[*Machine Translation Summit*, September 17-19, 1987, Hakone Prince Hotel, Japan]

HICATS/JE : A Japanese-to-English Machine Translation System Based on Semantics

Hiroyuki Kaji
Systems Development Laboratory, Hitachi, Ltd.
Asao-ku, Kawasaki 215, Japan

1. Introduction

We have been conducting a research and development aiming at practical machine translation systems. Our first product, HICATS/JE (Hitachi Computer Aided Translation System / Japanese to English), has been put on the market since May, 1986. The system can be used on HITAC M series computers / VOS3 operating system. It is intended to be utilized in translating documents primarily in scientific and technical field, e.g. manuals of products, research papers and patent abstracts.

The system has the following features:
(1) The semantics directed intermediate representation which bridges the structural gap between Japanese and English.
Disambiguation technique based on semantic features which improves the quality of translation.
(3) Implementation by production system approach which enables the sophisticated grammar to be efficiently developed.
(4) Utility functions and user friendly interface which facilitate cooperative use of the system and thus improve the total efficiency of translation process including pre- and post-editing.

This paper describes the core technologies developed for the system, i.e. the linguistic model, the translation process and the grammar description language.

2. Linguistic Model

Taking account of the characteristics of Japanese as well as the large structural gap between Japanese and English, we adopted a semantics directed dependency structure as the intermediate representation linking the source and target languages. We call it Dependency Graph. The primary meaning of a sentence is represented in a digraph of which nodes and labeled arcs represent concepts and semantic relationships among concepts respectively. The typical semantic relationships are case relationships, e.g. agent, object, goal, instrument, etc. The peripheral meaning conveyed in a sentence, e.g. tense, aspect, modality, emphasis, focus, etc., is represented as an attribute of a pertinent node.

The syntactic structures of the Japanese and English languages are quite different. If syntax directed intermediate representation such as a phrase structure was adopted, a complicated transfer component would be necessitated which depends on both the source and target languages. The dependency graph is comparatively language independent. For example, the following Japanese sentence [1] and its English equivalent [2] are mapped onto the same dependency graph [#].

```
  [1] KARE WA KUREYON DE E WO     KAITA.
      (he)   (crayon)  (picture) (drew)
  [2] He drew a picture with a crayon.
  [#] draw ( Agent: he, Object: picture, Instrument: crayon )
```

The dependency graph behaves as a pivot language for fairly large percentage of sentences. It is favorable from the viewpoints of not only the efficiency of development but also the extendability to other language pairs.

The Japanese language has such characteristics as flexible word order and variety of postpositional words. We can commonly find a number of sentences which are superficially different but have almost the same meaning. For example, the sentences [1] and [1'] have the same essential meaning, although they are different in the word order and the postpositional word.

```
    [1'] KUREYON DE KARE GA E WO     KAITA.
          (crayon)  (he)   (picture) (drew)
```

However, [1'] is also mapped onto [#]. Elimination of superficial differences in the intermediate representation enables the realization of a source language independent generation of target language.

Frequent omission of phrases is another characteristic of the Japanese language. On the other hand, omission is strictly restricted in English. Accordingly it is necessary to recover omitted elements in the intermediate representation. The dependency graph is favorable to this requirement, as it facilitates case pattern driven analysis.

3. Translation Process

The translation process is divided into the following steps: (l)morphological analysis, (2)syntactic analysis, (3)semantic analysis, (4)transformation of dependency graph, (5)syntactic generation and (6)morphological synthesis. The transformation of dependency graph has an actual effect just in case the Japanese sentence and its English equivalent has a difference at conceptual level. Fig. illustrates the translation process.

3.1 Morphological analysis

As a Japanese sentence is written without inserting delimiters between words, it is difficult to automatically segment a sentence into words. The segmentation is performed in depth first searching manner. Each pair of successive words is subjected to validity checking based on the part-of-speech adjacency matrix. Some of incorrect segmentations are rejected by this method. The morphological analysis component outputs only one solution. Ambiguity such as homograph and multiple parts of speech is maintained in the solution. It is disambiguated in the succeeding steps.

