ICOT Technical Memorandom: TM-0993

TM-0993

An Integrated Framework for Grammar Formalisms in a Unification-based Approach

by

H. Sano, F. Fukumoto & H. Onodera (Fujitsu)

December, 1990

ICOT

Mita Kokusai Bldg. 21F
4-28 Mita 1-Chome
Minato-ku Tokyo 108 Japan

(03)3456-3191 ~ 5
Telex ICOT J32964

Institute for New Generation Computer Technology

An Integrated Framework for Grammar Formalisms in a Unification-based Approach

SANO, Hirosi   FUKUMOTO, Fumiyo
INSTITUTE FOR NEW GENERATION COMPUTER TECHNOLOGY (ICOT)

ONODERA, Hiroshi
FUJITSU FACOM INFORMATION PROCESSING CORPORATION

Short abstract

Localized Unification Grammar (LUG) described here is a unification-based linguistic framework. This grammar formalism contains facilities for expressing structural ambiguity caused by a unification-oriented manner. Referring information about context, these uninterpreted linguistic constraints associated with topic-focus articulation, connectives, scopes with quantifier, and so on can be re-unified later. Thus, we have one way of expressing the ambiguity caused by compositional structure. With this formalism, the Japanese grammar has already been written independently of the task of the application domain.

Keywords : **syntax**, **grammatical formalisms**

Abstract

This paper describes a unification-based grammar formalism called Localized Unification Grammar (LUG), which uses a modular treatment of syntax and semantics. The form produced by LUG can be described as a complex constituent that is the result of composition and functional application. We describe the formalism of the grammar of using functional application to limit compositional ambiguities caused by unification-oriented structural description. This grammar formalism used in LUG contains facilities for expressing structural ambiguity. Referring information about context such as knowledge base about the world and pragmatic knowledge about words, these functional structures can be re-unified later. Thus, complete resolution of constituent structures depends on semantic-based and pragmatic-based accounts of subsequent information.

With this formalism, the Japanese grammar is written independently of the task of the application domain. In addition, parsing with the unification grammar assumes inversion of the analysis process.

By its bidirectional nature, the parser based on LUG can parse and generate natural language using the same grammar.

LUG, currently implemented in LINGUIST[1], has been doing well as a development and verification tool in the research of Japanese grammar with respect to computational linguistics. An experimental LUG grammar for current Japanese is operational.

1 Introduction

In general, unification-based grammar is called an augmented-phrase structure grammar because its core is context-free grammar [8]. In this grammar, grammatical category expression is not limited to atomic symbols. The basic grammar structures of this framework are partial descriptions of constituent structures formalized with feature structures. Different kinds of information are represented using this structure. Due to its uniformity, unification can be a powerful tool not only for constructing constituent structures but also for specifying semantic interpretation of natural language. Unification is a sort of constraint on the relations between structures. This fact leads to awkwardness when the syntactic form does not give enough information on its meaning. The reason for this is that the structures produced by unification-based grammar have to be described more or less directly.

Take for example,

(1)		Hitogomi	de	Jon	to	Lucy	wo	mitsuketa.	
	(	A crowded place	in	Jon	and	Lucy	(ACCusative)	spotted.	)

There are at least two ways to read this sentence,

(1–1) Jon and (I) spotted Lucy in a crowded place.

(1–2) (I) spotted Jon and Lucy in a crowded place.

Ambiguity in the attachment of postpositional phrase is a problem that has existed for a long time in the area of natural language processing. Structural ambiguity in postpositional phrases may modify, at least, nouns (1–1) and verb phrases (1–2). In (1–2), subject-conjunction relation supports the event of spotting. As compared with this reading, (1–2) says that the collective reading of the object holds the event. This structural ambiguity is a cause of inefficiency in processing, that is context-freeness.

Functional Structure

In the formalisms we propose, functional application taken in compositional structures is used to produce functional structures that can be re-unified later and not produce all possible compositional structures.

[1] Logic-Based Integrated Natural Language Processing System (LINGUIST) is a system being developed in the Sixth Research Laboratory at ICOT in Tokyo.

This integrated framework of grammar formalisms assumes that context and knowledge-based information are available no matter when they are needed for deciding a complete interpretation. The functional structures are marked by compositional relations associated with the grammatical relations that cause structural ambiguity.

