THE KLATTALK TEXT-TO-SPEECH CONVERSION SYSTEM

Dennis H. Klatt

Speech Incorporated
Box 169, MIT Branch Post Office, Cambridge, MA Ø2139
(Also at Mass. Inst. of Tech.)

ABSTRACT

A real time text-to-speech conversion system has been developed. Input is ordinary English spelling and/or simple numerical and algebraic expressions. Dynamic selection between a male or female output voice is under user control. The system executes a set of about 5ØØ letter-to-sound rules to guess at the pronunciation of words that do not match a carefully selected exceptions dictionary of about 15ØØ words. A very simple syntactic analyzer determines probable locations of phrase and clause boundaries in order to improve the naturalness and intelligibility of input sentences. The resulting phonemic representation is converted to speech by a synthesis-by-rule program and formant synthesizer. The rule program differs from others of this type in having an extensive set of segment duration rules and many detailed rules for the synthesis of consonant-vowel transitions.

INTRODUCTION

Speech synthesis technology has progressed to the point where a whole new generation of devices is finding its way into the marketplace. In applications where only a small number of words are to be spoken, real speech can be digitized, encoded, and stored for later decoding and playback on command. There are other applications where stored speech is impractical. Either the vocabulary is too large or the words must be composed into one of many possible well-formed English sentences. Examples include:

1. Reading machines for the blind
2. Speaking aids for the vocally handicapped
3. Talking computer terminals
4. Remote reception of electronic mail by phone
5. Teaching machines and training aids
6. Talking books to teach reading
7. Flexible data base inquiry systems
8. Remote access to information over the phone
9. Talking instrument panels
10. Hobbyist computers

A system for converting any typed English sentence into a spoken utterance is ideally suited for these applications. One such text-to-speech conversion device, Klattalk(1) , is the topic of this paper.

Text-to-speech conversion is usually thought of as a two-step procedure. The text is first converted to an abstract underlying linguistic representation consisting of phonemes, stress marks, and syntactic structure indicators. Then the sequence of phonemes is converted to sound by a set of rules that drive a vocal tract model, in this case a formant synthesizer.

TEXT-TO-PHONEME CONVERSION

The text-to-phoneme component includes modules for formatting input data into well-formed words, examining a small pronouncing dictionary, executing a set of letter-to-sound rules for words that do not match the pronouncing dictionary, and performing a rudimentary syntactic analysis of each input sentence.

Formatting Preprocessor

Any system that is to read arbitrary English text must be able to deal with "nonstandard" input words such as digit strings, abbreviations, and special symbols. In Klattalk, numbers are translated into the appropriate word sequence by an algorithm designed by Sharon Hunnicutt. Some common abbreviations are stored in an exceptions dictionary. Other abbreviations are pronounced as a word if they contain a vowel, but are pronounced by spelling out the letters otherwise. Unanticipated special symbols are ignored.

Letter-to-Phoneme Conversion

English spelling is complex but not arbitrary. If one knows something about the history of the language [7]. and if one is willing to treat some of the most common English words as exceptions, it is possible to formulate letter-to-phoneme rules that

(1) Copyright January 1, 1981 by Dennis H. Klatt.

CH 1746-7/82/0000 - 1589 $ 00.75 © 1982 IEEE

work correctly with a moderately high probability [3].

We use Hunnicutt's set of about 500 letter-to-phoneme rules [3], each of which converts a letter or letter sequence into a phoneme or phoneme sequence if the letter is in a particular symbol environment. Some rules are based on linguistic principles (e.g. describing the role of "silent e" in determining the sound of the previous vowel, or the role of consonant doubling on the sound of the previous vowel), while others simply take advantage of statistical tendencies.

The rules are successful about 95% of the time at the level of a phoneme, meaning that about 75% of the words are correctly analyzed. Common words that would be mispronounced are stored in an exceptions dictionary. This dictionary is searched before letter-to-phoneme rules are executed. If the word is found in the dictionary, it does not go through letter-to-phoneme conversion. Instead, the stored phonemic sequence is retrieved. With a 1500-word exceptions dictionary, it is possible to reduce the number of words containing phonemic errors in running text substantially. It has been found that only about one word in twenty contains a phonemic error when the system is reading novels.

The system contains a set of heuristic stress rules expressed in the same formalism as the letter-to-phoneme rules. Unstressed function words are placed in the exceptions dictionary. The stress rules do well most of the time. However, a stress rule error often results in a pronunciation that is far from correct, and the listener may have difficulty in recovering from such an error.

Exceptions Dictionary

We do not use a large pronunciation dictionary in Klattalk because of the storage costs involved. However, a small exceptions dictionary is included in order to correctly pronounce common words that are handled incorrectly by the letter-to-phoneme rules. The effective size of the exceptions dictionary is increased somewhat by not only matching each input word to the dictionary, but also by attempting to remove common affixes such as "-ing" from an unknown word prior to matching against the exceptions dictionary. This avoids having to store regular plurals and past forms, etc.

