A Computational Theory of Writing Systems

Richard Sproat
(AT&T Laboratories)

Cambridge University Press (Studies in natural language processing, edited by Branimir Boguraev), 2000, xviii+236 pp; hardbound, ISBN 0-521-66340-7, $59.95

Reviewed by
Kenneth R. Beesley
Xerox Research Centre Europe

Downloaded from http://direct.mit.edu/coli/article-pdf/27/3/464/1797839/coli.2000.27.3.464.pdf by guest on 11 August 2022

1. Introduction

Interest in the history, theory, and classification of writing systems has never been higher, and the last decade saw the publication of several worthy books on the subject, including the formidable one by Daniels and Bright (1996). At the same time, under the Unicode initiative, there has been solid progress in the definition of and, finally, the implementation of standards for computer encoding and rendering of scripts used around the world. However, the implications of this multi-lingual revolution for computational linguistics beyond the level of word-processing have not been well explored, and Richard Sproat's book *A Computational Theory of Writing Systems* is a welcome contribution.

In particular, Sproat's observations and theories are motivated and tested by years of work at AT&T on text-to-speech systems. This is not the first or the last time that the rigor of computational application, and the massive practical testing that it allows, will come back to shape theory.

2. Derivational geometry

Sproat provides very little in the way of an introduction to writing systems; after pointing the reader to reliable sources, he jumps boldly into his model of reading devices. The geometry of his model is traditional derivational phonology, involving a mapping through various levels from an underlying representation U to the surface phonology. Sproat makes two major claims:

1. Consistency: Every orthography corresponds to a single Orthographically Relevant Level (ORL) in the derivation; and
2. Regularity: $M_{ORL \rightarrow \Gamma}$, the mapping from the ORL to Γ, the spelling itself, is regular.

In the current academic world, derivational approaches to phonology are out of fashion, overshadowed by Optimality Theory (Prince and Smolensky 1993) and other mono-stratal models. The Two-Level finite-state model of computational morphology (Koskenniemi 1983) also eschews derivation. Nevertheless, a great deal of computational phonology and morphology continues to be done with derivational cascades of rewrite rules, which Johnson (1972) and Kaplan and Kay (1994) showed to be only

finite-state in power. Computational linguists from this tradition have challenged OT (Karttunen 1998; Gerdemann and van Noord 2000), assuming that as long as linguists stay within finite-state power, they can construct their grammars flatly or derivationally as they find most convenient and perspicuous.

The interesting contribution of Sproat to this ongoing debate is his argument that the derivation has explanatory power, modeling "orthographical depth." In this model, English orthography reflects a fairly deep ORL, Russian a shallower one, Belarusian a level slightly shallower than Russian, Spanish quite shallow, and so on.

Sproat's claim that his derivations are regular is testable and potentially disprovable; he is forthright in discussing challenges and apparent counterexamples. The second claim, that an orthography always represents a consistent level of representation, and not sometimes one level and sometimes another, is a stronger claim, especially given the messy history of borrowing and adapting scripts. Scripts, as Sproat himself points out, are contingent "artifacts" or technologies, not something inherently human like phonology. He argues credibly, however, that a viable everyday writing system must bear a "sensible relationship" to the language it represents, that we can expect a natural pressure in the direction of consistency.

3. Derivational breakdown

Sproat's $M_{ORL \rightarrow \Gamma}$ rules are divided into two main subgroups: M_{Encode}, which are (morpho)phonological mappings, and M_{Spell}, which are "autonomous spelling rules" or rules reflecting the conventions of the orthography itself. As the rules are regular, they can simply be composed together, and the result can be encoded as a finite-state transducer and applied bidirectionally.

$$M_{ORL \rightarrow \Gamma} = M_{Encode} \circ M_{Spell}$$

This distinction is easy to defend. M_{Spell} covers phenomena such as the conventions for representing phonologically long vowels in Dutch orthography, and parallel examples in other orthographies are easy to find.

Sproat then makes another, less-obvious, distinction, splitting up M_{Spell}, so that $M_{Spell_{map}}$ is a mapping, encoded as a finite-state transducer, but $M_{Spell_{constrain}}$ is encoded as a regular-language filter. Again, the two subsystems are regular and are composed together.

$$M_{Spell} = M_{Spell_{map}} \circ Id(M_{Spell_{constrain}})$$

As composition is defined only for transducers, the composition must technically involve the identity relation on the filter as shown.

The examples of $M_{Spell_{constrain}}$ involve alternate representations that appear in complementary distribution in the surface orthography. In Malagasy, the vowel /i/ is represented as either $\langle i \rangle$ or $\langle y \rangle$, with $\langle i \rangle$ occurring only in nonfinal position and $\langle y \rangle$ occurring only at the end of words. If M_{Encode} contains the rule

$$i \rightarrow \langle i \rangle \mid \langle y \rangle$$

that is, vowel /i/ is realized as either orthographical $\langle i \rangle$ or as $\langle y \rangle$, then $M_{Spell_{constrain}}$ would include the following regular filter to constrain the variants to appear only in appropriate contexts.

$$\neg[(\Sigma^* \langle i \rangle \ \#) \mid (\Sigma^* \langle y \rangle \ \neg\#)]$$

Downloaded from http://direct.mit.edu/coli/article-pdf/27/3/464/1797839/coli.2000.27.3.464.pdf by guest on 11 August 2022

Grammars that overgenerate and then filter in this way have an obvious OT flavor. However, while this approach would definitely seem to work, it is difficult to see how it differs substantially from the following two mapping rules.

$$i \rightarrow \langle y \rangle / \_ \# \quad \circ \quad i \rightarrow \langle i \rangle$$

