NYU/CRL QA System for QAC-2

Satoshi SEKINE Kiyoshi SUDO Yusuke SHINYAMA
New York University
715 Broadway, 7th floor, New York, NY 10003, USA
{sekine,sudo,yusuke}@cs.nyu.edu

Chikashi NOBATA Kiyotaka UCHIMOTO Hitoshi ISAHARA
Communications Research Laboratory
2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan
{nova,uchimoto,isahara}@crl.go.jp

1. Introduction

In this paper, we will describe NYU/CRL QA system for QAC-2 task-1. This is exactly the same system with the one we participated to QAC-1. We will describe the system and compare the evaluation result to the result of QAC-1.

2. NYU/CRL QA system

In this section our QA system and the evaluation result are described.

The NYU/CRL QA system consists of three components.

Question Examination (QE)
Examine the question sentence using question patterns. It creates information like keywords and NE type.
Text Retrieval (TR)
Based on keywords, texts that are expected to have answers are retrieved.
Answer Extraction (AE)
Among retrieved texts, answer strings will be searched using the information created by the QE component.

Note that we did NOT use the Mainichi 98 and 99 corpus in the knowledge creation etc by any means for any purpose. In the following description, `training QA data' means QAC dev data and CRL-QA data which will be explained later. We will describe each component in the following sub sections.

3. Question Examination

The input of this component is the question sentence and the output is the list of keywords, NE types of the answer, and several kinds of minor information. The sentence is first analyzed by morphological analyzer, JUMAN [JUMAN homepage], and our NE tagger.
Words are concatenated if it is a sequence of noun-prefix, nouns and noun-suffix or a NE expression, while the individual words remains (which are used as keyword etc with smaller scores).
Then question pattern rules are applied. Some examples are shown in Figure 1 and there are 129 patterns in the working system (however, as there are many `or's in the pattern, actual number would be very large).

The main purpose of the pattern matching is to find NE type expected by the question. The types can be more than one, as shown by the first two rules. Each MATCH line is matched against each bunsetsu by Perl's pattern matching. NEXTBUNSETSU indicates that the matching proceeds to the next bunsetsu. There are two kinds of patterns. One is to find the NE type directly. The first four patterns are this pattern. For example, if the first pattern matches, NE type ``organization'' is proposed. The other type is that the pattern find `center word' and the NE type is derived by the NE type specified by the word. The last two rules in the table is in this type. This is for the questions like ``「ＮＴＴデータ通信」

の社名変更後の会社名はなんですか。". The last rule in Figure 1 is used and extract the word ``会社" as the center word. Then the system look up our center word dictionary, which specifies relationship between nouns and its NE types. Using the dictionary, the system finally figures out the expected NE type includes `COMPANY'. The center word dictionary contains 16,431 entries and was compiled by hand based on Bunrui-Goi-Hyou and corpora. The rules have several attributes. PRIORITY to define the order of rule application, SCORE to indicate the likelihood of the NE type and GENERALIZATION to specify if the NE type can be generalized using the NE hierarchy.

```
RULE start
RULEID DOKO-ORGANIZATION
MATCH     ^(どこ|何処)
TYPE      組織名
SCORE     100.0
RULE end

RULE start
RULEID DOKO-LOCATION
MATCH     ^(どこ|何処|場所)
TYPE      地名
SCORE     100.0
RULE end

RULE start
RULEID KEN-HA-DOKO-PROVINCE
MATCH     (県|州|都道府県)と?は$
NEXTBUNSETSU
MATCH     ^(どこ|何処)(か|です|でし|。|？)
TYPE      都道府県州名
GENERALIZATION 0
PRIORITY 1
SCORE     10000.0
RULE end

RULE start
RULEID NANKASHO-N_LOCATION-0
MATCH     何(十|百|千|万|億|兆|何)*(か|箇|ヶ|ヵ|カ)所
TYPE      場所数
SCORE     1000.0
PRIORITY 1
RULE end

ULE start
RULEID    XXX-HA-DOKO
MATCH     [^にでの]は$
NEXTBUNSETSU
MATCH     ^(どこ|何処)(か|です|でし|。|？)
TYPEOF    HEAD 1
PRIORITY 2
RULE end

RULE start
RULEID    XXX-MEI-HA-NANI-PRE
MATCH     (名|名称)は$
NEXTBUNSETSU
MATCH     ^(何|なに|なん)(か|です|でし|。|？)
TYPEOF    PRE 1
PRIORITY 3
RULE end
```

