Functional Grammar and the Computer

John H. Connolly and Simon C. Dik, editors
(Loughborough University of Technology and University of Amsterdam)

Dordrecht: Foris, 1989, viii + 322 pp.
(Functional Grammar Series 10)
Paperbound, ISBN 90-6765-433-7

Reviewed by
Norman Fraser
University of Surrey

Linguistic theories can be divided into two varieties: those based on the *form* of linguistic expressions and those based on the *function* of linguistic expressions. Most present-day theories belong to the former variety. Because of their formal explicitness, the rule systems of these theories can be built into computer applications fairly straightforwardly, although they provide no indication of the contexts and ways in which they should be applied appropriately. Theories based on linguistic function address matters of appropriate usage but frequently fail to supply enough low-level formal detail to facilitate computer implementation. Halliday's *Systemic (Functional) Grammar* (Butler 1985) is an example of a functional theory that has served as the basis for some interesting computer systems, especially those concerned with text generation. This is probably less attributable to the functional credentials of Systemic Grammar than to its exceptionally useful descriptive tool, the *system network* (e.g., Mellish 1988). Connolly and Dik's volume presents the first major survey of NLP work based on another theory on functional principles, Dik's *Functional Grammar* (FG) (Dik 1978).

FG views words as predicates. The lexicon consists of a list of predicate frames. A linguistic expression, i.e., a string of predicates, is analyzed in terms of an abstract underlying predication. Predicates and predications make no reference whatsoever to the surface forms of expressions. Predications are related to surface orderings by means of expression rules. These constitute the nearest thing FG has to a syntactic component.

Functional Grammar and the Computer consists of six sections containing a total of 16 chapters. Section 1 is a good introduction by Dik to FG-based NLP and the papers in the collection. He locates the work reported in the book within the framework of a research program bearing the improbable name of *FG*C*M*NLU* (Functional Grammar Computational Model of the Natural Language User). I have no idea how to pronounce this. The book does not make clear whether *FG*C*M*NLU* is a single research project with coherent aims or simply the name by which all computational work is known within the FG community. A number of references suggest the former reading (e.g., Kahrel, p. 145; Connolly, p. 218) and this is made all the more plausible by the fact that 12 of the 15 authors are based in Amsterdam. However, many of the papers make no reference to *FG*C*M*NLU* and instead describe work carried out under rubrics such as *ASCOT, LEXALYSE, LIKE,* and *LINKS.*

Section 2 ("On Generating") consists of two papers devoted to sentence generation by Samuelsdorff and Bakker. Samuelsdorff presents program examples written in PROLOG2, whereas all of the other authors who provide program examples use standard Edinburgh PROLOG. Sentence generation is achieved by applying expression rules to predications.

Section 3 ("On Parsing") consists of four papers by Janssen, Gatward, Kwee, and Dignum. Janssen describes a parsing system that is based on a set of inverted expression rules. The algorithm described is acknowledged to be very inefficient. Gatward shows how a chart can be used to improve efficiency in FG parsing. Kwee maps expression rules onto phrase structure rules and then uses them to construct ATNs. He also makes use of a chart. Dignum describes another FG ATN. This section of the book is generally disappointing. This is not so much because the systems reported are so basic (all first implementations are equally basic), but rather because the text is peppered with a frustrating number of errors and misleading or mistaken observations. Misspellings such as "the Early algorithm" (Gatward, p. 89) are unfortunate but not seriously misleading. The same cannot be said of some of the attempts to characterize the present state of NLP. For example, regular readers of this journal will be surprised to learn that "no theoretical parsing strategies have been readily available for the design and development of practical parsing systems" (Kwee, p. 94). Unexpected sources are often cited in preference to more generally accepted authorities. For example, "the fundamental and conclusive authority in the field [of ATN parsing]...is Bates (1978)" (Kwee, p. 96). Dignum offers a 12-year-old *SIGART Newsletter* (No. 61, 1977) as a useful source on parsing systems. This may explain why he goes on to misrepresent the current state of the art so seriously: "Most of the parsers for database interfaces are designed for a particular domain. This means that the success of the parsing depends on the appearance of certain keywords in the text.... Little work, however, has been done on parsers that are context-independent" (p. 110). In spite of the overlapping concerns and approaches of the different chapters, there is very little cross-referencing. For example, Kwee's paper sits between papers that present in some detail FG chart parsing and FG ATN parsing. Kwee's paper also presents a simple ATN and active chart parsing approach to FG, but no reference is made to the neighboring papers. Instead Kwee concludes "I hope to have identified a number of blank spaces on the map of FG, certain points as yet unexplored, or insufficiently explored, that need (a first, or deeper) study in FG circles" (p. 105).

