Recoginizing Named Entities in Japanese Corpus by Incremental Deep Parsing (2)

Xinkai Wang[†], Masakazu Tateno[‡]

[†] Northern Jiao Tong University, Beijing, P. R. China and Fuji Xerox Co., Ltd.
[‡] Fuji Xerox Co., Ltd
Email: Tateno.Masakazu@fujixerox.co.jp

Abstract

This paper proposes a method to find organization names in Japanese corpus. The method consists of collecting the specific words, identifying specific parts of speech and describing layered rules to construct the names. We got good results by this method.

Keywords

Named Entity Extraction, Japanese organization names, Incrementa Deep Parsing

増深解析法による
日本語の固有名の認識(2)

王新凱[†]　　　　　舘野昌一[‡]

[†] 中国北方交通大学、富士ゼロックス株式会社
[‡] 富士ゼロックス株式会社
Email: Tateno.Masakazu@fujixerox.co.jp

あらまし　本稿は日本語の文書から組織名を認識する方法を提案する。この方法は、あらかじめ特定の語を収集し、品詞情報を用いて手作りの規則を記述しておき、後でそれらを順番に適用することにより、組織名を認識する。評価用データにより得たＦスコアは、88.35%であった。

1. Background

Named Entity Extraction (NEE) plays an essential role in information extraction systems and question answering systems. It is one of the subtasks of the Message Understanding Conference (MUC) and has been studied deeply. Organization names have been thought as one of the most difficult entities to extract [1, 2]. This paper focuses on recognizing organization names in Japanese corpus.
Although many of recent researches on NEE has been using machine learning methods[3, 4] and other statistical approaches, the method we use is based on the handcrafted rules. we wrote the layered rules to form organization names, in terms of syntactic features and contextal information of organization names in Japanese corpus.

One of the reasons of the difficulties to extract organization names is that there is no uniform property to be an organization name. For example, we think that “米国会計検査院” is an organization name, but “会計検査院” is the name and “米国” is a location name in the CRL truth file for IREX. “同市教委” would be a name because it identifies an entity of organization. But “同市教委” is not in the list of the organization names in the same truth file. So we feel inconsistency in the truth file that makes our comparing work more difficult. In this paper we will define our criteria for the organization names.

2. Our Criteria for Organization Names

We find that different people understand the definition of orgization names differently. Generally speaking, an organization is defined as “a group of persons organized for a particular purpose; an association”. Our criteria for organization names are as follows:

1) Types
We classified organization names (org) into three types.

Type I: org = word$_{1}$, …, word$_{n}$, *<u>suffix</u>*
Type II: org = word$_{1}$, …, word$_{n}$, | <u>suffix</u> |
Type III: org = word$_{1}$, …, word$_{n}$

In Type I, org consists of a sequence of words and a suffix. The suffix is always a part of org. In Type II, everything is the same as Type I except the suffix is not a part of org. In Type III, everything is the same as Type I except there is no suffix.

2) Determination of organization names
We think an organization name should be as long as possible.
org = org$_{1}$, org$_{2}$, …, org$_{n}$

This means that if there are several adjacent organization names, all of them comprise the new organization name.

3) Determination of location names

We think location names which appear in front of organization names should be parts of a new organization name.

org = loc_1, …, loc_m, org_1, …, org_n

4) If a sequence of words can be understood as a group of persons organized for a particular purpose by the human verifiers, it is classified as an organization name.

3. Incremental Rules

We use Xerox Incremental Parser (XIP, [5, 6]) that processes the layered rules on words and their parts of speech in the results of morphological analysis of the corpus to extract organization names. There are four types of rules defined in XIP such as *lexicon rules*, *disambiguation rules*, *chunking rules* and *dependecy rules*. Those rules are applied to the input in this order (Figure 1). Names are extracted gradually by applying multiple rules among the layers. The rules in the previous layer prepare the necessary information for the rules in the next layer. So the rules in every layer increase the information for the named entities. The followings are the examples of such rules.

