PATTERN MATCHING IN THE TEXTRACT INFORMATION EXTRACTION SYSTEM

Tsuyoshi Kitani† Yoshio Eriguchi†† Masami Hara††
Center for Machine Translation
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

In information extraction systems, pattern matchers are widely used to identify information of interest in a sentence. In this paper, pattern matching in the TEXTRACT information extraction system is described. It comprises a concept search which identifies key words representing a concept, and a template pattern search which identifies patterns of words and phrases. TEXTRACT using the matcher performed well in the TIPSTER/MUC-5 evaluation. The pattern matching architecture is also suitable for rapid system development across different domains of the same language.

1 INTRODUCTION

In information extraction systems, finite-state pattern matchers are becoming popular as a means of identifying individual pieces of information in a sentence. Pattern matching systems for English texts are reported to be suitable for achieving a high level of performance with less effort, compared to full parsing architectures [Hobbs et al. 92]. Among seventeen systems presented in the Fifth Message Understanding Conference (MUC-5), three systems used a pattern matcher as the main component for identifying patterns to be extracted [MUC-5 93]. A pattern matching architecture is appropriate for information extraction from texts in narrow domains since identifying information does not necessarily require full understanding of the text. The pattern matcher can extract information of interest by locating specific expressions defined as key words and phrasal patterns obtained by corpus analysis.

This paper describes a pattern matching method that first identifies concepts in a sentence and then links critical pieces of information that map to a pattern. The first step in pattern matching is a *concept search* applied in the TEXTRACT system of the TIPSTER Japanese microelectronics and corporate joint ventures domains [Jacobs 93a], [Jacobs 93b]. In this step, key words representing a concept are searched for within a sentence. The second step is a *template pattern search* applied in the TEXTRACT joint ventures system. A complex pattern to be searched for usually consists of a few words and phrases, instead of just one word, as in the concept search. The template pattern search recognizes relationships between matched objects in the defined pattern as well as recognizing the concept itself.

From the viewpoints of system performance and portability across domains, the TIPSTER/MUC-5 evaluation results suggest that pattern matching described in this paper is an appropriate architecture for information extraction from Japanese texts.

2 TIPSTER/MUC-5 OVERVIEW

The goal of the TIPSTER/MUC-5 project sponsored by ARPA is to capture information of interest from English and Japanese newspaper articles about microelectronics and corporate joint ventures.[1] A system must fill a generic template with information taken

†Visiting researcher from NTT Data Communications Systems Corp., email: tkitani@rd.nttdata.jp
††NTT Data Communications Systems Corp.

[1]Several ARPA-sponsored sites formed the TIPSTER information extraction project. The TIPSTER sites and other non-sponsored organizations participated in MUC-5.

from the text in a fully automated fashion. The template is composed of several objects, each containing several slots. Slots may have pointers as values, where pointers link related objects. Extracted information is expected to be stored in an object-oriented database [TIPSTER 92].

In the microelectronics domain, information about four specific processes in semiconductor manufacturing for microchip fabrication is captured. They are layering, lithography, etching, and packaging processes. Layering, lithography, and etching are wafer fabrication processes; packaging is part of the last stage of manufacturing. Entities such as manufacturer, distributor, and user, in addition to detailed manufacturing information such as materials used and the microchip specifications such as wafer size and device speed are also extracted in each process.

The joint ventures domain focuses on extracting entities, i.e. organizations, forming or dissolving joint venture relationships. The information to be extracted includes entity information such as location, nationality, personnel, and facilities, and joint venture information such as relationships, business activities, capital, and estimated revenue of the joint venture.

3 TEXTRACT ARCHITECTURE

TEXTRACT is an information extraction system developed as an optional system of the GE-CMU SHOGUN system [Jacobs 93a], [Jacobs 93b]. It processes the TIPSTER Japanese domains of microelectronics and corporate joint ventures. The TEXTRACT microelectronics system comprises three major components: preprocessing, concept search, and template generation. In addition to concept search, the TEXTRACT joint ventures system performs a template pattern search. It is also equipped with a discourse processor, as shown in Fig. 1.

