Named Entity Extraction from Japanese Broadcast News

Akio Kobayashi[†], *Franz J. Och*[‡] *and Hermann Ney*[*]

[†]NHK Science & Technical Research Laboratories, Japan
[‡]University of Southern California/Information Science Institute, USA
[*]Lehrstuhl für Informatik IV, Computer Science Department
RWTH Aachen - University of Technology, Germany
kobayashi-a.fs@nhk.or.jp, och@isi.edu, ney@informatik.rwth-aachen.de

Abstract

This paper describes a method for named entity extraction from Japanese broadcast news. Our proposed named entity tagger gives entity categories for every character in order to deal with unknown words and entities correctly. This character-based tagger has models designed by maximum entropy modeling. We discuss the efficiency of the proposed tagger by comparison with a conventional word-based tagger. The results indicate that the capability of the taggers depends on the entity categories. Therefore, the features derived from both character and word contexts are required to obtain high performance of named entity extraction.

1. Introduction

Recent progress of natural language processing (NLP) has enabled a variety of practical applications. For example, simple n-gram language models in a speech recognition system[1] provide real-time captioning services in broadcast news for hearing impaired people. However, such real-time applications raise a serious issue of maintenance. The systems must register new words observed in the latest news into their lexicons in order to cover the contents of upcoming events. Names of persons, companies and industrial products account for most of these new words. Named Entity (NE) extraction, an information retrieval technique, provides the solution since it can extract such meaningful expressions automatically from a large quantity of documents.

This paper describes a method for NE extraction from Japanese broadcast news. We propose a NE tagger which determines NE categories by evaluating every character. Maximum Entropy (ME) modeling, which is a very powerful method of representing properties of natural language, is applied for the models of the tagger to be realized.

In Section 2, we explain NE extraction and the categories of entities. The motivation for our character-based NE tagger is presented. In Section 3, conditional ME models for NE extraction and their features are briefly introduced. In Section 4, we discuss the efficiency of our proposed character-based NE tagger by comparison with a conventional word-based tagger, which determines the categories for every word.

2. Named Entity Extraction

2.1. Named Entity Extraction

NE extraction is for obtaining names of persons, countries, organizations and artifacts from documents. This area has been researched throgh contests such as MUC-7 (Message Understanding Conference) as part of information retrieval. A similar contest targeting Japanese NE tasks (IREX, Information Retrieval EXercise) was held in Japan[10]. Figure 1 illustrates a Japanese sentence including NEs (The meaning of the sentence is "Prime Minister Koizumi visited China this morning"). In the figure, "小泉 (*Koizumi*)" is a person's name and "中国 (*China*)" is the name of a location.

小泉/総理大臣/は/けさ/中国/を/訪問/し/まし/た/。

Figure 1: *Example of a Japanese sentence*

Table 1: *Categories*

category	explanation
artifact	industrial product, etc.
organization	organization / company
location	country / place
name	person
date	date
time	time
money	money
percent	percentage, ratio, etc.
none	non-NE

2.2. Categories

We defined eight unique NE categories according to the definition by the IREX (Information Retrieval EXercise) Committee[5][10](Table 1). The distinction between the IREX definition and ours is that a category is not tied with a word but with a character in our definition.

Each tag is extended by four markers; `start`, `middle`, `end` and `unique` positional markers are introduced so that entity boundaries can be marked. A `start` marker is attached to the beginning character of an entity. An `end` marker is also given to the last character of an entity with its category. If an NE consists of only one character, a `unique` marker is attached. A `middle` marker is given to all characters without any markers above.

2.3. Characteristics of Japanese

We confirm the motivation of our proposed character-based NE tagger by viewing characteristics in the Japanese language.

One of the characteristics of Japanese is linguistic agglutination, which means that there is no delimiter in Japanese sentences[1]. This fact shows that basic statistics (*e.g.*, word frequencies) cannot be obtained without morphological analysis[2]. Most popular Japanese morphological analyzers such as "Cha-Sen"[6] and "JUMAN[7]" produce a morpheme sequence from a sentence using morpheme and part-of-speech (POS) n-grams or rules written by hand. However, these analyzers cannot deal with *unknown* morphemes correctly because they yield morpheme sequences according to their morpheme-based lexicon. Analyzed morpheme sequences inevitably include incorrect morpheme boundaries.

Secondly, Japanese has many characters. One Japanese major machine-readable character set includes over 6,000 unique characters. Nagao and Mori[8] reported that character-based n-grams were important to determine morpheme boundaries. According to their survey, there are less frequent character pairs enclosing a morpheme boundary than pairs that do not enclose any boundaries. Nagata[9] studied a morphological analyzer using statistics of characters (*i.e.*, character n-grams). He demonstrated that these character-based statistics should be expected for efficient morphological analyzing. In particular, they are useful determining morpheme boundaries surrounding unknown morphemes.