3.2 Syntactic analysis 1)

The syntactic structure of a Japanese sentence can be conveniently grasped by governor-dependent relations among bunsetsu's. A bunsetsu is the minimal phrasal element which consists of a content word and function words succeeding it. After disambiguation of multiple parts of speech and determination of the governor type and the dependent type of each bunsetsu, the syntactic dependency structure is analyzed.

In most cases, a number of solutions are syntactically possible. One method to this problem is to obtain all the solutions in the syntactic analysis step and then choose the best one after semantically interpreting each solution. It will be effective, if the system can utilize various kinds of real world knowledge to evaluate the plausibility of each solution. However, it is quite difficult to construct such a knowledge base for the whole domain. We adopted a practical method to obtain the most plausible syntactic structure by incorporating heuristic rules based on semantic features into syntactic analysis. This method is applied not only to structural ambiguities but multiple parts of speech, omitted or reduced expressions, etc.

Here we show some examples of semantic feature based processing.

(1) Disambiguation of multiple parts of speech

```
   [3] TEKISUTO WO YOMIKOMI, FAIRU NI KAKUNOUSURU.
        (text)     (read)     (file)  (store)
```

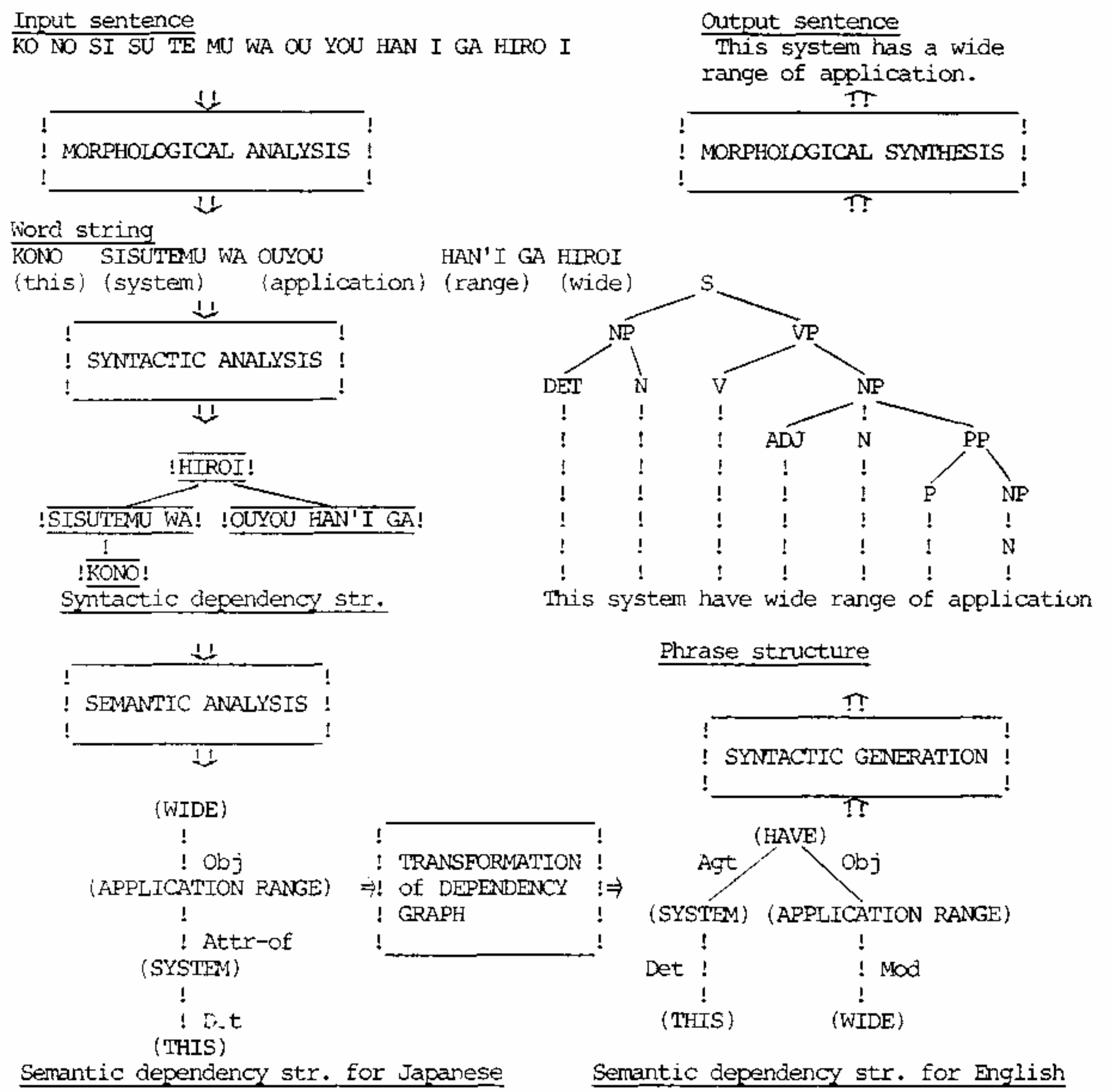
Input sentence
KO NO SI SU TE MU WA OU YOU HAN I GA HIRO I
MORPHOLOGICAL ANALYSIS
Word string
KONO SISUTEMU WA OUYOU HAN'I GA HIROI
(this) (system) (application) (range) (wide)
SYNTACTIC ANALYSIS
HIROI
SISUTEMU WA
OUYOU HAN'I GA
KONO
Syntactic dependency str.
SEMANTIC ANALYSIS
(WIDE)
Obj
(APPLICATION RANGE)
Attr-of
(SYSTEM)
Det
(THIS)
Semantic dependency str. for Japanese
TRANSFORMATION of DEPENDENCY GRAPH
Output sentence
This system has a wide range of application.
MORPHOLOGICAL SYNTHESIS
S
NP
VP
DET
N
V
NP
ADJ
N
PP
P
NP
N
This system have wide range of application
Phrase structure
SYNTACTIC GENERATION
(HAVE)
Agt
Obj
(SYSTEM)
(APPLICATION RANGE)
Det
Mod
(THIS)
(WIDE)
Semantic dependency str. for English

