REFERENCE

Feigenbaum, Edward A. and Pamela McCorduck 1983 *The Fifth Generation: Artificial Intelligence and Japan's Computer Challenge to the World*. Addison-Wesley, Reading, MA.

Harold Somers is a lecturer at UMIST, and a member of the machine translation research group there, with a special interest in Japanese MT. He recently spent 15 months in Japan as a visiting research fellow, first at NTT's Basic Research Laboratory, and then at Toshiba Corporation's R&D Center, during which time he was able to closely investigate Japanese NLP activities. Somers's address is: Centre for Computational Linguistics, UMIST, P.O. Box 88, Manchester M60 1QD, England. E-mail: hls@ccl.umist.ac.uk.

NATURAL LANGUAGE UNDERSTANDING

James Allen
(University of Rochester)

Menlo Park, CA: Benjamin/Cummings, 1987, xvi+574 pp.
ISBN 0-8053-0330-8; (hb)

Reviewed by
Michael Kac
University of Minnesota

Since the appearance of Tennant (1981), there have been a number of books intended to serve as introductions to computational linguistics, including Harris (1985), Grishman (1986), and Winograd (1983). The book under review is directed at much the same audience and is, in my opinion, the best so far. This is not to say that it is flawless; but its weaknesses are due more to the nature of the subject than to any failings of the author. (I should note, however, that I have not used it in a course, and so cannot report on student reactions to it.) The book is in four parts, entitled "Syntactic Processing", "Semantic Interpretation", "Context and World Knowledge", and "Response Generation", supplemented by two appendices, one on logic and one on symbolic computation; each chapter ends with a set of problems. It is broad enough in its coverage of a variety of approaches to give the reader a sense of the field of computational linguistics in its present state while managing to avoid a mere "Cook's Tour" approach (Pullum 1984).

The author is conscientious about regularly going into certain problems at a level of detail that makes for more than just a superficial overview; at the same time, the range of topics covered is sufficiently broad that anyone wanting to base a course on a cover-to-cover reading of the book had better have a full year in which to do it. Allen does not discuss how to actually formulate algorithms in some selected language, though many Lisp-like data structures appear throughout and the overall perspective is quite clearly that of contemporary AI. (The second appendix covers the essential ideas underlying Lisp and Prolog, though without taking the

form of a primer for either language.) A nice theoretical balance is maintained, and if there is any axe-grinding it is exceedingly subtle.[1]

The treatment of syntax takes context-free grammar as the starting point and then considers both top-down and bottom-up approaches to parsing. Two different formalisms are pursued in some detail, namely transition networks and logic grammars, both with and without augmentations that take them outside the class of CFG-equivalent devices. One whole chapter is devoted to handling movement, and includes discussion of how (at the cost of increasing the size of the grammar) use of a hold facility can be avoided by using Gazdar-style slash categories. The problem of non-determinism is also discussed, and there is an overview of Marcus's parser.[2]

The section on semantics deals partly with representational issues (including what logical forms should look like), partly with the specifics of designing algorithms for semantic interpretation, and partly with a variety of open questions (including the especially thorny problem of correctly interpreting nominal compounds). A notion of "merging", closely akin to unification, is proposed as a basic operation for semantic interpretation and a detailed example showing the interaction of a syntactic parser and semantic interpreter is presented.

The section on context and world knowledge deals first with knowledge representation (a topic that is, of course, of more than just linguistic relevance), with problems in establishing reference, and with inferences based on knowledge. Anaphora is dealt with in two places, once in the chapter on reference (where the focus is on intrasentential relations), and once in the chapter on discourse structure (where the focus is on "long-distance" anaphoric relations, such as the use of a pronoun to refer to something mentioned some time earlier in a dialog). Finally, the last part contains a chapter on question-answering systems and one on "generation",[3] the latter of note in part because of the attention it gives to systemic grammar.

I very much like the overall plan of the book (as outlined above) and think that the level of detail into which it goes is about right for a text of this kind. From the organization, I infer that the presumed audience is computer scientists with little background regarding natural language rather than linguists with an interest in applying what they know in a computational domain, which seems reasonable enough: in my own introductory CL course, the former group vastly outnumbers the latter (but then computer scientists as a group must outnumber linguists by an astounding factor anyway). I wonder, however, if the time is not fast approaching when we are going to have to make some linguistics beyond the introductory level a prerequisite for courses in CL, which would eliminate the need for a lot of background that Allen (and others, such as Harris) are forced to try to cram in as best they can. My own prejudices and disciplinary allegiances may be showing

here, but I am dubious of the effectiveness of trying to incorporate a crash course in linguistic fundamentals into a course in CL. To be fair, I also think that logic at a level that Allen presupposes (and that linguists do not automatically acquire as a part of their training) ought to be required, too, plus some formal language and automata theory.

