Translation by Understanding: A Machine Translation System LUTE
Hirokato NOMURA, Shoza NAITO, Yasuhiro KATAGIRI, and Akira SHIMAZU
NTT Basic Research Laboratories, Musashino-shi, Tokyo, 180, Japan

Abstract
This paper presents a linguistic model for language understanding and describes its application to an experimental machine translation system called LUTE. The language understanding model is an interactive model between the memory structure and a text. The memory structure is hierarchical and represented in a frame-network. Linguistic and non-linguistic knowledge is stored and the result of understanding the text is assimilated into the memory structure. The understanding process is interactive in that the text invokes knowledge and the understanding procedure interprets the text by using that knowledge. A linguistic model, called the Extended Case Structure model, is defined by adopting three kinds of information: structure, relation, and concept. These three are used recursively and iteratively as the basis for memory organization. These principles are applied to the design and implementation of the LUTE which translates Japanese into English and vice versa.

1. Introduction
Since the early 1970s, a variety of approaches to language understanding have been proposed. In particular, the importance of knowledge organization has been emphasized, and linguistically structured knowledge such as Script [1] and knowledge representation frameworks such as Frame [2] and Semantic Network [3] have been proposed. At the same time, the linguistic approach has been adopted to reveal the discourse structure, the cognitive approach has attempted to explain phenomena such as focus, topic and intention, and the formal semantic approach has been used to establish semantics based on the logical model theory.

We propose an interactive model between the memory structure and the text (or utterance) as a language understanding model. In the model, knowledge stored in the memory structure plays the principal role such that the text invokes knowledge and the understanding system interprets the text using that knowledge. The knowledge consists of linguistic knowledge and non-linguistic knowledge. They are closely related each other and incorporated into the memory structure simultaneously. As a result of understanding, the system assimilates the meaning structure of the text into its memory structure. The bases for representing the knowledge are structure, relation and concept which are the fundamental components for constructing and representing the memory structure including the meaning structure of a sentence. For the purpose of clear definition of linguistic information, a linguistic model, called the Extended Case Structure model (ECS), which is capable of treating the structures of complex sentences, is provided.

These principles have been applied to the design of a new version of the experimental machine translation system called LUTE (Language Underst aer, Translator and Editor) [4]. This paper deals mainly with the current Japanese-English version of LUTE (LUTE-JE version-1) [5]. LUTE has following processing characteristics: 1) Not only syntactic but also semantic relations (dependencies) between modifiers and modificants are analyzed simultaneously. 2) All kinds of information such as syntactic patterns, meaning structures, lexical items, and knowledge are represented in a uniform framework, called Frame-Net work. 3) Analysis produces a "most plausible meaning structure" based on the prediction of syntactic structures and the integration of semantic structures. 4) Transfer is realized as a general framework for manipulating the frame network.
situation in which the utterance is carried out. Information in the LSP is used for filling in missing cases, and resolving anaphora. A discussion of the detailed procedure for the assimilation would be beyond the scope of this paper.

3. Extended Case Structure Model

3.1 General Framework

The Extended Case Structure Model (ECS) is a linguistic model for representing the meaning structures of the text. Thus the ECS presents a representation schema for the episodic memory. Figure 1 shows its fundamental construction. The traditional case structure (Fillmorean type) is a structure for a unit sentence which consists mainly of relations between nouns and a verb. This is not sufficient to represent structures of real sentences which sometimes have complex noun phrases, compound sentences, etc. Also, the ECS has to have facilities for representing other structures involving relations between a noun and a noun, a verb and a verb, etc. The ECS has been designed to integrate those structures into one linguistic model. Its nature is hierarchical as to the compoundness of constituents, iterative as to conjunction, and recursive as to embedding. Using these formalisms, the syntactic and semantic structures of sentences can be represented uniformly and correctly.

3.2 Semantic Structure In ECS

There are two types of semantic structures, composite and primitive structures. A composite structure is made by integrating semantic structures using semantic relations. A primitive structure, by definition, cannot be divided into further substructures. In general, a single word corresponds to a primitive structure, and a phrase corresponds to a composite structure. Since syntactic information can also contribute to define meaning structures, each semantic structure simultaneously incorporates not only meaning information but also syntactic information.