In the preprocessor, Japanese text is segmented into primitive words tagged with their parts of speech by a Japanese segmentor called MAJESTY [Kitani and Mitamura 93], [Kitani 91]. Then, proper nouns, along with monetary, numeric, and temporal expressions

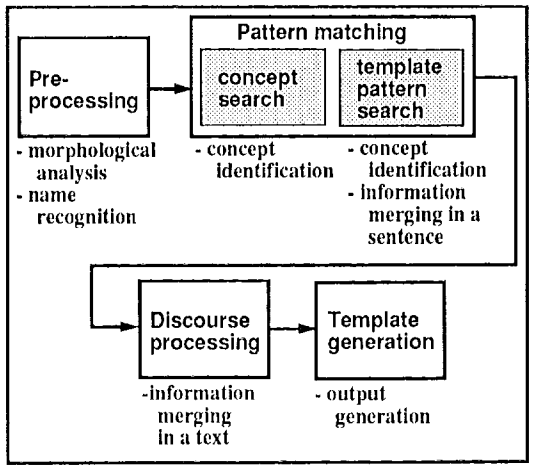

Fig. 1: Architecture of the TEXTRACT joint ventures system

are identified by the name recognition module. The segments are grouped into units which are meaningful in the pattern matching process [Kitani and Mitamura 94]. Most strings to be extracted directly from the text are identified by MAJESTY and the name recognizer in the preprocessor.

The concept search and template pattern search modules both identify concepts in a sentence. The template pattern search also recognizes relationships within the identified information in the matched pattern. Details of the pattern matching process are described in the next section.

The discourse processor links information identified at different stages of processing. First, implicit subjects, often used in Japanese sentences, are inherited from previous sentences, and second, company names are given unique numbers necessary to accurately recognize company relationships throughout the text [Kitani 94]. Concepts identified during the pattern matching process are used to select an appropriate string and filler to go into a slot. Finally, the template generation process assembles the extracted information necessary to create the output described in Section 2.

4 PATTERN MATCHING IN TEXTRACT

4.1 Concept search

Key words representing the same concept are grouped into a list and used to recognize the concept in a sentence. The list is written in a simple format: *(concept-name word1 word2 ...)*. For example, key words for recognizing a dissolved joint venture concept can be written in the following way:

```
(DISSOLVED 提携解消 整理 消滅)
                or
(DISSOLVED dissolve terminate cancel).
```

The concept search module recognizes the concept when a word in the list exists in the sentence. Using such a simple word list sometimes generates an incorrect concept. For example, a dissolved concept is erroneously identified from an expression "*cancel* a hotel reservation". However, when processing text in a narrow domain, concepts are often identified correctly from the simple list, since key words are usually used in a particular meaning of interest in the domain.

During the Japanese segmentation process in the preprocessor, a key word in the text tends to be divided into a few separate words by MAJESTY, when the word is not stored in the dictionary, For example, the compound noun "提携解消" consists of two words, "提携" (joint venture) and "解消" (dissolve). It is segmented into the two individual nouns using the current MAJESTY dictionary. Thus, when the compound word "提携解消" is searched for in the segmented sentence, the concept search fails to identify it. To avoid this segmentation problem, adjacent nouns are automatically put together during the concept search process.

This process allows, by default, partial word matching between a key word and a word in the text. Therefore, "提携" and "業務提携" both meaning "a joint venture" can be identified by a single key word "提携". However, due to the nature of partial matching, the key word "シリコン" (Silicon) matches "二酸化シリコン" (Silicon dioxide), which is a different type of film reported in the microelectronics domain. This undesirable behavior can be avoided by attaching ">" to the beginning or "<" to the end of key words. Thus, "> シリコン <" tells the matcher that it requires an exact word matching against a word in the text.

4.2 Template pattern search

4.2.1 Template pattern matcher

The template pattern matcher identifies typical expressions to be extracted from the text that frequently appear in the corpus. The patterns are defined as pattern matching rules using regular expressions.

The pattern matcher is a finite-state automaton similar to the pattern recognizer used in the MUC-4 FASTUS system developed at SRI [Hobbs et al. 92]. In TEXTRACT, state transitions are driven by segmented words or grouped units from the preprocessor. The matcher identifies all possible patterns of interest in the text that match defined patterns. It must ignore unnecessary words in the pattern to perform successful pattern matching for various expressions.

4.2.2 Pattern matching rules

Fig. 2 shows a defined pattern in which an arbitrary string is represented as "@string" along with its corresponding English pattern.[2] Specifically, a variable starting with "@CNAME" is called the company-name variable, used where a company name is expected to appear. For example, "@CNAME_PARTNER_SUBJ" matches any string that likely includes at least one company name acting as a joint venture partner and functioning as a subject in the sentence.

