# PHONOLOGICAL RULE IMPLEMENTATION IN SPEECH RECOGNITION

CHARLES HOEQUIST, JR.

University of Cambridge
Linguistics Department
Cambridge CB3 9DA
United Kingdom

## ABSTRACT

This paper compares the implementation of different types of phonological rules in a system providing limited dialect normalization. Dialect normalization will be sketched briefly, as a means of simplifying the speaker-normalization task.

Two phonological-rule implementations are compared: a representative of parsing by finite-state deterministic automata similar to those in Koskeniemmi [1] and of context-free phrase-structure rules like those proposed by Church [2].

## INTRODUCTION

Automatic speech recognition (ASR) devices for continuous speech are forced to take account of rule-governed variation in the acoustic signal in a way that isolated-word recognizers are not. In the latter, recognition can be treated as a problem for sophisticated pattern matching. Continuous-speech recognizers, on the other hand, must attempt to cope with the inevitable acoustic variation resulting from, among other things, language-specific regularities governing the realization of segments or syllables in specific environments, that is, phonological rules.

The idea that rule-governed variation in the speech signal can best be handled by some analogue to a linguist's phonological rules is not in itself new. Several recognition devices which grew out of the ARPA project in 1971-1976 used rules to expand base dictionaries into dictionaries containing (it was hoped) all possible phonetic realizations of the dictionary's words [3]. With a realistically large rule set and large vocabulary, this sort of expansion is likely to become impractical. The implementations discussed here run in the other direction, that is, rules are applied to a labeled, segmented input string to produce candidates for matching to a fixed set of lexical entries. Our interest here is, however, not the direction the rules run, but the constraints on the power of the rule formalism and on constraining their application.

## LINGUISTICS AND SPEECH RECOGNITION

Linguists' phonological descriptions of the last quarter-century have been overwhelmingly cast in the form of a single set of context-sensitive transformational rules, that is, rules which are capable of rewriting the phrase markers making up a string, in this case phonetic segments. The direction of operation should be irrelevant, so lexical items can be transformed into surface strings and vice versa.

Such a phonological grammar component has at least two major problems, concerning dialect and the formal power of the rules. It is impractical to try to design a single pandialectal set of rules for a recognition system to map inputs onto a single lexicon. Though it is imaginable that one set of rules could be written that would correctly map an input string onto the intended string of lexemes, it would in the process generate a considerable number of false mappings due to the application of rules which, by virtue of belonging to a different dialect, played no part in the production of the input. Training to an individual speaker might get around this problem (assuming for the moment that the mapping from one dialect's phonological system to another's is isomorphic), but at the cost of sacrificing generality that could be captured by the proper rules. This can be handled by the use of dialect normalization, in which an initial training phase establishes not only idiosyncratic but dialectal characteristics of the speaker, and uses these to determine what subset of rules is the most suitable. This strategy is used in the ASR section of the Cambridge Alvey project.

But even if the rules in a phonological component are marked for which dialect they apply to, the context-sensitive transformational rule formalism is itself problematic. This is because it is in some cases impossible to uniquely reconstruct the pre-transformation string, because of the ability of transformational rules to modify phrase markers. In practice, this can lead to a given phonetic surface string being traceable to several possible phonological strings, analogous to syntactic ambiguity (something similar can happen with context-free rules as well, when several structures are assigned to a given segment string; but in practice the problem is much less severe).

As in much recent work in syntax, one solution is to restrict the power of rules. As examples of context-sensitive transformations versus formally more constrained rules, two phonological parsers will be discussed here, one using exclusively context-free rules, the other allowing some context specification. The respective strengths and weaknesses of the two systems will serve to illustrate some of the requirements a phonological grammar needs to fulfill.

## PHONOLOGICAL PARSERS

### Context-free rules in a phrase-structure phonology

Using only context-free rules in the phonological component of a recognizer might seem hopeless at first; phonological processes are heavily context-dependent. As will be seen, the 'context-free' system discussed here does not ignore context, but encodes it in such a way that a context-sensitive formalism is claimed to be unnecessary.

