with SWESIL?), it might have been better to wait until more care could be taken.

Although the title leads one to assume the authors will concern themselves with "word experts," which usually implies lexical procedures in Small's sense (they cite him several times), it turns out that nothing in DLT can be construed as such. The dictionary entries are intended to be numerous, but the authors themselves proscribe large entries (p. 145), or even ones wherein procedures appear ("There is no level of abstraction within the LKB in which meaning can be viewed as separate from words" (p. 118), and other pro-modularity arguments). DLT's semantic rules seem much more akin to Wilks's preference semantics than to Small's word expert procedures; and syntactic parsing in DLT, it turns out, is based on dependency grammar (p. 105) and is quite independent of any semantic processing.

More serious are certain scientific and technical claims that are un(der)supported. Here, of course, one tends to leave the realm of reviewing the book and move on to critiquing the work behind it. As this may not be in the purview of a book review proper, I will constrain myself to two points. First, the "lexical taxonomy" issue, for all the attention paid to it, is glossed over. In particular, while the authors promote the "good" results, identification of noun and verb hierarchies, achieved by Calzolari and Amsler, they fail to mention the problems that Amsler, at least, brings up: the noun hierarchies tend to attach themselves to the verb hierarchies eventually (which seems to violate a structural assumption built into DLT) and, worse, hierarchies seem quite inappropriate for any other part of speech (e.g., adjectives and adverbs).

Second, it is perhaps most unfortunate that the writers, who discount results reported on the basis of small prototypes (Part III, Chapter 1), are reduced to just such claims themselves—testing but seven phrases (p. 172) and one single sentence (p. 176) using a system with a "Semantic Dictionary [that] contains some 800 headwords" (p. 205). True, such a vocabulary is an order of magnitude larger than that found in most academic AI/NLP systems; but it is still an order of magnitude below the 10,000 that the authors themselves think reasonable (p. 145)—which is in turn one or two orders of magnitude below that probably required for such uses as they anticipate for DLT, if claims like those of Walker and Amsler (based on several months of *New York Times* text) are to be believed. Regardless, the trivial number of test cases put to SWESIL prove absolutely nothing, notwithstanding statements like "some interesting successes have already been achieved" (p. 205). (I grade SWESIL at 79% on the 7-phrase test (p. 172). At least they are honest: many AI/NLP reports neglect to mention errors at all, and fewer yet quantify anything.)

In short, the authors are inconsistent about the evaluation of other NLP systems against their own, and

too much of this book talks about DLT in the future tense (again, not problems unique to this book). It was written too early, and apparently in too great a hurry. For the most part, it is interesting reading for the uninformed; there is no news for the professional. But for this I could have liked it, whether or not I agree with their approach.

Jonathan Slocum is staff scientist at Symantec, a software firm that includes AI applications—notably Q&A, a file management system with a natural language interface—among its offerings. Previously, Dr. Slocum directed the successful METAL machine translation project at the University of Texas. Slocum's address is: Symantec, 1302 Darter Lane, Austin, TX 78746. E-mail: slocum@cs.utexas.edu

---

KNOWLEDGE SYSTEMS AND PROLOG: A LOGICAL APPROACH TO EXPERT SYSTEMS AND NATURAL LANGUAGE PROCESSING

Adrian Walker (ed.); Michael McCord, John F. Sowa, and Walter G. Wilson
(IBM T.J. Watson Research Center and IBM Systems Research Institute)

Reading, MA: Addison-Wesley, corrected reprinting, March 1987, xii + 475 pp.
ISBN 0-201-09044-9, $35.50 (hb)

Reviewed by
Stan C. Kwasny
Washington University

Books written by committees—where everyone writes a chapter—usually suffer from problems of readability, continuity, and stylistic variations that are difficult for a reader to overcome. Such were my expectations in reviewing this book. To the credit of the committee that produced it, none of these expectations were met. On the contrary, I found this book to be among the very best Prolog-based descriptions of expert systems, natural language processing, and knowledge systems.

The book takes a formal, logical approach to its subject. Every opportunity is taken to demonstrate in concise terms the relationship of Prolog with logic. Too many Prolog textbooks fail to point out connections with classical logic where possible. They approach Prolog as a conventional programming language. An important and obvious aspect of Prolog is thus missed: that Prolog, since it is based on logic, permits relatively easy translation from logical form to program. Moreover, given the close relationship between logical form and program, various mixtures of declarative and procedural styles can be incorporated into problem solutions. To emphasize one to the exclusion of the other does not tell the whole story.

