

BOOK REVIEWS

AN INTRODUCTION TO FORMAL LANGUAGE THEORY

Robert N. Moll, Michael A. Arbib, and A. J. Kfoury
with contributions by James Pustejovsky
(University of Massachusetts, Amherst; University of
Southern California; Boston University; Brandeis
University)

New York, NY: Springer-Verlag, 1988, x+203 pp.
ISBN 0-387-96698-6 and 3-540-96698-6, \$39.90
Hardbound

Reviewed by
Geoffrey K. Pullum
University of California, Santa Cruz

This book announces itself in its preface (p. vii) as “the first textbook to combine the topics of formal language theory traditionally taught in the context of programming languages with an introduction to issues in computational linguistics”. This is an interesting idea; interesting enough to have made me turn to the book with some anticipation; but I was disappointed. The natural language material (virtually all syntax) is segregated in the last two chapters (52 pp.), drafted by James Pustejovsky (henceforth *P*). Moll, Arbib, and Kfoury (henceforth *MAK*) offer in the first seven chapters (143 pp.) a standard (and quite brief) introduction to formal language theory with no natural language perspective at all (except for a brief and slightly confusing illustration of the phrase structures for two English sentences on pages 20–21).

The main problem with the book is that *P*’s two chapters reflect little of the formal rigor for which the first seven chapters have laid the groundwork. It is hard to believe *MAK* worked over *P*’s chapters with a critical eye. By the standards of the material in the first seven chapters, the last two are vague at best, and frequently confusing or even confused.

The two sections of the book have entirely different philosophies even in bibliographical policy. Neither section gets it quite right. *MAK* adopt school textbook style, avoiding literature citations. They mention about a dozen standard names (in order of appearance: Chomsky, Backus, Naur, DeMorgan, Kleene, Cocke, Kasami, Younger, Earley, Turing, Church, Cantor, Gödel, and Post), but they cite no specific literature except for five introductory books (Aho and Ullmann’s *The theory of parsing, translation, and compiling*, Minsky’s *Computation: Finite and infinite machines*, and three books of their own). I think this is inappropriate in a university-level textbook; students should be given directions to at least some of the more important

primary literature in computer science journals. *P*’s two chapters, on the other hand, have their own list of references, more like a linguistics article; but this list includes several works with scant formal content that will mean little to the student of computer science.

MAK’s treatment of the standard topics in formal language theory is clear and effective. Their presentations of proofs and introductions of new concepts frequently have an appealing freshness and directness. They cover a good selection of the most central topics in formal language theory: context-free (CF) and context-sensitive grammars and languages, the Chomsky hierarchy, closure properties of languages, regular expressions and finite-state languages, pushdown automata and their equivalence to CF grammars, normal forms for CF grammars, the CKY and Earley algorithms, Turing machines, linear bounded automata, halting and undecidability, and parsing, including top-down parsing (with LL grammars) and bottom-up parsing (with LR grammars). They also have a chapter on the algebraic approach more favored by European theoretical computer scientists: fixed-point principles, representing CF grammars by equations, and so on.

P’s two chapters attempt to cover the aims of linguistic theory (generative grammar), generative capacity as applied to natural language grammars (a section that *P* calls “The generative power of natural languages”, a mistaken turn of phrase, since languages do not generate anything), ATNs, lexicalism and X-bar theory, generalized phrase structure grammar (a section riddled with misstatements), and government and binding theory (GB). The lack of rigor is unmistakable throughout.

On page 9, in *MAK*’s section, the distinction between a node and its label is clear (“Every node that is not a leaf is labeled with a variable”), but on pages 177–178, *P* confuses nodes with their labels (“the node *N*”; “the number of ‘bars’ associated with a node”). On page 178, *P* refers to “having *n* bars” as an equivalence relation (he may mean “has the same number of bars as”). After presenting on page 80 the X-bar rule

$$\bar{N} \rightarrow \text{Spec } \bar{N}$$

on page 180, *P* tells the reader, gratuitously and bafflingly:

Although the notation may be new, the structures here are familiar. The rule for \bar{N} , for example, is a slightly different version of the following NP rule.

$$\text{NP} \rightarrow \text{Det A N PP}$$

But in fact the structures induced by these rules are strikingly different, as the reader who had understood even the first ten pages of this book could not fail to see. What is more, vital issues in syntax and semantics have

been held to hinge on whether the X-bar structure (corresponding to $NP \rightarrow Det\ Nom$ rather than $NP \rightarrow Det\ A\ N\ PP$) is correct in the constituent structure \bar{c} laims it makes. P also states that the categories \bar{N} , \bar{V} , \bar{A} , and \bar{P} "correspond to words that are two-place predicates" (p. 178), which cannot be right, since one-place verbs like *elapse* and three-place verbs like *give* also appear in \bar{V} constituents.

