Basic Japanese Text Analysis Technology as a Platform for Knowledge Extraction

Kenji Imamura†, *Kuniko Saito, and Hisako Asano*

Abstract

In this article, we introduce a framework for the basic analysis of text content, with particular regard to the mechanisms of morphological analysis, named entity recognition, and dependency parsing of the Japanese language. These text-analysis tools are needed to gather useful knowledge and information from the abundant text resources available on the Internet.

1. Three text analysis techniques

Although information circulating on the Internet can take many different forms such as images and audio content, it is centered on text data such as HTML (hypertext markup language) and word processor documents. To effectively utilize the information contained in this text data, we first need to analyze what is written in the text. The techniques introduced in this article are basic ones that can be applied to analyze and utilize the information contained not only in documents on the Internet but also in ordinary text data. An overview of these techniques is shown in **Fig. 1**.

1.1 Morphological analysis

Because a computer sees text simply as a sequence of individual characters, we must first determine which part of the text corresponds to which word. Morphological analysis is the process of segmenting the input text into words. Additional information about words such as their parts of speech and their pronunciations are simultaneously supplied.

1.2 Named entity recognition

Even when a text has been segmented into words, there are many cases where a collection of multiple words has a special meaning. Named entity recognition is the process of extracting the names of persons, locations, organizations, and the like (called *named entities*) from word sequences. Because named entities are often keywords, which are strongly connected with the topic of the text content, they are useful for information retrieval.

1.3 Dependency parsing

To grasp the semantic information contained in a text, such as who did what, one must ascertain the structure of the text. Dependency parsing is the process of analyzing the syntactic structure of Japanese sentences by looking at the modifying relationship between phrases.

In the following sections, we introduce the frameworks of these techniques and our implementations of them.

2. Morphological analyzer: JTAG

In languages like Japanese, there are no spaces between words. Therefore, it is necessary to figure out which words are present in the text and where these words begin and end. In morphological analysis, words that are already known are kept in a dictionary. This dictionary is consulted to segment the input text into the most likely word strings. If additional information is included in the dictionary, such as parts of speech (e.g., noun or verb and the type of word) and word pronunciations, then various types of information can be associated with each word.

In general, morphological analysis involves sepa-

† NTT Cyber Space Laboratories
Yokosuka-shi, 239-0847 Japan
Email: imamura.kenji@lab.ntt.co.jp

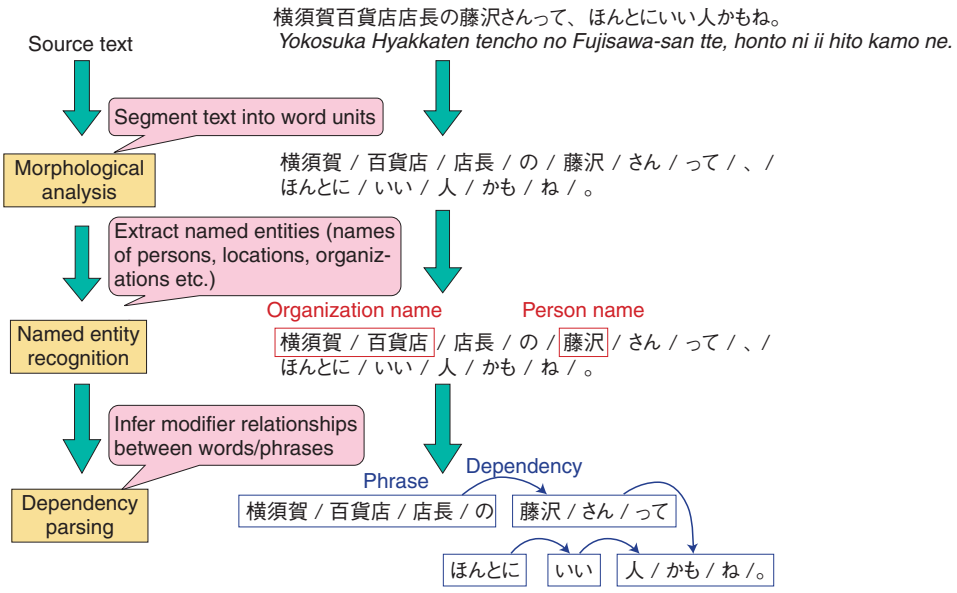

Fig. 1. Overview of basic text analysis technology.