The pattern "は | が:strict:P" tells the pattern matcher to identify the word, where "は" or "が" are grammatical particles that serve as subject case markers. The default type "strict" requires an exact string match, whereas "loose" allows a partial string match. Partial string matching is useful when compound words must be matched to a defined pattern. A joint venture, "提携:loose:VN", whose part of speech is verbal nominal, matches compound words such as "企業提携" (corporate joint venture) as well as "提携" (joint venture).

[2]This English pattern is used to capture expressions such as "XYZ Corp. created a joint venture with PQR

```
(JointVenture1 6
   @CNAME_PARTNER_SUBJ
   は|が:strict:P
   @CNAME_PARTNER_WITH
   と:strict:P
   @SKIP
   提携:loose:VN)
```
(a) A matching pattern for Japanese

```
(JointVenture1 3
   @CNAME_PARTNER_SUBJ
   create::V
   a joint venture:NP
   with::P
   @CNAME_PARTNER_WITH)
```
(b) A matching pattern for English

Fig. 2: A matching pattern for (a) Japanese and (b) English

The first field in a pattern is the pattern name followed by the pattern number. The pattern number is used to decide whether or not a search within a given string is necessary. To assure efficiency with the pattern matcher, the field designated by the number should include the least frequent word in the entire pattern ("提携" for Japanese and "a joint venture" for English in this case).

4.2.3 Pattern selection

Approximately 150 patterns were used to extract various concepts in the Japanese joint ventures domain. Several patterns usually match a single sentence. Moreover, since patterns are often searched using case markers such as "は", "が", and "と", which frequently appear in Japanese texts, even a single pattern can match the sentence in more than one way when several of the same case markers exist in a sentence. However, since the template generator accepts only the best matched pattern, choosing a correctly matched pattern is important. The selection is done by applying three heuristic rules in the following order:

- select patterns that include the most number of matched company-name variables in which there is at least one company name,
- select patterns that consume the fewest input segments (the shortest string match), and
- select patterns that include the most number of variables and defined words.

Another important feature of the pattern matcher is that rules can be grouped according to their concept. A rule name "JointVenture1" in Fig. 2, for example, represents a concept "JointVenture". Using this grouping, the best matched pattern can be selected from matched patterns of a particular concept group instead of choosing from all the matched patterns. This feature enables the discourse and template generation processes to look at the best information necessary when filling in a particular slot.

5 EXAMPLE OF THE INFORMATION EXTRACTION PROCESS

This section describes how concepts and patterns identified by the matcher are used for template filling. Concepts are often useful to fill in the "set fill" (choice from a given set) slots. An entity type slot, for example, has four given choices: COMPANY, PERSON, GOVERNMENT, and OTHER. The matcher assigns concepts related to each entity type except OTHER. Thus, from the given set, the output generator chooses an entity type corresponding to the identified concept. There are cases when discourse processing is necessary to link identified concepts and patterns. The following text: "X Inc. created a joint venture with Y Corp. last year. X announced yesterday that it terminated the venture." is used to describe the extraction process illustrated in Fig. 3.

In the preprocessing, two company names in the first sentence "X Inc." and "Y Corp." are identified either by MAJESTY or the name recognizer. In the first sentence, the template pattern search locates the JointVenture1 pattern shown in Fig. 2. Now, the JOINT-VENTURE concept between "X Inc." and "Y Corp." is recognized. In the second sentence,

Inc."

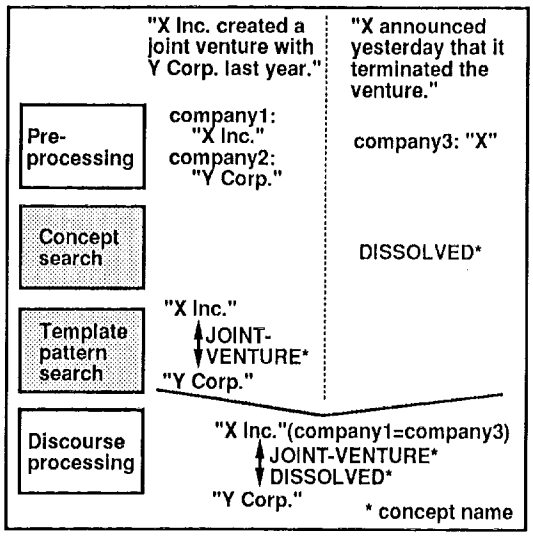

Fig. 3: Example of the information extraction process

the company name "X" is also identified by the preprocessor.[3] Next, the concept "DISSOLVED" is recognized by the key word *terminate* in the concept search. (The key word list is shown in Section 4.1.) After sentence-level processing, discourse processing recognizes that "X" in the second sentence is a reference to "X Inc." found in the first sentence. Thus, the "DISSOLVED" concept is joined to the joint venture relationship between "X Inc." and "Y Corp.". In this way, TEXTRACT recognizes that the two companies dissolved the joint venture.