The language of P's chapters is full of impressionistic turns of phrase; "we can think of X as Y ", " X acts as Y ", and " X functions as Y " occur frequently. Differentiating GB from transformational grammar, P asserts: "In GB, the logical form (or semantics) of a sentence is considered to be just as important as the surface structure of a sentence (the words that we actually hear)" (p. 188). This appears to suggest that a crucial difference between GB and transformational grammar lies in a value judgment, namely how "important" logical form is held to be. It also manages to confuse logical form (which is syntactic) with semantics, s-structure with PF, and sentences with utterances.

Yet these distinctions are highly germane to the material discussed. For example, if surface structure contains "the words that we actually hear", it is not clear that there is any coherent interpretation of P's long excursus (pp. 155–159) on that old chestnut of theoretical syntax, the putative blocking of the rule

want to \rightarrow *wanna*

by *wh*-movement traces. P relies on a distinction between two kinds of empty string: e "is an empty category, in the sense that we can't hear it spoken" (p. 157), but it is not the empty string λ introduced by MAK on page 3, because, crucially, $\lambda \neq e$ (p. 158). The perceptive student will ask what the formal distinction is between $[_{NP}\lambda]$ and $[_{NP}e]$, and will find no answer. (Linguistically, there is published evidence that the explanation P seeks to present is not adequate anyway; see Postal and Pullum (1982).)

On page 195, P talks of "a notion of 'closeness' based not on distance in a contiguous string, but on the nearness in a tree". He then offers this definition of such a "closeness" relation:

C-command: Node α c-commands node β if the first branching node dominating α also dominates β .

This is extraordinarily vague for a definition in a textbook on formal language theory. The *if* should be *iff*; *first* is not defined; *branching node* is not defined; *dominates* is not defined (or used anywhere else in the book, as far as I could determine: typically, P has not checked whether MAK use the notion of a dominance relation when talking about trees); and it is not made clear whether α and β have to be distinct nodes, or whether one can dominate the other. In addition, the reference to "closeness" remains obscure: there is no limit to how far a c-commanding node may be from a c-commanded node, either in paths through the tree or symbols in the string (α may be indefinitely far down a

chain of unit productions, and β may be embedded anywhere in an arbitrarily complex structure). Whatever else c-command may be useful for, it does not guarantee "closeness" between α and β .

The foregoing are just a few of a very large number of examples that could be cited of inadequacies in the natural language portion of this book. The casual exposition of basic natural language syntax presented in the last two chapters ill befits a text on how to study languages and grammars with mathematical precision.

It is vital for the future of computational linguistics that linguists' study of languages and grammars should not remain at this level of informality and confusion. We need a textbook that deals with the grammars and properties of natural languages and programming languages with equal seriousness. This one does not satisfy that description.

REFERENCE

Postal, P.M. and Pullum, G.K. 1982 The Contraction Debate. *Linguistic Inquiry*, 13:1, 122–138.

Geoffrey K. Pullum is a professor of linguistics at the University of California, Santa Cruz, where he is also Dean of Graduate Studies and Research. He holds a doctorate from the University of London in general linguistics. His recent research has been in the theory of natural language syntax and in mathematical aspects of theoretical and computational linguistics. Pullum's address is: Cowell College, UCSC, Santa Cruz, CA 95064. E-mail: gkp@ling.ucsc.edu.

ATTRIBUTE-VALUE LOGIC AND THE THEORY OF GRAMMAR

Mark Johnson
(Brown University, Providence RI)

Stanford, CA: Center for the Study of Language and Information, 1988, xi+162 pp. (CSLI lecture notes 16)

(Distributed by the University of Chicago Press)
ISBN 0-937073-37-7, \$37.50 (hb)
ISBN 0-937073-36-9, \$15.95 (sb)

Reviewed by
Robert Kuhns
Arthur D. Little, Inc.

In *Attribute-Value Logic and the Theory of Grammar*, Mark Johnson has written an important work in the area of formal syntax. Johnson's underlying thesis that attribute-value-based grammars can be viewed as systems of logic permits a detailed analysis of their formal, computational, and linguistic properties. Although this book does not propose a specific linguistic theory, it does have direct implications for research into attribute-value grammars, which represent a major trend in contemporary grammatical theory.