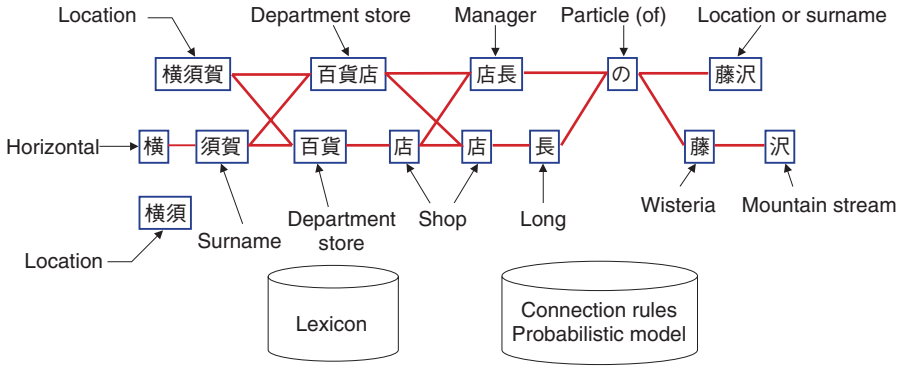

Fig. 2. Example of morphological analysis.

rating the text into words while obtaining word candidates from a dictionary. It also checks whether word candidates can be connected to each other and evaluates the cost of word connection. An example is shown in **Fig. 2**. There are two types of analyses: in one, the connection possibilities are expressed by rules and in the other, they are expressed by a probabilistic model (described below).

The morphological analyzer JTAG that we have developed [1] is a rule-based analyzer. The dictionary contains not only parts of speech and word pronunciations, but also semantic categories based on the Japanese lexicon Nihongo GoiTaikei [2]. It includes entries for approximately 900,000 words, but can nevertheless operate at high speed to process huge amounts of text.

While JTAG is designed specifically for Japanese morphological analysis, the multilingual morphological analyzer developed at Cyber Space Laboratories can also analyze other languages that have no gaps between words, such as Chinese and Korean (as well as English). It performs highly accurate analysis by using a probabilistic model [3].

3. Named entity recognizer: NameLister

When a document has been segmented into individual words, it can often contain clusters of multiple words that have a special meaning. For example, the individual meanings of the three words 日本 (*nihon*),

Table 1. Type of named entities.

Types	Meaning, examples
Person	Surnames, first names, nicknames, etc.
Location	Names of countries, addresses, building names, etc.
Organization	Organizations such as companies and social groups
Artifact	Product names, titles of films, books, etc.
Date	“2008”, “January 1st”, “last year”, etc.
Time	“8:30”, “half past eight”, etc.
Money	“200 yen”, “$3000”, etc.
Percent	“30%”, “50%”, etc.

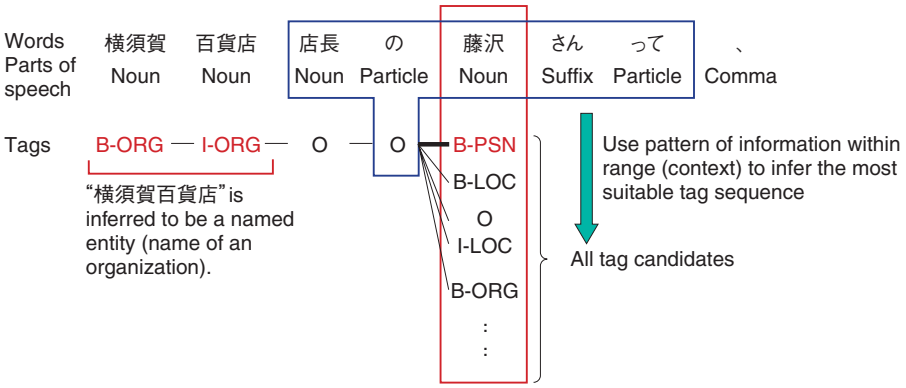

Fig. 3. Searching for named entity sequences.

