婚	OTHER	名詞-一般-S	*	*	I
i	社	OTHER	名詞-一般-B	名詞-接尾-一般-S	名詞-一般-S	O
$i+1$	会	OTHER	名詞-一般-E	名詞-固有名詞-組織-E	名詞-接尾-一般-S	
$i+2$	の	HIRAG	助詞-連体化-S	助詞-格助詞-一般-S	*	

Figure 4.2. Example: features for unknown word identification

4.2 Word sense disambiguation method for unknown word's POS guessing

We introduce word sense disambiguation method for the POS guessing. Word sense disambiguation is the problem of determining in which sense a word having a number of distinct senses is used in a given sentence. The problem is solved by classifiers based on machine learning and/or statistics. The approaches utilize the contextual information as features for the classifiers.

For example, consider the word "bass", two distinct senses of which are:

1. a type of fish
2. tones of low frequency

and the sentences "The bass part of the song is very moving" and "I went fishing for some sea bass". To a human it is obvious the first sentence is using the word "bass" in sense 2 above, and in the second sentence it is being used in sense 1.

The problem is changed to determine the POS of the extracted word in this section. We use support vector machines [80] as the classifier. We introduce "one v.s. rest method" for the binary classifier (support vector machines) to adopt for the multi-class problem.

The features for the classifier are composed by the point-wise *n*-best outputs of Japanese morphological analysis, which is used in the unknown word identification. Figure 4.3 shows an example of the feature set. Two character contexts on both sides are referred to. The character types in the extracted word are also introduced. In the example, to infer the POS for the word "及び腰", the classifier uses the features appearing within the solid lines.

Char.	Char. Type	POS(Best)	POS(2nd)	POS(3rd)
権	OTHER	名詞-一般-E	名詞-接尾-一般-S	*
は	HIRAG	助詞-係助詞-S	*	*
及	OTHER	接続詞-B	*	*
び	HIRAG	接続詞-E	*	*
腰	OTHER	名詞-一般-S	*	*
で	HIRAG	助詞-格助詞-一般-S	助動詞-S	動詞-自立-S
、	OTHER	記号-読点-S	*	*

Figure 4.3. Example: features for unknown word's POS guessing

4.3 Experimental results

4.3.1 Recall evaluation for unknown word identification

Now, we evaluate recall of our method. We used *RWCP text corpus*[61] as the gold standard. We set up four datasets based on the frequency in the corpus and the hit number of a web search engine which is

shown in table 4.1 (see also appendix B).

Table 4.3 shows the four datasets. The words of which the hit number is lower than the threshold are regarded as unknown words. We evaluate how many unknown words in the corpus are identified. The direction of morphological analysis is fixed from the beginning of sentence (BOS) to the end of sentence (EOS). The direction of chunking is also fixed forward (from BOS to EOS).

Table 4.3. Data sets for recall evaluation

dataset	# of word in lexicon	(rate)	# of unknown word in corpus	(rate)
freq. 1 w/o lex	28576	(11.6%)	17396	(1.87%)
freq. 1 w/ lex	217103	(88.5%)	17396	(1.87%)
goo 1000	108471	(44.2%)	9814	(1.06%)
goo 10000	52069	(21.2%)	33201	(3.60%)
whole ipadic	245247	(100.0%)	646	(0.07%)

We perform five fold cross validation. In the selection of unknown word candidates, we see to it that the training data and the test data do not share any unknown word. We evaluate recall and precision 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 conducts only for recall, since it is difficult to make fair judgment of precision in this setting. The precision 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”. 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 4.4. Results: Unknown word identification – recall evaluation

dataset	Recall	Precision
freq. 1 w/o lex	53.5%	69.0%
freq. 1 w/ lex	56.5%	71.4%
goo 1000	56.2%	75.2%
goo 10000	74.7%	82.7%

Table 4.4 shows the result of recall evaluation. Though the unknown word rate is less than 2% in the dataset “freq. 1 w/o lex” and “freq. 1 w/ lex”, the method achieves over 50% recall. The differences between the result of “goo 1000” and “goo 10000” are caused by the differences of the size positive example – namely the number of unknown words in the data.

Figure 4.4 and 4.5 show the relation between recall and word frequency. The line indicates the recall on the whole data. The low frequent words can be extracted in nearly same accuracy as the whole data. Most of preceding works cannot extract low frequent words, since these methods are based on the word frequency in the text. The proposed method can extract the low frequent word, if the word is not registed in the lexicon of the morphological analyzer. Again, the preceding works, which is not based on frequency but hand maintained patterns, can extract only words in specific character type patterns. The proposed method can extract words in any character type patterns.

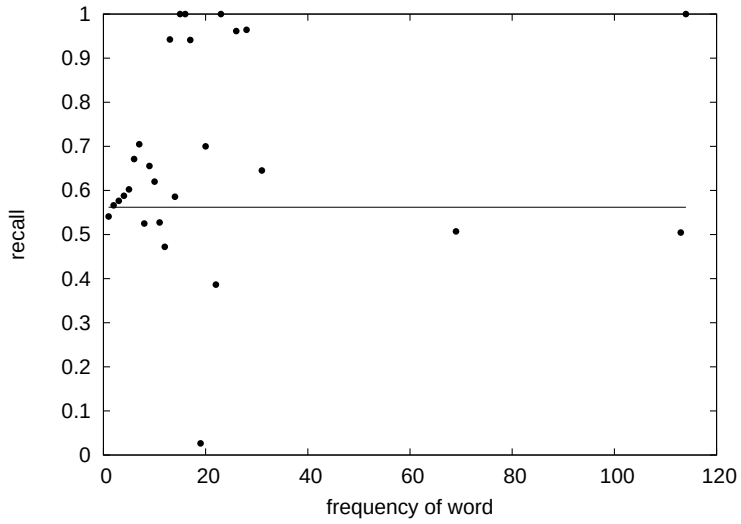

Figure 4.4. Results: Unknown word identification – frequency and recall on “goo 1000”

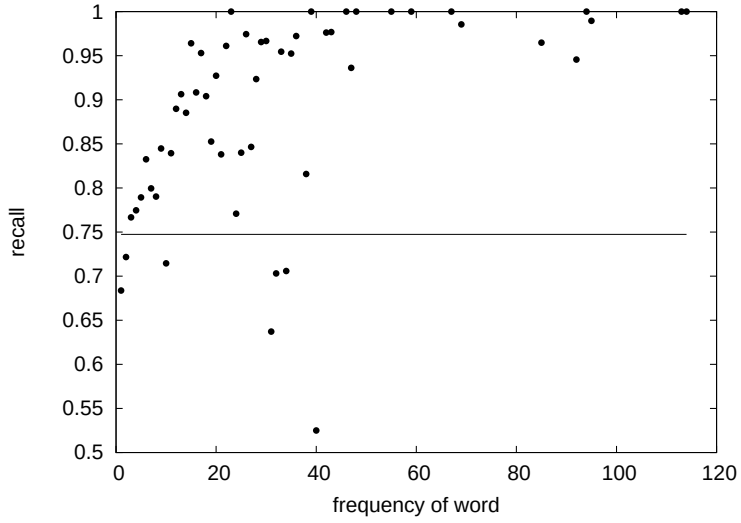

Figure 4.5. Results: Unknown word identification – frequency and recall on “goo 10000”

Table 4.5 and 4.6 show the recall of each POS in the dataset “goo 1000” and “goo 10000”. While the recall is not so good for the words which include compounds such as organization names and collocated case particles, it achieves high score 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 ambiguity errors between conjugational suffixes and auxiliary verbs.

4.3.2 Precision evaluation for unknown word identification

Now, we evaluate precision of the proposed method. We perform unknown word identification on newspaper articles and patent texts.

Table 4.5. Results: Unknown word identification – recall by POS “goo 1000”

POS	# of sample	Recall
名詞-一般	1899	43.4%
名詞-固有名詞-組織	1566	38.2%
名詞-固有名詞-人名-姓	1552	78.6%
名詞-固有名詞-人名-名	1540	78.3%
名詞-固有名詞-一般	1135	53.2%
動詞-自立	747	44.9%
名詞-固有名詞-地域-一般	501	69.6%
名詞-固有名詞-人名-一般	242	83.8%
名詞-サ変接続	218	55.5%
名詞-引用文字列	100	50.0%
名詞-形容動詞語幹	97	54.6%
形容詞-自立	47	19.1%
副詞-助詞類接続	41	51.2%
名詞-固有名詞-地域-国	36	41.6%
副詞-一般	20	20.0%

Table 4.6. Results: Unknown word identification – recall by POS “goo 10000”

POS	# of sample	Recall
名詞-一般	9009	67.1%
名詞-固有名詞-人名-名	3938	86.8%
名詞-固有名詞-組織	3800	63.8%
名詞-固有名詞-人名-姓	3717	90.4%
動詞-自立	3446	73.4%
名詞-サ変接続	2895	87.5%
名詞-固有名詞-地域-一般	1911	79.3%
名詞-固有名詞-一般	1864	58.3%
名詞-形容動詞語幹	624	83.2%
名詞-固有名詞-地域-国	449	88.4%
名詞-固有名詞-人名-一般	387	80.9%
形容詞-自立	208	47.6%
副詞-助詞類接続	191	72.8%
助詞-格助詞-連語	165	55.8%
名詞-接尾-助数詞	119	73.1%

Unknown word identification in newspapers

Firstly, we examine unknown word identification experiment in newspaper articles. We use articles of *Mainichi Shinbun* in January 1999 (116863 sentences). Note that, the model is made based on *RWCP text corpus* [61], which consists of articles of *Mainichi Shinbun* in 1994 (about 35000 sentences).

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

Concerning with compound words, we reject the words which do not match any constituent of the dependency structure of the largest compound word. Figure 4.6 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.

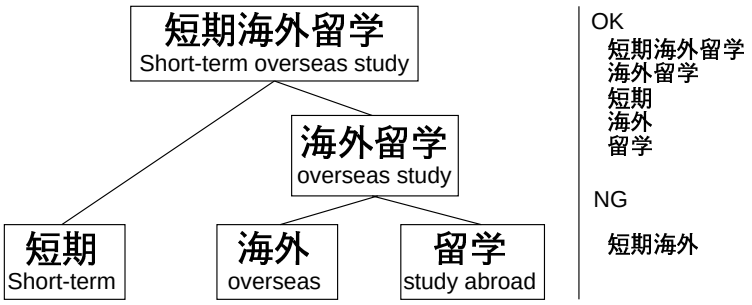

Figure 4.6. Judgement for compound words

We evaluate two models: “goo 1000” and “goo 10000”. The direction of morphological analysis is fixed from the beginning of sentence (BOS) to the end of sentence (EOS). We make two settings for the direction of chunking, forward (from BOS to EOS) and backward (from EOS to BOS).

Table 4.7. Results: Unknown word identification – precision for newspaper

Dataset/ Chunking Direction	# of identified words Token	Type	Precision
“goo 1000”/Forward	58708	19880	94.6%
“goo 1000”/Backward	59029	19658	94.0%
“goo 10000”/Forward	142591	41068	95.3%
“goo 10000”/Backward	142696	41035	95.5%

Table 4.7 shows the precision for newspaper articles. It shows that the proposed method achieves around 95% precision in both models. There is almost no difference in the several settings of the direction and the contextual feature.

Unknown word identification from patent texts

Next, we examine word identification experiment patent texts. We use patent texts (25084 sentences), which are OCR recognized. We evaluate models by the number of extracted words and precisions as in the preceding experiment. The extracted tokens may contain errors of the OCR reader in this experiments. 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 unknown words and OCR Errors.