Section 4 ("On the Lexicon and Its Utilization") consists of four papers. Kahrel considers the trade-off between storing lexical items whole and manufacturing them by processes of derivation. (In FG terms this is a question of which words belong to the *lexicon* and which to the *fund*.) The next three papers are concerned with the automatic generation of a lexicon and knowledge base from a machine-readable version of the *Longman Dictionary of Contemporary English* (*LDOCE*). Voogt–van Zutphen describes a system for generating a FG lexicon; Vossen shows how a type hierarchy of FG predicates can be constructed; Meijs presents a system for generating FG's equivalent of a semantic net. These three papers together constitute the most consistent and effectively cross-referenced part of the collection.

I found Section 5 ("On Knowledge Representation and Logic") the least useful in the collection. Connolly's paper is misplaced toward the end of the book. It presents a superficial overview of AI and a brief and confusing attempt to describe the process of grammar writing in terms of knowledge engineering. The presentation of a few lines of PROLOG code seems to be designed to show that it is possible to encode FG rules in a programming language. As an introductory chapter this might work, but following the sections on generation and parsing that themselves contained partial program listings, this paper appears to be redundant. Rijkhoff's paper investigates the process of finding referents for term phrases. Weigand presents an analysis of modality in dialog in terms of possible world semantics. Dik's paper shows how it is possible to reason about the FG predicates and their interrelations. For example, he presents approaches to reasoning about hyponymy, order, converse relations, gradability, etc.

Section 6 ("On Translation") consists of a single paper by van der Korst in which he presents a FG MT system. The principles of the system are very simple. The predication underlying a linguistic expression is language-neutral; therefore the theory provides a ready-made interlingua. If it is possible to parse and to generate, then it is also possible to translate. Of course, this is an oversimplification, since different languages typically use different subsets of the set of possible predicates. Thus, paraphrasing relations between predications are necessary. Van der Korst provides a lot of useful examples to illustrate the problems and achievements of his system.

The verdict: This is an important book, since it begins to sketch what a computational version of FG might look like. It is very important for people working in functional paradigms such as FG to bring their insights about language use to the design of NLP systems, which will have real users. However, the book is ultimately disappointing for a number of reasons. It has the feel of a collection of disparate papers that are united in their debt to Dik (1978) rather than by their participation in a coherent research program. The papers are inadequately cross-referenced and display many needless inconsistencies of style (e.g., "PROLOG" vs. "Prolog"; endnotes vs. footnotes). The papers build very few bridges between computational FG and what is going on in the rest of NLP. As we have noted, some of the attempts to do so misfire. Perhaps most disappointing of all, the volume fails to raise what ought to be the most interesting question: what, if any, are the distinctive benefits of functional theories such as FG for NLP?

References

Bates, Madeleine (1978). "The theory and practice of augmented transition network grammars." In *Natural Language Communication with Computers,* edited by Leonard Bolc, 191–259. Springer-Verlag.

Butler, Christopher S. (1985). *Systemic Linguistics: Theory and Applications.* Batsford.

Dik, Simon C. (1978). *Functional Grammar.* North-Holland. (Third printing, 1981, Foris.)

Mellish, Christopher S. (1988). "Implementing systemic classification by unification." *Computational Linguistics,* **14**(1), 40–51.

Norman Fraser is a research fellow at the University of Surrey. He is currently researching dependency parsing and dialog management. His address is: Social and Computer Sciences Research Group, University of Surrey, Guildford, Surrey, GU2 5XH, U.K.; e-mail: norman@soc.surrey.ac.uk.

A Computational Model of Metaphor Interpretation

James H. Martin
(University of Colorado at Boulder)

Boston: Academic Press, 1990,
xxiii + 229 pp.
(Perspectives in Artificial Intelligence 8)
Hardbound, ISBN 0-12-474730-2

Reviewed by
Dan Fass
Simon Fraser University

A Computational Model of Metaphor Interpretation, a revised version of Martin's 1988 Ph.D. thesis, describes a computer program called MIDAS that contains an approach