Fig. Translation Process

The part of speech of 'YOMIKOMI', which may be verb or may be noun, can be determined by semantic feature checking. If 'YOMIKOMI' was a noun, it should constitute a conjunctive structure with 'FAIRU'. But the semantic features of 'YOMIKOMI' and 'FAIRU' are different. That is, 'YOMIKOMI' is [+action] but 'FAIRU' is [-action]. Accordingly the assumption of conjunctive structure is rejected.

(2) Determination of governor

```
   [4] HANKEI GA 5cm NO EN WO   EGAKU.
       (radius)        (circle) (describe)
```

The solution that 'HANKEI GA' is governed by 'EGAKU' is syntactically possible. This solution implies that 'HANKEI' is the agent of 'EGAKU'. But it is unlikely, as 'HANKEI' is [-operation_ability]. Furthermore 'HANKEI' is [+attribute], which implies that it is likely to co-occur with a quantitative expression like '5cm'. Therefore not 'EGAKU' but '5cm' is chosen as the governor of 'HANKEI'.

(3) Determination of conjunctive structure

[5] NIHONGO NO KAISEKI TO EIGO NO SEISEI
(Japanese) (analysis) (English) (generation)

The correct conjunctive structure can be chosen by examining the syntactic and semantic similarity between conjunctive noun phrases. The second constituent of the conjunctive structure is either 'EIGO' or 'EIGO NO SEISEI' of which head noun is 'SEISEI', while the head noun of the first constituent is 'KAISEKI'. Both 'KAISEKI' and 'SEISEI' are [+action] but 'EIGO' is [-action]. Accordingly 'SEISEI' is preferred to 'EIGO' for the counterpart of 'KAISEKI'. The modification scope of 'NIHONGO NO' is either 'KAISEKI' or 'KAISEKI TO EIGO NO SEISEI'. The former is preferred to the latter, as 'EIGO NO', which modifies 'SEISEI', is a bunsetsu of the same kind as 'NIHONGO NO'.

