PERFORMANCE COMPARISON OF COMPONENT ALGORITHMS FOR THE PHONEMICIZATION OF ORTHOGRAPHY

Jared Bernstein
Telesensory Speech Systems
Palo Alto, CA 94304

Larry Nessly
University of North Carolina
Chapel Hill, NC 27514

A system for converting English text into synthetic speech can be divided into two processes that operate in series:

1) a text-to-phoneme converter, and

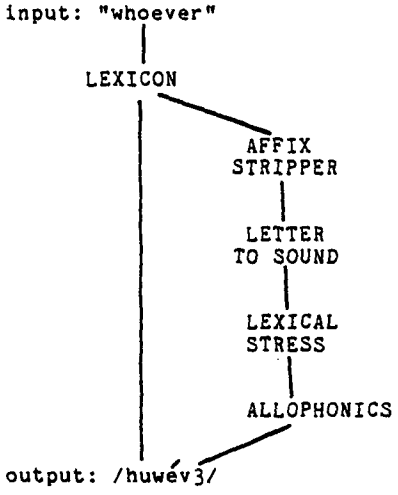
2) a phonemic-input speech synthesizer.

The conversion of orthographic text into a phonemic form may itself comprise several processes in series, for instance, formatting text to expand abbreviations and non-alphabetic expressions, parsing and word class determination, segmental phonemicization of words, word and clause level stress assignment, word internal and word boundary allophonic adjustments, and duration and fundamental frequency settings for phonological units.

Comparing the accuracy of different algorithms for text-to-phoneme conversion is often difficult because authors measure and report system performance in incommensurable ways. Furthermore, comparison of the output speech from two complete systems may not always provide a good test of the performance of the corresponding component algorithms in the two systems, because radical performance differences in other components can obscure small differences in the components of interest. The only reported direct comparison of two complete text-to-speech systems (MITALK and TSI's TTS-X) was conducted by Bernstein and Pisoni (1980). This paper reports one study that compared two algorithms for automatic segmental phonemicization of words, and a second study that compared two algorithms for automatic assignment of lexical stress.

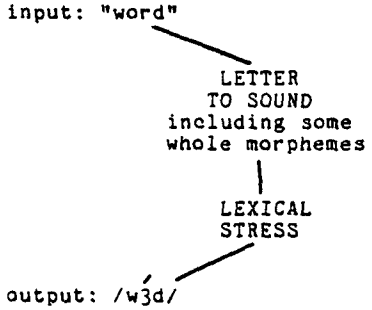
Only three systems for text-to-phoneme conversion have been reported in detail: McIlroy's (1974) Votrax driver, Hunnicutt's (1976) rules for the MITALK system, and the NRL rules developed by Elovitz and associates (1976). Liberman (1979), Hertz (1981), and Hunnicutt (1980) have described more recent systems, but have not published rule sets.

One fairly standard approach to automatic phonemicization of words has the following component parts:



Several research systems are of this general design, including Allen's MITALK system, the TTS-X prototype at Telesensory Systems, and Liberman's proper name phonemicizer.

The most popular text-to-phoneme design is the NRL approach, which has only two components, of which only the first is presented in detail and evaluated by Elovitz. The original NRL system is:



The very great advantage of the NRL approach is the unified treatment of letter sequences, affixes, and whole words. There is exactly one pass through a word, left to right, in which the maximum string starting with the leftmost unphonemicized character is matched. These strings are sometimes whole words, sometimes affixes, and sometimes consonant or vowel sequences or word fragments like "BUIL". The main constraint of the system is its greatest attraction: the unity and simplicity of the code that scans the word and accesses a single table of letter strings. In contrast to this, the MITALK system, for instance, has one module and associated table structure for lexical decomposition of whole words, another module for stripping common affixes, and a third module for translating consonant and vowel sequences that remain in the pseudo-root of the word.

STUDY ONE

Study One reports a comparison of two routines for translating orthographic letters into segmental phonemes: Hunnicutt@TSI and NRL@DEC.