Table 4.8. Results: Unknown word identification – precision for patent texts

Dataset/ Chunking Direction	# of identified words Token	Type	Accuracy Correct	Wrong	OCR Error
“goo 1000”/Forward	56008	12263	83.9%	15.4%	0.7%
“goo 1000”/Backward	56004	10505	89.2%	10.0%	0.8%
“goo 10000”/Forward	97296	16526	85.6%	13.7%	0.7%
“goo 10000”/Backward	98826	15895	87.0%	11.8%	1.2%

Table 4.8 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 word identification, the backward direction is effective for this task.

Identified examples

Figure 4.7. Examples of identified unknown words with 10 preceding & succeeding characters

Figure 4.7 shows examples of identified unknown words. Our method can correctly identified words which cannot be identified by the character type pattern-based method, such as “風び（し）” and “セーフティー（ネット）”. Moreover, we can also extract compound verbs such as “たたき出し”. For newspapers, errors occur frequently in the score of sports games (Sumo etc...). For patent texts, errors occur frequently in mathematical and chemical expressions. There are several peculiar prefixes such as “該”(correspond) or “前記”(above-mentioned) in the patent texts. The prefixes tend to attach the succeeding unknown words and cause errors. Introducing small size annotated texts in patent texts helps to reduce the errors.

4.3.3 Evaluation for word segmentation

Next, we evaluate how our method improves word segmentation accuracy. In the preceding experiments, we do chunking with tags in table 4.2. By annotating B and I tags to known words and rejecting O tag, we can do word segmentation with unknown word processing. *RWCP text corpus* [61] is used for the experiment. We define single occurrence words as unknown words in the corpus – namely “freq. 1 w/ lex” data. We use *ipadic* [50] as the lexicon in the morphological analyzer. The unknown words are excluded from the lexicon in this experiment. 50% of the corpus is reserved for Markov model estimation. 40% of the corpus is used for chunking model estimation. 10% of the corpus is used for evaluation. As the baseline model for comparison, we make simple Markov model using 50% and 90%

of the corpus. As the topline model, we make simple Markov model with a lexicon which includes the defined unknown words. Table 4.9 illustrates the models. The result is in table 4.10. It shows that the unknown word processing improves word segmentation accuracy.

Table 4.9. Data sets for word segmentation with unknown word processing

Corpus	50%	40%	10%
# of unknown words	8274	7485	1637
# of words	461137	368587	92222
Baseline (50%)	Markov model	ignored	test
Baseline (90%)	Markov model	Markov model	test
Proposed Method	Markov model	chunking model	test
Topline	Markov model	Markov model	Markov model/test

Table 4.10. Results: unknown word identification – word segmentation

	Rec.	Prec.	F-Measure
Baseline (50%)	97.7%	96.5%	97.1
Baseline (90%)	97.8%	96.6%	97.2
Proposed method	98.5%	98.1%	98.3
Topline	99.2%	98.8%	99.0

4.3.4 Evaluation for unknown word's POS guessing

Next, we evaluate the POS guessing module. The method is shown in section 4.2. We used polynomial kernel degree 2 for support vector machines. Features for the classifier are shown in figure 4.3. *RWCP text corpus*[61] is used for the experiment. We define single occurrence words as unknown words in the corpus – namely "freq. 1 w/ lex" data. We use *ipadic* [50] as the lexicon in the morphological analyzer. The unknown words are excluded from the lexicon in this experiment. 50% of the corpus is reserved for Markov model estimation. 40% of the corpus is used for POS guessing model estimation. 10% of the corpus is used for evaluation. Table 4.11 shows the data set.

The result is in table 4.12. POS for Japanese person names (名詞-固有名詞-人名-姓 and 名詞-固有名詞-人名-名) are analyzed in good accuracy, since the contextual information helps to determine the POS. However, the total accuracy is not so good accuracy. One problem is caused by the possibility-based POS tagset. For example, verbal noun "名詞-サ変接続" in IPADIC POS tagset means that the word behaves as both simple noun and verbal noun. If the word occurs as a verbal noun in the sentence, the classifier can identify the word as a verbal noun. Nevertheless, if the word occurs as a simple noun, the classifier cannot identify that the word can also behave as a verbal noun. The state-of-the-art word sense ambiguation method utilizes unlabeled data for the modeling. We would like to solve the posibility-based POS tagset problem by introducing such models in my future work.

Table 4.11. Data set for POS guessing

Corpus	50%	40%	10%
# of unknown words	8274	7485	1637
# of words	461137	368587	92222
Proposed Method	Markov model	POS guessing model	test

Table 4.12. Results: POS guessing for extracted words

	Prec.	Rec.	# in test data
名詞-一般	0.65	0.65	599
名詞-固有名詞-人名-名	0.92	0.86	182
動詞-自立	0.82	0.75	131
名詞-固有名詞-組織	0.58	0.31	122
名詞-固有名詞-人名-姓	0.83	0.65	110
名詞-固有名詞-一般	0.34	0.31	102
名詞-固有名詞-地域-一般	0.82	0.56	100
名詞-サ変接続	0.75	0.32	94
副詞-一般	0.50	0.13	30
名詞-固有名詞-人名-一般	0.33	0.12	26
名詞-形容動詞語幹	0.78	0.70	20
名詞-引用文字列	0.60	0.16	19
副詞-助詞類接続	0.40	0.21	19
形容詞-自立	0.25	0.05	19
名詞-接尾-一般	0.00	0.00	14
感動詞	0.20	0.22	9
名詞-副詞可能	0.00	0.00	6
名詞-接尾-助数詞	0.50	0.20	5
Other	0.00	0.00	28
All(F-measure 0.611)	0.551	0.687	1637

4.3.5 Comparison with the related works

Unknown word processing is one crucial step in Japanese morphological analysis. In the preceding works, there are two sorts of strategies for unknown words processing: offline strategy and online strategy. Figure 4.8 illustrates the offline and online strategy for unknown word processing.

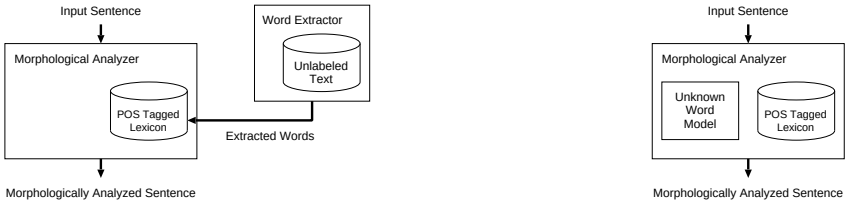

Figure 4.8. Offline strategy and online strategy

The offline strategy can be defined as the word extraction problem for the POS tagged lexicon. The word extraction is done offline from the large scale text database, then the extracted words are registed into POS lexicon which will be used in the models for known words. We use this strategy. The approaches in the word extraction strategy can be classified by the type of the clue to extract the word.

One is the pattern-based approach, which utilizes the pattern of character types and/or characters. Ikeya [51] presents a method to find unknown word boundaries for strings composed by only *kanji* characters. The method also uses 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 on unknown word identification in newspaper domain. Nevertheless, it can retrieve only words which match the defined character type patterns.

The other is the statistics-based approach which utilizes the frequency in the texts. Mori [67] presents a method in which frequent strings in texts are identified as unknown words using statistical methods with smoothing technique. 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[1] on EDR corpus. However, the method cannot identify low frequency unknown words.

The online strategy can be defined as the approaches with unknown word modeling. The approaches utilize the likelihood of the word formation from the arbitrary character sequence.

Nagata [53] classifies unknown word types based on the combination of character types in an unknown word. They defined likelihood by the normal distribution of the word length for each word type. The most likely POS and word sequences are estimated by the likelihood of word type and the contextual POS information. The method achieves 42.0% recall and 66.4% precision on EDR corpus[2].

Uchimoto [44] [41] [42] presents Maximum Entropy based methods. They extract all strings less than six characters in a sentence. Then, they do morphological analysis based on words in lexicon and extracted strings. The method is also based on the features of the character types and the contextual POS information. The method [44] achieves 82.40% recall on Kyoto corpus in which they evaluate both word boundaries and top-level POSs of unknown words[3].

4.4 Related works in other languages

The occurences of unknown word can be easily identified in English texts. Therefore, only unknown word's POS guessing is a research topic in such languages. One approach for unknown word's POS guessing is to use suffixes or surrounding context of unknown words [70]. Weischedel [60] introduces conditional probability using the ending form and the existence of hyphenation and capitalization. Brants [71] introduces the linear interpolation of fixed length suffix model in the trigram tagger.

Machine learning method is also introduced for POS guessing, which also uses suffix and or surrounding context of unknown words as the feature for the learner. Mikheev [4] introduces rule-based methods. Orphanos [22] introduces decision trees. Nakagawa [74] uses support vector machines.

For Chinese language, Chen [37] introduces a method using statistical methods and human-aided rules. Their method achieves 89% precision and 68% recall on CKIP lexicon. Zhang [46] shows a method with role 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% for the Chinese person names recognition in *the People's Daily*. Our method is extended role tagging using redundant outputs of morphological analysis and support vector machine-based chunking.

4.5 Summary

We proposed a novel method to identify unknown word occurrences in Japanese texts. Our method is based on a pattern recognition method which is cascading a morphological analyzer and a chunker. Firstly, a Markov model-based makes outputs with n-best candidate. Secondly, SVM-based chunker captures patterns where unknown words occur. Our method isn't based on frequency. Therefore, the method can identify low frequency unknown words. We evaluated the method in several texts. The results show that the method is applicable in the practical use such as the patent text.

Zhang [46] shows a method with role 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.

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

[2] Seed lexicon size 45,027 words, training corpus size 100,000 sentences, test corpus size 100,000 sentences.

[3] Seed lexicon size 180,000 words, training corpus size 7,958 sentences, test corpus size 1,246 sentences (out-of-vocabulary rate 17.7%)

Their method achieves 69.88% presicion and 91.65% for the Chinese person names recognition in *the People's Daily*. Our method is extended role tagging using redundant outputs of morphological analysis and support vector machine-based chunking.

The proposed POS guessing method is still simple. The POS tagset in *IPADIC* is composed by possibility-based POSs. For example, "名詞-副詞可能" (Adverbial Noun) means the word can be used as both adverb and common noun. Therefore, a simple POS guessing method cannot discriminate "副詞-一般" (General Adverb) and "名詞-副詞可能" (Adverbial Noun) by one example sentence in which the word behave as adverb. Our future work should solve the problem.

Chapter 5

Filler Filtering

The time to hesitate is through.
– **Jim Morrison**(1943-1971)

In this chapter, we argue about problems in morphological analysis of transcriptions of spoken language. Matsumoto and Den [85] present four problems in morphological analysis of spoken language – "Filler", "Disfluency", "Ungrammaticality" and "Peculiar representation in spoken language". Their article proposes a method based on mixed statistical models of spoken and written language. *UniDic* project [83] has now been started for compiling an electronic dictionary usable for morphological analysis of both written and spoken languages. The *UniDic* lexicon will cover "Peculiar representation in spoken language" such as contractions. Furthermore, the statistical model of *UniDic* is estimated from "Corpus of Spontaneous Japanese" [38] – hereafter CSJ Corpus. *UniDic* is expected to cope with two problems namely "Ungrammaticality" and "Peculiar representation in spoken language".

Nevertheless, the first two problems – "Filler" and "Disfluency" – cannot be solved by these methods. Statistical morphological analyzers based on Markov model make errors at the positions where fillers or disfluencies occur. Since most analyzers are trained on written language, the contextual information is broken at such a position. Thus, automatic filler/disfluency identification methods are in demand.

We introduce a chunking method for filler/disfluency identification. The method is based on a pattern recognition method discussed in chapter 3. "Filler" and "disfluency" are regarded as targets for the pattern recognition method. Filler and disfluency tagged corpus is available in CSJ Corpus. We compose filler filter based on the CSJ corpus.