The most developed example of a context-free phonological parser is to be found in Church [2]. There, a segment lattice (with no word- or syllable-boundary specifications) serves as the input to a chart parser operating with a set of context-free phrase structure rules. The parser outputs a hierarchical structure with a string of segments at its lowest level, dominated by nodes identifying syllable components, syllable boundaries, and stress-foot boundaries. This serves in its turn as input to a syllable-based lexicon.

Though the parser would be interesting to linguists simply on the grounds of the use of nonlinear phonological structure, the hierarchy of units is equally important for its restriction of the parser's power. The encoding of phonological processes, many of them contextually dependent, into context-free rules is done as a byproduct of the hierarchical structuring implicit in the chart parser. The contextual dependencies which obtain among successive segments are encoded into the higher-level units (syllables and feet), thus enabling the parser to handle many context-conditioned processes without requiring the generative capacity of a context-sensitive system. For example, the rule in English which aspirates voiceless stops in syllable-initial position would be expressed as a phrase structure rule such as

$$\text{onset} \rightarrow p^h \mid t^h \mid k^h$$

expanding some nonterminal symbol for syllable onset into aspirated voiceless stops. Thus, when parsing an input string, such segments would be labeled as syllable onsets, which in turn would allow the parser to hypothesize syllable boundaries for dictionary lookup.

This is fine, as far as it goes. However, the system (at least as Church presents it) suffers from two types of problem, one specific and system-oriented, one more general and based on the philosophy of context-free parsing.

The first difficulty can be characterized as a perfect-input requirement. This refers to the system's requirement of perfect, fine-grained phonetic labeling, and the consequences both of mislabeling, or even of labeling in too broad a fashion. Let us look at the initial-aspiration rule again. If the recognizer's front end ever fails to recognize a syllable-initial voiceless stop as aspirated, the parse fails. It is unreasonably optimistic to expect flawless performance from any front end. This difficulty weighs heavily, because a proper parse in this system depends on the accurate labeling of such phonetic detail. The problem can be evaded by making the rules more general, but this has the accompanying cost of dramatically increasing the number of valid parses formed from any given string.

Even if the initial segmenting and labeling process were to work as well as is necessary, the second problem, linked to context-free parsing itself, would remain. A parser of the sort described above decides whether a given string is allowed (that is, can be assigned a structure), given a particular rule set. Because this rule-set is composed of context-free phrase-structure rules, no rewriting of segment labels is allowed. Since segment labels cannot be altered, the phonological parser's job here is really to judge whether its input is phonotactically possible. Its alterations of input therefore only consist of the insertion of nonterminal symbols, in this case syllable and foot elements. This fails to address the existence of phonological processes involving segment insertion (e.g. epenthetic [t]) and deletion (e.g. disappearance of schwa vowels). Such rule-governed alterations in the segments are also troublesome for their ability to yield surface realizations which violate the phonotactics of English. To avoid rejection of such strings, a recognition system would have to broaden its allowable input, either by listing numerous alternative realizations of a word in the lexicon, yielding a system like that produced by the lexicon expansion rules mentioned above, or by increasing the number of rules to allow several optional rewritings of the proper non- terminal symbols. Either alternative increases the number of parses of a string and so reduces the advantage of context-free over context-sensitive rules.

### Context-sensitive rules

The implementation used here as an example of context-sensitive parsing is the 'two-level' parser developed first by Koskenniemi [1]. Though his intention was to use it for morphological decomposition of an input string, it has qualities which suit it for phonological rule implementation as well.

The core of two-level parsing is its encoding of rules into nondeterministic, finite-state automata, which simulate context-sensitive rules. Since context-sensitive rules are formally more powerful than finite-state machines, it would be possible to write rules to generate strings which could not be correctly analyzed by any finite-state device. However, phonological processes do not seem to ever produce such outputs. These automata can be envisioned as moving simultaneously along two 'tapes' (hence the name 'two-level'), one representing the input and one representing a graph through the lexicon, which has a tree structure such that a node contains information about whether a lexical entry can end at that point, and if continuation is possible, what lexical characters can follow.