We do not assume a language-independent universal semantic representation. Thus, it is necessary to define a proper ECS for each language: Japanese ECS (J-ECS) [6] for Japanese language and English ECS (E-ECS) [7] for English language. In the translation process from Japanese into English, the analysis procedure generates a J-ECS for a Japanese sentence, and the transfer procedure generates an E-ECS corresponding to the J-ECS.

3.3 Semantic Relation in ECS

Semantic relation connects semantic structures and builds a larger semantic structure, ranging from a word structure to a sentence structure. Figure 2 shows types of semantic relations, and each of them can be explained briefly as follows:

1) Noun relation: Relationship between nouns; Examples are whole-part, upper-lower, possession, material, etc.
2) Case relation: Relationship between a case element and a predicate; Examples are object, agent, instrument, place, etc.
3) Embedded relation: Relationship between an embedded sentence and a noun phrase, which can be categorized into three types: a) case relation between a modified noun phrase and the predicate in a modifier embedded sentence, b) noun relation between a modified noun phrase and a noun phrase in a modifier embedded sentence, and c) an appositive or subsidiary relation between a modified noun phrase and a modifier embedded sentence.
4) Conjunctive relation: Relationship between sentences; Examples are cause-result, time-advance, assumption, etc.

3.4 Concepts in ECS

Concepts are associated with structures mentioned above. Among them, concepts associated with word structures represent word meanings which appear when the words are used in a sentence. A word meaning is represented by principal concepts, supplementary concepts, and their semantic dependences. Principal and supplementary concepts are defined by using semantic categories, and prepared for nouns, adverbs, verbs, adjective-verbs, and modalities as shown in Figure 3. Semantic dependencies are defined by using semantic relation frames and semantic structure frames. Semantic categories, semantic relation frames, and semantic structure frames have the following characteristics: 1) There are two types of concepts: prototype and instance. Prototypes play a part of selectional constraint to define semantic dependency structures. Instances show an assimilated structure which satisfies the selectional constraints. 2) They show semantic commonness and analogy between two structures. This allows the system to share information and to provide facilities for paraphrase. 3) Semantic categories make up a hierarchical structure. This provides the system with inheritance ability and information sharing.

4. Dictionaries, Knowledge and Their Representation

4.1 Dictionary

There are two types of dictionaries in LUTE. Mono-lingual dictionaries are used in analysis and generation, while bi-lingual dictionaries are used in transfer. Mono-lingual dictionaries have the following information about words and concepts: 1) how the word is expressed, 2) how the word is used in the syntax of a sentence, and 3) what concept the word corresponds to. Bi-lingual dictionaries have information on the correspondence of concepts in different languages, and will be explained in section 6. (Note that concepts are defined here by associating structures which are generally languages dependent.) Figure 4 shows the contents of a word dictionary.

A word meaning can be regarded as an entry to the conceptual knowledge description. The LUTE dictionaries contain the following semantic information:

1) Semantic category (for word meanings): Principal concepts associated with the word meaning. Those for nouns and adverbs are used as selectional constraints in semantic relation analysis. Those for predicates are used to analyse modality.
2) Case frame (for predicate word meanings): Constraints and case relations which are applied to construct unit sentence semantic structures. There are three types of case frames: intrinsic for each predicate word meaning, common for several predicate word meanings, optional for outer case relations.
3) Noun relation frame (for noun word meanings): Constraints and semantic relationships which are applied to construct semantic structures made up of two nouns. Case frames are also used as a kind of object relation frames for predicate-type nouns.
4) Event relation frame (for predicate word meanings): Constraints and semantic relationships to be applied to construct complex sentence semantic structures. An example is the relation between the verb in a main clause and the verb in a subordinate clause.
5) Heuristics (for semantic categories and relation frames): This is used for resolving ambiguity of semantic categories, semantic relations, and semantic structures by linguistic information such
4.2 Knowledge

4.3 Frame-Network

5. Extended Case Analysis

6.1 Flow and Control in ECA
The third part describes the structure type to be produced by the process. It is represented by a two-dimensional vector, whose first element between modifier and modificant in the construction is analyzed using a grammar (CFG) rule augmented with a semantic relation analysis of the structure sequence which satisfies the above condition. This describes whether this structure pattern can be applied to the structure type in the ECS. An example of structure patterns for a unit sentence is shown in Figure 6.