5.1 Target phenomena

Now, we explain the target phenomena – fillers and disfluencies respectively.

Fillers are utterances to fill pauses in speech. A typical filler is a prolonged monophthong accompanied with flat pitch. Short words like "あの"(/ano/, *uh*..) and "それで" (/sorede/, *then*..) are also regarded as fillers when they are accompanied by the prolongation of the last vowels and flat pitch. In several dictionaries for Japanese morphological analysis, fillers are listed as a closed set. Still, it is difficult to detect them by simple morphological analysis.

Disfluency is a word or a fragment of a word which is pronounced and corrected later. Disfluencies cannot be listed as a closed set like fillers because we cannot predict which words will be re-pronounced.

In the CSJ Corpus, fillers and disfluencies are annotated as figure 5.1. Table 5.1 shows the tagset for fillers and disfluencies. Disfluencies classified as content words or function words depending on the type of word that is re-pronounced.

Table 5.1. Filler/disfluency tags in CSJ Corpus

Tag	Description
D	Disfluency (content words)
D2	Disfluency (functional words)
F	Filler

(F えーと) 音声 と 文字 (D2 の) と の 関係 を
/(E-TO) ONSEI TO MOJI (NO) TO NO KANKEI WO/
relationship between voice and character

(D さ) 三十秒間 に おける
/(SA) SANJUUBYOUKAN NI OKERU/
in thirty seconds

Figure 5.1. Example: fillers and disfluencies

5.2 Pattern recognition method for filler filtering

In this section, we present a pattern recognition method for filler/disfluency identification. The goal is to automatically annotate of fillers and disfluencies in plain transcriptions of speech data. The method is based on the following three steps:

1. The input transcription of utterance is analyzed with redundant outputs by a statistical morphological analyzer;
2. The transcription is segmented into characters. Each character is annotated with the character type and POS tags of the top *n*-best answers given by the statistical morphological analyzer;
3. Using annotated features, a chunker based on support vector machines detects fillers and disfluencies

Firstly, we do morphological analysis for input transcriptions of utterance with redundant outputs. When a filler or disfluency occurs in a sentence, the statistical morphological analyzer will face difficulty in analyzing it. Because the analyzed output will have a number of possible analyses at the position where the filler or disfluency occurred. The ambiguities are extracted as features for the chunker at each character position. Finally, a support vector machine-based chunker retrieves fillers or disfluencies that cannot be identified by the morphological analyzer.

Repetition of pronunciation should be introduced as a feature to identify disfluencies. We define a feature of *differences of pronunciation* of hiragana characters appearing within a fixed context. Measure of differences of pronunciation is defined as the following four values (table 5.2). We introduce the feature between the current position and the position within 2 preceding and succeeding context characters. Figure 5.2 shows an example of extracted features [1] [2].

The tagset for chunking is based on the IOB2 model (table 5.3) [47]. Filler tags and chunk tags are paired. The paired tags are annotated to each character by the chunker based on extracted feature. We do chunking deterministically either from the beginning or the end of the sentence.

Figure 5.2 illustrates the chunking procedure when we use two character contexts with POS and character type information. We also use the two preceding filler tags which are already estimated. In the example, to estimate the filler tag "O" at the position i, we use the features within the solid lines.

[1] Because of spaces, the POS information is simplified and translated into English.
[2] Most of morphological analyzer define "Filler" as POS and have the filler entries in their lexicon.

Table 5.2. Feature values for differences of pronunciation

Tag	Description
1	the same pronunciation
2	only consonant difference
3	only vowel difference
0	otherwise

Position	Character	Character Type	Diff. of Pronunciation				POS(best)	POS(2nd)	POS(3rd)	Filler tag
			-2	-1	+1	+2				
$i-2$	い	HIRAG	0	0	0	1	Filler-S	Adj-S	Verb-S	D-B
$i-1$	短	OTHER	0	0	0	0	Adj-B	Adj-B	Prefi x-S	O
i	い	HIRAG	1	0	3	0	Adj-E	Adj-E	Filler-S	O
$i+1$	え	HIRAG	0	3	0	0	Filler-B	*	*	
$i+2$	ー	KATAK	0	0	0	0	Filler-E	*	*	

Figure 5.2. Extracted features

5.3 Experimental results

The CSJ Corpus is used as the gold standard for our experiments. The data which only contain non-verbal events are removed. Training data are 80% (77058 utterances, 54 speakers) which contain (D):7102, (D2):433 and (F):39504. Test data are 20% (18360 utterances, 13 speakers) which contain (D):1633, (D2):137 and (F):9299. For evaluation, we use recall, precision and F-measure ($\beta = 1$).

We use a beta release of *UniDic* [83] and *ipadic* [50] as the lexicon for statistical morphological analyzer. *Yamcha* [72] as the support vector machine-based chunker. To perform the experiment for open data, the words which occurred only in the test data are removed from the lexicon of the morphological analyzer. The features for the chunker are two character contexts with character types and POS tags of the *3*-best answers which are given by the morphological analyzer. The two preceding filler tags are also introduced as dynamic features. Repetition of pronunciation is introduced as a feature to identify disfluencies (table 5.2). We introduce the feature between the current position and the position within 2 preceding and succeeding context characters. Figure 5.2 shows the example of features for chunking.

We evaluate two settings: with or without features of *differences of pronunciation*. We also evaluate two settings for chunking directions: forward ($\rightarrow$) and backward ($\leftarrow$). The results of experiments are shown in table 5.4. The chunking direction backward achieves slightly better accuracy than the chunking direction forward. As a baseline, we use a simple morphological analyzer based on *UniDic* which contains filler entries. The simple morphological analyzer is 91.5 F-measure for filler detection. Our method achieves better F-measure (93.7) for filler identification. Our method can also identify disfluencies of content words with 52.5 F-measure. When we add the feature *differences of pronunciation*, the accuracy of the disfluencies of function words is slightly improved, however, the overall accuracy is degraded.

We also perform an experiment with *ipadic*. The result is table 5.5. Because of the size of lexicon (*ipadic* 240,000 words, *UniDic* 18,000 words), *ipadic* reduces out-of-vocabulary words in the transcriptions. Then, the method with *ipadic* achieves better accuracy for disfluencies of content word than the

Table 5.3. Chunk tags for fillers and disfluencies

Chunk tag	Description
B	Beginning of Filler or Disfluency
I	Inside of Filler or Disfluency (including End)
O	Outside of Filler or Disfluency

Table 5.4. Results: Filler and disfluency identification with *UniDic*

without *diff. of pronunciation*						
	Chunking Direction →			Chunking Direction ←		
Tag	Rec.	Prec.	F.	Rec.	Prec.	F.
(D)	59.9%	41.4%	48.9	69.1%	46.6%	52.5
(D2)	7.3%	32.3%	11.9	8.0%	35.5%	13.0
(F)	93.3%	93.8%	93.6	93.7%	93.7%	93.7
All	87.3%	83.0%	85.1	87.7%	84.8%	86.3
with *diff. of pronunciation*						
	Chunking Direction →			Chunking Direction ←		
Tag	Rec.	Prec.	F.	Rec.	Prec.	F.
(D)	60.9%	40.4%	48.6	61.1%	44.7%	51.6
(D2)	10.2%	25.5%	14.4	10.2%	25.5%	14.6
(F)	93.0%	93.6%	93.3	92.6%	93.4%	93.0
All	87.3%	82.2%	84.6	86.9%	83.6%	85.3

Table 5.5. Results: Filler and disfluency identification with *ipadic*

without *diff. of pronunciation*						
	Chunking Direction →			Chunking Direction ←		
Tag	Rec.	Prec.	F.	Rec.	Prec.	F.
(D)	55.9%	63.9%	59.6	54.6%	64.3%	59.1
(D2)	5.8%	33.3%	9.9	7.3%	34.5%	12.0
(F)	92.8%	93.3%	93.0	92.9%	93.0%	93.0
All	86.3%	89.3%	87.7	86.2%	89.2%	87.7
with *diff. of pronunciation*						
	Chunking Direction →			Chunking Direction ←		
Tag	Rec.	Prec.	F.	Rec.	Prec.	F.
(D)	56.0%	62.7%	59.1	56.5%	63.0%	59.6
(D2)	11.7%	31.4%	17.0	13.1%	35.3%	19.1
(F)	92.8%	93.6%	93.2	93.0%	93.5%	93.2
All	86.4%	89.1%	87.7	86.7%	89.0%	87.8

method with *UniDic*. The result shows disfluencies are identified as out-of-vocabulary strings excluding fillers. Features to discriminate disfluencies from unknown words are necessary for our method. It will be investigated in our future work.

5.4 Summary

Spontaneous speech processing is one of striking growth field in natural language processing. We newly proposed a task "filler filtering" in morphological analysis of transcripted spontaneous spoken language. We introduced a pattern recognition method for the task. Filler filtering accuracy is quite good enough for practical use. However, the disfluency identification accuracy is not still adequate. The method cannot discriminate between the disfluency and unknown word. Our future work should address the problem.

Chapter 6

Maintenance Schema for Word Segmented Corpus

Consistency is the last resort of the unimaginative.
– **Oscar Wilde**(1854-1900)

In languages that do not provide word boundaries in texts (e.g., Chinese and Japanese), consistent definition of word formation in the lexicon is crucial to natural language processing. However, there is no consensus on the definition of word delimitation in different annotated corpora of these types of languages. Linguistic databases, to keep multiple definitions of word delimitation, are much in demand as the requirement for word delimitation tends to differ depending on linguists and/or application areas.

To make matters worse, nominal compound word of Japanese consist of various kinds of nouns with prefixes and suffixes. In many cases, nominal compounds form different types of nouns from their constituent nouns, such as proper nouns or technical terms. Since complicated compounds have syntactic structure within themselves, such structure should be specified in the linguistic resources.

In this chapter, we present a use of relational database for consistent development and maintenance of linguistic resources of Japanese language, that is lexicons and annotated corpora. The database is also used for building statistical models for Japanese language analyzers. This chapter describes two major facilities of the current implementation.

First, the system maintains two sorts of data in the databases: lexicons and annotated corpora. The lexicons provide grammatical information as well as canonical form and the construction of the entry words. In order to keep consistency of word definition in the lexicon and the annotated corpus, we define the relationship between the words in those two linguistic resources through the relational database. It is necessary to ensure changes in one resource to be reflected to another. Such maintenance is automatically achieved by the use of relational database.

Second, we have to cope with multiple definition of word delimitation, as different applications call for different definition of word delimitation. To accommodate more than one definition of word formation in the lexicon, we facilitate a hierarchical definition of word formation of compounds in the database. The lexicon with the most fine-grained word definition is taken as the base lexicon, and the compounds are defined as binary tree structure consisting of the lower level entries. Users may select the grain size of the words so as to meet their requirements. Meanwhile, there are many contracted words in spoken language. Contracted words are derived from two or more words. We treat the original forms of contracted words as well in this framework.

Section 6.1 represents word delimitation definitions of Japanese used in the database. Section 6.2 shows the database schema for Japanese POS tagged corpora. Section 6.3 presents the compound word lexicon.

6.1 Word delimitation definitions of Japanese language

This section presents word delimitation definitions of Japanese language which is maintained in the database.

In Japanese language, definition of word boundaries varies according to gramatical theories and applications, and there is no common standard. Each word delimitation definition may have its own POS tagsets. For Japanese natural language processing, two POS tagsets are widely used, which are adopted by Japanese morphological analyzers. One is Masuoka-Takubo POS tagset[73] adopted by morphological analyzer *Juman*[66]. The other is IPA POS tagset[50] adopted by morphological analyzer *ChaSen*[84]. *ChaSen* output is much fine grained than *JUMAN* output. Figure 6.1 shows the difference of word delimitation definitions between *JUMAN* and *ChaSen*.