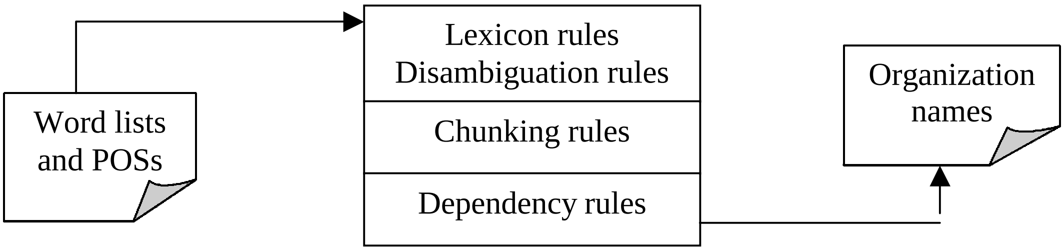

Figure 1 Layers of rules
Lexicon rules and Disambiguation rules are applied then Chunking rules are applied to detect and chunk organization names. Dependency rules separate organization names from chunking trees at last.

1). Lexicon rules Lexicon rules attach additional information to the input that is the result of the morphological analysis. For example,

学校 = noun [osfn_concrete=+].

shows that a noun is attached as a part of speech and osfn_concrete is attached as a feature of the word ("学校"), where the word is a lemma in the input.

2). Disambiguation rules Disambiguation rules select a part of speech for a word that has multiple parts of speech depending on the contexts. For example,

place, fname = place | place_suffix |.

shows that a word with place and fname as parts of speech is realized as place for the feature when the next word has place_suffix as a feature.

3). Chunking rules Chunking rules make a group on the sequence of words and associate a node with new features on the group. For example,

onamep[orgn=+] = ?[onamep_head_possible:+], ?[region:general].

shows that a word with onamep_head_possible and the next word that has a feature region with general as its value are grouped as a chunk with onamep as the part of speech and orgn as the feature. The following example shows this chunk.

```
                                     TOP
  +-------+---------------------+---------+-------+----------+------------+-----+
  |       |     |       |       |         |       |          |            |     |
NOUN     AUX  NOUN     NOUN   AUX       NOUN     AUX       ONAMEP       AUX  SYMBOL
  +       +     +       +       +         +       +     +-----------+     +     +
  |       |     |       |       |         |       |     |           |     |     |
宝剣岳    の   宝剣   山荘    から     ふもと     の   NOUN      PLACE   へ    。
                                                        +           +
                                                        |           |
                                                      ホテル      千畳敷
```

4). Dependency rules Dependency rules detect the relationships among words, their parts of speech and chunks. For example,

| *?#1[onamep:+]* | *oname_struct(#1)*

This rule extracts all chunks with a feature onamep and assigns the dependency name called oname_struct. In the sentence above, this rule gives the result as follows:

ONAME_STRUCT（ホテル 千畳敷）

4. Examples of applying incremental rules to resolve problems

We found that two types of organization names are more difficult to extract than others. In these cases, it is very important to design the incremental rules carefully.

1). Abbreviation names which appear in personal titles For example, the word "通産" in "通産相" is an abbreviation name of the organization. It is hard to detect whether this is a name or not. We think "通産相" is the second type of names in our name criteria. So "相" is the suffix of the organization name. We use following rules to detect and extract this name.

相 += noun[titleid =+].
…

```
onamep = ?[noun:+]  | ?[titleid:+] |.
…
| ?#1[onamep:+] |  oname_struct(#1)
```

Firstly we collect this type of suffixes as many as possible, and attach the special feature called “titleid” to them. Then a chunking rule in the next layer is applied to detect and chunk the organization name. Finally the dependency rule extracts the name. The following example shows this chunk.

```
                          TOP
        +----------+---------+---------+-------+
        |          |         |         |       |
      FNAME      ONAMEP     NOUN      AUX   SYSMBOL
        +          +         +         +       +
        |          |         |         |       |
 ・・  橋本        NOUN       相        は      、        ・・
                   +
                   |
                  通産

 ONAME_STRUCT(通産)
```