The rules check whether the current input character can be matched to the current lexical character. If all the rules allow the current pairing, the next character is taken from the input and checked for any allowable matches in the lexicon, and the process is repeated. This continues until the end of the input is reached (a successful parse) or no pairing of lexical and input characters is possible at some point. In contrast to Church's parser, the rules are context-sensitive, and the parse is left-to-right.

This shares with the context-free system the advantage of not needing word boundaries to be specified in advance, and so avoiding the problems caused by the reliance of a whole-word matcher on reliable word boundaries. It differs, however, in the set of strings it will pass and in the types of rules that it can express. The first difference is due to the structure of the two-level parser, and is not connected to context-sensitivity in rules. Church's parser first clusters the input segments into higher-level units, which are then used in lexical search. In the two-level system, the search is an integral part of parsing the input string. Since the rules are comparing input and lexical strings, a parse may fail simply because one of the input words is not present in the lexicon (compare this with the lexicon-independent output of the Church parser). The advantage of this is that many unusable analyses (consisting of phonotactically legitimate nonwords) are filtered out early. The second difference is that which we have been emphasizing, that rules are allowed which rewrite segment labels, thus allowing for the restoration of deletions, the undoing of neutralizations, etc.

## CONCLUSION

It is the case that many phonological processes often expressed in a context-sensitive formalism can be expressed in a more restricted system. However, the inability of context-free phonological rules to alter the segments which constitute their input puts them at a severe disadvantage when faced with the output of those processes in speech production which alter segment identity (e.g. neutralization) or create surface violations of phonotactic rules through insertion or deletion of segments. Though it is possible to create a recognizer without context-sensitive rules between the input and the lexicon (recognizers can function without rules at all), the variation handled by the rules must be taken care of somewhere else. In this regard, it is significant that the context-free implementation discussed above includes a 'canonicalizer' level (Church, pp. 44-45), which removes phonetic detail and and tries to recover altered or deleted segments prior to lexical access. This latter amounts to putting in an extra set of rules to perform operations involving rewriting and inserting of phrase markers, i.e. context-sensitive transformational rules.

Nonetheless, a context-sensitive formalism cannot be said to be optimal. The very power which enables it to undo the effects of processes which defeat context-free rules also overgenerates mappings between the input and the lexicon. The obvious next step is to find ways to prune these mappings. Two methods are already in use. One is that of automatically checking the lexicon to see whether it contains the segment sequences which the rules produce, and ceasing to follow any hypothesized output which contains nonwords. The other is the reliance on an early decision as to the speaker's dialect to determine which subset of the existing rules will be applied to an input string, rather than simply trying to handle all dialects with one large rule set. Another possibility would be to apply some rules only when the system found evidence of fast-speech phenomena. Further reduction in the number of hypotheses could be achieved by the use of syntactic knowledge, to forbid sequences of lexical items that cannot be syntactically parsed.

## REFERENCES

[1] Koskenniemi, Kimmo. "Two-Level Morphology", PhD dissertation, University of Helsinki, 1983. Printed in "Texas Linguistic Forum", vol. 22, June 1983, 1-164.

[2] Church, Kenneth W. "Phrase-Structure Parsing: A Method for Taking Advantage of Allophonic Constraints", MIT dissertation, distributed by IULC, June 1983.

[3] Shoup, June E. Phonological Aspects of Speech Recognition. in: W. Lea (ed.), "Trends in Speech Recognition", New Jersey: Prentice-Hall, Inc., 1980, 125-138.

## ACKNOWLEDGMENTS

This paper is based on work carried out as part of the Linguistics Department's component (SERC grant GR/D/42405) of Alvey research project MM1/069 on Automatic Speech Recognition, which involves Cambridge University, the MRC Applied Psychology Unit, and Standard Telecommunication Laboratories Ltd.

Se 23.2.4