3.3 Semantic analysis [1)]

Most of the function words, which express semantic relationships among content words, have multiple meanings. The semantic features play an important role in determining the semantic relationships.

Obligatory cases are analyzed by consulting the case patterns. A case pattern prescribes the set of cases that a predicate inherently requires. Each case is marked by a postpositional word (surface case marker) and has a restriction on the semantic features of nouns. The specification of surface case markers is often sufficient to determine the deep cases. However, ambiguity is sometimes caused, by dropping of surface case markers. For example, an auxiliary postpositional word ('WA', 'MO', etc.) to express a certain nuance can be substituted for a case marking postpositional word. In an embedded sentence structure, the modified noun isn't accompanied with the postpositional word to mark its case on the modifying predicate. The semantic feature restriction is utilized to determine the deep cases for the sentences mentioned above. An example is given below.

[6] KEKKA WO KAKUNOUSURU ERIA
(result) (store) (area)

In this sentence, 'ERIA' can be both the agent and the goal syntactically. The latter is preferred to the former, as 'ERIA' is not only [-function] but [+place].

In optimal case analysis, the semantic features are still more important. For example, 'DE' has such a meaning as Place, Instrument, Cause, Manner, etc.

[7] KEISANKI DE MANYUARU WO HON'YAKUSURU. ----- Instrument
(computer) (manual) (translate)

[8] BUNPOU NO KAKUCHO DE SEIDO GA KOUJOUSITA.-----Cause
(grammar) (extension) (accuracy) (improve)

The semantic features [+tool] of 'KEISANKI' and [+volition] of 'HON'YAKUSURU' imply that 'KEISANKI' is the instrument of 'HON'YAKUSURU', while the semantic features [+action] of 'KAKUCHO' and [-volition] of 'KOUJOUSITA' imply that 'KAKUCHO' is the cause of 'KOUJOUSITA'.

A syntactic dependency does not always exist between words having a semantic relationship. Such a structure is caused by topicalization of case modifier, zero-pronominalization in complex sentence, etc. The semantic features and the case patterns give a clue to recognize the structure mentioned above. An example is shown below.

[9] KIKAI HON'YAKU WA JITSUYOUKA GA MUZUKASII.
(machine) (translation) (put to practical use) (difficult)

'KIKAI HON'YAKU' is syntactically one of the dependents of 'MUZUKASII' but semantically the object of 'JITSUYOUKA'. The semantic relationship is inferred from the features [+action] of 'JITSUYOUKA' and [+tough] of 'MUZUKASII'.

3.4 Transformation of dependency graph

The dependency graph is a fairly language-independent representation. However, the effect of an expression peculiar to the source language is sometimes left in the structure. In such a case, structural transformation is done in order to obtain a target language oriented representation. It is crucial for generating a natural sentence of the target language.

Japanese is a BE-type language while English is a DO-type language. A typical transformation is that from a BE-type language oriented structure to a DO-type language oriented structure. An example is seen in Fig.

Unification of concepts is another important transformation. An example is given below.

```
   [10] KOURITSU GA   YOI     ===    efficient
        (efficiency) (good)
```

The two nodes 'KOURITSU' and 'YOI' are merged into one. Because one English lexical unit expresses the same concept as the combination of two Japanese lexical units does.

3.5 Syntactic generation [2)]

English is a language which has strict restrictions on the word order. Therefore a phrase structure grammar is suitable for generating English sentences. The mapping from the dependency graph onto a phrase structure is rather straightforward, as the dependency graph representation is transformed into English oriented one in the preceding step. A phrase structure tree is generated in top down and recursive manner. Some of the attributes of the node in the dependency graph play an important role in selecting the sentence style.

One of the features of the syntactic generation is that the structure of English is selected independently of that of Japanese. For example, a sentential phrase in Japanese may be translated into a simple noun phrase, and vice versa. Such a flexible structure selection is realized without any transfer or conversion rules.

