Statistically-Driven Computer Grammars of English: The IBM/Lancaster Approach

Ezra Black, Roger Garside, and Geoffrey Leech (editors)
(IBM T.J. Watson Research Center and Lancaster University)

Amsterdam: Editions Rodopi
(Language and Computers: Studies in
Practical Linguistics, edited by Jan
Aarts and Willem Meijs, Volume 8),
1993, ix + 248 pp.
Paperbound, ISBN 90-5183-478-0,
$58.50, Dfl 100.00

Reviewed by
Dekai Wu
Hong Kong University of Science and Technology

Statistical computational linguistics is entering a consolidation phase, signaled by the appearance of book-length tracts devoted to single, coherent projects. After several years of vigorous research in the area, gains along a number of dimensions have solidified into established methodology. Although *Statistically-Driven Computer Grammars of English* (henceforth *SDCGE*) is an edited collection, its presentation of a broad range of issues, spanning methodologies from linguistic evaluation to statistical training, is tightly and lucidly integrated.

SDCGE is the product of a five-year collaboration between IBM T.J. Watson Research Center's Continuous Speech Recognition Group and Lancaster University's Unit for Computer Research on the English Language, to produce a wide-coverage English parser with the aid of statistical grammar-training techniques. The project initially targeted the AP newswire, but eventually lowered its aim to IBM computer manuals. As motivation, Ezra Black opens the book with an introduction summarizing three experiments demonstrating the poor accuracy of "traditional" parsing systems, most notably a 1992 evaluation of 35 parsers against the Penn Treebank, which yielded only 22% average correctness (by a generous criterion) by the only seven participants willing to report their results.

The project incorporates two complementary statistical methodologies. Grammar development is driven by ranking error types by frequency over the corpus, and fixing the most frequent ones first. Evaluation is performed by taking the maximum-likelihood parse for each sentence as the grammar's prediction, rather than the "traditional" approach of allowing many possible parses, as with chart parsers. This condition is crucial, since otherwise the accuracy (recall) of any parser can be arbitrarily inflated by simply outputting more parses per sentence.

One of the most important services of *SDCGE* to its readers is its assemblage of invaluable anecdotes for those seeking to avoid the costly pitfalls that are easily encountered in this field, especially where human evaluation and processing of corpora are involved. The authors' experience with various levels of treebanking detail, along with the tradeoffs involved, is described in chapters by Roger Garside and Anthony McEnery, and Elizabeth Eyes and Geoffrey Leech. Treebanking proceeds by verifying the output from the CLAWS automatic tagging system, followed by human "skeleton parsing," where sentences are bracketed into flat parse trees labeled from a minimal set of nonterminal types. User interfaces for post-editing tags and creating skeleton parses, optimized for throughput, are both described. The authors set out four criteria to be traded off in designing tagsets and nonterminals: the productivity of the human evaluators, attainable accuracy level, attainable uniformity of analysis across evaluators, and linguistic validity.

The authors' own design decisions on the CLAWS2a tagset, the 17 nonterminals, and the approximately 500 English grammar rules are described in concrete detail, largely in the book's longest chapter, in which Black covers the IBM grammar. Along the way, numerous examples extracted from various corpora are used to illustrate how inadequacies in the coverage of current grammatical theory (e.g., ellipsis, coordination, argument structure) are dealt with in practical ways. The grammar is sufficiently broad as to provide a good departure point for those wishing to construct their own systems.

The one unfortunate omission is that the chapter describes the entire grammar in standard rewrite-rule notation, but goes on in the subsequent chapter (again by Black) to explain that their system employed a different, more compact feature-based representation instead. The feature representation reduces the number of grammar rule variations that must be remembered or explicitly stored, and provides the linguistic motivation for the equivalence classes ("mnemonics") that are later used to reduce the number of parameters to be trained. The features themselves are listed in an appendix, but the grammar is not given in this form, leaving the reader to reconstruct the feature-based grammar from the rewrite rules.

The book then moves on to a nice treatment of the training algorithm and issues, by John Lafferty. Between fields as (formerly) disparate as pattern recognition and linguistics, it is always a difficult balancing act to treat topics in detail comprehensible to a novice of one area while not burdening specialists with too much to sift through. The inside-outside algorithm is introduced here at a level suitable for novices; its relationship to the general EM (expectation-maximization) algorithm is omitted. The essential ideas needed to extend the algorithm to handle feature-based mnemonics are outlined, though again the inexperienced reader will encounter difficulty reconstructing the details.

The concluding chapter by Black, Garside, David Magerman, and Salim Roukos draws up the results. For sentences of 1–23 words (the authors' primary objective), the maximum-likelihood parse for 64–69% of the sentences is "correct," meaning there are no constituent crossings compared against the treebank parse, measured using randomly extracted test sets from the treebank. A "correct" parse, but not necessarily the maximum-likelihood parse, is produced for about 94% of the sentences. The chapter also describes some current directions, notably the idea of a history-based grammar, in which parse tree derivation probabilities are conditioned on variables other than a constituent's immediate parent (most importantly the parent's head). Because the resulting parameter space is too large in its full form, a decision tree is induced to capture the most significant conditioning factors. The preliminary experiment significantly improved the prediction accuracy on labeling of nonterminals, which is not included in the above definition of "correct."

SDCGE displays a refreshing honesty about its empirical results. The authors are forthcoming about the relative modesty of their "correctness" criteria. The many design errors—most remedied, some not—and their consequences are described. An enormous amount of labor has gone into the project, allowing the results to stand by themselves.

As implied by its title, the book firmly restricts its claims and attention to English. However, computational linguists interested in other languages would do well to ignore this focus, since much of the material is applicable to other languages. Only the specific rules must be overlooked (particularly Chapter 4), leaving an intact thread on tagging, treebanking, and probabilistic context-free grammars.

SDCGE succeeds as an introductory guide to the methodology of building probabilistic context-free grammars, both for linguists moving toward corpus-based techniques and for computer scientists seeking advice on the tagging and treebanking end. It is not a textbook, but may be the closest thing so far. Its polemics in favor of statistical methodology may be too much for some linguists, but the wealth of concrete detail more than makes up for this.

Dekai Wu received his Ph.D. from the University of California at Berkeley in 1992, working in probabilistic computational linguistics and natural language processing. He is assistant professor at the Hong Kong University of Science and Technology, where he directs the SILC project on statistical Chinese–English translation. Wu's address is: Department of Computer Science, University of Science and Technology, Clear Water Bay, Hong Kong; e-mail: dekai@cs.ust.hk