ADAPTATION OF MONTAGUE GRAMMAR TO THE REQUIREMENTS OF QUESTION-ANSWERING

S. P. J. Landsbergen

Philips Research Laboratories
Eindhoven - The Netherlands

Abstract

In this paper a new version of Montague Grammar (MG) is developed, which is suitable for application in question-answering systems. The general framework for the definition of syntax and semantics described in Montague's 'Universal Grammar' is taken as starting-point. This framework provides an elegant way of defining an interpretation for a natural language (NL): by means of a syntax-directed translation into a logical language for which an interpretation is defined directly.

In the question-answering system PHLIQA 1 [1] NL questions are interpreted by translating them into a logical language, the Data Base Language, for which an interpretation is defined by the data base. The similarity of this setup with the Montague framework is obvious. At first sight a QA system like this can be viewed as an application of MG. However, a closer look reveals that for this application MG has to be adapted in two ways.

Adaptation 1. MG is a generative formalism. It generates NL sentences and their logical forms 'in parallel'. In a QA system a parser is needed: an effective procedure which assigns to an input question the syntactic structure that is required for the translation into the logical language. The MG framework has to be changed in such a way that for each grammar within that framework a parser can be defined.

Adaptation 2. The logical language used in MG contains a term for every referential word. The Data Base Language of a QA system is restricted in this respect, which is caused by the fact that the data base only contains knowledge about a restricted subject-domain. Therefore the translation from NL into the Data Base Language is partial. An extension of MG is needed which shows how a subset of NL sentences can be interpreted by means of a translation into a restricted logical language.

Adaptation 2 is only briefly discussed here, as it results in a framework which has already been described extensively in [1].

The main part of this paper is devoted to adaptation 1. A new syntactic framework is

proposed, which can be summarized as follows.

- The syntactic rules (M-rules) operate on labeled trees (or equivalently: labeled bracketings) instead of strings as in MG. Successful application of M-rules - starting with basic terms - leads to a surface tree of a sentence. (This kind of extension of MG has already been proposed by Partee and others, for different reasons than for making parsing possible)

- A context-free grammar G_{cf} defines the class L_{cf} of trees that are allowed as arguments and results of the M-rules. So the class of surface trees defined by the M-rules is a subset of L_{cf}.

- An M-rule R_i is a pair $<C_i, A_i>$, where C_i is a condition on n-tuples of trees $<t_1, \ldots, t_n>$ and A_i is an action, applicable to any tuple for which C_i holds, and delivering a tree t.

Each rule R_i must obey the following conditions:

(i) C_i and A_i are effective procedures.

(ii) From R_i an inverse rule $R_i^{-1} = <C_i^{-1}, A_i^{-1}>$ can be derived such that C_i^{-1} and A_i^{-1} are effective procedures and:

$$C_i(<t_1, \ldots, t_n>) \Longrightarrow C_i^{-1}(A_i(<t_1, \ldots t_n>))$$

$$C_i^{-1}(t) \Longrightarrow C_i(A_i^{-1}(t))$$

(iii) t is bigger (has more nodes) than any t_i in the tuple $<t_1, \ldots, t_n>$.
Special, simple, cases of M-rules are the context-free rules of G_{cf}.

For this type of grammar a parser can be designed which operates in two steps:

1) an ordinary context-free parser, based on G_{cf}, which assigns surface trees to sentences.

2) a procedure that applies inverse M-rules in a top-down fashion to these surface trees.

The parser is successful for a given sentence if a surface tree can be assigned to it by 1) and if this surface structure can be broken down into basic expressions by procedure 2). In that case the resulting derivation structure of M-rules is input for the translation into the logical language.

It is proved that such a parser is an effective procedure and that it assigns to a sentence exactly those syntactic structures that the generative rules would assign. The proof is first given for a finite set of rules and is then extended to grammars with rule-schemes defining an infinite set of rules. Rule-schemes are needed because the grammar contains an infinite set of syntactic variables. The reservation has to be made that the parser generates only one of the infinitely many derivations of a sentence that differ only in their choice of variables.

The power of the new framework is discussed. It is shown how Montague's PTQ grammar might be reformulated in it. The parser is compared with the parser written by Friedman and Warren for that grammar.

Finally, conditions are discussed that have to be added to the framework in order to make an effective translation into natural language possible.

[1]   W. J. H. J. Bronnenberg et al. - The question-answering system PHLIQA 1.
To appear in L. Bolc (ed.): Natural Language Question Answering Systems. Carl Hanser Verlag, München & Wien; Macmillan, London.