At a level of fine detail, there are a number of glitches of various kinds, ranging from solecisms like confusion of term and referent (e.g.: "For example, the NP *the fish* is ambiguous between being a single fish and being many fish" (p.11)) to badly chosen examples. As an instance of the latter I offer the sentence *It's impossible for you to have seen the moon in our car, because it wouldn't fit,* given on p.256 as a case of an inherently anomalous NP (*the moon in our car*) losing its anomaly in context. I would have thought that the sentence in question shows how world knowledge would enable one to know that the sequence *the moon in our car* was *not* an NP in this context, or at least not one in which the prepositional phrase functions as a restrictive modifier of the head (in which case the anomaly is not there to begin with). I would add, less as a criticism than as a suggestion for amplification in a later edition (which may well come about), that the discussion on non-intersective adjectives on p. 291f take into account the extensive treatment of the semantics of such items to be found in Keenan and Faltz (1985). Finally, I should mention that numerous pages of the copy I received had large black smudges in the margins.

The foregoing reservations aside, I think that this is a good effort at a systematic and comprehensive introduction to the field of CL.

REFERENCES

Grishman, R. 1986 *Computational Linguistics.* Cambridge University Press.

Harris, M.D. 1985 *Introduction to Natural Language Processing.* Reston Publishing Company, Reston, VA.

Heidorn, G.; Jensen, K.; Miller, L.A.; and Chodorow, M.S. 1982 The EPISTLE Text-Critiquing System. *IBM Systems Journal* 21:305–326.

Keenan, E. and Faltz, L.M. 1985 *Boolean Semantics for Natural Language.* Reidel, Dordrecht, Netherlands.

Pullum, G.K. 1984 If It's Tuesday, This Must be Glossematics. *Natural Language and Linguistic Theory* 2:151–156.

Tennant, Harry 1981 *Natural Language Processing.* Petrocelli Books, New York, NY and Princeton, NJ.

Winograd, Terry 1983 *Language as a Cognitive Process, Vol.1: Syntax.* Addison-Wesley, Reading, MA.

NOTES

1. I do, however, disagree with the following remark: "It is important to realize that you do not store features and make agreement checks just so that you can detect illegal sentences. Making judgements on grammar is not a goal of computational linguistics" p. 10. Systems like CRITIQUE, né EPISTLE (Heidorn et al. 1982) are intended to do precisely that (among other things).

2. I have a stylistic quibble with Allen's reference on p. 178 to Mitch [sic] Marcus; full names should be used in formal exposition.

3. I use quotes here since, strictly speaking, *production* would be a better term. The word *generate* in its technical sense really means nothing more than "define", and its use as a synonym for *produce* merely perpetuates ancient conceptual confusions.

Michael Kac is professor of linguistics at the University of Minnesota, where he teaches an introductory course in computational linguistics. His current research interests include parsing and applications of formal language theory in linguistics. His address is: Department of Linguistics, 142 Klaeber Court, University of Minnesota, Minneapolis, MN 55455. E-mail: kac@umn-cs.arpa.

A NATURAL LANGUAGE INTERFACE FOR COMPUTER-AIDED DESIGN

Tariq Samad
(Carnegie-Mellon University)

Reviewed by
Bruce Ballard
AT&T Bell Laboratories

This short book describes an implemented natural language system called **Cleopatra**, which, in its 7,000 lines of Franz Lisp, seeks to provide an interface for use in a CAD system. Since the existing application involves circuit design, the work is reminiscent of the SOPHIE system (Brown and Burton 1975). An example sentence that illustrates the level of sophistication toward which Cleopatra strives is, "What is the voltage at n13 after the voltage at n1 reaches 2 v and before the voltage at n23 reaches 3 v?" Most of the book discusses the system's treatment of syntax, in accordance with the finding that "the pragmatics and the semantics required for as limited a sub-domain as ours are very small subsets of the pragmatics and semantics required generally".

Cleopatra is predicated on the belief that "the design of a natural language interface for CAD cannot be based on prior assumptions about which English constructions will be useful" and the further suggestion that "the complexity and diversity of the CAD domain . . . render previous approaches to natural language interfaces of little practical utility for our purposes." From the evidence supplied in the book, however, I remain to be convinced that the subset of the CAD domain addressed to date is in fact more complex or more diverse than what has been studied and reported on in the literature. Neither do I believe the techniques presented, though interesting in their own right, render the chosen domain more approachable than a thoughtful attempt at using existing techniques would have.