This book *does* tell the whole story, and gives the reader a good sense of Prolog's flexibility in addressing difficult issues across a spectrum of problems. Care-

fully, and with a pedagogically attractive style, it discusses topics related to expert systems (e.g., representation, rules, shells, explanation, and confidence factors) as well as topics related to natural language processing (e.g., conjunctions, inflection, left-recursion, question-answering, and scoping). All of this combines to provide a very concrete, yet readable, introduction to these fields. It achieves this through examples, woven with explanations that are clear. All coded examples have just enough odd details to reassure us that they have survived a process of close scrutiny and debugging.

Chapter 2 is of special note. "A Prolog to Prolog" by John Sowa is perhaps the best informal introduction to Prolog I have ever read. I found it both thorough and complete, but written in a style that makes it pleasant and delightful to learn. Pure Prolog techniques are applied to several classic and novel artificial intelligence problems. Simple, direct comparisons are made to standard logic (with quantifiers), to Pascal record structures, and to relational data bases. Unification is explained with a few simple examples and criteria for reversible predicates are given.

Chapter 3 discusses more advanced programming techniques in Prolog. Here, the book departs slightly from its concentration on classic artificial intelligence problems to discuss the more mundane issues of Prolog itself: data structures, unification, control structures, labels, and metavariables. Later in the chapter, techniques for representing and searching graphs and trees are presented along with a worked-out example of alpha-beta pruning. Throughout the book, principles of good Prolog programming style are emphasized. At the conclusion of this chapter, these are summarized into a neat list of 11 guidelines. Warnings about side effects and the judicious use of cuts are similarly presented.

Syllog (from "syllogism") is described in Chapter 4; it represents an expert system shell embedded in Prolog. It features some interesting ideas for implementing the interpreter based on *backchain iteration* which, when run, leads eventually to a fixed point representing the answer. Several sample expert systems built from Syllog are also presented in the chapter. PLANTDOC, for example, is a MYCIN-like system that diagnoses disease in garden plants. It serves to illustrate reasoning using confidence factors, frames, and explanation techniques.

In Chapter 5, the book presents a score of methods for constructing syntactic structures from inputs, a particular semantic model, and a set of techniques for mapping syntactic structures to semantic forms. It demonstrates the ease with which input sentences can be coerced into logical expressions of meaning (expressed in "Logical Form Language"). Much of this chapter has appeared in recent research articles by McCord and, to his credit, he is able to clearly present solutions to some rather complicated issues.

In the chapter, definite clause grammars (DCGs) are introduced first. Since these are so clean and elegant in their simplicity, they nicely introduce some of the mechanisms for natural language processing in Prolog. Next, modular logic grammars (MLGs) are introduced and shown to allow for a greater degree of modularity between syntax and semantics. The bulk of the chapter is dedicated to deriving solutions to a variety of problems using MLG. The range of issues and the degree of detailed discussion are impressive. In the conclusion, a natural language question-answering system for a university data base is presented.

I have very few criticisms of the book. It is based entirely on IBM Prolog with little or no effort to provide information about other implementations of Prolog. An appendix is provided which serves as a type of reference manual for IBM Prolog. Perhaps the book could be more generic, but then I feel it would undermine its reliance on concrete examples that the reader can confidently run. Most readers will not find this to be a difficult problem and the slight effort required to overcome the problem is well worth it. I found very few errors and all were very minor (e.g., missing existential quantifier on page 49). They never detract from what the author is saying.

I was impressed with this book. Much of the credit for this certainly belongs with the editor, Adrian Walker, who does a marvelous job of weaving together a coherent book from the contributions of his co-authors. The credentials of the supporting cast are equally impeccable. I found many enjoyable segments in the book. Although an appendix is provided which discusses the logical basis of Prolog and Syllog, I think it would be difficult to use this book for the logical fundamentals of Prolog, as the authors suggest. However, it is certainly an important reference and could easily be used as a textbook in support of a course on artificial intelligence, Prolog programming, expert systems, or natural language processing.

Stan Kwasny holds a visiting appointment with the Department of Computer Science and is affiliated with the Center for Intelligent Computer Systems at Washington University in St. Louis, MO. He has over 15 years experience conducting research in computational linguistics. His address is: Washington University, Campus Box 1045, St. Louis, MO 63130. E-mail: sck@wucs1.wustl.edu

---

BRIEFLY NOTED

CATEGORIAL GRAMMAR