電信 (*denshin*), and 電話 (*denwa*) are Japan, telegraph, and telephone, respectively. However, when strung together as 日本電信電話, they form the company name NTT. Because there is no limit to the number of patterns that can be formed by combining words, these require additional processing besides morphological analysis.

Our named entity recognizer NameLister extracts eight different types of named entity, as listed in **Table 1** [4]. These named entities are likely to constitute keywords, which are strongly related to the subject of the text, and hence provide essential information for search purposes.

To extract named entities from a sequence of words, NameLister introduces the concept of *tags*. The starting words of named entities are given beginning-mark tags (B-type name), the second and subsequent words are given continuation-mark (intermediate) tags (I-type name), and words that are not part of named entities are given other (O) tags. Introducing the concept of tags lets us extract named entities if the most suitable tags are applied to each word. This is called *sequential tagging* (or *sequential labeling*).

The sequential tagging of NameLister is performed in such a way that the most optimal tag sequence is searched for from all combinations of tags using a probabilistic model called Conditional Random Fields (CRF) [5]. This probabilistic model is built from a certain quantity of correctly tagged text (corpus) by automatically learning the appearing patterns of named entities. As a simple example, in the pattern XXさん (XX-*san*), it is highly likely that the characters represented by XX correspond to a person’s name, while the characters 鈴木 (*suzuki*) are most likely to correspond to a person’s name regardless of the context. The probabilistic model is a numerical expression of these likelihoods. In languages, however, there is almost no limit to the number of such patterns, so it is impossible to cover all possible names. By automatically learning from real examples, NameLister produces numerical values for subtle likelihoods that are not noticed by humans.

Named entity recognition is implemented by searching for the most likely tag sequences from among all combinations of tags while consulting the probabilistic model. An example of such a search is shown in **Fig. 3**. This figure illustrates the tagging of the word 藤沢 (*fujisawa*). The words that appear

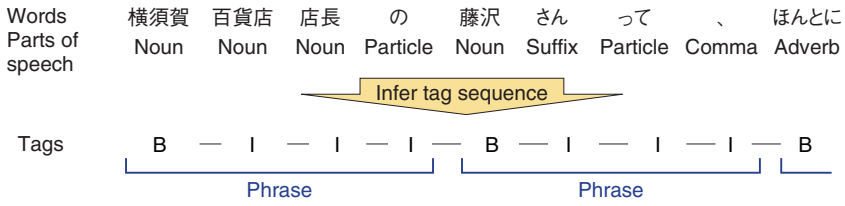

Fig. 4. Phrase chunking using sequential tagging.

before and after it (i.e., the context) are used to calculate the likelihood of each tag, and the most likely tag is then selected. In this example, the characters さん (*-san*) appear immediately after *fujisawa* and the character の (*no*) appears immediately before, so the tag with the greatest likelihood is that of a word starting a personal name (B-PSN). Because the optimal tag sequence is determined by taking the context into consideration, the word *fujisawa* would have been correctly analyzed as a location name if it had appeared in the input text 横須賀から藤沢まで電車で行きました (*yokosuka kara fujisawa made densha de ikimashita* (I went from Yokosuka to Fujisawa by train.)).

Since the sequential tagging used in NameLister can be similarly applied to other languages besides Japanese, it currently enables named entities to be extracted from Chinese, Korean, and English texts as well.

4. Dependency parser: Jdep

Morphological analysis and named entity recognition are word-level analysis functions. On the other hand, dependency parsing is an analysis function that operates at the sentence level. The ultimate ideal is to perform a variety of processes that understand the meaning of the text. However, in order to understand what text means, we must first ascertain its grammatical structure. Dependency parsing is a function for analyzing the structure of a text.

