Computational Morphology: Practical Mechanisms for the English Lexicon

Graeme D. Ritchie, Graham J. Russell, Alan W. Black, and Stephen G. Pulman
(University of Edinburgh, University of Geneva, University of Edinburgh, and University of Cambridge and SRI International, Cambridge)

Cambridge, MA: The MIT Press
(ACL–MIT Press Series in Natural Language Processing, edited by Aravind K. Joshi, Karen Sparck Jones, and Mark Y. Liberman), 1992,
x + 291 pp.
Hardbound, ISBN 0-262-18146-0, $32.50

Reviewed by
Evan L. Antworth
Summer Institute of Linguistics

In spite of the title of this book, the authors stress that it "is *not* a general review of computational work on morphology." Rather, it describes a specific project to build a morphological parser and large lexicon of English. This project, done at the Universities of Edinburgh and Cambridge between 1983 and 1987, became part of a Common Lisp software package now called the Alvey Natural Language Tools. The subtitle of the book, *Practical Mechanisms for the English Lexicon*, is more indicative of the authors' aims, namely to develop linguistically sound tools for building computational lexicons. While their proof of concept is done by developing a lexicon of English, the theoretical basis and the software tools are intended to be applicable to a wide variety of languages.

Intentionally drawing on the work of others, their morphological analyzer is based on Kimmo Koskenniemi's (1983) two-level model of morphology. In the two-level model, the rule component handles spelling alternations that occur at the boundaries of morphemes, and the lexicon lists all lexical forms and specifies the morphotactic structure of words. Both the rules and the lexicon are computationally implemented using *finite-state machines*. The authors' implementation of the rule component of the two-level model follows Koskenniemi's work quite closely. Their major innovation is to implement a rule compiler that translates spelling rules written in a high-level linguistic notation into finite-state transducers, which is the representation actually required by the rule interpreter. The compiler places some significant limitations on the expressive power of the rule notation, however; neither the Kleene star construct nor optional elements are permitted in rule contexts. The compiler also runs rather slowly, making it less useful for interactively developing a set of rules. An appendix to the book contains a list of the sixteen spelling rules used in the English description.

The authors' implementation of the two-level lexicon is a significant innovation from Koskenniemi's original design (as well as other implementations based on it such as KIMMO (Karttunen 1983) and PC-KIMMO (Antworth 1990)). In Koskenniemi's model, morphotactics are handled by *continuation classes* that specify for each morpheme in the lexicon the classes of morphemes that can follow it. This system is simple and computationally efficient, but breaks down when one tries to handle

co-occurrence constraints between morphemes that are not contiguous. The design is also less than ideal because morphotactic information is intertwined in the structure of the lexicon, rather than stated in rules separate from the lexicon.

The present authors have removed all morphotactic information from their lexicon and have instead encoded it in a word grammar that is implemented as a *feature grammar* or *unification grammar*. Linguistic entities are represented as complex *categories*, composed of feature specifications. By exploiting underspecification of categories, rules can elegantly capture linguistic generalizations. The notation also permits use of *feature value variables* that are filled in by the process of unification. This is the mechanism used to copy values from one category in a parse tree to another.

While feature-based word structure rules are sufficiently powerful to describe English word structure, the authors have also implemented a system of *feature-passing conventions* in the word grammar. Feature-passing conventions permit certain regularities of English morphology to be expressed more perspicuously than can be done just with word structure rules. One limitation of the feature-passing conventions is that they operate only on binary branching rules (the authors claim that it is rarely necessary to divide an English word into more than two immediate constituents). Thus the conventions are stated in terms of three entities: *mother*, *left daughter*, and *right daughter*.

The first feature-passing convention, the Word Head Convention, captures the familiar observation that the category of an English word is determined by the category of its right daughter (for instance, the final suffix). The convention permits the grammar writer to define a set of WHead features and requires that "the values of the WHead features in the mother must be the same as the values of the corresponding WHead features in the right daughter." For example, if PLURAL is declared as a WHead feature and the final suffix of a word (the right daughter) has the feature +PLURAL, the whole word acquires the feature +PLURAL. The second feature-passing convention, the Word Daughter Convention, handles cases not covered by the more general Word Head Convention, namely cases in which the category of the whole word is determined by the left daughter. These two conventions are the main mechanism by which a whole word acquires the features of its constituent morphemes. This is a significant advance over Koskenniemi's original design, which identified individual morphemes in a word but provided no direct way to infer the syntactic category of the whole word.

The third feature-passing convention, the Word Sister Convention, uses a special feature STEM to subcategorize affixes for the kind of stems to which they can attach. For instance, the suffix *+ness* is specified to attach only to an uninflected adjective; thus *happiness* is allowed, but **happierness* is disallowed because the adjective is inflected, and **arriveness* is disallowed because the stem is not an adjective. Thus the Word Sister Convention is used to account for much of the morphotactic structure that in Koskenniemi's original model would be handled by continuation classes.

Besides mechanisms for morphological parsing, the authors have also implemented several kinds of *lexical rules* that are intended to capture regularities and generalizations among lexical entries. They permit the lexicon compiler to write simpler entries, which are pre-processed into an expanded form that is actually used by the morphological rules. Lexical rules have both a theoretical and practical role: they permit statement of linguistic generalizations and they give the user tools for increasing the efficiency of writing and maintaining a lexicon. One type of lexical rule, called a Completion Rule, adds predictable information to individual entries. For instance, entries for nouns can be written without specifying the feature number; a Completion Rule will add a number feature with a value of singular, capturing the fact that nouns are singular by default.

The latter chapters of the book cover the description of English and various issues and details related to the implementation of the software. The appendixes include notational formalisms, feature definitions, spelling rules, lexical entries, and sample output from the system.

In sum, this book meets its goal of both providing a sound theoretical foundation and producing a set of practical software tools. The marriage of the two-level model and feature grammar makes possible an elegant and powerful description of English.

References

Antworth, Evan L. (1990). *PC-KIMMO: A Two-Level Processor for Morphological Analysis.* Occasional Publications in Academic Computing No. 16. Dallas, TX: Summer Institute of Linguistics.

Koskenniemi, Kimmo (1983). *Two-Level Morphology: A General Computational Model for Word-Form Recognition and Production.* Publication 11, Department of General Linguistics, University of Helsinki, Finland.

Karttunen, Lauri (1983). "KIMMO: a general morphological processor." *Texas Linguistics Forum*, **22**, 165–186.

Evan L. Antworth is a member of the Summer Institute of Linguistics, and spent seven years in the Philippines with SIL. He is associate editor of SIL's series Occasional Publications in Academic Computing. He is the author of a book on PC-KIMMO, an implementation of two-level morphology for personal computers. Antworth's address is: Summer Institute of Linguistics, 7500 West Camp Wisdom Road, Dallas, TX 75236. e-mail: evan@sil.org