KNOWLEDGE REPRESENTATION
AND MACHINE TRANSLATION

Susumu Sawai, Hiromichi Fukushima,
Masakatsu Sugimoto and Naoya Ukai

Development Division, Computer Systems
Fujitsu Limited
Nakahara-ku, Kawasaki
Japan

This paper describes a new knowledge representation called
"frame knowledge representation-0" (FKR-0), and an experi-
mental machine translation system named ATLAS/I which uses
FKR-0.
The purpose of FKR-0 is to stored information required for
machine translation processing as flexibly as possible, and
to make the translation system as expandable as possible.

1. INTRODUCTION

Preliminary research on machine translation (MT) started soon after computers
became available.

MT research was prevalent in the USA during the early 1960s. However, the con-
clusions of the ALPAC report published in 1966 opposed funding for MT research
and resulted in the general discontinuance of MT research in the USA[1].

The effort is more concerted in countries where MT systems are more necessary than
in the USA. For example, the use of both French and English in Canada and the
multilingual use of formal documents in the EEC present pressing demands for
practical MT systems.

The SYSTRAN system (produced by Latsec Incorporated) has been applied in the
following areas:
1) Private companies in Canada use it for translating engineering documents from
    English into French.
2) NASA used it to communicate with the crews of the Apollo and Soyuz spaceships,
    translating between Russian and English.
3) The EEC uses it for examining the feasibility of other MT systems[2].

Other systems currently being used include the METEO system which translates
English weather reports into French in Canada, and the WEIDNER and LOGOS systems
produced by private firms in the USA. Recently, there has been a revival of
interest in MT systems in the USA, partly because of significant advances being
made in artificial intelligence (AI) research.

The future development of MT systems is ensured by the total integration of high-
performance computers, new man-machine interface designs, new software methodolo-
gies, and progress in knowledge engineering[3].

The language barrier in Japan in far greater than in the EEC or Canada, because
Japanese is an isolated language. There is a large demand for document translation
in Japan.

The fifth generation system project in Japan aims to build the artificial intelli-
gence machine that will process natural languages. This machine will perform
language translation at least 90% of it automatically, such that the cost of a
translation job could be reduced by 70%.

2. PROBLEMS AND SOLUTION

2.1 Methodological problems in a machine translation system

Basically, a machine translation system consists of three components: a dictionary (lexicon), grammar (translation rules), and the translation program (algorithm).

The major methodological problem in machine translation systems is how to separate the translation program from the grammatical rules. The advantage of this separation is that the program can be used for various languages and grammars without modification; that is, it is language-independent. However, there are practical problems in separating the grammar from the program, including difficulties in formulating complex rules for linguistic data and avoiding large storage requirements or heavy computation loads[4].

2.2 Solution by knowledge representation methodology

Artificial intelligence research on natural languages and knowledge representation progressed rapidly during the 1970s. In AI, "knowledge representation" is a combination of data structures and interpretive procedures that leads to "knowledgeable" behavior. A new type of machine translation system conceived by Drs. Y. Wilks and R. Schank appeared in the early 1970s. This type of system translates input text into the knowledge representation of semantic primitives intended to be language-independent[1].

At present, the major knowledge representation techniques are predicate logic, procedural representations, semantic networks, production systems (PS), and frames. In procedural representations, knowledge is contained in procedures (programs). The basic idea of production systems is a database consisting of rules, called production rules, in the form of condition-action pairs. A frame is a predefined internal relation.

This paper proposes an efficient knowledge representation method using frame-techniques to solve the above-described problems in machine translation systems.

3. FKR-0

Figure 1 shows the framework of frame knowledge representation-0 (FKR-0)[5].

In the FKR-0 knowledge representation method, a production system is combined with a procedual representation and is systematized into a state transition network. Rule representation frames and control frames are provided for the efficiency of system operation.

3.1 Rule representation frame

Figure 2 shows Jackendoff's semantic representation of verbs. Because Jackendoff is a linguist, he did not propose any machine translation system, but his semantic representation provides a good frame work with a clear indication of the relationship between the actor and the action. It indicates that the verb "OPEN" can take two noun phrases; that the subject can be either of two noun phrases, NP (1) or NP (3); and that NP (2) is an instrument, "INST"[7].
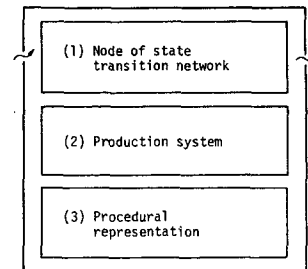

Fig. 1
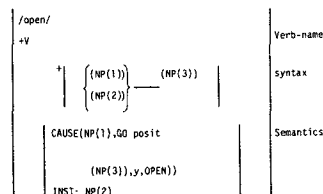Framework of knowledge
representation FKR-0.