Using the functional facility, a complex constituent structure of the above type can be illustrated conceptually as follows:

$$\left[inmanner(S,J), \left(\begin{array}{c} ind(J) \\ sort(J, \text{``John''}) \end{array}\right), \left(\begin{array}{c} ind(L) \\ sort(L, \text{``Lucy''}) \end{array}\right), ind(S), sort(S, \text{``spotted''}), arg(S,[A,L])\right]$$

This functional structure allows a structural ambiguity of the form inmanner(S,J) to be resolved, permitting access to information that keeps track of context. In the example above, when 'John' introduced into the context before sentence (1) is uttered, it is not necessary to introduce a new parameter or semantic object corresponding to 'John'. Due to the information given by the context, only the subject-conjunction relation can hold the event of spotting as follows:

$$\left[and(A, \mathbf{John}), \left(\begin{array}{c} ind(L) \\ sort(L, \text{``Lucy''}) \end{array}\right), ind(S), sort(S, \text{``spotted''}), arg(S,[A,L])\right]$$

2 LUG as a integrated framework

2.1 Integration

Localized Unification Grammar (LUG) combines the configurational aspect of phrase structure base with a functionally oriented framework that assumes functions to be universal primitives of language. The addition of unification-based grammar to functional structure leads to reduction of combinatorial explosion in interpreting a sentence. Linguistic phenomena having a functionally oriented nature which do not fit into given configurations of sentence are represented in terms of a way of a functional application. The functional application consists in a mapping of phrase structures onto functional relations. The adoption of the LUG enables us to develop an integrated system of semantic and pragmatic interpretation that depends on context.

2.2 The LUG form

In a unification-based grammar, different kinds of information are represented using the same representation as feature-value pairs. These feature-value pairs are commonly specified using sets of equations. In LUG form, each equation is of the form **feature(value)** and bundles of equations use the form of list structure. The LUG form has a uniform structure called basic triple as follows.

$$\left(\begin{array}{c} CAT \\ \left[\begin{array}{l} SYN : list\ of\ attribute - value\ pairs \\ REL : list\ of\ attribute - value\ pairs \\ F : subcategorization\ list \end{array} \right] \end{array} \right)$$

Figure 1: LUG form

CAT is the rule identifier that is treated as a nonterminal symbol in grammar rules, and the structure as a whole is declared to be of CAT. CAT is specified for SYN, REL and F features. Each SYN and REL is a list of attribute-value pairs. REL contains a quasi-variable that can be referred through sharing, but SYN does not. A characteristic of LUG formalism is the use of a quasi-variable. By letting the REL part contain a quasi-variable that stands for a category expression, the LUG form takes advantage of having a functional representation that can be reconfigured or re-unified later.

The constituent structure produced by the grammar of LUG supplies a quasi-logical form in which some of the syntactic and semantic ambiguity is partially resolved and not completely compositional parsing of a sentence. In fact, the constituent structure produced by syntactic rule application in LUG contains a part of the semantic information and can be used in building logical forms as semantic interpretation of sentences. Even this structure could be utilized in processing pragmatic treatment. It should be emphasized that representations of lexicon also employ the LUG notation for entries in the dictionary and that morphological derivation rules used to synthesize new entries are written in the form of LUG.

F is a list of complements that are directly subcategorized by the category CAT and can be a help in dealing with free word order and omission of complements. F consists of two lists, each containing elements corresponding to syntactic restriction posed on complements that the verb dominates.

2.3 Interpretation

The constituent structure expression corresponding to uncertain functional application can be described in terms of a tagged form which consists of a tag-feature basic triple pair. The tag-feature, being equivalent to other features, is as follows:

tag-attribute(X,Y)

Figure 2: Tag-feature

Where, each X and Y is a quasi-variable, the former stands for a head phrase, a phrase whose dependency to be resolved is supported by the latter. Syntactic-semantic interactions generalize tag-features relating a basic triple with positioning of them over linguistic phenomena. Thus, the tag form can describe a constituent structure to which a reduction process must be applied.

Given enough information about context, for example, the reference of an entity should be resolved with the information and the way in which the interpretation rules work. Constraints imposed on a functional structure are checked in the process of interpretation of a sentence with its constituent structure. It is possible to attach the interpretation rules to the nodes associated with syntactic rules so as to reduce some of the most common types of syntactic ambiguity on parsing such attachments as spacio-temporal modifiers.

2.4 Syntactic/Semantic description of Japanese

LUG may produce structures that are independent of the utterance situation in certain respects. Although the noun phrases marked with case markers are filled in from the syntactic point of view, postpositional phrases and other attachment are held ambiguous. Take for example, the of-type feature being one of tag-features is organized around relative phrases. Since Japanese has no relatives, the dependence of a head noun on some verb complements to its right is ambiguous. It depends on the semantic nature of the head noun, so constituent structure is so constructed that it has nothing to do with syntactic structure base. Quantifiers represented by inmanner feature as a tag-features are left floating. We have four types of tag-feature that influence configurational ambiguities and are as follows:

Table 1: tag-features

Tag features	**of-type**	**inmanner**	**attach**	**co-refer**

Apart from being a very useful feature(Of-type, Inmanner and Co-refer) for standing for compositional ambiguity, they reflect phrase state being determined under the control of sentence level such as S-Structure, introduced to explain how syntactic dependencies contribute to semantic interpretations.