Suppose that the second sentence is replaced with another sentence: "Shortly after, X terminated a contract to supply rice to Z Corp.". Although it does not mention the dissolved relationship nor anything about "Y Corp.", the system incorrectly recognizes the dissolved joint venture relationship between "X Inc." and "Y Corp." due to the existence of the word *terminate*. When this undesirable matching is often seen, more complicated template patterns must be used instead of the simple word list. A dissolved concept, for example, could be identified using the following template pattern:

```
(Dissolved1 2
   @CNAME_PARTNER_SUBJ
   dissolve | terminate | cancel::V
   @skip
   venture::N
   @CNAME_PARTNER_WITH).
```

Then, discourse processing must check if companies identified in this pattern are the same as the current joint venture companies in order to recognize their dissolved relationship.

6 OVERALL SYSTEM PERFORMANCE

A total of 250 newspaper articles, 100 about Japanese microelectronics and 150 about Japanese corporate joint ventures were provided by ARPA for use in the TIPSTER/MUC-5 system evaluation. Five microelectronics and six joint ventures systems were presented in the Japanese system evaluation at MUC-5.[4] Scoring was done in a semi-automatic manner. The scoring program automatically compared the system output with answer templates created by human analysts, then, when a human decision was necessary, analysts instructed the scoring program whether the two strings in comparison were completely matched, partially matched, or unmatched. Finally, it calculated an overall score combined from all the newspaper article scores. Although various evaluation metrics were measured in the evaluation [Chinchor and Sundheim 93], only the following error and recall-precision metrics are discussed in this paper. The basic scoring categories used are: correct (COR), partially correct (PAR), incorrect (INC), missing (MIS), and spurious (SPU), counted as the number of pieces of information in the system output compared to the possible (answer) information.

(1) Error metrics

- Error per response fill (ERR):

$$\frac{wrong}{total} = \frac{INC + PAR/2 + MIS + SPU}{COR + PAR + INC + MIS + SPU}$$

[3] When it is an unknown word to the preprocessor, the discourse processor identifies it later.

[4] These numbers include TEXTRACT, an optional system of GE-CMU SHOGUN.

Table 1: Scores of TEXTRACT and two other top-ranking official systems in TIPSTER/MUC-5

domain	ERR	UND	OVG	SUB	REC	PRE	P&R
TEXTRACT (JME)	59	43	28	12	51	63	56.3
System A (JME)	58	30	38	14	60	53	56.3
System B (JME)	65	54	24	12	40	66	50.4
TEXTRACT (JJV)	50	31	23	12	60	68	63.8
System A (JJV)	54	36	27	12	57	64	60.1
System B (JJV)	63	51	23	12	42	67	52.1

TEXTRACT's scores submitted to MUC-5 were unofficial.
JME: Japanese microelectronics domain
JJV: Japanese corporate joint ventures domain

- Undergeneration (UND):

$$\frac{MIS}{possible} = \frac{MIS}{COR + PAR + INC + MIS}$$

- Overgeneration (OVG):

$$\frac{SPU}{acutual} = \frac{SPU}{COR + PAR + INC + SPU}$$

- Substitution (SUB):

$$\frac{INC + PAR/2}{COR + PAR + INC}$$

(2) Recall-precision metrics

- Recall (REC):

$$\frac{COR + PAR/2}{possible}$$

- Precision (PRE):

$$\frac{COR + PAR/2}{acutual}$$

- P&R F-measure (P&R):

$$\frac{2 * REC * PRE}{REC + PRE}$$

The error per response fill (ERR) was the official measure of MUC-5 system performance. Secondary evaluation metrics were undergeneration (UND), overgeneration (OVG), and substitution (SUB). The recall, precision, and F-measure metrics were used as unofficial metrics for MUC-5.

Table 1 shows scores of TEXTRACT and two other top-ranking official systems[5] taken from the TIPSTER/MUC-5 system evaluation results [MUC-5 93].[6] TEXTRACT performed equally with the top-ranking systems in the two Japanese domains.