Fig. 2
Jackendoff's semantic
representation

Figure 3 is an example of the
rule representation frame
used in FKR-0 for the Japanese
verb "指定する" (to specify).
The frame shows:
1) the verb name, which is
   the name of a node in
   the state transition
   network;
2) the relationship between
   the verb and one or
   more noun phrases;
3) the conditional process
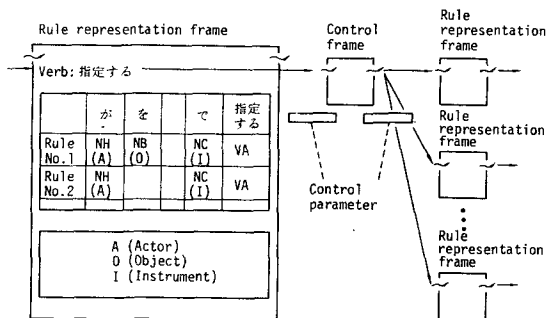   to be performed after
   the rules are applied.

Fig. 3
Communication of frames

This conditional process includes the judgement of the conditions required for
calling other frames.

Current FKR-0 specifications do not have the "cause" concept included in
Jackendoff's semantic representation, however, procedural representation is
planned for future FKR-0 editions.

3.2 Control frame

The FKR-0 system has control frames which supervise the rule representation frames
discussed in Sec. 3.1. Each pair of adjacent frames communicates by a control
parameter as shown in Fig.3.

The roles of the control frame are to resolve one rule frame into several subrule
frames and to control the calling sequence of these frames. This is one solution
which overcomes the disadvantages inherent in production system (PS) methodology;
i.e. inefficency of program execution and opaqueness of the control flow. A con-
trol frame addresses the rule frame by means of the contents of the control para-
meter. If the next frame is not specified, control returns to the top-level con-
trol frame called the "control nucleus".

3.3 Grammatical rules

ATLAS/I is an experimental machine translation system in which grammatical rules
are specified in FKR-0 representation. ATLAS/I currently has several control
frames and the several types of rule representation frames. These are examples:

1) Noun frame

   Example: ((STATE (NOUN   ) → STATE (CONTROL1)))
             (R (COND (NH X4   ) ND) (PARM (O   W1)))

   A noun phrase "H4" is formed by combining a noun denoting a human being "NH"
   (太郎) and postposition "X4" (が). The function "PARM" is the mapping
   ·function from graph to graph which can be used by the analysis, translation,
   and synthesis processes.

2) Verb frame

   Example: ((STATE (OPEN   ) → STATE (CONTROL5)) DEMON)
             (R (COND (H4 C7 VO) A')   (PARM (I'_VO W@1@2@3))))
             DEMON:PROC; /*ACTOR AND INST ARE SLOTS.*/
             ACTOR=WORD(2);/*A WORD(*) IS THE CONTENTS.*/
             INST=WORD(3); /*OF THE STACKS.   */
             END;

The surface case structure "H4 + C7 + VO" is changed to a deep case structure "A' + I' + VO". DEMON is a procedure. A noun phrase "C7" is a combination of the concrete noun "NC" (カギ), and postposition "X7" (で). If the surface case structure is "H4 + C7 + VO", a verb "VO" (open) has an agent case "A'" and an instrument case "I'". A variable "WORD" designates a slot in the stack. Figure 4 shows the structure of the grammatical rules.
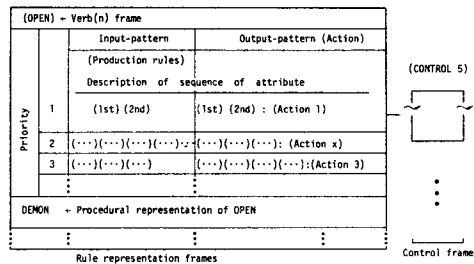

Fig. 4
Structure of the grammartical rules

Fig. 5
Model of ATLAS/I

3.4 Model of ATLAS/I

Figure 5 is a simplified model of ATLAS/I which includes an input tape, an output tape, a stack, a control section, a dictionary, a register, and grammatical rules (rule representation frames and control frames). When scanning an input tape, the stack is used as a table for temporary storage; at reduction, it is used as a table with attributes and equivalents. The dictionary is a table with words, attributes, and equivalents and is used as a table for lexical rules. The word "太郎", for example, is stored as (太郎 , noun, Taro) in the dictionaries. The character strings "太郎がカギで開ける。", for example, are stored in the input tape. The control parameter is set in the register.
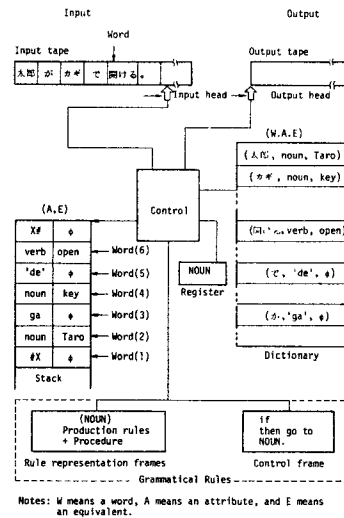
3.5 Initial state of ATLAS/I Model

"NOUN" is set in the register as an initial value. Grammatical rules have pre-defined values. The input head points to the left most position of the input tape. The output tape is blank. The output head points to the leftmost position of the output tape. The initial value of the slots in the stack is (#X, φ), meaning the top of the sentence and null string (φ).