2). Maxium matching organization names For exmple, “長江集団公司” appeared in the Japanese newspaper is an oranization name, while “長江集団” is a name if no other suffixes appear. This means that we should have some incremental rules to match an organiztaion name as long as possible. We use these rules to achieve it.

```
公司 += noun[ osfn =+, osfn_suffix =+ ].
…
place, fname = place | osfn |.                  // for 長江
…
noun[osfn=+] = ?[osfn:+] | ?[ osfn_suffix:+] |.  // for 集団公司
…
onamep[orgn=+]= place, ?[osfn:+];?[orgn:+].     // Semicolon for disjunctive
…
| ?#1[onamep:+] |  oname_struct(#1)
```

Firstly we see “公司” as the suffix (“osfn_suffix”) of an organization names’suffix (“osfn”). Then a disambiguation rule will correct the POS of “長江” from “fname” to “place”; and a chunking rule in the next layer will identify the suffix of the organization name. Then “集団公司” is grouped into a noun structure with the feature of “osfn”. In the next step, a chunking rule will chunk the place name and the chunk with “osfn” into an organization name. Finally, a dependecy rule will extract the organization name. The following chart shows this process.

```
                TOP
                 +
                 |
               ONAMEP
     +--------------------+
     |                    |
   PLACE                 NOUN
     +              +--------------+
     |              |              |
   長江            NOUN           NOUN      ・・・
                    +              +
                    |              |
                   集団           公司
```

ONAME_STRUCT(長江　集団　公司)

5. Results and Comparison

We used Mainichi Newspaper of January 1 to 10 of 1995 to write the rules comprehensively. Then we tried and fixed the rules using that of January 5th and 12th. Then we applied the rules to the task of NEE of IREX in 1999. Table 1 shows the statistics on them. The first two rows are the statistics of the results after fixing the rules. The third row shows the statistics without any modification of the rules before applying them to the task. Recall rate became worse while Precision rate became better. One of the reasons of the leaks is the new patterns that are not captured as the organization names in the training sets. As a result, the rules extracted parts of the name instead of the whole name. For example, our method found ロシア軍 instead of ロシア軍司令部. On the other hand, the noise is reduced because the handcrafted rules may capture the generic nature of the organization names.

	Result	Correct	Truth	P(%)	R(%)	F(%)
950105	482	443	471	91.91	94.06	92.97
950112	650	579	609	89.08	95.07	91.98
IREX99	430	402	480	93.48	83.75	88.35

Table1 Statistics of the results

Table 2 shows the results with other methods. Our result is comparable with others ([3], [7]).

Methods	Done by	F (%)
Linguistic Method with Handcrafted rules	Li, Tateno	88.35
Decision List Learning and Maximum Entropy	Utsuro, Sassano, Uchimoto	84.70
Support Vector Machine	Isozaki	78.70

Table2 Comparison of F scores

6. Future work

Our incremental rules are easy to extract the target names and do not need the huge corpus, which are usually the requirement of statistic-based appoarches. However, the rules should be edited by hand. We hope to combine statistic-based method with incremental rules in the next step.

References

1. Shiren Ye, Tat-Seng Chua, Liu Jimin, (2002), An Agent-based Approach to Chinese Named Entity Recognition, In Proc.of COLING2002.
2. Palmer D. D. (1997), A Trainable Rule-Based Algorithm for Word Segmentation, In Proc. of *35th of ACL & 8th conf. Of EACL*, 321-328.
3. Hideki Isozaki, Janpanese Named Entity Recognition Based on meta Rules and Decision Tree Learning, In *情報処理学会論文誌, Vol. 43, No. 5, May 2002*.
4. Satoshi Sekine, Ralph Grishman, Hiroyuki Shinnou, A Decision Tree method for Finding and Classifying Names in Japanese Texts, In WVLC98.
5. Salah Aït-Mokhtar, Jean-Pierre Chanod, and Claude Roux. "Robustness beyond shallowness: incremental deep parsing", In Natural Language Engineering, 8(2):121--144, 2002.
6. Xinkai Wang, Mazakazu Tateno, Research on Rule-based Name Extraction for Organizations in Simplified Chinese Texts, In Proc of *FIT 2002*, Tokyu.
7. 宇津呂　武仁，颯々野　学，内元　清貴，″正誤判別規則学習を用いた複数の日本語固有表現抽出システムの出力の混合″，自然言語処理，第 9 巻，第 1 号，pp.65-100, January 2002.