Since the TEXTRACT microelectronics system did not include a template pattern search or discourse processor to help differentiate between multiple semiconductor processes of the same kind, it reported only one object for each kind of manufacturing process, even when multiple objects of the same kind existed in the article. This resulted in the lower scores in the microelectronics domain than those of the joint ventures domain.

This pattern matching architecture is highly portable across different domains of the same language. The TEXTRACT microelectronics system was developed in only three weeks by one person by simply replacing joint venture concepts and key words with representative microelectronics concepts and key words.

7 CONCLUSION

In the Japanese microelectronics and corporate joint ventures domains, TEXTRACT performed equally with the top-ranking official systems at the TIPSTER/MUC-5 system evaluation. Although performance of pattern matching must be evaluated, the high performance of TEXTRACT suggests that the pattern matcher worked well in extracting information from the text. The pattern matcher

[5]TEXTRACT processed only Japanese text, whereas the two other systems processed both English and Japanese text.

[6]TEXTRACT's scores submitted to MUC-5 were unofficial. It was scored officially after the conference. The official scores showed slight differences from unofficial ones.

has not been tested to languages other than Japanese. It is expected to work to other languages with some minor modifications given that the input is segmented into primitive words tagged with their parts of speech.

The TEXTRACT Japanese microelectronics system was developed in only three weeks by one person. In spite of its simplicity, it showed the high performance. This result also suggests that the pattern matching architecture is highly portable across similar domains of the same language, thus facilitating rapid system development. Developing and maintaining TEXTRACT's pattern matching based architecture is easier and less complex than that of a full parsing system, as experienced in the early stage of SHOGUN system development [Jacobs 93b].

Corpus analysis took about half of the development time, since only a KWIC (Key Word In Context) list and a word frequency tool were used to acquire the concept-word lists and the template patterns. Using good statistical corpus analysis tools will shorten the development time and promise a high performance. The tools should not only collect patterns of interest with context, but also give statistical data to show how well defined patterns are working when they are applied in the system.

At MUC-5 meeting, P&R F-measure of one of the top-ranking systems was claimed to be close to the human performance [Jacobs 93b].[7] To match the system performance of a pattern matching system to human performance, the preprocessor must recognize expressions to be extracted at nearly 100% accuracy given that other components simply merge information and generate output.

Acknowledgements

The authors wish to express their appreciation to Jaime Carbonell, who provided the opportunity to pursue this research at the Center for Machine Translation, Carnegie Mellon University. Thanks are also due to Teruko Mitamura and Michael Mauldin for their many helpful suggestions.

[7] The human performance was estimated to be recall and precision of about seventy to eighty.

References

[Chinchor and Sundheim 93] Chinchor, N. and Sundheim, B. (1993). MUC-5 Evaluation Metrics. *Notebook of the Fifth Message Understanding Conference (MUC-5).*

[Hobbs et al. 92] Hobbs, J., Appelt, D., et al. (1992). FASTUS: A System for Extracting Information from Natural-Language Text. *SRI International, Technical Note No. 519.*

[Jacobs 93a] Jacobs, P. (1993). TIPSTER/SHOGUN 18-Month Progress Report. *Notebook of the TIPSTER 18-Month Meeting.*

[Jacobs 93b] Jacobs, P. (1993). GE-CMU: Description of the Shogun System Used for MUC-5. *Notebook of the Fifth Message Understanding Conference (MUC-5).*

[Kitani 91] Kitani, T. (1991). An OCR Post-processing Method for Handwritten Japanese Documents. In proceedings of *Natural Language Processing Pacific Rim Symposium*, pp. 38-45.

[Kitani and Mitamura 93] Kitani, T. and Mitamura, T. (1993). A Japanese Preprocessor for Syntactic and Semantic Parsing. In proceedings of *Ninth IEEE Conference on Artificial Intelligence for Applications*, pp. 86-92.

[Kitani and Mitamura 94] Kitani, T. and Mitamura, T. (1994). An Accurate Morphological Analysis and Proper Name Identification for Japanese Text Processing. *Journal of Information Processing Society of Japan, Vol. 35*, No. 3, pp. 404 - 413.

[Kitani 94] Kitani, T. (1994). Merging Information by Discourse Processing for Information Extraction. In proceedings of *Tenth IEEE Conference on Artificial Intelligence for Applications*, pp. 412-418.

[MUC-5 93] (1993). System Descriptions. *Notebook of the Fifth Message Understanding Conference (MUC-5).*

[TIPSTER 92] (1992). Joint Venture Template Fill Rules. *Plenary Session Notebook of the TIPSTER 12-Month Meeting.*