Hunnicutt@TSI is the affix stripper and letter to sound rules as discribed in AJCL Microfiche 57, and implemented in MACRO-11 in Telesensory Systems' TTS-X prototype text-to-speech system. Hunnicutt's system was modified only slightly in translation, and about 20 rules were added. The system starts from the right end of the word and identifies as many suffixes as it can from a table of about 140 suffixes, proceeding toward the beginning of the word until either the remainder (pseudo-root) of the word has no vowel or fewer than three letters, or no more suffixes can be matched. Next, a similar proceedure works from the beginning of the word, matching as many prefixes as it can from a table of about 40 prefixes. Finally, the pseudo-root of the word is scanned left to right twice, once translating the consonants, and next translating the vowels.

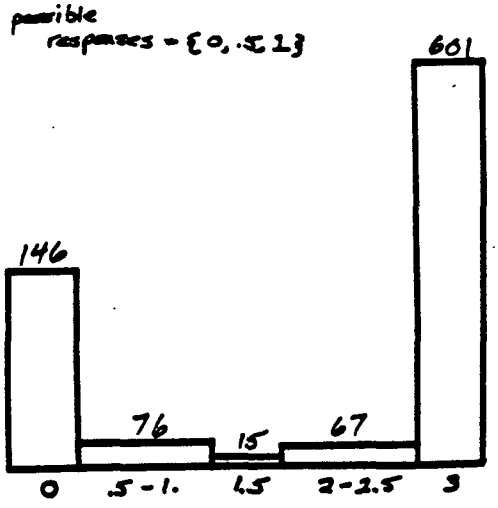
NRL@DEC is a system implemented by Martin Minnow at Digital Equiptment Corp. The whole system is somewhat more elaborate that the original NRL system, but the letter to sound module and its mode of operation are basically as described by Elovitz et alia, with 20 or 30 rules added. The NRL rules include about 60 very common whole words, as well as about 25 rules that handle various environments for three prefixes and fifteen suffixes.

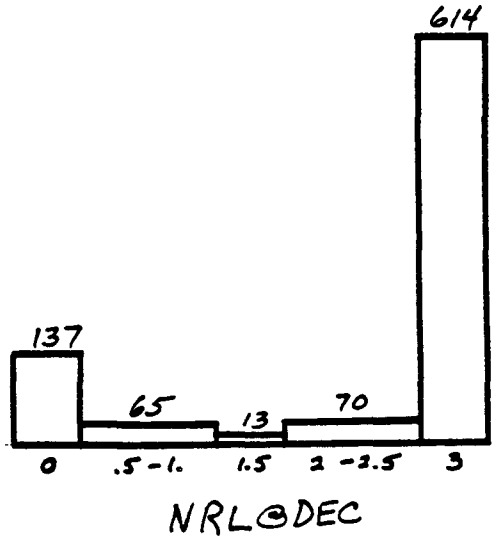
A set of 865 words was processed both by the Hunnicutt@TSI affix stripper and letter to sound rules, and by the NRL@DEC letter to sound rules including the affix rules and the word fragments. The 865 words comprised approximately every fiftieth word of the Brown Corpus (Kucera & Francis, 1967) in frequency order, starting from about the 400th most frequent word: "position." The lexicon of the TSI system was disabled, and none of the whole words in the NRL rules was in the set of 865.

Since the output from both subsystems was tapped before stress assignment, vowel reduction, and any allophonics were performed, the criterion of correctness was "does this phonemicization represent any acceptable pronunciation of the spelled word, assuming one can assign stress correctly and then reduce vowels appropriately." Thus, a phonemicization consistent with any possible word class for that spelling, or any 'regular' regional pronunciation was to be accepted. Three judges (two phoneticians and a phonologist) were given printed copies of the two resulting phonemic transcriptions; both were in fairly transparent broad phonemic form. The judges chose among three possible responses to each word: 1 = correct; .5 = close or questionable; and 0 = wrong. Cross judge consistency can be seen from the bimodal distribution of summed scores in Figure 1.