A structure pattern consists of three parts: 1) the condition for applying the pattern, 2) the procedure for semantic structure analysis, and 3) newly integrated structure type. The first part describes whether this structure pattern can be applied to the structure sequence. The second part performs a semantic relation analysis of the structure sequence which satisfies the above condition. The third part describes the structure type to be produced by the above procedure. A structure pattern might be viewed as a context-free grammar (CFG) rule augmented with a semantic relation analysis. In this case, the condition part corresponds to the right hand side of the CFG rule, the integrated structure type part corresponds to the left hand side of it, and the procedure part can be seen as a procedure to derive the left hand side from the right hand side.

5.3 Structure Pattern

A structure pattern is a package of knowledge for predicting syntactic constructions between pairs of modifiers and modificants among the constituent structures of a sentence. Based on this prediction, an analysis procedure is invoked to analyze their semantic structures. If this analysis succeeds, the modifier/modificant pair is integrated into a new unified structure. Structure patterns are assigned to each structure type in the ECS. An example of structure patterns for a unit sentence is shown in Figure 6.

A structure pattern consists of three parts: 1) the condition for applying the pattern, 2) the procedure for semantic structure analysis, and 3) newly integrated structure type. The first part describes whether this structure pattern can be applied to the structure sequence. The second part performs a semantic relation analysis of the structure sequence which satisfies the above condition. The third part describes the structure type to be produced by the above procedure. A structure pattern might be viewed as a context-free grammar (CFG) rule augmented with a semantic relation analysis. In this case, the condition part corresponds to the right hand side of the CFG rule, the integrated structure type part corresponds to the left hand side of it, and the procedure part can be seen as a procedure to derive the left hand side from the right hand side.

5.3 Semantic Structure Analysis

For each constituent construction predicted, the semantic relation between modifier and modificant in the construction is analyzed using semantic relation frames. Depending on the differences in structure types of the modifier/modificant pair, different types of semantic relations can be analyzed. In addition, the word meanings of the word structure and the categories for the integrated structure can also be analyzed.

Semantic relation analysis can be explained by the analogy of a key and key-hole. A modifier has a number of possible key-holes, and a modifier can be regarded as the key which can match it. The procedure is to search for the best matching key hole for the key. The shapes of keys and key-holes are determined by syntactic (case particles) and semantic (semantic category) information.

The score given to the integrated structure represents the degree of syntactic and semantic mismatch recognized in the integration process. It is represented by a two-dimensional vector, whose first argument is for syntactic mismatch, and second is for semantic mismatch. At each stage of analysis, if syntactic constraint is not satisfied, two points are added to the syntactic mismatch score, and if it is satisfied by modal particles, one point is added to it. As for semantic constraints, if they are not satisfied, two points are added to the semantic mismatch score, and if they are satisfied through inheritance of semantic categories, one point is added to it.

5.4 Case Analysis

Case analysis is the process of matching a case instance and prototype cases in the case frame and of selecting the best matched prototype case. Then, the value of the case relation between the case instance and the predicate is determined to be the case relation of the selected prototype case.

A modifier element may have co-case slots. It is true that some modifiers are strongly associated with particular word meanings of predicate words. For example, a verb "hiku (Billy)" has multiple meanings, and its exact meaning in a sentence is determined when it occurs simultaneously with object cases such as "kaze wo hiku (the wind to catch a cold)", "jisho wo hiku (the dictionary) consult a dictionary" and "denwa wo hiku (the telephone to establish a telephone service)". When a modifier element definitely determines the word meaning of a modificant element, it is not efficient to test all possible word meanings of the modificant. Therefore, if the same case slot is shared by both a modifier and a modificant element, the meaning which shares this same case slot is selected as the word meaning of both elements without analyzing another possibilities.

5.5 Modality analysis [8]

The classification of modality information and the procedure for analyzing them have presented in the reference thus we will describe here only the outline. Modality analysis consists of the following three modules combined with case analysis and conjunctive analysis.

(1) Pre-case-analysis: A modality which causes a change in the case structure is analyzed at this stage. The case frame to be assigned to the predicate is modified by utilizing the result of this analysis before starting the case analysis. As for semantically ambiguous auxiliary verbs which are also related to the modification of the case structure, their role is only predicted at this stage, and after case analysis, the likelihood of the prediction is evaluated.

(2) Post-case-analysis: A modality whose analysis requires case structure information is analyzed at this stage as follows:

a) If the category of the modality expression is unique, this category is assigned to the meaning structure.

b) If a daemon (a procedure to resolve ambiguities using heuristics) is attached to the modality expression, it performs the following three tasks:

i) determining the operational scope of the modality, ii) adding the impliciative meaning caused by the modality word.