4.1 What is dependency parsing?

In Japanese, the structure of a sentence is normally represented in terms of two types of element: constituent phrases (called *bunsetsu*) and the dependency between phrases including the modifying relationship. For example, the sentence 望遠鏡でカゴを持った少女を見た (*Bōenkyō de kago wo motta shōjo wo mita* (With the telescope, I saw a girl carrying a basket) is split into five phrases: 望遠鏡で / カゴを / 持った / 少女を / 見た (*Bōenkyō de* (with the telescope) / *kago wo* (basket) / *motta* (carrying) / *shōjo wo* (girl) / *mita* (*saw*)). Since 望遠鏡 depends on 見た, this sentence is interpreted as being about seeing a girl through a telescope. If it were mistakenly analyzed as 望遠鏡 depending on 持った (*motta*), then the sentence would be interpreted as being about carrying a basket in a telescope, which would be highly unusual. Thus, ascertaining the syntactic structure of a sentence is an essential step in figuring out the meaning of the sentence (i.e., who did what?).

4.2 Jdep framework

The dependency parser Jdep performs phrase chunking by means of sequential tagging in the same way as NameLister. An example is shown in **Fig. 4**. When it separates phrases, each word is tagged to indicate whether it is the beginning of a phrase (B) or a continuation of a phrase (I). The phrase sequence can then be obtained by extracting sequences of the form B-I-I-.

When the phrases have been determined, the next job is to determine which phrase modifies which phrase. In a narrow sense, this is called *dependency parsing*. Sequential tagging is also used here. An example is shown in **Fig. 5**. In the case of dependency parsing, each phrase is tagged with the relative position of the phrase that it modifies. In the example shown in Fig. 5, the phrase 藤沢さんって depends on the phrase 人かもね, so it is tagged 3D (modifies the third phrase after this one).

4.3 Analysis of CGM documents

On the Internet, there are a huge number of text documents published directly by users in the form of weblogs (blogs) and the like. These are known as consumer generated media (CGM). Content of this sort contains many colloquial elements and includes independent phrases that are not related to any others, such as emoticons and filler expressions like えーっと (*ētto*) and … (silence). By introducing a *self-dependency* tag to the sequential tagging, Jdep can clearly specify which phrases are independent and do

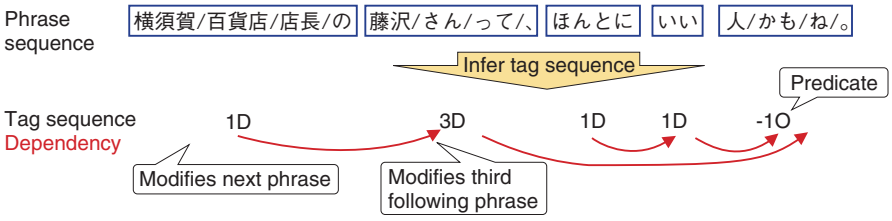

Fig. 5. Dependency parsing using sequential tagging.

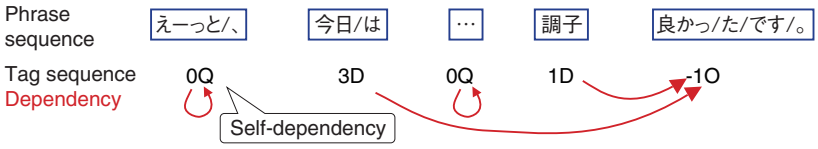

Fig. 6. Dependency parsing of a sequence including independent phrases.

not modify anything. An example of the analysis of a sentence that includes independent phrases is shown in **Fig. 6**. The introduction of this function has made it possible to extract only the grammatical structures that directly influence the meaning of the sentence when analyzing CGM documents [6].