These surveys indicate that characters are essential features of NE extraction. In the following section, we give a detailed description of the NE tagger which makes decisions of NE categories for every character using statistics of characters.

3. NE Extraction Using Maximum Entropy Modeling

3.1. Formulation of NE Extraction

NE extraction is defined as an optimization problem:

$$\mathbf{t}^* = \arg\max_{\mathbf{t}} \prod_j P(t_j|t_{j-1}, C_{j-m}^{j+m}) \tag{1}$$

The problem is to find the optimum category sequence $\mathbf{t}^*$ using a statistical model $P(t_j|t_{j-1}, C_{j-m}^{j+m})$. Here, t_j denotes a category, and C_{j-m}^{j+m} denotes a certain context associated with t_j. In this paper, C_{j-m}^{j+m} represents either a sequence of words or a sequence of characters.

3.2. Maximum Entropy Modeling

We briefly introduce maximum entropy modeling[2] in order to realize the statistical models in Equation (1). Let a pair (x, y) be a co-occurrence observed in training data. y is a symbol derived from an observation and x is a sequence of observations. Define $f_i(x, y)$ $(i = 1, \ldots, I)$ as a binary "feature" function representing characteristics of the event (x, y).

$$\mathcal{F} = \{f_i : (x, y) \longmapsto \{0, 1\}, i \in \{1, \ldots, I\}\} \tag{2}$$

where $f_i(x, y)$ returns 1 only if (x, y) matches an adequate condition assigned to $f_i(x, y)$. Given a set of features and training data, the solution of the conditional ME modeling is

$$P_{ME}(y|x) = \frac{\exp \sum_i \lambda_i f_i(x, y)}{\sum_{y'} \exp \sum_j \lambda_j f_j(x, y')} \tag{3}$$

where λ_i is a model parameter determined by the Generalized Iterative Scaling algorithm[4] under the following constraints.

$$E_{P_{ME}}[f_i] = E_{\tilde{P}}[f_i] \tag{4}$$

$E_{P_{ME}}[f_i]$ denotes the expectation of f_i on P_{ME}. $\tilde{P}$ represents the empirical distribution over given training data.

3.3. Features

3.3.1. Character Context Features

Initially, character context features of ME models are introduced.

Let c_j be a character with a category t_j to be predicted. For an event (C_j, t_j), C_j is a sequence of characters $\{c_{j-n}, \ldots, c_j, \ldots, c_{j+n}\}$. The character context feature is defined as follows:

$$f_i(C_j, t_k) = \delta(t_k, t_{k'}) \times \delta(C_j, C_{j'}) \tag{5}$$

where $\delta(x, x')$ denotes Kronecker's delta function and it returns 1 only if $x = x'$ while it returns 0 otherwise. A character context feature has either a sequence of $c_{j-n}, \ldots, c_j$ as a left context or a sequence of $c_j, \ldots, c_{j+n}$ as a right context. A feature has $n (n = 1, 2, \ldots)$ characters, therefore, features with various context length can be activated at once.

3.3.2. Word Context Features

Word context features are analogously introduced into ME modeling. The definition of the word context feature is

$$f_i(C_j^w, t_k) = \delta(t_k, t_{k\prime}) \times \delta(C_j^w, C_{j'}^w) \tag{6}$$

where C_i^w is a context including a word sequence $\{w_{l-m}, \ldots, w_l, \ldots, w_{l+m}\} (m = 0, 1, \ldots)$. In the sequence, w_l is a word including the character c_j. Then, w_{l-1} is a predecessor word of w_l, and w_{l+1} represents a successor. Note that each word context feature establishes a link between a word sequence and a category. Word contexts contain information on word boundaries as a prior knowledge while character contexts never include them. The drawback of using word contexts is that unknown words are not treated correctly.

3.4. Word-based Tagger

The word-based taggers are commonly used in Japanese NE extraction. Borthwick[3] and Uchimoto[11] performed experiments on word-based NE taggers using ME models for IREX tasks. In these studies, the features of the ME models were based on word contexts and several lexical properties (parts of speech, etc.). These properties were induced from word sequences produced by the "JUMAN" morphological analyzer. We construct a word-based tagger using an ME model with word context features as a baseline for comparison with our character-based tagger.