(3) Post-conjunctive-analysis: Following the conjunctive analysis between the subordinate clause and the main clause, this module is activated to determine whether the modality in the main clause also operates on the subordinate clause. For negation in the main clause, the transfer of negation is considered. Testing whether or not the modifier event is subsidiary to the occurrence of the main event is accomplished using the semantic relation frames assigned to the predicate of the main clause.

5.6 Determination of Word Meaning

Word meaning is an entry from a word to the conceptual network consisting of dictionary information and knowledge. Since a word has multiple word meanings, it is possible that the word might have multiple entries. The information available for the determination of word meaning in the accumulated situation (discourse information) and the accumulated word meanings (accumulated concepts). If no such information is available, a default value is borrowed as the most likely word meaning. In the early stage of semantic relation analysis, tentative word meanings are assumed. These word meanings may not
be accurate because they have been determined solely by the local analysis. It is possible that some of the rejected meanings at this stage might be more adequate as the exact word meanings for a given word in the context of the entire sentence. Therefore, the system must retain all possible word meanings as candidates so that it can change the meanings after obtaining enough information to determine the exact meaning.

5.7 Determination of Category

At the stage of building a meaning structure for a sentence, categories for each constituent structure are also determined. Categories for a structure are assigned the same as the categories of the head constituent. But if a structure is exocentric, categories for the structure can be obtained by some operation on its constituent substructures. For example, the category of "no mona no hetai (a toy soldier)" is human (therefore, animate).

In order to determine the categories of a semantically ambiguous structure or an exocentric structure, an attached procedure is invoked. For example, the Japanese noun "tame (目的)" is ambiguous because it has two categories, purpose and cause. To resolve this ambiguity, a daemon is invoked after the noun phrase containing "tame" is analyzed. This daemon performs the following heuristics:

1) If "tame" is followed by both a case particle "ni (に)" and a modal particle "h"(う)", then the semantic category of the modal predicate is determined to be "purpose".
2) If "tame" is succeeded by an embedded sentence and the predicate shows a perfective aspect (that is, the end part of the embedded sentence contains the auxiliary verb "ta (た)" or "teiru (為いる)"), or the semantic category of the predicate in "state", the category is determined to be "cause".
3) Otherwise, "purpose".

6. Transfer

6.1 Transfer Functions

Discrepancies among ECS's for different languages arise for several reasons. One is essential in nature. We believe that syntactic information should be preserved as far as possible in ECS. But semantically equivalent information is often reused differently in the syntax of different languages. Conceptual systems are also difficult in different language communities. These differences must be reflected in ECS's.

Transfer process should fill these gaps between the ECS's of two different languages. At the stage of building a language transfer system for Japanese to English, conceptual systems are also different in different language communities. These differences must be reflected in ECS's.

Transfer process should fill these gaps between the ECS's of two different languages. The transfer stage from Japanese to English, structures, relations and concepts in J-ECS are transferred into those in E-ECS. Since concepts and relations are integrated into structures, the transfer of concepts and relations is performed at the same time as the transfer of structures.

6.2 Transfer of elements of ECS

In the course of the transfer processes, ECS's in the source language are converted by recursively traversing original structures from top nodes, and creating corresponding target structures. So, the transfer process consists of transforming original structures into ECS's, i.e., concepts that make up the ECS and relations which hold among them.

There are cases which don't suit this scheme well, and hence require special treatment. They are idiosyncratic to lexical items and specific procedures have to be triggered by certain concepts included in the original structures. Idiosyncratic transformations include:

1) deletion: certain structures in the source structures are deleted and no counterpart structures are embodied in target structures; for example, compound structures are transferred into primitive structures, as in the transfer from "Sakura wa tama (百合花)

2) addition: certain structures that have no counterpart in the source structures are added to target structures; for example, primitive structures are transferred into compound structures, as in the transfer from "Samidore (朝日)

3) modification: source structures are non-trivially changed in the process of transfer, as in the transfer from Japanese adjective sentence "Iwae ga ou (大がお)

The transfer of concepts consists of 1) transfer of semantic categories, and 2) transfer of word meanings. A transfer dictionary for a pair of languages is prepared to give information on the correspondence between concepts in both languages. An entry of the dictionary consists of a pair of frames, that is, a source concept frame, a target concept frame, and a mediating frame. Since concept correspondence is in general not one-to-one, there may be several target concepts corresponding to one source concept and vice versa. Mediating frames provide information on conditions to make it possible to choose among alternatives. Concepts that would trigger idiosyncratic procedures have the information in the dictionary in the form of transfer rules.