5. Applicable fields of this technology

The above three techniques can be applied to the processing of various kinds of text on the Internet. For example, morphological analysis could be applied to search engines that can take account of the meaning of words, instead of simply considering string matching. This would prevent a search for 京都 (*Kyōto* (Kyoto)) from returning hits containing the characters 東京都 (*Tōkyō-to* (Tokyo metropolis)). Named entity recognition can extract expressions that are useful as keywords, so it is useful for improving the ranking and quality of search results. Dependency parsing can determine relationships such as who did what. It can thus be applied to data mining fields such as discovering what it is that has suddenly stopped working when large amounts of text are written about "stopping".

6. Future work

Ideally, we would like to make a computer able to understand and process the meaning of any text published on the Internet. The three basic analysis techniques introduced in this article are essential elements for the semantic analysis processes. However, even when they are used together, they still cannot understand more complex relationships like "who did what where, when, and how?" for example. Another problem is that written documents often omit elements that are salient to the reader, but not to the computer. Therefore, we are still a long way from having computers that completely understand and process written documents. In the future, we will continue to study analysis techniques such as reference omission resolution and semantic comprehension of text contents, starting with the Japanese language.

References

[1] T. Fuchi and S. Takagi, "Japanese Morphological Analyzer using Word Co-occurrence—JTAG," Proc. of COLING-ACL, pp. 409–413, 1998.
[2] S. Ikehara, M. Miyazaki, S. Shirai, A. Yokoo, H. Nakaiwa, K. Ogura, Y. Ooyama, and Y. Hayashi, "Nihongo GoiTaikei [Japanese Lexicon]," Iwanami Shoten, 1997 (in Japanese).
[3] K. Saito and M. Nagata, "Multi-Language Named-Entity Recognition System based on HMM," ACL 2003 Workshop on Multilingual and Mixed-language Named Entity Recognition, 2003.
[4] http://nlp.cs.nyu.edu/irex/
[5] J. Suzuki, E. McDermott, and H. Isozaki, "Training Conditional Random Fields with Multivariate Evaluation Measures," Proc. of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics, pp. 217–224, 2006.
[6] K. Imamura, G. Kikui, and N. Yasuda, "Japanese Dependency Parsing Using Sequential Labeling for Semi-spoken Language," Proc. of the 45th Annual Meeting of the Association for Computational Linguistics Companion Vol. Proc. of the Demo and Poster Sessions, pp. 225–228, 2007.

Kenji Imamura

Senior Research Scientist, Speech, Acoustics and Language Laboratory, NTT Cyber Space Laboratories.

He received the B.E. degree in engineering from Chiba University, Chiba, and the Ph.D. degree in engineering from Nara Institute of Science and Technology, Nara, in 1985 and 2004, respectively. He joined NTT in 1985. He was with the ATR Spoken Language Communication Research Laboratories, Kyoto, from 2000 to 2006. His research interests include natural language processing, especially machine translation, parsing, and spoken-language processing. He is a member of the Institute of Electronics, Information and Communication Engineers of Japan, the Information Processing Society of Japan (IPSJ), and Applied Natural Language Processing.

Kuniko Saito

Research Engineer, Speech, Acoustics and Language Laboratory, NTT Cyber Space Laboratories.

She received the B.E. and M.E. degrees in chemistry from the University of Tokyo, Tokyo, in 1996 and 1998, respectively. She joined NTT Information and Communication Systems Laboratories in 1998. Her research focuses on part-of-speech tagging, named entity recognition, and term extraction. She is also interested in multilanguage processing for Asian languages such as Japanese, Korean, and Chinese. She is a member of the Association for Natural Language Processing (NLP) and IPSJ.

Hisako Asano

Senior Research Scientist, Speech, Acoustics and Language Laboratory, NTT Cyber Space Laboratories.

She received the B.E. degree in information engineering from Yokohama National University, Kanagawa, in 1991. She joined NTT Information Processing Laboratories in 1991. Her research interests include natural language processing, especially morphological analysis, information extraction, and text analysis for text-to-speech synthesis. She is a member of NLP and IPSJ.