4. Experiments

4.1. Training & Test Data

All training and test data were taken from manuscripts of NHK broadcast news. Sentences in both sets of data were partitioned

[1] In Figure 1, morpheme boundaries are described by "/" characters for convenience

[2] A "morpheme" and a "word" have nearly the same meaning in this paper.

into words by "Cha-Sen" morphological analyzer. The training data had 24,433 sentences (1.25M words and 2.0M characters). There were 14,914 unique words observed more than two times in the training data. The data had 2,500 unique characters observed more than once. The test data had 996 sentences (48k words and77k characters). There were 2,525 NEs in total. The remaining characters or words were treated as an unknown symbol in the ME models.

4.2. Taggers and Models

We constructed the following taggers for experiments.

- **Baseline:** Word-based tagger using the ME model with word context features. A feature has one, two or three words in its left/right contexts.
- **Tagger1:** Character-based tagger using the ME model with character context features. A feature has one, two, three or four characters in its contexts.
- **Tagger2:** Character-based tagger using the ME model with the word context features. A word feature has one or two words in its contexts.
- **Tagger3:** Character-based tagger using the ME model with both character and word context features above.

In GIS training, frequencies of events in the training data were simply discounted with a constant value of 0.1 to obtain smoothed ME models. The number of iteration in GIS training was fixed at 200 times. All parameters associated with the context length were determined by preliminary experiments.

4.3. F-measure Evaluation

Recall is defined as a measure of the numbers of correct NEs from the correct documents and calculated by

$$Recall : R = \frac{Number\ of\ correctly\ marked\ up\ NEs}{Number\ of\ NEs\ in\ the\ reference} \tag{7}$$

Precision is defined as a measure of the numbers of correct NEs from all NEs and calculated by

$$Precision : P = \frac{Number\ of\ correctly\ marked\ up\ NEs}{Number\ of\ marked\ up\ NEs} \tag{8}$$

F-measure is defined as a harmonic mean between recall and precision:

$$F = \frac{2PR}{P+R} \tag{9}$$

In this paper, F-measure is used as a criterion for results.

4.4. Results

4.4.1. Overall Results

Overall results of the extraction experiments are shown in Table 2. **Tagger1**, which used the ME model with character context features, did not achieve better performance than the word-based tagger **Baseline**. It is clear that the features derived from word contexts are more powerful than those from character contexts since the word contexts include information on word boundaries explicitly. **Tagger1**, nevertheless, gave almost the same recall score as **Baseline**. This result suggests that the character context features tend to overgenerate NE candidates. **Tagger3**, which used both character and word context features, exceeded the performance of **Baseline**. Consequently, it is reasonable for an efficient NE extraction technique to generate surplus NE candidates using character context features while the word context features suppress excess candidates.

Table 2: *Overall extraction results*

	# features	Precision(%)	Recall(%)	F
Baseline	335k	85.74	82.42	83.87
Tagger1	1.0M	80.06	82.69	81.36
Tagger2	345k	78.6	79.80	78.97
Tagger3	1.4M	86.42	85.43	85.92

Detailed results are shown in Table 3. **Baseline** gave the best performance of F-measure on tagging the NEs in *location* and *artifact* categories. In contrast, **Baseline** did not exceed **Tagger1** in the performance when *organization* names were tagged. It is possible that entity structures affect the difference in performance. For example, the names in the *location* category are typically expressed as concatenations of proper nouns associated with prefectures, cities and streets. The word context features naturally detect the linkages between these words more powerfully than the character context features do. In contrast, names in the *organization* category, in particular company names include many invented words and special notations. For such entities, the character context features are more efficient than the word context features since the ME models with the character context features are designed to express how entities are formed from characters. Hence, features that reflect the characteristics of the entities are needed for efficient NE extraction.

4.4.2. Extraction of Unseen NEs

We compared the performance tagging unseen NEs. There were a total of 624 unseen NEs (397 unique entities) in the test data. Recall was regarded as a measure of the capability of the taggers obtaining the unseen NEs since it is difficult to identify the extracted NEs as the unseen NEs in the reference. Overall results shown in Table 4 shows that the performance of **Tagger1** was inferior to that of **Baseline** whereas character-based taggers were expected to deal with the unseen NEs successfully because of character-based entity tagging. Table 5 shows that all the character-based taggers gave disappointingly poor performance tagging the NEs in the *artifact* category. The degraded performances probably resulted from the length of NEs. The average length of NEs included in the *artifact* category was 5.82 characters per entity, which was longer than the average length of NEs in all categories (4.82).

Tagger3 improved the performance of **Tagger1** for tagging the NEs in *organization*, *location* and *artifact* categories. The result indicates that it is useful for the ME model to take word-based and character-based features together for most categories.