JUMAN	積極的に	融資	して	いる
ChaSen	積極 | 的 | に	融資	し | て	いる

Figure 6.1. The differences of word delimitation - *JUMAN* v.s. *ChaSen*

The lexicon, which has multiple definitions of word delimitation, is in demand. Our databases are developed to maintain lexicon and corpora with *UniDic* POS tagset[83] and word delimitation definitions. *UniDic* introduces four layers of word delimitation definitions to cover demands of many domains. Below, we show base idea and word delimitation definitions of the *UniDic*.

6.1.1 Word delimitation definitions in *UniDic*

There is no word delimitation definition which can be sharable among separate fields. In the *UniDic* scheme, they define four layers of word delimitation definitions as follows:

- Layer 0: Morphemes
  Layer 0 defines not “word” but “morpheme”. POS tag is not defined at this layer.
- Layer 1: Simple words
  Layer 1 defines the smallest unit of words without compound. We define the layer 1 as the basis of word definition and forms the units for compound words.
- Layer 2: Compound words
  The words of layer 2 are defined by layer 1 as following two rules:
  - matches following regular expression:
    `(prefix)*(Noun|Adjective|Verb)+`
    `(suffix)*`
  - and for each adjacent word pair, left side word has dependency with right side word.
- Layer 3: Named entities
  Named entity, idiom and collocation.

Layer 0 defines morphemes. One Chinese character(Kanji) is defined as one morpheme. For morphemes, which are composed by Katakana or Hiragana characters, are segmented into minimal units. The unit in layer 0 has no POS tag information, because these are defined not as words but as morphemes.

Layer 1 defines base words to make compound words. The words in layer 1 are minimal units which can put POS information. *UniDic* POS tagset is designed for the unit. Note that, POS tag names are defined in Japanese language. In this chapter, we use English POS name translated from the Japanese original tag name.

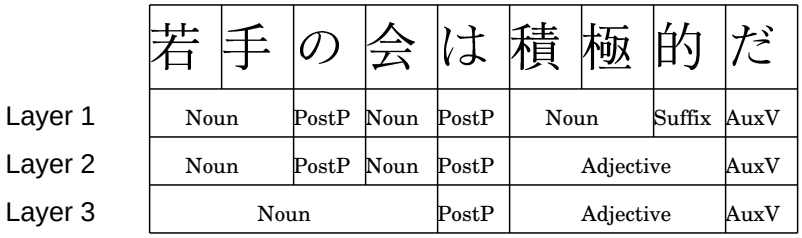

Figure 6.2. The difference of word delimitation - *UniDic*

A word of layer 2 is composed by words of layer 1. Second rule restricts the word of layer 2 within a left branching structure. The composition comes from the fact that a right branching structure restricts "*rendaku*" phenomena[1] and accent moving[24]. Figure 6.3 shows an example of *rendaku*. On one hand, the left branching structure of compounds is defined as one word in layer 2 definition, which has two *rendaku* phenomena. On the other hand, the right branching structure of compounds is defined as two words, then one *rendaku* is restricted beacuse of the right branching. Then, the unit of layer 2 is suited for putting form – Japanese reading – and accent informations.

Layer 2 is designed for not only speech and acoustic processing but also chunking. When, we do chunking from layer 1, IOB2 model fits the branching structure. IOB2 model, which is widely used in NP or named entity chunking, is a model to annotate following tags on fine grained word sequence:

- *B*: the beginning of the unit
- *I*: the inside of the unit
- *O*: the outside of the unit

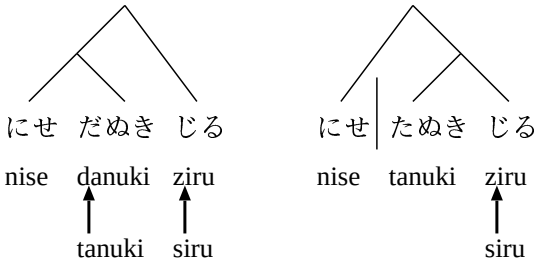

Figure 6.3. Dependency structures and "rendaku"

Figure 6.4 shows relation between branching structures and layer 2 chunks. The word, which do not match the regular expression, are put tag *O*. When a word match the regular expression, the word, which is left element of a subtree, is put tag *B*, otherwise the word is put tag *I*. In this sense, chunking based on IOB2 tag is suit for layer 2 definition.

Layer 3 defines much longer units like named entities, idioms and collocations. The definition has no relation with the dependency structure of word. Then, the relation between BIO tag and dependency structure is weakened. Figure 6.5 shows layer 3 chunks and IOB2 tags.

[1]The process voices an initial voiceless consonant of the second member in a certain class of compound words.

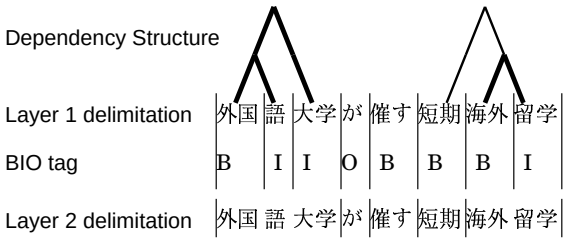

Figure 6.4. Layer 2 and IOB2 tag

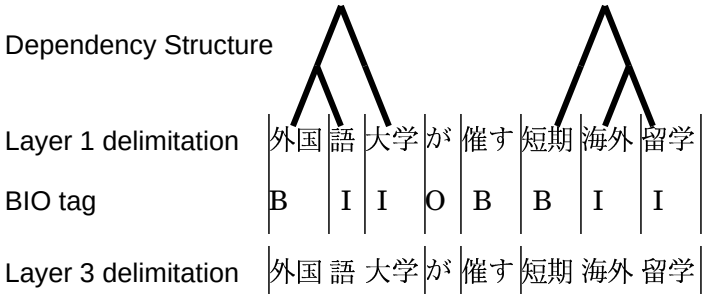

Figure 6.5. Layer 3 and IOB2 tag

6.2 Maintenance schema for Japanese POS tagged corpora

When we develop POS tagged corpora based on *UniDic* schema, we must maintain the information on each of the layers. POS tag informations of layers will be overlapping on original texts.

Stand-off annotation [25] enables us to cope with the overlapping problem. Stand-off annotation is the method separating markup from the material marked up.

First, we present former works of stand-off annotation framework. Second, we present our database schema for Japanese POS tagged corpora.

6.2.1 Stand-off annotation

Stand-off annotation framework is initially formalized in a field of SGML. We illustrate the framework by the example of Henry et al. [25].

Consider following marking sentence structure:

```
..
<w id='w12'>Now</w><w id='w13'>is</w><w id='w14'>the</w>
..
<w id='w27'>the</w><w id='w28'>party</w><w id='c4'>.</w>
..
```

We can mark sentences in a separate document as follows:

```
..
<s xml-link='sample' href="#ID(w12)..ID(c4)"></s>
<s xml-link='sample' href="#ID(w29)..ID(c7)"></s>
..
```

Their application enables us to see this documet collection as a single stream with the words nested inside the sentences:

```
..
<s>
<w id='w12'>Now</w><w id='w13'>is</w><w id='w14'>the</w>
..
<w id='w27'>the</w><w id='w28'>party</w><w id='c4'>.</w>
</s>
<s>
..
</s>
..
```

They showed three reasons why separating markup from the material marked up:

1. the base material cannot copy to introduce markup because of read-only and/or very large
2. the markup may involve multiple overlapping hierarchies
3. distribution of the base document may be controlled, but the markup is intended to be freely available

Our strong reason to introduce stand-off annotation is to cope with overlapping problem of Japanese POS information.

Stand-off annotation is widely introduced in the field of corpus maintenance [57]. Bird et al. [62] proposed *Annotation Graphs* which allows to encode various information in a structure. In XML framework, stand-off annotation is formalized as XPointer [63].

6.2.2 Stand-off annotation framework for Japanese POS tagged corpora

We introduce stand-off annotation framework for Japanese POS tagged corpora in order to solve three problems. First is to maintain POS tagged corpora based on multiple word delimitation definitions. Second is to permit coexistence POS information and other phonetic informations in corpora. Third is to keep consist word delimitation definition between lexicons and corpora.

Figure 6.6 shows the stand-off annotation for Japanese POS tagged corpora. We define character ID for each character in the original text. POS tag information are defined by "Begin ID", "End ID" and "Tag information". Original text and tag information are stored in different tables.

Stand-off framework enable us to keep multiple word delimitation definition on one text. Moreover, we can add other information to the corpora without overlapping restriction.

When we maintain Japanese POS tagged corpora, it is difficult to keep consistency of word delimitation definition. To cope with this problem, we introduce links between corpora and lexicons. Figure 6.7 shows the links between corpora and lexicons. The links make the word delimitation in tagged text consistent with the lexicon.

Note that, practical word lexicon keeps more information – base, form (i.e. Japanese reading), pronunciation, conjugation information and so on. *CFORM* stands for conjugation form. On the relational databases, we define "Word ID" and "CFORM ID" as primary key. Then, tag information is represented by pointers to the primary key. We use character ID as the anchor of pointers. When we maintain corpora of spoken language, the anchors are defined as time and track on audio data.

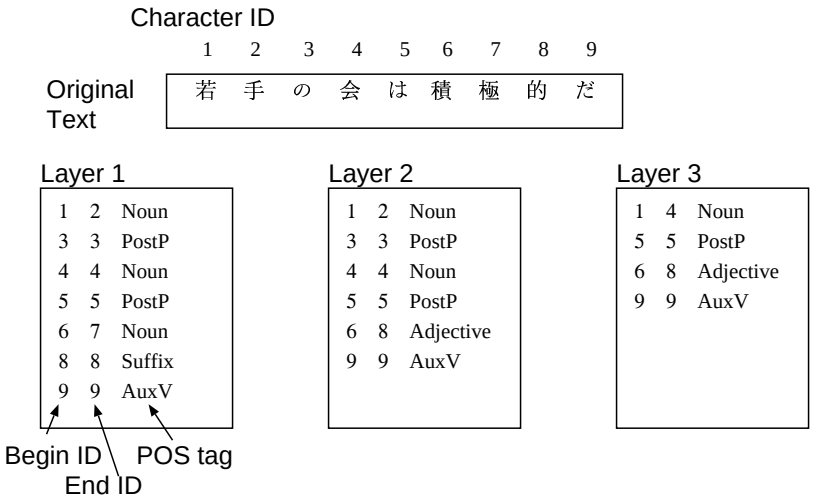

Figure 6.6. Stand-off annotation for several word delimitation definitions

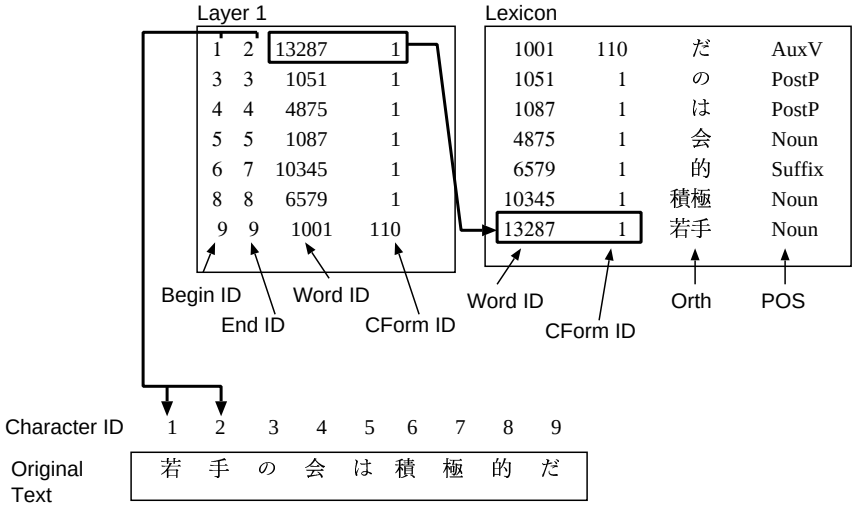

Figure 6.7. Linking between corpora and lexicons

6.3 Compound word lexicon – relationships among multiple word delimitation definitions

In the preceding section, we present how to deal with multiple word delimitation definition for one text. In this section, we present how to deal with the relationships among these definitions. We maintain the relationship as compound word lexicon with dependency structure. First, we classify words based on the dependency structure of the words. We show the definitions of categories for the words. Second, we present the database schema to deal with the dependency structure in compound words. Third, we present XSLT usage to extract composed words in a compound word.