FIGURE 1

Summed Scores from 3 Judges on 865 words





Another, more diagnostic way to view the results is to present the number of words that fall into each cell of a 2X2 grid formed by the Hunnicutt@TSI rating vs. the NRL@DEC rating, as shown in Figure 2. Figure 2 omits the 26 words that had a summed score of 1.5 for either of the two letter to sound systems.

FIGURE 2

Hunnicutt@TSI \ NRL@DEC	<1.5	>1.5
<1.5	a: 127	b: 90
>1.5	c: 69	d: 553

If the rule sets were equivalent, the grid would have zeroes in cells b and c. If one rule set were a super-set of the other, you would get a zero in cell b or cell c, but not both. Most of the 553 words in cell d are regular, or else are common exceptions (like "built"). Most of the 127 words in cell a are obviously exceptional (e.g. "minute, honor, one, two").

Examination of the 159 words distributed between cells b and c yields the payoff. Of the 69 words that Hunnicutt@TSI got right and NRL@DEC missed, nearly half are correct by virtue of the extensive affix stripping in Hunnicutt's algorithm. Among these 69 words in cell c are "mobile, naval, wallace, likened, coworkers, & reenacted."

Of the 90 words that NRL@DEC got right and Hunnicutt@TSI got wrong, only about 15 are definitely due to NRL's word fragment rules. Six of the 90 words are in cell d just because NRL does not strip suffixes the way that Hunnicutt's rules do. These six words are "november, visited, preferably, presidency, september, & oven."

In general, both algorithms get about 25% wrong on this lexically flat sample of 865 word types. About 15% of the words are incorectly phonemicized by both subsystems. This might suggest that 15% wrong may be a state of the art performance level for segmental phonemicization of word types by sets of 400 rules.

STUDY TWO

Study Two compared the performance of two algorithms for assignment of lexical stress to words. Both of the algorithms were coded in MACRO-11 and ran in different versions of TSI's TTS-X prototype text-to-speech system. The first algorithm is Hunnicutt's lexical stresser, which is described in detail in AJCL Microfiche 57. Hunnicutt's algorithm is an adaptation of Halle's cyclic stress rules for English. The adaptations include adjustments for the less specified input to the rules (e.g. the part of speech of the root is unknown), and the number of stress levels specified in the output is reduced, presumably because the Klatt synthesizer it was designed to drive only used two stress levels. Hunnicutt also added stress rules that depended on the occurance of certain classes of suffixes. Hunnicutt's rules require several pointers and a suffix table, they sometimes pass through a word several times in the manner of Chomsky & Halle's (1968) rules, and they occupy about 3K bytes of executable code in their TSI version.

The second algorithm is a simplified version of a stress rule proposed in Hill & Nessly (1973). We will refer to this rule as Nessly's default, since it is the default case of Nessly's full stress algorithm. Nessly's default stress is quite similar to Latin stress and to the "first approximation" stress rule discussed twoard the beginning of Chomsky & Halle's chapter three (1968, pp.69-77). The main differences between Nessly's default rule and Chomsky & Halle's "first approximation" are:

(1) No word class information is used in Nessly's default, so verbs are stressed as nouns.

and (2) What constitutes a "strong cluster" (which contains a tense vowel or a closed syllable end) is different. Nessly's default is indifferent to vowel length or tensity.

Nessly's default rule can be outlined as follows:

```
if(number of syllables = 1)
        stress it.

if(number of syllables = 2)
        stress left syllable.

else
   skip the last syllable.

   if(next-to-last is closed)
           stress it.

   else
           stress third from last.

   (place alternating 2nd stresses
    on syllables to the left.)
```

The MACRO-11 version of this rule requires about 150 bytes of executable code, and accepts one pointer to the last vowel in the word. It passes through the word once, right to left, and it does very well assigning correct stresses (in caps) to "LUminant" vs. "maLIGnant," for example.