Table 2 below presents a few examples:

Table 2: Phrasal dependence

inmanner	postpositional phrase	(located in X)
attach	spacio-temporal adverb	(located in X′)
co-refer	topicalized phrase	(located in X″)
	conditional phrase	

For example, topicalized phrases are searched first for possible discourse entities. The phrase marked with 'Co-refer' should be preferred over the phrase marked with 'Inmanner'. The phrase marked with 'Attach' takes the middle position. These features are set to reflect a close relationship between the syntax and semantics of linguistic expressions. For the space, the basis underlying relation between linguistic description of Japanese and the formalism we offer does not appear in preliminary version.

3 Applicability

Localized Unification Grammar, whose syntactic notation is the same as DCG, is a kind of context-free structure grammar. The left-hand side of each rule consists of a non-terminal only except rule corresponding to the lexical entry. Any of the grammar rules have certain arguments specified by the LUG framework. Even rules for entries have the same arguments. The grammar rules are translated into Horn clauses using two kinds of translator, by adding extra arguments to nonterminals representing difference lists of tokens. Thus the grammar rules are written as Prolog assertions. The parser for analysis works bottom-up, the parser for generation works top-down.

3.1 Implementation

LUG is currently implemented in LINGUIST, an experimental natural language processing system being developed in the framework of logic programming. Grammar rules for Japanese fragments in LINGUIST are written in the form of LUG and parser is on a PSI-II (Personal Sequential Inference machine developed at ICOT) with about 600 grammar rules containing those for morphological analysis. About 1100 words of verbal lexicon and about 200 words of adverb are accumulated around the grammar. The property of LUG formalism is under evaluation with several copula.

As we have mentioned, the LUG form separates the context-dependent aspect from the context-dependent aspect by using a tag-feature in a uniform expression. Finding the appropriate relation as functional relation is difficult problem. Resolution technique and processing strategy requiring a lot of semantic and pragmatic knowledge are an integral part of artificial intelligence. We are now developing a strategy to remove ambiguity from the LUG form.

4 Summary

We have described the integrated framework of our proposed LUG. The LUG form shares unification-based phrase structure grammar with functional application associated with combinatorial properties of phrases. It differ from phrase structure grammar in restricting compositional natures to minimal structures. Thus, we have one way of expressing the ambiguity caused by compositional structure. Notice that we use the term in this paper in the wide sense of a grammar system that associates not only syntactic analysis but also semantic interpretation and pragmatic processing. These functional structures can be re-unified later using both contextual and knowledge-based information.

To take advantage of the generally compositional nature of the unification mechanism, functional application is assigned to the constituent structure associated with phrases. The formalisms we have presented here for the grammar can avoid structural explosion and serve as an integrated framework for semantic interpretation of sentences based on a knowledge-based approach for natural language processing. The complex structures produced by LUG provide enough information to compose semantic interpretations, and are regarded as of great value in the efficiency processing of natural language.

References

[1] Fernando C.N. Pereira, David H.D. Warren. (1980), *Definite Clause Grammars for Language Analysis – A Survey of the Formalism and a Comparison with Augmented Transition Networks*, Artificial Intelligence 13, 1980, pp 231–278.

[2] Fred Popowich. (1989), *Tree Unification Grammar*, Proc. of the Fourth Conference of the European Chapter of the ACL, 1989.

[3] Kevin Knight. (1989), *Unification: A Multidisciplinary Survey*, ACM Computing Surveys, Vol.21, No.1, March 1989

[4] Lauri Karttunen. (1989), *RUG:Regular Unification Grammar* Proc. of Fourth Conference of the European Chapter of the ACL, 1989.

[5] Lauri Karttunen, Martin Kay. *Parsing in a free word order language* Natural language parsing.

[6] Martha E. Pollack, Fernando C.N. Pereira. (1988), *An Integrated Framework for Semantic and Pragmatic*, Proc. of 26th Annual Meeting of the ACL, 1988.

[7] Marc Moens, Jo Calder, Ewan Klein, Mike Reape, Henk Zeevat. (1989), *Expressing generalizations in unification-based grammar formalisms*, Proc. of Fourth Conference of the European Chapter of the ACL, 1989.

[8] Michael Barlow. (1988), *Unification and Agreement*, Report No. CSLI-88-120, February 1988.

[9] Michael Gerlach, Michael Sprenger. (1988), *Semantic Interpretation of Pragmatic Clues: Connectives, Modal Verbs, and Indirect Speech Acts*, Proc. of the 12th International Conference on Computational Linguistics, 1988.

[10] Paul Schmidt. (1988), *A Syntactic Description of German in a Formalism Designed for Machine Translation*, Proc. of the 12th International Conference on Computational Linguistics, 1988.

[11] Robert C. Moore. (1989), *Unification-Based Semantic Interpretation*, Proc. of 27th Annual Meeting of the ACL, 1989.