3.6 Flow of ATLAS/I Model

3.6.1 Phase A: sentence processing

In phase A, one sentence of a text is translated.

1) State (1): scan

The words in the input tape are scanned by the input head and the dictionary is accessed to determine the attributes and equivalents. When found, these attributes and equivalents are stored in the stack, and then the input head

advances one word to the right. When the input tape is scanned, the stack is used as a table by the control section.

The control section scans the words of the input tape. If a period "." is encountered, it stops scanning and stores (X#, φ) in the stack.

The rule representation frame that is pointed to by the control parameter is referenced, and the control causes a translation from state (1) to state (2).

2)   State (2): reduction and code generation

The control section checks if the slots in the stack are (#X, φ), (SS, a character string), and (X#, φ). If so, there is a transition from state (2) to state (4); if not, the control section checks whether the attributes in the stack match those in the input pattern of the production rule (P). If not, the control causes a transition to the state specified by the rule representation frame.

If matched rule (P) does not exist and if the rule representation frame does not specify the new state, there is a transition to the default state specified by the top-level control frame called the "control nucleus". If matched rule (P) exists, the equivalents (SE) in the stack whose attributes (SA) match the attributes of the rule (P) input pattern are used as parameters of the rule (P) action function. Figure 8 is a general diagram of the organization of the grammar. The input and output patterns are organized according to the sequence of attributes. The action function of rule (P) pops the equivalents (SE) and attributes (SA) from the stack, and pushs the attributes of the rule (P) output pattern and the character strings created by this action function as the new slots (SA', SE'), whose number equals that of the rule (P) output pattern. The control returns to state (2).

3)   State (3): Frame transition

The control checks the control frame, determines the name of the next rule representation frame, and stores this name in the register. After selection of this rule representation frame control passes to state (2).
Remark: The control frame determines the termination of phase A. At termination, the control causes a transition to phase B.

4)   State (4): accept

Control pops the character string of the slot (SS, a character string) from the stack and writes this string into the output tape.

3.6.2 Phase B: text processing

The text is translated in phase B. Control continues phase (A) until the input head arrives at the rightmost position of the input tape.

All text translation processes end when the input head arrives at the right-most position of the input tape.

4.   ATLAS/I MACHINE TRANSLATION SYSTEM

ATLAS/I is currently used in the limited application of translating software-related reports from Japanese into English. The number of sentence translation patterns is gradually increasing through the addition of grammatical rules and vocabulary.

4.1 Japanese to English machine translation

Machine translation involves three stages:
input of the original Japanese text, trans-
lation, and postediting of the translated
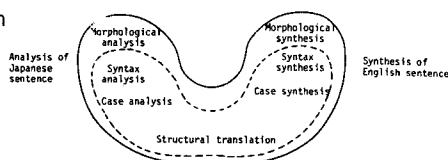English text (see Fig. 6).

Fig. 6
Flow of ATLAS/I

ATLAS/I currently integrates three processes: analysis of an original Japanese
sentence, structural translation from Japanese into English, and synthesis of
the English sentence. The "case" of noun phrases, which is the relation of noun
phrases to verbs, is checked while the syntax is analyzed. This case analysis is
performed with reference to the production rules. When the matching rule is found,
case and syntactic synthesis of the English are performed, and the synthesized
English sentence is printed. These production rules are defined in FKR-0.

The machine translation processes achieved through the use of FKR-0 are delincated
in Fig.6 by dotted lines.

5. CONCLUSION

The major methodological problem in a machine translation system is how to separate
the translation program from the grammatical rules. In ATLAS/I, grammatical rules
are stored in the new form of knowledge representation called FKR-0. In FKR-0, a
production system is combined with a procedural representation and systematized
into a state transition network. Rule representation frames and control frames
are provided for efficient system operation.

ATLAS/I is currently operational with about 5000 words and 400 grammatical rules
for translating software-related reports from Japanese into English.

FKR-0 allows the system to gradually increase the number of sentence patterns, and
this expansion is currently underway in the ATLAS/I system.   .

6. ACKNOWLEDGEMENT

REFERENCES

[1] Barr, A. and Feigenbaum, E.A.: The Handbook of Artificial Intelligence Vol. 1
    (William Kaufmann, Inc., Los Altos, California, 1981) 231-238.

[2] Hutchins, W.J.: Progress in Documentation-Machine Translation and Machine-
    Aided Translation, Journal of Documentation, 34, 2 (Jun. 1978)

[3] Nagao, M.: Machine Translation (In Japanese), Journal of Information Process-
    ing, Vol. 20, No. 10 (Oct. 1979) 896-902

[4] Bruderer, H.E.: Handbook of Machine Translation and Machine-aided Translation
    (North-Holland, Amsterdam, 1977)

[5] Sawai, S., Sugimoto, M. and Ukai, N.: Knowledge Representation and Machine
    Translation, FUJITSU Sci. Tech. J., 18, 1 (Mar. 1982) 117-134

[6] Jackendoff. R.: Toward an Explanatory Semantic Representation, Ling. Inq., 7
    (1976) 89-150.