For testing the stress algorithms, a sample of 430 words was selected. These 430 words were all the items of five or more characters that had frequencies of 40 ppm through 34 ppm (inclusive) in the Brown corpus. The segmental phonemicization was done by Hunnicutt's rules in TSI's TTS-X prototype. The automatically produced segmental phonemicizations that the stress algorithms operated on were rejected only if they did not have the correct number of syllables. Thirteen of the 430 words were phonemicized with the wrong number of syllables. Another 54, or 13%, of the 430 were one syllable words, which were allways assigned correct

stress. Stress assignments were judged by the first author. The results on the remaining 417 words of the sample were:

	Correct	Wrong
Hunnicutt/Halle	308	109
Nessly default	303	114

So, on these words, the two algorithms perform at about the same level of accuracy, which is about 25% wrong on a lexical sample.

DISCUSSION

In both studies, very simple algorithms performed about as well as algorithms of vastly greater complexity. In the case of the letter-to-sound algorithms (Hunnicutt@TSI and NRL@DEC), the difference in complexity is primarily in the procedure for checking the rules against the word. Hunnicutt's rules themselves are only a little more complicated than the NRL rules. Presumably, with some modification, most of Hunnicutt's rules could be modified to run within a one-pass NRL procedure.

The stress algorithms tested in Study Two present a very great contrast in both number of rules and procedure for rule application. If Nessly's default rule is like a simplified version of Chomsky & Halle's "first approximation" stress rule, and if Hunnicutt's algorithm is fairly close to Chomsky & Halle's full lexical stress rules (with noun-root assumed), then our data suggest that the epicyclic accretion that produced Chomsky & Halle's full set of stress rules from their "first approximation" has gained almost nothing in lexical coverage.

We have reported performance in terms of percent wrong on samples of word types from the Brown corpus. It seems that an appropriate measure of performance that reflects what people feel when they hear a text-to-speech system is AVERAGE WORDS BETWEEN ERRORS (AWBE). We would like to end this paper by giving AWBE for a simple text-to-phoneme system with a 25% error rate in both letter-to-sound conversion and lexical stressing, and a lexicon with 1500 words.

If the lexicon is in parallel with the letter to sound and stress rules, and the performance of the letter to sound rules and the stress rules are independant, an overall error rate of about 7% can be expected. This would translate into an AWBE of 13.3.

ACKNOWLEDGEMENTS

The authors gratefully acknowlege valuable help from Martin Minow, Peter Maggs, Margaret Kahn, and Julie Lovins.

REFERENCES

J.Bernstein & D.Pisoni (1980) "Unlimited text-to-speech system: Description and evaluation of a microprocessor based device," IEEE ICASSP-80 Proceedings.

N.Chomsky & M.Halle (1968) THE SOUND PATTERN OF ENGLISH, Harper-Row, New York.

H.Elovitz, R.Johnson, A.McHugh, & J.Shore (1976) "Letter-to-sound rules for automatic translation of English text to phonetics," IEEE Trans. on Acoustics, Speech, and Signal Processing, vol. ASSP-24, no. 6.

S.Hertz (1981) "SRS letter to sound rules," IEEE ICASSP-80 Proceedings.

S.Hunnicutt (1976) "Phonological rules for a text-to-speech system" AJCL Microfiche 57.

S.Hunnicutt (1980) "Grapheme to phoneme rules: a review" KTH SLT-QPSR 2-3/1980, Stockholm.

H.Kucera & W.Francis (1967) COMPUTATIONAL ANALYSIS OF PRESENT DAY AMERICAN ENGLISH, Brown U. Press, Providence.

M.Liberman (1979) "Text-to-speech conversion by rule and a practical application," Proceedings of the Ninth International Congress of Phonetic Sciences, Copenhagen.

M.McIlroy (1974) "Synthetic English speech by rule," Bell Telephone Laboratories Memo.

K.Hill & L.Nessly (1973) "Review of The Sound Patten of English," LINGUISTICS 106: 57-101.