In the compound word lexicon, we do not mention about morpheme layer 0. Because maintaining the relationshops between layer 0 and layer 1 is cumbersome but no use for many users. We use layer 1

as basis and annotate relationships with layer 2 or layer 3.

6.3.1 Categories for compound words

To keep the relationship between longer and shorter word delimitation definition, we define the dependency structure of compound word in the lexicon, in which compound words are supposed to have binary dependency structures. Then the relationship between longer and shorter word delimitation definition are defined as parental relation on the dependency tree. We defined the classification of compound words from the dependency tree. Table 6.1 shows the categories for the classification.

Table 6.1. Category for compound words

Category	Specification
B	Basis
N	Compound
P	Parallel Compound
C	Contracted Word
X	Fragment of Compound Word
S	Collocation Pair without Dependency

Below, we exemplify the definitions of categories.

Dependency structures of compound words

We represent a compound word with the constituent words as a binary dependency structure. For an adjecent pair of words which compose a compound word , if left side constituent word has dependency with right side one, then we put category “N” for the compound word and annotate pointers to two constituent words on the compound word. When the constituent words can be also divided into constituent words, we recursively define the parental relationship for the constituent words.

To examine the descendant on the dependency structure via databases, we can discriminate whether the compound word has left or right branching structure. The discrimination is crucial information to identify layer 2 words of *UniDic* schema. Figure 6.8 exemplifies left and right branching structure of compound words.

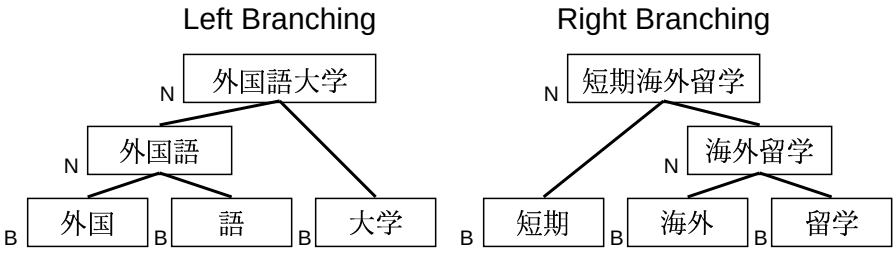

Figure 6.8. Example: left and right branching of compound words

Binarization of ternary dependency structure

Because of parallel structure, ternary structures can occur in the dependency structure of compound words. Nevertheless, we restrict within binary structures to represent compound words. Then, we extract ternary structure into left branching binary structure. The extracted nodes are put category “P”. Figure 6.9 shows binarization of ternary tree structure.

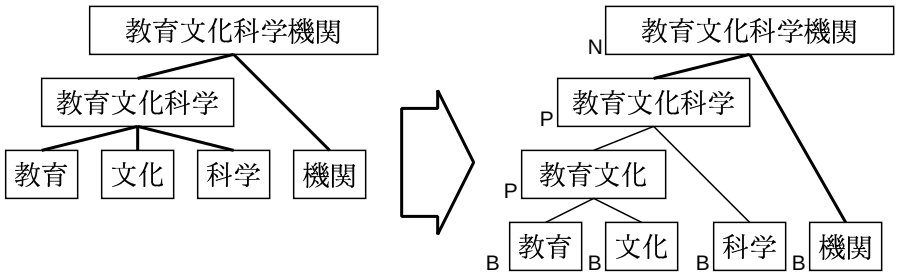

Figure 6.9. Example: binarization of ternary structure

Contracted words

Contracted words are peculiar to spoken language. Contracted expression “ちゃう” (/chau/) is composed by “て” (/te/) and “しまう” (/shimau/). Our databases treat these contracted expressions with category “C”. Figure 6.10 exemplify the expression “chau”.

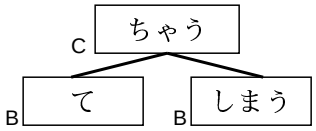

Figure 6.10. Example: contracted word “ちゃう”

Collocation, idiom and named entity

When we annotate for collocations, idioms and named entities, some constituent words may form a compound word, which we should not be registerd in any layers of word definitions. In such case, we put category label “X” for the intermediate compound words. Figure 6.11 is an example of a compound case particle. The intermediate word “ついて” is not registered in any layers.

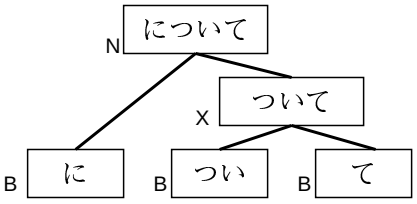

Figure 6.11. Example: compound case particle “について”

Collocation without Dependency

There are some collocation without dependency in the lexicon. These words are registed because of accuracy of Japanese morphological analysis. Figure 6.12 shows an example of numeral suffix “goushitsu”. The constituent words of this collocation have neither dependency nor parallel relationship. Nevertheless, this collocation has a trigger role of word formation for numeral expression. We put category “S” for these words.

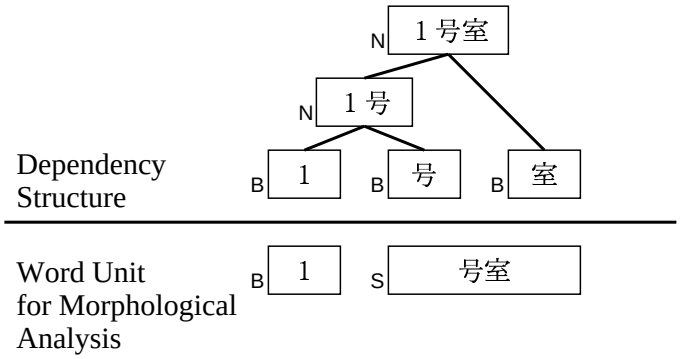

Figure 6.12. Example: suffix “号室” and the dependency structure

6.3.2 Compound word lexicon on relational database

We presented the methods to represent word relationships by pointers. We keep these pointers on our relational databases as they stand. Table 6.2 shows compounding word table on the relational databases. To simplify, we represent “Word ID” for the anchor of pointers. On our practical databases, the anchor is pair of “Word ID” and “CFORM ID” to transact Japanese conjugation.

Annotators are using GUI to annotate compound relationships. In left below or right below fields, the GUI shows up possible prefix or suffix matching words. The annotator selects most suitable segmentation by top down strategy.

Table 6.2. Compoud word lexicon on relational databases

Word ID	Category	Orthography	Left Const.	Right Const.
1	N	教育文化科学機関	2	7
2	P	教育文化科学	3	6
3	P	教育文化	4	5
4	B	教育	-	-
5	B	文化	-	-
6	B	科学	-	-
7	B	機関	-	-

6.4 Summary

We proposed a use of relational database for language resource maintenance. The method enables us to keep consistency between POS tagged corpus and lexicon. Furthermore, the method enables us to keep multiple word units with the relationships between compounds and constituents. We have not develop new POS tagged corpus. We only use the method for modification of an existing corpus. Then, the consistency of POS tagged corpus is improved. The morphological analyzer, which is based on the POS tagged corpus, is also improved.

Chapter 7

Conclusions

If a man writes a book, let him set down only what he knows.
I have guesses enough of my own.
– **Johann Wolfgang von Goethe**(1749-1832)

7.1 Summary

Our contribution by this thesis is development of the dictionary – *IPADIC* [50] – used by Japanese morphological analyzer *ChaSen* [84]. The approach we adopted here to this problem has the following characteristics.

- We divided the problem into three subproblems: models for known word, models for unknown word and corpus maintenance schema.
- Models for known word;
  We proposed three extensions for Markov model-based Japanese morphological analysis: lexicalized POS, position-wise grouping and selective trigram. The extensions covered unique phenomena in Japanese language: function words, conjugation, contracted words and fine-grained POS tagset.
- Models for unknown word;
  We proposed an offline model for unknown word processing in Japanese morphological analysis. The unknown words are extracted from large-scale unlabeled texts in advance. We newly introduced a pattern recognition method to identify unknown word occurence in the texts and introduced word sense disambiguation method to determine POSs for the extracted words.
- Corpus maintenance schema;
  We proposed a use of relational database to maintain POS tagged corpora and lexicons. The corpora and lexicons are connected by the relationships in the relation database. Therefore, the relational database enables us synchronous transaction between the lexicons and the corpora. Therefore, the risk of discrepancy in the corpus is reduced by the proposed method. We also proposed an maintenance schemata for compound word lexicon, which enable us to keep multiple definitions of word boundary.

We also discuss some related tasks: Japanese named entity extraction and morphological analysis for transcription of spoken language.

- Japanese named entity extraction;
  We focused attension on the word boundary discrepancy problem in Japanese named entity extraction. We proposed character-based chunking to solve the problem. We also introduced point-wise *n*-best answers of Japanese morphological analyzer as the features for chunker. Therefore, the chunking model becomes robust and covers the discrepancy problem.

- Morphological analysis for transcription of spoken language;
  We focused attension on fillers and disfluencies in spoken language. We proposed filler filtering for the transcriptions, which based on a pattern recognition method.

The proposed method for Japanese named entity extraction achieves the 2nd best accuracy for IREX dataset. The best method [39] is an successor, which introduce *Bunsetsu* features for our method. The idea of filler filtering is newly introduced for morphological analysis for transcription of spoken language. The proposed method helps to maintain large-scale speech corpus.

7.2 Open problems

Japanese morphological analysis is preliminary but difficult task. Although we think that the investigations repoted in this thesis present some significant progress, further research on this problem is clearly still needed. Other issues not investigated completely in this thesis and some possible solutions include:

POS guessing for unknown words

We proposed a simple method for unknown word's POS guessing. There are several sophisticated method for other languages as we reviewed in section 4.4. We should introduce such method for Japanese unknown word processing. However, the method will not solve a problem which is derived from the possibility-based POS tagset. For example, an adverbial noun "名詞-副詞可能" in IPADIC POS tagset means that the word behaves as both simple noun and adverb. If the word occurs as an adverb in the sentence, the classifier will identify the word as an adverb. If the word occurs as a simple noun, the classifier will identify that the word as a simple noun. We should consider both examples – namely adverb and simple noun – to estimate the POS of the word as an adverbial noun. One of solution will be use of unlabeled data to collect many examples in which the word is used as both types.

Building compound word lexicon

We designed a compound word lexicon and a maintenance schema for the lexicon and tagged corpus. However, we have not built up the compound lexicon yet. Even if we introduce the maintenance schema, to build the compound lexicon manually is cumbersome. Practically, unsupervised or semi-supervised method will be necessary for the problem. We would like to introduce a bootstraping method with the frequency in unlabeled data and/or the hit number on the web search engine for the problem.