Transfer of relations consists of transfer of four types of relations described in 3.3. Correspondence information is also placed in the transfer dictionary. But information on case relation transfer are stored as verbal concepts, since they might be specific to individual verbs or classes of verbs.

6.3 Transfer process

The transfer process is essentially a manipulation of frame networks. A rule-based system was devised to facilitate easy specification of the complex patterns of network manipulations. An example of the transfer rule is shown in Figure 7. Similar to structure patterns, a transfer rule consists of three parts: a matching part, execution part, and a return part. The matching part specifies the conditions under which the rule should be invoked. It also contains variables, which are bound during matching process and the information will be passed to and used in the later stage when the matching is successful. The execution part specifies the transfer of substructures and concepts, value assignment to the variables, further conditional branching, and other operations. Lisp code can be involved in this part. The return part specifies the target structure that has to be constructed and returned on the basis of the application of the entire rule.
The frame system presented here has a class-instance hierarchy, which adopts an "object-oriented" style of implementation for the frame network manipulation in the transfer process. Transfer rules specifying how the network should be handled are written for each type of structures. These are converted into executable forms, and attached to class frames of the structure as methods. When the top node of the input ECS is given a "transfer" message, corresponding methods in the class frame, to the instances of which the top node belongs, will be invoked and handle the network as is specified in the original rules.

7. LUTE Experiments

The LUTE is an experimental machine translation system between Japanese and English developed by applying the investigations mentioned above. The dictionary of each language has about 3000 entries. It has been implemented on a Symbolics Lisp-machine by using ZetaLisp. The size of the system is 850KB of programs and 4MB of dictionaries and knowledge.

LUTE was not developed for practical use but to provide a part of the computer environment, RIESOLUTION (Reciprocal Environment for the Study of Language Understanding, Translator & Editor), on which theoretical works concerning computational linguistics can be examined. As a result, RIESOLUTION contains many facilities for man-machine interaction via a multi-window screen and consists mainly of a frame editor and facilities for conducting program executing. In this environment, it is possible to perform translation experiments such as analyzing texts, transferring the meaning structures, generating phrases and sentences, developing dictionaries, editing knowledge-base and examining programs both separately and simultaneously. For example, LUTE can regenerate a sentence of the source language, while showing the deleted parts in the source sentence, from a meaning structure produced by the analysis of a source sentence. Also, any intermediate representation can be modified to examine the transfer and generation as a whole or a part. Since all of the data are represented in a frame network, this environment system provides a general framework for frame-manipulation facilities. A snapshot of the translation experiment is shown in Figure 8.

References
[1] Schank, R. C., and Abelson, H. P. Scripts, Plans, Goals, and Understanding, ADD, 1977.
[2] Minsky, M. L., A Framework for Representing Knowledge, in The Psychology of Computer Vision, Winston (ed.), McGraw Hill, 1975.
[3] Quillian, M. R., Semantic Memory, in Semantic Information Processing, Minsky (ed.), MIT Press, 1968.
[4] Nomura, H., Towards the High Ability Machine Translation, Joint Japanese-European Work Shop on Machine Translation, Nov. 1983.
[5] Nomura, H., Experimental Machine Translation Systems LUTE, Second Joint Japanese-European Work Shop on Machine Translation, Dec. 1985.
[6] Shinmzu, A., Naito, S., and Nomura, H., Japanese Language Semantic Analyzer based on an Extended Case Frame Model, Proceedings of the International Joint Conference on Artificial Intelligence, Aug. 1983.
[7] Hida, H., Ogura, K., and Nomura, H., A Case Analysis Method Cooperating with ATNG and its Application to Machine Translation, Proceedings of the 10th International Conference on Computational Linguistics, July 1984.
[8] Naito, S., Shinmzu, A., and Nomura, H., Classification of Modality Function and its Application to Japanese Language Analysis, Proceedings of Association for Computational Linguistics, July 1985.

The child acquired the concept of mass.

The child can solve the problem.

The child that acquired the concept of mass can solve the problem.

The child that acquired the concept of mass can solve the problem.

Fig. 8 Snapshot of an experiment on the LUTE.