The exceptions dictionary also includes a small set of unstressed function words such as "and", "of", "the", etc. These words would be stressed incorrectly if handled by the letter-to-sound rules.

Syntactic Analysis

Syntactic structure symbols are important determiners of sentence stress, rhythm, and intonation. The syntactic analyzer has the task of finding ends of clauses and ends of noun phrases. Fortunately, many clause boundaries and locations of other syntactic breaks are marked by commas in the text. For clause boundaries that are not so marked, some can be recognized through use of a set of special clause-introducing words such as "because". The break between a noun phrase and verb phrase is the hardest to detect reliably, but can be guessed at with some success by including common verbs in the exceptions dictionary.

The difficult cases (such as words that can be nouns or verbs) are far too difficult for simple heuristics -- they would require a syntactic analyzer that essentially "understood" the text. Unfortunately, current automatic text parsers have little semantic knowledge, and not only have noun/verb problems, but also discover many syntactic ambiguities (multiple parsings of the same word string) that we, as readers, do not even see because the semantically correct reading is seen first. Thus a powerful syntactic analyzer is not included in Klattalk.

Example

An example of the use of some of these symbols is provided in Figure 1. The output from the text-to-phoneme system for the sentence "The old man sat in a rocker" has been computed by (1) selecting a sequence of phonemes to represent the sound pattern of each word, (2) inserting stress symbols to indicate which vowels receive primary lexical stress, and (3) inserting syntactic boundary information, in this case a phrase boundary symbol, to break up the speech into phonological phrases for easier listening.

```
   DH AX   ' OW L D   M ' AE N  )

 S ' AE T   IH N   AX   R ' AA K RR .
```

Figure 1. Output representation from the text-to-phoneme module for the input text, "The old man sat in a rocker".

PHONEME-TO-SPEECH RULES

The phoneme-to-speech algorithm of Klattalk is divided into a phonological component, a phonetic component, and a synthesizer component. The abstract linguistic representation discussed above serves as input to the phonological component of the synthesis-by-rule program. The output from the phonological rules is a string of phonetic segments, with each segment being assigned a stress feature and duration. Fundamental frequency is also specified by rules of the phonological component.

Phonological Component

Rules contained in the phonological component are summarized below. These rules constitute an especially important part of any system designed to synthesize sentences. Without them, sentences have to be synthesized very slowly in order to overcome the deleterious perceptual effects of improper sentence rhythm, intonation, and stress pattern.

Stress Rules. The phonological component assigns a feature STRESS (value = Ø or 1) to each phonetic segment in the output string. The default value is Ø (unstressed). Vowels preceded by a ' or ! stress symbol in the input are assigned a value of 1. Consonants preceding a stressed vowel are also assigned a value of 1 if they are in the same morpheme and if they form an acceptable word-initial consonant cluster. Segmental stress is used in rules that determine segmental duration, fundamental frequency, plosive aspiration duration, and formant target undershoot.

Rules of Segmental Phonology. The segmental phonological rules are extremely important. They are not "sloppy speech" rules, but rather rules that aid the listener in hypothesizing the locations of word and phrase boundaries. Examples of segmental phonological rules include glottal stop insertion, introduction of postvocalic allophones of /r/ and /l/ that differ from syllable-initial allophones, and glottalization of word-final /t/ under some circumstances.

Fundamental Frequency. A "hat-pattern" strategy for specification of a fundamental frequency (FØ) contour [6] involves four steps:

1. Define a baseline FØ contour consisting of a gradual fall
2. Determine syntactically-conditioned times of rises to a plateau above the baseline and falls back to the baseline
3. Add in local increases in Fo due to lexical stress
4. Add in local perturbations due to segmental factors such as vowel height and consonant voicing

The influence of syntactic structure is to shift the FØ value up to a hat-pattern plateau above the baseline on the first stressed syllable of a syntactic unit, and to shift back down to the baseline on the last stressed syllable of the syntactic unit. The algorithm results in a dramatic fall-rise FØ contour between syntactic units. Fundamental frequency motions are used by a speaker to indicate aspects of syntactic structure, to highlight semantically important words, and to indicate emotions (of course the text-to-speech system is not capable of conveying different emotions or detecting locations appropriate for semantic emphasis).

Segmental Duration. The phonological component specifies inherent durations for each phonetic segment type of English and executes a set of rules that modify the inherent durations. The rules operate within the framework of a model of durational behavior which states that each rule tries to effect a percentage increase or decrease in the duration of the segment, but segments cannot be compressed shorter than a certain minimum duration.

Durational phenomena covered by the rules include pause insertion, clause-final lengthening, phrase-final lengthening, word-final lengthening, polysyllabic shortening, word-initial consonant lengthening, shortening of unstressed segments, and interactions between adjacent segments [4].