4.4.3. Numbers of Features

Considering numbers of the features in the ME models, the efficiency of the taggers is discussed. As shown in Table 2, **Baseline** achieved comparably high performance with a smaller number of features than others while **Tagger3** gave a small im-

Table 3: *Extraction results (F-measure)*

	org.	pers.	loc.	art.	date	time	money	prcnt.
Baseline	77.74	83.20	**88.36**	**85.47**	92.51	56.90	81.16	87.38
Tagger1	82.80	79.05	80.44	79.08	92.86	65.98	**86.96**	**89.00**
Tagger2	74.38	78.46	80.65	78.62	92.86	65.31	81.16	81.16
Tagger3	**83.17**	**87.14**	87.60	84.14	**93.20**	**68.04**	84.06	87.56

Table 5: *Extraction results (unseen NEs, recall)*

	org.	pers.	loc.	art.	date	time	money	prcnt.
Baseline	7.97	43.27	51.41	**14.06**	66.67	88.89	64.71	88.24
Tagger1	17.39	41.52	32.20	1.56	66.67	88.89	**76.47**	94.12
Tagger2	10.87	43.86	42.94	6.25	66.67	77.78	64.71	70.59
Tagger3	**20.29**	**56.14**	**57.06**	3.13	66.67	88.89	70.59	**100.00**

Table 4: *Overall extraction results (unseen NEs)*

	Recall(%)
Baseline	38.78
Tagger1	34.29
Tagger2	35.26
Tagger3	45.03

provement over **Baseline** for its increased number of features. This suggests that the features of **Tagger3** describes redundantly the ME model. Since the number of features has an impact on computational costs of GIS training, an efficient feature selection technique is required.

5. Conclusion

This paper described a method of NE extraction from Japanese broadcast news using maximum entropy modeling. We discussed efficiencies of both word-based and character-based taggers considering features of the ME models. The word-based tagger is powerful for NE extraction, while the character-based tagger is also useful for tagging the unseen NEs in several categories. The character-based tagger using the ME model with both word and character context features obtained the best performance for F-measure.

6. Acknowledgements

The authors would like to thank members of the Lehrstuhl für Informatik IV, at RWTH Aachen for their helpful suggestions and provision of computational equipment and tools.

7. References

[1] A. Ando, T. Imai, A. Kobayashi, H. Isono, and K. Nakabayashi. "Real-Time Transcription System for Simultaneous Subtitling of Japanese Broadcast News Programs". *IEEE Transactions on Broadcasting*, pages 189–196, September 2000.

[2] A. Berger, S. D. Pietra, and V. D. Pietra. "A Maximum Entropy Approach to Natural Language Processing". *Computational Linguistics*, 22:39–71, 1996.

[3] A. Borthwick. "A Japanese Named Entity Recognizer Constructed by a Non-speaker of Japanese". In *proceedings of the IREX Workshop*, pages 187–193, 1999.

[4] J. Darroch and D. Ratcliff. "Generalized Iterative Scaling for Log-Linear Models". *The Annals of Mathematical Statistics*, pages 1470–1480, 1972.

[5] IREX Committee. *proceedings of the IREX workshop*, 1999.

[6] Y. Matsumoto, A. Kitauchi, T. Yamashita, Y. Hirano, H. Matsuda, K. Takaoka, and M. Asahara. *"Morphological Analysis System ChaSen version 2.1 Manual"*. Nara Advanced Institute of Science and Technology, December 2000.

[7] Y. Matsumoto, S. Kurohashi, Y. Taeki, and M. Nagao. *"Japanese Morphological Analyzing System: Juman"*. Kyoto University and Nara Institute of Science and Technology, 1997.

[8] M. Nagao and S. Mori. "A New Method of N-gram Statistics for Large Number of N and Automatic Extraction of Words and Phrases from Large Text Data of Japanese". In *COLING-94*, pages 611–615, 1994.

[9] M. Nagata. "Automatic Extraction of New Words from Japanese Texts using Generalized Forward-Backward Search". In *Empirical Methods in Natural Language Processing*, pages 48–59, 1996.

[10] S. Sekine and Y. Eriguchi. "Japanese Named Entity Extraction Evaluation -Analysis of Results". In *International Conference on Computational Linguistics*, pages 25–30, 2000.

[11] K. Uchimoto, Q. Ma, M. Murata, H. Ozaku, M. Utiyama, and H. Isahara. "Named Entity Extraction Based on A Maximum Entropy Model and Transformation Rules". *Natural Lanugage Processing*, pages 64–90, August 2000. (in Japanese).