An important problem in English sentence generation is word selection for a concept having multiple English equivalents. A mechanism is incorporated into the generator to select an appropriate word by consulting the dictionary in which selectional restrictions are specified on semantic feature basis.

3.6 Morphological synthesis

The leafs of the phrase structure tree generated in the preceding step contain the words which constitute the output sentence. Furthermore each node of the tree has some attributes, including tense, aspect and number, which it inherited from the corresponding node of the dependency graph. The output sentence is obtained by synthesizing the word form according to the attributes.

4. Grammar Description Language [3)]

A practical machine translation system requires a very large grammar. The variety and complexity of linguistic phenomena prohibits completing the entirety of the grammar. Not only the dictionary but also the grammar needs to be continually enhanced and maintained. Any practical machine translation systems cannot be realized without an efficient grammar writing system. We have developed an grammar description language (GDL) having powerful description capability and high comprehensibility. The translation system has been implemented in a form of software to interpret the grammar written in GDL.

4.1 Data structure and grammatical rule

GDL handles a graph composed of nodes with attributes and labeled arcs. And a grammatical rule is written in a form of graph transformation rule.

A node corresponds to a bunsetsu or content word, and the following are treated as attributes of a node: function word, part of speech, inflection form, governor type, dependent type, semantic feature, usage code, tense, aspect, modality, number, definiteness, nuance, etc. Attributes are classified into two types: scalar and set. A scalar-type attribute has only one value for each node. A set-type attribute has a number of values for each node. An Arc is labeled with both the surface case marker and the deep case code.

A grammatical rule consists of a condition part and an action part. The condition part specifies a subgraph pattern, and the action part specifies transformation to be made on the subgraph. Conditions are described on the attributes as well as the linking topology. Some conditions can be optional. Moreover, arbitrary number of repetitions of a subpattern can be included. The transformation of a subgraph consists of the following operations: creation, duplication and deletion of a node, alteration of the linking topology, and addition, duplication and deletion of attribute values.

4.2 Structuring of grammar and application control

A grammar is composed of various rules. The order of their application is determined by what linguistic phenomenon they relate to. GDL has facilities to structure the grammar and effectively control the application of the rules.

A grammar is decomposed into subgrammars. A subgrammar is a ordered set of rules relating to a particular linguistic phenomenon, e.g. disambiguation of multiple parts of speech, determination of the dependent type, deep case analysis for obligatory cases, etc. A number of control parameters are given to each subgrammar to specify how to apply the rules. For example, there are four options on the mutual relation among the rules, i.e. Exclusive, Concurrent, Dependent and Unrelated. Furthermore, it is possible to divide the rules of a subgrammar into subsets. The subset is called compound rule. The control parameters are also given to each compound rule. Thus GDL enables a flexible application control of the grammatical rules.

5. Concluding Remarks

The current system cannot cope with all sentences. However, it outputs a passable sentence, if a suitably pre-edited sentence is input. Accordingly the so-called sublanguage or controlled language approach is crucial for putting the system to practical use. We should take great interest in the development of a proper sublanguage and its supporting tools.

Acknowledgements: I would like to acknowledge Dr. Jun Kawasaki, Mr. Koichiro Ishihara and Dr. Tan Watanabe of SDL (Systems Development Laboratory) for their valuable advice and constant encouragement. Thanks are also due to the cooperation of Mr. Kikuo Yoshimura, Ms. Atsuko Koizumi, Mr. Takao Usui, Mr. Koichi Sakamoto, Mr. Jun'ichi Matsuda and other members at SDL and Software Works, Hitachi, Ltd.

References

1. H.Kaji and A.Isatsu: Dependency Structure Analysis of Japanese for Machine Translation, Proc. 30th Annual Convention IPS Japan, pp.1579-1580(1985)
2. H.Kaji and Y.Nitta: English Sentence Generation from Conceptual Dependency Diagram, Proc. 28th Annual Convention IPS Japan, pp.907-908(1984)
3. H.Kaji, K.Yoshimura and T.Usui: Grammar Description Language for Japanese-English Machine Translation System ATHENE/N, Proc. 31st Annual Convention IPS Japan, pp.1347-1348(1985)