Phonetic Component

The phonetic component of the rule program produces a set of 2Ø synthesizer control parameters every pitch period. The rules contained in the phonetic component are too varied and complex to be described in detail. The rules begin by assigning a target value to each parameter for each phone. Smooth transitions between target values are computed by other rules that take into account features of adjacent phones. Then there are a set of special rules that delay voicing onset in voiceless stops and attach a burst to plosives and affricates. The advantages of this program over others are that many acoustic-phonetic details have been included in the rules so as to give significantly higher intelligibility than any other system.

Formant Synthesizer

The formant synthesizer module converts 2Ø input control parameters into gains and difference equation constants to control a special-purpose synthesizer chip. The synthesizer configuration is a somewhat simplified version of a synthesizer program that has been published in the Journal of the Acoustical Society of America [5].

SYSTEM EVALUATION

A text-to-speech system can be evaluated along a number of dimensions. The output speech should be intelligible, natural sounding, and easy to comprehend (i.e. one should not have to concentrate on hearing the words to the extent that the meaning of a sentence is forgotten).

Few systems have been evaluated systematically along these dimensions. However, both the MITalk system [1], and the Telesensory Systems system [2] have undergone extensive tests for intelligibility and listening comprehension. The results indicate that comprehension can approach that of a single reading of the same material by a human talker, but that intelligibility of unexpected words is not as good as that of natural speech. The Klattalk will soon undergo the same testing protocol. It is hoped that word intelligibility will be significantly higher than has been reported thusfar.

FUTURE PLANS

The Klattalk text-to-speech system performs well when the input text is properly analyzed into the right pattern of phonemes, stress assignments, and syntactic phrases. However, errors in text-to-phoneme analysis can be quite disturbing to the listener.

Thus we will examine ways to improve the performance of the text-to-phoneme component, either by improving the letter-to-sound rules or by augmenting the exceptions dictionary. However experience shows that it is not easy to make the letter-to-sound rule module significantly better, and it takes a very large number of words added to the exceptions dictionary to make a significant percentage improvement because there are so many moderately frequent words in English.

Both approaches to improved performance would benefit from access to a very large phonemic dictionary of English. We have begun to assemble such a dictionary and to check the pronunciations through careful listening to the synthetic output. Our aim is to have available a dictionary of at least 20,000 common words combined with common affixes.

Another possible approach is to develop a "morpheme" dictionary [1]. In this type of system, words are broken down into their constituent meaningful elements, or morphs. For example, "hothouses" consists of "hot+house+s" and "scarcity" consists of "scarce+ity". Allen *et al*. [1] have found that a 15,000 morpheme dictionary can be used to assign a pronunciation to over 150,000 English words by finding a morph covering for each word. Words encountered that do not decompose into morphs of this dictionary are sent to a letter-to-phoneme rule system, but this only happens to about 2 words in a 100 in a typical text.

Morphemic decomposition is essential for words like "hothouse" because any reasonable letter-to-phoneme rule system will always pronounce the letter sequence "th" in the way that is correct for words like "thousand". However, a morpheme based system has to be carefully tuned and appended with exceptions to prevent words like "scarcity" from being pronounced as "scar+city" (as it was in early versions of Allen's system).

Commercial Implementation

The Klattalk text-to-speech system is currently a laboratory system that runs in real time on a PDP-11/60 and a Lincoln Labs LDSP fast digital processor. A microcomputer-based version of the system should be implemented soon through a license agreement with Digital Equipment Corporation. Commercial products based on this design will become available shortly thereafter.

REFERENCES

1. Allen, J., Hunnicutt, S., Carlson, R. and Granstrom, B. (1979), "MI talK-79: The 1979 MIT Text-to-Speech System", *ASA-50 Speech Communication Papers*, J.J. Wolf and D.H. Klatt (Eds.), The Acoustical Society of America, NY, NY, 507-510.
2. Bernstein, J. and Pisoni, D.B. (1980), "Unlimited Text-to-Speech System: Description and Evaluation of a Microprocessor Based Device", *Proceedings ICASSP-80*, IEEE Catalog No. 80CH1559-4, 576-579.
3. Hunnicutt, S. (1980), "Grapheme-to-Phoneme Rules: A Review", *Speech Transmission Laboratory QPSR 2-3/1980*, 38-60, Royal Institute of Technology, Stockholm, Sweden.
4. Klatt, D.H. (1979), "Synthesis by Rule of Segmental Durations in English Sentences" in *Frontiers of Speech Communication Research*, B. Lindblom and S. Ohman (Eds.), 287-300, Academic Press.
5. Klatt, D.H. (1980), "Software for a Cascade/Parallel Formant Synthesizer", *J. Acoust. Soc. Am. 67*, 971-995.
6. Maeda, S. (1974), "A Characterization of Fundamental Frequency Contours of Speech", *M.I.T. Quarterly Progress Report No. 114*, 193-211.
7. Venezky, R.L. (1970), *The Structure of English Orthography*, Mouton, The Hague.