Adoptation for other languages

The proposed method is aimed for Japanese. We believe that the method is also applicable for Chinese or other languages. We introduced the known word model for Chinese and English[48]. We adopt error-driven method for feature selection. We also introduced the unknown word model for Chinese [8] [49]. We would like to make a practical morphological analyzer for Chinese, too.

Fine-grained named entity hierarchy tagset

Sekine proposed a hierarchical named entity tagset [68]. The number of tags is over 150. Simple chunking methods cannot extract entities with such tags, because the tagged corpus is smaller compared with the number of the named entity tags. One possible solution is splitting the problem into two

subproblem: token extraction and tag estimation: Firstly, a chunker extracts named entity tokens with coarse-grained tags in the hiearchical structure. Then, a classifier estimates fine-grained tags. Large scale unlabeled corpora will be necessary for the classifier, since training data for the fine-grained tags is very sparce.

References

[1] A. Borthwick. *A Maximum Entropy Approach to Named Entity Recognition*. PhD thesis, New York University, 1999.

[2] A. Kitauchi and T. Utsuro and Y. Matsumoto. Probabilistic Model Learning for Japanese Morphological Analysis by Error-driven Feature Selection(in Japanese). *Transaction of Information Processing Society of Japan*, 40(5):2325–2337, 1999.

[3] A. McCallum and W. Li. Early Results for Named Entity Recognition with Conditional Random Fields. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 188–191, 2003.

[4] A. Mikheev. Automatic Rule Induction for Unknown-Word Guessing. *Computatinal Linguistics*, 23(3):405–423, 1997.

[5] A. Ratnaparkhi. A Maximum Entropy Model for Part-of-Speech Tagging. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP '96)*, pages 133–142, 1996.

[6] A. Ratnaparkhi. *Maximum Entropy Models for Natural Language Ambiguity Resolution.* PhD thesis, University of Pennsylvania, Philadelphia, PA., 1998.

[7] B. Sundheim. Overview of Results of the MUC-6 Evaluation. In *Proc. of the Sixth Message Understanding Conference (MUC-6)*, pages 13–31, 1995.

[8] C. Goh and M. Asahara and Y. Matsumoto. Chinese Unknown Word Identification Using Position Tagging and Chunking. In *41st Annual Meeting of the Association for Computational Linguistics, Interactive Poster/Demo Sessions, Companion volume of the Proc. (ACL 2003)*, pages 197–200, 2003.

[9] D. Cutting and J. Kupiec and J. Pedersen and P. Sibun. A Practical Part-of-Speech Tagger. In *Proc. of the Third Conference on Applied Natural Language Processing (ANLP '92)*, pages 133–140, 1992.

[10] D. D. Palmer and D. S. Day. A Statistical Profile of the Named Entity Task. In *Proc. of Fifth ACL Conference on Applied Natural Language Processing (ANLP '97)*, pages 190–193, 1997.

[11] D. Klein and J. Smarr and H. Nguyen and C. D. Manning. Named Entity Recognition with Character-Lever Models. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 180–183, 2003.

[12] D. Ron and Y. Singer and N. Tishby. Learning Probabilistic Automata with Variable Memory Length. In *Proc. of the Seventh Annual ACM Conference on Computational Learning Theory (COLT '94)*, pages 35–46, 1994.

[13] D. Roth and D. Zelenko. Part of Speech Tagging Using a Network of Linear Separators. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Proc. of the Conference (COLING-ACL '98)*, pages 1136–1142, 1998.

[14] D. Wu and G. Ngai and M. Carpuat. A Stacked, Voted, Stacked Model for Named Entity Recognition. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 200–203, 2003.

[15] E. Brill. A simple rule-based part of speech tagger. In *Proc. of the Third Conference on Applied Natural Language Processing (ANLP '92)*, pages 152–155, 1992.

[16] E. Brill. Transformation-Based Error-Driven Learning and Natural Language Processing: A Case Study in Part-of-Speech Tagging. *Computational Linguistics*, 21(4):543–565, 1995.

[17] E. F. Tjong Kim Sang. Introduction to the CoNLL-2002 Shared Task: Language-Independent Named Entity Recognition. In *Proc. of Sixth Conference on Natural Language Learning (CoNLL-2002)*, pages 155–158, 2002.

[18] E. F. Tjong Kim Sang and F. De Meulder. Introduction to the CoNLL-2003 Shared Task: Language-Independent Named Entity Recognition. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 142–147, 2003.

[19] F. De Meulder and W. Daelemans. Memory-Based Named Entity Recognition using Unannotated Data. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 208–211, 2003.

[20] F. Jelinek. *Statistical Methods For Speech Recognition*. The MIT Press, 1998.

[21] F. Sha and F. Pereira. Shallow Parsing with Conditional Random Fields. In *Proc. of Human Language Technology and North American Chapter of Association for Computational Linguistics (HLT-NAACL 2003)*, pages 134–141, 2003.

[22] G. Orphanos and D. Christodoulakis. POS Disambiguation and Unknown Word Guessing with Decision Trees. In *Proc. of the Ninth Conference of the European Chapter of the Association for Computational Linguistics (EACL-99)*, pages 134–141, 1999.

[23] H. Isozaki and H. Kazawa. Efficient Support Vector Classifiers for Named Entity Recognition. In *Proc. of the 19th International Conference on Computational Linguistics (COLING 2002)*, pages 390–396, 2002.

[24] H. Kubozono. *The Organization of Japanese Prosody. Studies in Japanese Linguistics.* Kurosio Publishers, Inc., 1991.

[25] H. S. Thompson and D. McKelvie. Hyperlink semantics for standoff markup of read-only documents. In *Proc. of SGML Europe '97*, 1997.

[26] H. Schütze and Y. Singer. Part-of-Speech Tagging Using A Variable Memory Markov Model. In *Proc. of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL '94)*, pages 181–187, 1994.

[27] H. Schmid. Probabilistic Part-of-Speech Taggin Using Decision Trees. In *Proc. of the International Conference on New Methods in Language Processing(NeMLaP-1)*, pages 44–49, 1994.

[28] H. Schmid. Improvements In Part-of-Speech Tagging With an Application To German. In *Proc. of EACL SIGDAT workshop*, pages 47–50, 1995.

[29] H. Yamada and T. Kudo and Y. Matsumoto. Japanese Named Entity Extraction Using Support Vector Machine (in Japanese). *Transaction of Information Processing Society of Japan*, 43(1):44–53, 2002.

[30] I. Hendrickx and A. van den Bosch. Memory-based one-step named-entity recognition: Effects of seed list features, classifier stacking and unannotated data. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 176–179, 2003.

[31] IREX Committee, editor. Proc. of the IREX workshop, 1999.

[32] J. D. Kim and S. Lee and H. Rim. HMM Specialzation with Selective Lexicalization. In *the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP-VLC '99)*, pages 121–127, 1999.

[33] J. Kazama and Y. Miyao and J. Tsujii. A Maximum Entropy Tagger with Unsupervised Hidden Markov Models. In *Proc. of the Sixth Natural Language Processing Pacific Rim Symposium (NLPRS 2001)*, pages 333–340, 2001.

[34] J. Lafferty and A. McCallum and F. Pereira. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proc. 18th International Conf. on Machine Learning (ICML 2001)*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[35] J. Mayfield and P. McNamee and C. Piatko. Named Entity Recognition using Hundreds of Thousands of Features. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 184–187, 2003.

[36] Japan Electronic Dictionary Research Institute. EDR Electronic Dictionary User's Manual (in Japanese), 1994.

[37] K. Chen and W. Ma. Unknown Word Extraction for Chinese Documents. In *Proc. of the 19th International Conference on Computational Linguistics (COLING 2002)*, pages 169–175, 2002.

[38] K. Maekawa and H. Koiso and S. Furui and H. Isahara. Spontaneous Speech Corpus of Japanese. In *Proc. of Second International Conference on Language Resource and Evaluation (LREC 2000)*, pages 947–952, 2000.

[39] K. Nakano and Y. Hirai. Japanese Named Entity Extraction Using Bunsetsu Feature (in Japanese). In *IPSJ SIG Notes NL-156*, pages 7–14, 2003.

[40] K. Takeuchi and N. Collier. Use of support vector machines in extended named entity. In *Proc. of Sixth Conference on Natural Language Learning (CoNLL-2002)*, pages 119–125, 2002.

[41] K. Uchimoto and C. Nobata and A. Yamada and S. Sekine and H. Isahara. Morphological Analysis of the Spontaneous Speech Corpus. In *Proc. of the 19th International Conference on Computational Linguistics (COLING 2002)*, pages 1298–1302, 2002.

[42] K. Uchimoto and C. Nobata and A. Yamada and S. Sekine and H. Isahara. Morphological Analysis of a Large Spontaneous Speech Corpus in Japanese. In *Proc. of 41st Annual Meeting of the Association for Computational Linguistics (ACL 2003)*, pages 479–488, 2003.

[43] K. Uchimoto and Q. Ma and M. Murata and H. Ozaku and M. Utiyama and H. Isahara. Named entity extraction based on a maximum entropy model and transformation rules (in Japanese). *Journal of Natural Language Processing*, 7(2):63–90, 2000.

[44] K. Uchimoto and S. Sekine and H. Isahara. The Unknown Word Problem: a Morphological Analysis of Japanese Using Maximum Entropy Aided by a Dictionary. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2001)*, pages 91–99, 2001.

[45] K. W. Church. A stochastic parts program and noun phrase parser for unrestricted text. In *Proc. of The Second Conference on Applied Natural Language Processing (ANLP '88)*, pages 136–143, 1988.

[46] K. Zhang and Q. Liu and H. Zhang and X. Cheng. Automatic Recognition of Chinese Unknown Words Based on Roles Tagging. In *Proc. of First SIGHAN Workshop on Chinese Language Processing*, pages 71–77, 2002.

[47] L. A. Ramshaw and M. P. Marcus. Text chunking using transformation-bases learning. In *Proc. of the third Workshop on Very Large Corpora (VLC '95)*, pages 83–94, 1995.

[48] M. Asahara. Extended Statistical Model for Morphological Analysis. Master's thesis, Nara Institute of Science and Technology, Japan, 2000.

[49] M. Asahara and C. Goh and X. Wang and Y. Matsumoto. Combining Segmenter and Chunker for Chinese Word Segmentation. In *Proc. of Second SIGHAN Workshop on Chinese Language (Bakeoff papers)*, pages 144–147, 2003.

[50] M. Asahara and Y. Matsumoto. *IPADIC Users Manual*. Nara Institute of Science and Technology, Japan, 2003.

[51] M. Ikeya and H. Shinnou. 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, 2000.

[52] M. Munoz and V. Punyakanok and D. Roth and D. Zimak. A learning approach to shallow parsing. In *the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP-VLC '99)*, pages 168–178, 1999.

[53] M. Nagata. A Part of Speech Estimation Method for Japanese Unknown Words using a Statistical Model of Morphology and Context. In *37th Annual Meeting of the Association for Computational Linguistics (ACL '99)*, pages 277–284, 1999.

[54] N. A. Chinchor. Overview of MUC-7/MET-2. In *Proc. of Message Understanding Conference MUC-7*, 1999.

[55] N. Chomsky. Three models for the description of language. *IRE trans. on Information Theory*, 2(3):113–124, 1956.

[56] N. Chomsky. On certain formal properties of grammars. *Information and Control*, 2(2):137–167, 1959.

[57] N. Ide and P. Bonhomme and L.Romary. XCES: An XML-based Encoding Standard for Linguistic Corpora, 2000.

[58] O. Bender and F. J. Och and H. Ney. Maximum Entropy Models for Named Entity Recoginition. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 148–151, 2003.

[59] R. Munro and D. Ler and J. Patrick. Meta-Learning Orthographic and Contextual Models for Language Independent Named Entity Recognition. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 192–195, 2003.