Figure 1 Question pattern

Also, in the question examination component, keywords are identified.The keywords are used in both text retrieval and question extraction.Keywords include most kinds of nouns, adjectives, adverbs, verbs andunknown words.The keywords have scores based on POS type, IDF, and if it is center words or not. Keyword expansion is done using synonym dictionary, which contains 46,619 group of words, and the synonym words have relatively lower scores than the original words in the following processing. In the training phase, we figured out that having too many keywords is rather harmful, so the keywords are trimmed based on the score, number of keywords and overlappings to other keywords.

Several minor information is extracted, as well,

which includes the context of interrogative word, the following word of interrogative in order to find suffix of number expressions (for example, ``メートル" in ``全長は何メートルですか。"), if the question is asking the definition of a word, if the question is asking alias. Such information is used in the various places of the following processing.

4. Text Retrieval

Text retrieval is basically done by something like Boolean search against paragraphs of articles (rather than the articles). In the search, the more kinds of keywords appear in the text, the more score the text gets. Only when the score is the same (i.e. the same number of kinds of keywords appears), the scores prepared in the question examination are used. We found, in the training phase, that there is an optimal number of text used in the following process. The text are deleted based on the number of text retrieved, absolute score difference to the top text, ratio difference to the top text.

5. Answer Extraction

The answers are extracted from the retrieved texts. The sentences in the texts were analyzed fully automatically by JUMAN and our NE taggers in advance. We used two NE taggers. One is Maximum Entropy based NE tagger using IREX's 8 NE definitions, trained by CRL-NE data. The other is rule-based system using 140 NE types, in which the NE dictionaries and rules are created by hand. The NE entities appeared in the previous paragraphs are also used in the answer extraction. The NE hierarchy is designed by our selves [Sekine et. al 02] and available in the Web [ENE hoepage].

The words in the retrieved texts which are tagged as nouns (except some special kinds of nouns), unknown words, NEs and sequence of nouns are taken as answer candidates. The system calculate scores for each candidate based on the distance from keywords, NE type, inclusion of center words, suffix, expression within brackets, if the question is asking alias and similarity of the context to the context in the question. We tried to use distance in terms of dependency, but we figured out the word distance performs better than the dependency distance using the training QA data.

6. Evaluation Result

The QAC-2 task-1 evaluation result of our system compared to our QAC-1 task-1 result is shown in Table 1.

Task	Answer	Correct	MRR
QAC1 task1	1000	121	0.39
QAC2 task1	1000	128	0.402

Although there are minor changes in the task, it seems that the level of difficulty of two tasks (QAC1, task 1 and QAC2 task 1) are about the same.

References

[IREX Committee 1999] IREX Committee, 1999, *Proceedings of the IREX Workshop*.

[Sekine and Isahara 2000] Satoshi Sekine, Hitoshi Isahara, 2000, “IREX: IR and IE Evaluation Project in Japanese”, *Proceedings of the LREC-2000 conference*.

[Sekine et.al 2002] Satoshi Sekine, Kiyoshi Sudo, Chikashi Nobata, 2002, “Extended Named Entity hierarchy”, *Proceedings of the LREC-2002 conference*.

[ENE Homepage] Extended Named Entity Hierarchy Homepage, *http://nlp.cs.nyu.edu/ene/*.

[Juman Homepage] Juman Homepage (Juman Ver.3.61), *http://pine.kuee.kyoto-u.ac.jp/nl-resource/juman.html*