Here the phoneme /i/ is first mapped to $\langle y \rangle$ at the end of words, and elsewhere, any leftover /i/ is simply mapped to $\langle i \rangle$. If orthographical $\langle y \rangle$ and $\langle i \rangle$ are excluded from the domain, then the following equivalence holds.

$$\begin{aligned} &\neg[\Sigma^*(\langle y \rangle \mid \langle i \rangle)\ \Sigma^*] \ \circ\ i \rightarrow \langle i \rangle \mid \langle y \rangle \ \circ\ \neg[(\Sigma^* \langle i \rangle\ \#) \mid (\Sigma^* \langle y \rangle\ \neg\#)] \\ &\quad \equiv \quad \neg[\Sigma^*(\langle y \rangle \mid \langle i \rangle)\ \Sigma^*] \ \circ\ i \rightarrow \langle y \rangle\ /\_\ \# \ \circ\ i \rightarrow \langle i \rangle \end{aligned}$$

The filter component $M_{Spell_{constrain}}$ is also invoked for handling the alternation of Greek nonfinal σ vs. final ς, and for the contextual variant shapes of Arabic (what Unicode calls the character vs. glyph distinction). Again, it is difficult to see why these same phenomena could not be handled with mappings or transducers rather than filters, according to the taste and convenience of the linguist.

4. Planar regular language

Sproat expands the normal notion of regular language, consisting of strings of linearly concatenated symbols, to planar regular languages, which allow a richer set of concatenation operations, including left concatenation ($\rightarrow$), right concatenation ($\leftarrow$), downwards concatenation ($\downarrow$), upwards concatenation ($\uparrow$), and surrounding concatenation ($\odot$). Illustrated on Chinese, this notation allows for the grouping of semantic radicals and (semi-reliable) phonetic elements into traditional Chinese morphograms. The same mechanism is applied, even more successfully, to Korean Hankul (Hangul), where letter units are arranged into syllable-sized glyphs, and to Devanagari and Pahawh Hmong. Planar regular expressions are therefore the mechanism proposed for notating the relative placement of glyphs in all "Small Linguistic Units," where variation from the macroscopic order of the script is possible.

5. Conclusion

Any theory of writing systems takes on a huge task, ultimately beyond the ability of any single human being. Sproat's references are intimidatingly wide, and his examples and experiments include Chinese, Korean, English, Russian, Belarusian, Croatian, Mayan, Manx Gaelic, Egyptian Hieroglyphics, Syriac, Malagasy, and others. He applies his theory to classification, adaption of writing systems, spelling reform, and the psycholinguistics of reading. Yet he's repeatedly candid about the challenges to and limitations of his theory (e.g., calligraphy is beyond the pale), and he is to be commended for pointing out the areas needing further research and confirmation.

Individual readers will of course want to see the theory tested on their own favorite writing systems; I, for one, wanted to see how the model might apply to Arabic orthography, and how the model might shed light on the debates about rival proposed orthographies in the field, for example, for Bantu languages. Critics will want to test the claims of Regularity on reduplicating languages, and the claims of Consistency against the amount of ad hoc "lexical marking" needed to support it. But Sproat has laid out a testable theory, and in the best scientific spirit he has even provided listings of lexicons and derivation rules for English, contrasting a deep ORL solution à la

Downloaded from http://direct.mit.edu/coli/article-pdf/27/3/464/1797839/coli.2000.27.3.464.pdf by guest on 11 August 2022

Chomsky and Halle (1968) with a shallower solution. To complete the picture, the AT&T finite-state libraries and a set of programming formalisms called Lextools are now available on the Web, which further facilitates the reproduction of his English examples and testing on other languages.[1]

All in all, this book is commendable, and it is highly recommended for any serious student of writing systems.

References

Chomsky, Noam and Morris Halle. 1968. *The Sound Pattern of English*. Harper and Row, New York.

Daniels, Peter T. and William Bright. 1996. *The World's Writing Systems*. Oxford University Press, Oxford.

Gerdemann, Dale and Gertjan van Noord. 2000. Approximation and exactness in finite state Optimality Theory. In: Jason Eisner, Lauri Karttunen and Alain Thériault, editors, *Finite-State Phonology: Proceedings of the Fifth Workshop of the ACL Special Interest Group in Computational Phonology (SIGPHON)*. Luxembourg, August 2000, 34–45.

Johnson, C. Douglas. 1972. *Formal Aspects of Phonological Description*. Mouton, The Hague.

Kaplan, Ronald M. and Martin Kay. 1994. Regular models of phonological rule systems. *Computational Linguistics*, 20(3): 331–378.

Karttunen, Lauri. 1998. The proper treatment of optimality in computational phonology. *Proceedings of FSMNLP'98, International Workshop on Finite-State Methods in Natural Language Processing*, Bilkent University, Ankara, Turkey. (cmp-lg/9804002.)

Koskenniemi, Kimmo. 1983. Two-level morphology: A general computational model for word-form recognition and production. Publication 11, University of Helsinki, Department of General Linguistics.

Prince, Alan and Paul Smolensky. 1993. Optimality Theory: Constraint interaction in generative grammar. RuCCS Technical Report 2, Rutgers Center for Cognitive Science, Rutgers University.

Kenneth R. Beesley is a Principal Scientist at the Xerox Research Centre Europe. He works mostly in finite-state morphological analysis and generation. Beesley's address is Xerox Research Centre Europe, 6, chemin de Maupertuis, 38240 Meylan, France; e-mail: Ken.Beesley@xrce.xerox.com.

1 *http://www.research.att.com/sw/tools/fsm/*, *http://www.research.att.com/sw/tools/lextools/*

Downloaded from http://direct.mit.edu/coli/article-pdf/27/3/464/1797839/coli.2000.27.3.464.pdf by guest on 11 August 2022