[60] R. Weischedel and M. Meteer and R. Schwartz and L. Ramshaw and J. Palmucci. Coping with Ambiguity and Unknown Words through Probabilistic Models. *Computatinal Linguistics*, 19(2):359–382, 1993.

[61] Real World Computing Partnership. RWC Text Database, 1995. `http://www.rwcp.or.jp/wswg/rwcdb/text/`.

[62] S. Bird and M. Liberman. A Formal Framework for Linguistic Annotation. *Speech Communication*, 33:23–60, 2001.

[63] S. DeRose and E. Maler and R. Daniel Jr. XML Pointer Language (XPointer) Version 1.0, 2001. http://www.w3.org/TR/xptr/.

[64] S. Ikehara and M. Miyazaki and S. Shirai and A. Yokoo and H. Nakaiwa and K. Ogura and Y. Ooyama and Y. Hayashi. *Goi-Taikei – A Japanese Lexicon CDROM*. Iwanami Shoten, Tokyo, 1999.

[65] S. Kurohashi and M. Nagao. Building a Japanese Parsed Corpus while Improving the Parsing System. In *Proc. of The First International Conference on Language Resources and Evaluation (LREC 1998)*, pages 719–724, 1998.

[66] S. Kurohashi and T. Nakamura and Y. Matsumoto and M. Nagao. Improvements of Japanese Morphological Analyser JUMAN. In *Proc. of International Workshop on Sharable Natural Language Resources*, pages 22–28, 1994.

[67] S. Mori and M. Nagao. Word Extraction from Corpora and Its Part-of-Speech Estimation Using Distributional Analysis. In *The 16th International Conference on Computational Linguistics (COLING '96)*, volume 2, pages 1119–1122, 1996.

[68] S. Sekine and K. Sudo and C. Nobata. Extended Named Entity Hierarchy. In *Proc. of the Third International Conference on Language Resource and Evaluation (LREC 2002)*, pages 1818–1824, 2002.

[69] S. Sekine and Y. Eriguchi. Japanese Named Entity Extraction Evaluation - Analysis of Results. In *Proc. of the 18th International Conference on Computational Linguistics (COLING 2000)*, pages 25–30, 2000.

[70] S. Thede. Predicting Part-of-Speech Information abou Unknown Words using Statistical Methods. In *36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics, Proc. of the Conference (COLING-ACL '98)*, pages 1505–1507, 1998.

[71] T. Brants. TnT – a statistical part-of-speech tagger. In *Proc. of the Sixth Conference on Applied Natural Language Processing, (ANLP-2000)*, pages 224–231, 2000.

[72] T. Kudo and Y. Matsumoto. Chunking with Support Vector Machines. In *Proc. of the Second Meeting of North American Chapter of Association for Computational Linguistics (NAACL-2001)*, pages 192–199, 2001.

[73] T. Masuoka and Y. Takubo. *Kiso Nihongo Bunpou – kaitei-ban –*. Kurosio Publishers, Inc, 1992.

[74] T. Nakagawa and T. Kudo and Y. Matsumoto. Unknown Word Guessing and Part-of-Speech Tagging Using Support Vector Machines. In *Proc. of the Sixth Natural Language Processing Pacific Rim Symposium (NLPRS 2001)*, pages 325–331, 2001.

[75] T. Nakagawa and T. Kudo and Y. Matsumoto. Revision Learning and its Application to Part-of-Speech Tagging. In *40th Annual Meeting of the Association for Computational Linguistics, Proc. of the Conference (ACL 2002)*, pages 497–504, 2002.

[76] T. Utsuro and M. Sassano and K. Uchimoto. Combining Outputs of Multiple Japanese Named Entity Chunkers by Stacking. In *Proc. of the Conference on Empirical Methods in Natural Language Processing (EMNLP 2002)*, pages 281–288, 2002.

[77] T. Zhang and D. E. Johnson. A robust risk minimization based named entity recognition system. In *Proc. of Seventh Conference on Natural Language Learning (CoNLL-2003)*, pages 204–207, 2003.

[78] T. Zhang and F. Damerau and D. E. Johnson. Text Chunking using Regularized Winnow. In *Proc. of the 39th Annual Meeting of the Association for Computational Linguistics (ACL 2001)*, pages 539–546, 2001.

[79] T. Zhang and F. Damerau and D. E. Johnson. Text chunking based on a generalization of Winnow. *Journal of Machine Learning Research*, 2:615–637, 2002.

[80] V. N. Vapnik. *Statistical Learning Theory*. A Wiley-Interscience Publication, 1998.

[81] W. Daelemans and J. Zavrel and K. van der Sloot and A. van den Bosch. MBT: Memory-Based Tagger, version 1.0, Reference Guide. Technical Report ILK Technical Report ILK-0209, University of Tilburg, The Netherlands, 2002.

[82] W. Daelemans and J. Zavrel and S. Berck. MBT: A MemoryBased Part of Speech Tagger-Generator. In *Proc. of the Fourth Workshop on Very Large Corpora (VLC '96)*, pages 14–27, 1996.

[83] Y. Den and T. Utsuro and A. Yamada and M. Asahara and Y. Matsumoto. Design of an Electric Dictionary Suitable for Spoken Language Research (In Japanese). In *Proc. of the Second Spontaneous Speech Science and Technology Workshop*, pages 39–46, 2002.

[84] Y. Matsumoto and A. Kitauchi and T. Yamashita and Y. Hirano and H. Matsuda and K. Takaoka and M. Asahara. Morphological Analysis System ChaSen version 2.3.3 Manual. Technical report, Nara Institute of Science and Technology, 2003.

[85] Y. Matsumoto and Y. Den. Morphological Analysis of Spoken Language (In Japanese). In *IPSJ SIG Notes NL-143*, pages 49–54, 2001.

[86] Y. Takemoto and T. Fukushima and H. Yamada. A Japanese Named Entity Extraction System Based on Building a Large-scale and High-quality Dictionary and Pattern-maching Rules. *Transaction of Information Processing Society of Japan*, 42(6):1580–1591, 2001.

Appendix

Chapter A

IPA POS tagset

名詞-一般	Noun, General
名詞-固有名詞-一般	Noun, Proper Noun, General
名詞-固有名詞-人名-一般	Noun, Proper Noun, Name of a person, General
名詞-固有名詞-人名-姓	Noun, Proper Noun, Name of a person, Family name(for Japanese)
名詞-固有名詞-人名-名	Noun, Proper Noun, Name of a person, First name(for Japanese)
名詞-固有名詞-組織	Noun, Proper Noun, Organization
名詞-固有名詞-地域-一般	Noun, Proper Noun, Name of a place, General(Excluding Name of a country)
名詞-固有名詞-地域-国	Noun, Proper Noun, Name of a place, Name of a country
名詞-代名詞-一般	Noun, Pronoun, General
名詞-代名詞-縮約	Noun, Pronoun, Contracted Form
名詞-副詞可能	Noun, Adverbial
名詞-サ変接続	Noun, Verbal(“suru”)
名詞-形容動詞語幹	Noun, Adjective Noun(“na”)
名詞-ナイ形容詞語幹	Noun, Adjective Noun(“nai”)
名詞-数	Noun, Number
名詞-非自立-一般	Noun, Dependent, General
名詞-非自立-副詞可能	Noun, Dependent, Adverbial
名詞-非自立-助動詞語幹	Noun, Dependent, a stem of an auxiliary verb
名詞-非自立-形容動詞語幹	Noun, Dependent, Adjective Noun(“na”)
名詞-特殊-助動詞語幹	Noun, Abnormal, a stem of an auxiliary verb
名詞-接尾-一般	Noun, Suffix, General
名詞-接尾-人名	Noun, Suffix, Name of a person
名詞-接尾-地域	Noun, Suffix, Name of a place
名詞-接尾-サ変接続	Noun, Suffix, Verbal(“suru”)
名詞-接尾-助動詞語幹	Noun, Suffix, a stem of an auxiliary verb
名詞-接尾-形容動詞語幹	Noun, Suffix, Adjective Noun(“na”)
名詞-接尾-副詞可能	Noun, Suffix, Adverbial
名詞-接尾-助数詞	Noun, Suffix, Auxiliary numeral
名詞-接尾-特殊	Noun, Suffix, Abnormal
名詞-接続詞的	Noun, Conjuncitive
名詞-動詞非自立的	Noun, Verbal(dependent)
接頭詞-名詞接続	Prefix Particle, adjunct to noun
接頭詞-数接続	Prefix Particle, adjunct to number
接頭詞-動詞接続	Prefix Particle, adjunct to verb
接頭詞-形容詞接続	Prefix Particle, adjunct to adjective

動詞-自立	Verb, Independent
動詞-非自立	Verb, Dependent
動詞-接尾	Verb, Suffi x
形容詞-自立	Adjective, Independent
形容詞-非自立	Adjective, Dependent
形容詞-接尾	Adjective, Suffi x
副詞-一般	Adverb, General
副詞-助詞類接続	Adverb, adjunct to postposition
助詞-格助詞-一般	Postposition, Case, General
助詞-格助詞-引用	Postposition, Case, Quotation
助詞-格助詞-連語	Postposition, Case, Compound words
助詞-係助詞	Postposition, Dependent
助詞-副助詞	Postposition, Adverbial
助詞-並立助詞	Postposition, Parallel
助詞-終助詞	Postposition, Sentence End
助詞-副助詞／並立助詞／終助詞	Postposition, Adverbial/Parallel/Sentence End
助詞-連体化	Postposition, Substantive
助詞-副詞化	Postposition, adjunct to onomatopée
助詞-特殊	Postposition, Abnormal
助詞-間投助詞	Postposition, Interjectional
助動詞	Auxiliary Verb
感動詞	Interjection
記号-一般	Symbol, General
記号-アルファベット	Symbol, Alphabet
記号-句点	Symbol, Period
記号-読点	Symbol, Comma
記号-空白	Symbol, Space
記号-括弧開	Symbol, Open parenthesis
記号-括弧閉	Symbol, Close parenthesis
フィラー	Filler
その他-間投	Others, Interjectional

Chapter B

Data for Unknown Word Processing

In this chapter, we present about the method to define unknown words in the text. There is no standard data for the unknown word processing. Low frequent words are excluded from lexicon and defined as unknown words in most of preceding works. In this thesis, we define the unknown words by the following two sorts of criteria.

- the word frequency in the corpus
- the hit number on the web search engine

Since the unknown word is defined by the frequency or the hit number, the unknown word contains both simple word and compounding word.

B.1 Criteria: the word frequency in the corpus

Most of preceding works define the unknown words based on the word frequency in the corpus. We can classify their methods based on the presence of the lexicon.

One method is based on only corpora. The word, whose frequency in the corpora is one, is defined as the unknown word. The word, whose frequency in the corpora is more than one, is defined as the known word. Nagata [53] introduced the method for their experiments. We call "freq. 1 w/o lex" in this thesis.

The other method is based on corpora and lexicon. The unknown word is defined by the frequency in the corpora. The known word is defined as a large-scale lexicon in which the defined unknown words are excluded. Uchimoto [44] introduced the method for their experiments. We call "freq. 1 w/ lex" in this thesis.

Both methods are reproducible settings. However, we cannot say that the methods models the real situation.

B.2 Criteria: the hit number on the web search engine

When we already have a large-scape lexicon, we cannot prepare enough size of corpora to estimate frequencies of all words in the lexicon. Then, we introduce the web search engines for the frequency estimation. We use the hit number on the search engine as the frequency of a word. We define the unknown words according to the hit number. We use a web search engine *goo* [1]. We use *ipadic* [50][2] as a lexicon and RWCP text corpus [61] as the corpus.

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

[2] The lexicon contains about 240,000 entries by the stem form.

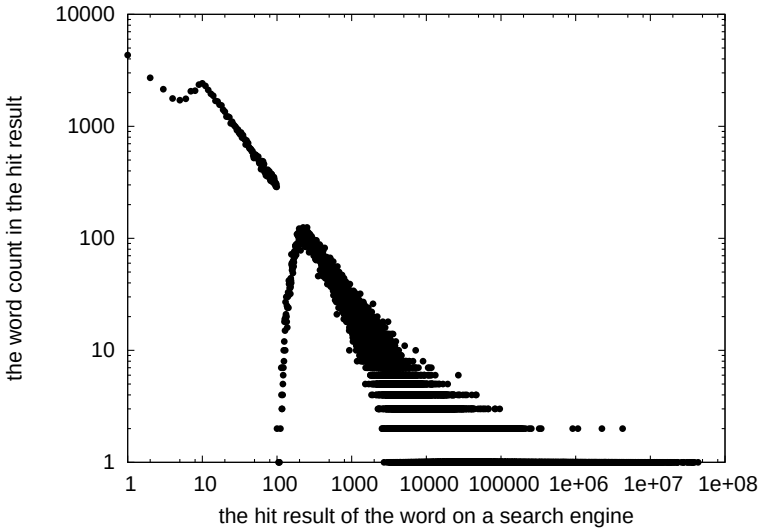

Figure B.1. The distribution of the hit number

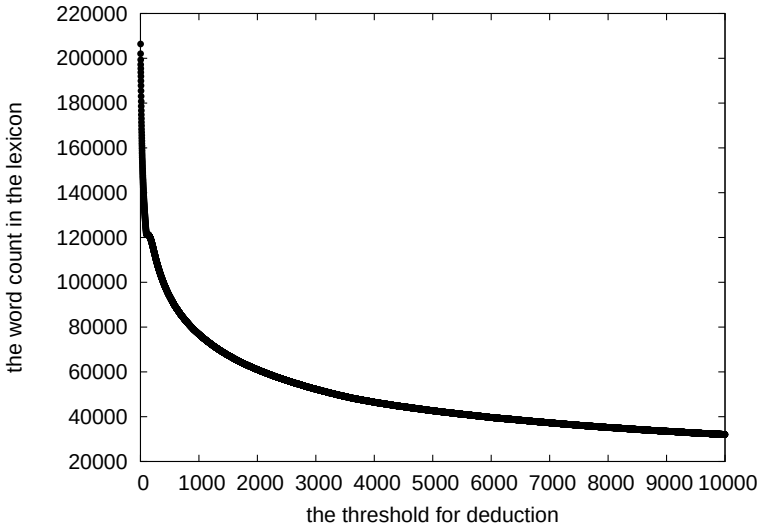

Figure B.2. The threshold for deduction and the word count in the lexicon

Figure B.1 shows the distribution of the hit numbers of the words in *ipadic*. X-axis presents the hit number and Y-axis presents the number of word according to the hit number on the X-axis. Note that, the number of word is degraded at the hit number 100. The reason will be that the round up is begun at the hit number.

We decrease the entries in the lexicon according to the hit numbers. The word with the lower hit number is removed from the lexicon in first. Figure B.2 shows the relation between the hit number and the number of the entries in the lexicon. Figure B.3 shows the relation between the hit number and the known word rate in the corpus. Note that, we do not remove the word with the hit number 0, because the word might be stop word on the search engine.

When we remove the word whose the hit number is less than 1001, the size of lexicon becomes about

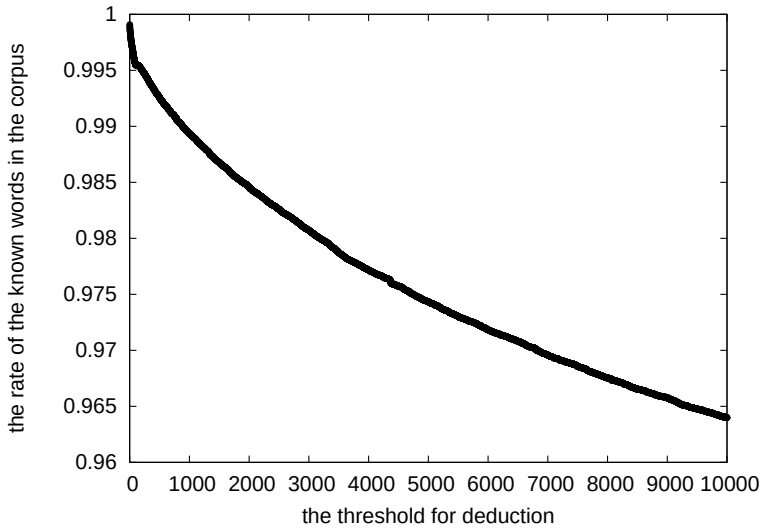

Figure B.3. The threshold for deduction and the rate of the known words

1/3 and the known word rate in the corpus becomes about 99%. When we remove the word whose the hit number is less than 10001, the size of lexicon becomes about 1/6 and the known word rate in the corpus becomes about 96%.

We call the dataset with the threshold 1000 as “goo 1000” and the dataset with the threshold 10000 as “goo 10000”. When we do 5-fold cross validation over these two datasets, simple data splitting causes the problem that an unknown word can be stored in both training and test data. Then, firstly, we make sentence groups which contain same unknown words. Secondly, we compose 5 data whose sizes are nealy same. Thirdly, we do cross validation based on the data.

This setting is not reproducible. However, the setting reflects the actual situation much than the preceding works.

List of Publications

Major Publications

Journal Papers

(1) M. Asahara, R. Yoneda, A. Yamashita, Y. Den, and Y. Matsumoto, “Maintenance schema of Japanese POS tagged corpora” (in Japanese) Transaction of Information Processing Society of Japan, Vol.43 No.07, pp.2091-2097, July 2002.

(2) M. Asahara, and Y. Matsumoto, “Extended Statistical Model for Morphological Analysis” (in Japanese) Transaction of Information Processing Society of Japan, Vol.43 No.03, pp.685-695, March 2002.

International Conferences (Reviewed)

(3) M. Asahara, and Y. Matsumoto, “Japanese Named Entity Extraction with Redundant Morphological Analysis” In Proc. of Human Language Technology and North American Chapter of Association for Computational Linguistics (HLT-NAACL 2003), pp.8–15, May 2003.

(4) M. Asahara, and Y. Matsumoto, “Filler and Disfluency Identification Based on Morphological Analysis and Chunking” In Proc. of ISCA and IEEE Workshop on Spontaneous Speech Processing and Recognition, Tokyo, Japan, pp.163–166, April 2003.

(5) M. Asahara, R. Yoneda, A. Yamashita, Y. Den, and Y. Matsumoto, “Use of XML and Relational Databases for Consistent Development and Maintenance of Lexicons and Annotated Corpora” In Proc. of the Third International Conference on Language Resource and Evaluation (LREC 2002), Las Palmas, Canary Islands, Spain, pp.1372–1378, May 2002.

(6) M. Asahara, and Y. Matsumoto, “Extended Models and Tools for High-performance Part-of-Speech Tagger” In Proc. of the 18th International Conference on Computational Linguistics (COLING 2000), Saarbrücken, Germany, pp.21–27, July 2000.

International Workshops

(7) M. Asahara, C. Goh, X. Wang, and Y. Matsumoto, “Combining Segmenter and Chunker for Chinese Word Segmentation” Proc. of Second SIGHAN Workshop on Chinese Language (Bakeoff papers), pp.144–147, July 2003.

(8) M. Asahara, R. Yoneda, and Y. Matsumoto, “Use of a Relational Database in the Development and Maintenance of Linguistic Resources for Statistical Japanese Morphological Analysis” In Proc. of the IRCS Workshop on Linguistic Databases, Philadelphia, pp.24–31, December 2001.

(9) M. Asahara, R. Yoneda, H. Matsuda, Y. Tsuboi, K. Takaoka, and Y. Matsumoto, “Corpus Maintenance Schema for Statistical Morphological Analysis of Chinese and Japanese (in Japanese)”, Chunichi Taiyaku Koopasu no Kouchiku to Ouyou Kenkyuu, International Symposium, September 24, 2001.

Local Workshops (Domestic)

(10) M. Asahara, and Y. Matsumoto, “Filler and Disfluency Identification Based on Morphological Analysis and Chunking” Gengo-shori-gakkai Nenzi Taikai Ronbun-shu, pp.651–654, March 2003.

(11) M. Asahara, and Y. Matsumoto, “Unknown Word Identification in Japanese Text Based on Morphological Analysis and Chunking” (in Japanese) IPSJ SIG Notes, 2002-NL-154, pp.47–54, March 2003.

(12) M. Asahara, and Y. Matsumoto, “Japanese Named Entity Extraction with Redundant Morphological Analysis” (in Japanese) IPSJ SIG Notes, 2002-NL-153, pp.49–56, January 2003.

(13) M. Asahara, R. Yoneda, A. Yamashita, Y. Den, and Y. Matsumoto, “Maintenance schema of Japanese POS tagged corpus based on Relational Databases” (in Japanese) JSAI, SIG-SLUD-A103, pp.7–12, March, 2002.

(14) M. Asahara, and Y. Matsumoto, “Error-driven extensions of Statistical Learning Models for POS tagging” (in Japanese) IPSJ SIG Notes, 2000-NL-139, pp.25–32, September 2000.

(15) M. Asahara, and Y. Matsumoto, “Extended Hidden Markov Model for Japanese Morphological Analyzer” (in Japanese) IPSJ SIG Notes, 2000-NL-137, pp.39–46, June 2000.

(16) M. Asahara, and Y. Matsumoto, “An integrated method: *Bunsetsu* chunking and morphological analysis (in Japanese)” Gengo-shori-gakkai Nenzi Taikai Ronbun-shu, pp.505–508, March 1999.

Book Chapter

(17) 浅原 正幸，米田 隆一，松田 寛，坪井 祐太，高岡 一馬，松本 裕治.「統計的中日形態素解析のための品詞タグつきコーパス管理システム」「中日対訳語料庫的研制与応用研究論文集」徐一平, 曹大峰 (編), 外語教学与研究出版社, 北京, pp.84–106, September 2002.

Award

(18) 平成 15 年度 情報処理学会 山下記念研究賞,「日本語固有表現抽出における冗長的な形態素解析の利用」“Japanese Named Entity Extraction with Redundant Morphological Analysis” (in Japanese) IPSJ SIG Notes, 2002-NL-153, pp.49–56, January 2003.

Other Publications

International Conferences (Reviewed)

(19) C. Goh, M. Asahara, and Y. Matsumoto, “Chinese Unknown Word Identification Using Position Tagging and Chunking” ACL 2003: 41st Annual Meeting of the Association for Computational Linguistics, Interactive Poster/Demo Sessions, Companion volume of the Proceesings, pp.197–200, July 2003.

Local Workshops (Domestic)

(20) C. Goh, M. Asahara, and Y. Matsumoto, “Chinese Unknown Word Identification Based on Morphological Analysis and Chunking” IPSJ SIG Notes, 2003-NL-155, pp.7-12, May 2003.

(21) Y. Den, T. Utsuro, A. Yamada, M. Asahara, and Y. Matsumoto, “Design of an Electric Dictionary Suitable for Spoken Language Research (In Japanese)” Proc. of the Second Spontaneous Speech Science and Technology Workshop, pp. 39-46, 2002

(22) Y. Den, and M. Asahara, “An Integrated Environment for Maintaining Various Language Resources Built on Relational DatabaseDesign of an Electric Dictionary Suitable for Spoken Language(In Japanese)” Proc. of the First Spontaneous Speech Science and Technology Workshop, pp. 39-46, 2001