# A Model and a System for Machine Recognition of Speech

D. RAJ REDDY, LEE D. ERMAN, and RICHARD B. NEELY

*Abstract*—This paper presents a model for machine recognition of connected speech and the details of a specific implementation of the model, the HEARSAY system. The model consists of a small set of cooperating independent parallel processes that are capable of helping in the decoding of a spoken utterance either individually or collectively. The processes use the "hypothesize-and-test" paradigm. The structure of HEARSAY is illustrated by considering its operation in a particular task situation: voice-chess. The task is to recognize a spoken move in a given board position. Procedures for determination of parameters, segmentation, and phonetic descriptions are outlined. The use of semantic, syntactic, lexical, and phonological sources of knowledge in the generation and verification of hypotheses is described. Preliminary results of recognition of some utterances are given.

## Introduction

Most papers on speech recognition conclude by saying that it is necessary to use higher level linguistic cues to obtain acceptable recognition. The terms context, syntax, semantics, and phonological rules are used but attempts to utilize these sources of knowledge have not been successful because of the ill structuredness of these concepts. This paper represents a summary of several years of investigation to formulate an information processing model that would lead to efficient recognition of speech and in which the role of various sources of knowledge would be well defined.

At the 1969 spring meeting of the Acoustical Society, we presented several papers on the structure of a speech recognition system that was used to recognize a list of 500 isolated words and a syntax-directed connected speech-recognition system using a finite state grammar and a 16-word vocabulary (Vicens [37], Reddy [31], Neely [22]). Six amplitude and zero-crossing parameters of the incoming utterance were sampled every 10 ms and segmented. The segments were labeled to specify the phonetic class; the syntax was used for sentence analysis and word boundary determination, and prelearned acoustic and phonetic segmental descriptions of lexical items were used for word recognition.

Several inherent limitations were apparent even as we developed the system. First, the vocabulary had to be reduced to 16 words because of word boundary ambiguity problems. For example, the word "large" had to be changed to "big" because of assimilation of the reduced vowel of "the" into the semivowel /l/ of "large" in the utterance: "Pick up the large block."

Second, we had to overcome the limitations of the syntax-directed methods. One could not blindly parse from left to right; rather, we had to locate anchor points from which parsing could proceed both backwards and forwards. This was necessary to compensate for machine errors in earlier stages and to compensate for the idiosyncrasies in speaker performance such as introduction of spurious words, repetition of words, and inclusion of hmm- and ha-like sounds.

Third, the simple hierarchical structure in which output from one process forms the input to the next was not adequate for the task. Errors introduced in each process tend to have multiplicative effect, i.e., if each of four processes introduced 10 percent errors, the cumulative error would be 34 percent. Further, the lack of feedback and feedforward of the simple hierarchical model meant any errors that got through were uncorrectable. The main virtue of the system was that it was the first demonstrable system to use syntactic and lexical constraints to recognize connected speech sentences (such as: "Pick up the big block at the bottom right corner").

For the past four years the authors have been attempting to develop a model and a system for connected speech recognition that did not suffer from the limitations mentioned previously, and that would serve as a research tool for speech-recognition research over a wide range of tasks. The following sections present the resulting model and an outline of the system implemented on a PDP-10 computer.

## The Model

We were interested in developing a system capable of recognition of connected speech from several speakers with graceful error recovery, in close to real time, and easily generalizable to operate in several different task domains. We started with several requirements for the model.

1) Contributions of syntax, semantics, context, and other sources of knowledge towards recognition should be clearly evaluatable. Exactly what and how much does each contribute towards improving the performance of the system?

2) The absence of one or more sources of knowl-

Manuscript received April 30, 1972. This research was supported in part by the Advanced Research Projects Agency of the Department of Defense under Contract F44620-70-C-0107 and monitored by the Air Force Office of Scientific Research.

D. R. Reddy and L. D. Erman are with the Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pa. 15213.

R. B. Neely was with the Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pa. 15213. He is now with the Xerox Palo Alto Research Center, Palo Alto, Calif. 94505.

edge should not have a crippling effect on the performance of the model. That semantic context should not be essential for perception is illustrated by overheard conversations among strangers. That syntactic or phonological context should not be essential is illustrated by conversations among children. That lexical representation is not essential is illustrated by our recognition of new words and nonsense syllables.

3) When more than one source of knowledge is available, interactions between them should lead to a greater improvement in performance than is possible to attain by the use of any subset of sources of knowledge.

4) Since the decoding process is errorful at every stage, the model must permit graceful error recovery.

5) Increases in performance requirements, such as the real time requirement, increase in vocabulary, modifications to the syntax, or changes in semantic interpretation, should not require major reformulation of the model.

The model we have arrived at to satisfy these requirements consists of a small set of cooperating independent processes capable of helping in the decoding process either individually or collectively and using the "hypothesize-and-test" paradigm.

Each of the processes in our model is based on a particular source of knowledge, e.g., syntactic, semantic, or acoustic-phonetic rules. Each process uses its own source of knowledge in conjunction with the present context (i.e., the presently recognized subparts of the utterance) in generating hypotheses about the unrecognized portions of the utterance. This mechanism provides a way for using (much talked about but rarely used) context, syntax, and semantics in the recognition process.

The notion of a set of independent parallel processes, each of which is capable of generation and verification of hypotheses, is needed to satisfy the requirements 1) and 2) mentioned previously. In our model, the absence of a source of knowledge implies deactivating that process, and recognition proceeds (albeit more slowly and with lower accuracy) using the hypotheses generated by the remaining processes. The independence of the processes permits us to deactivate a source of knowledge and measure how and by how much that source of knowledge improves the system.

The need for parallel processes can be derived from the real-time performance requirement. If the system is to ever approach human performance, it must be able to answer trivial questions as soon as they are uttered (some times even before they are completed). This implies that various processes of the system should be able to operate on the incoming data as soon as they are able to do so without waiting for the completion of the whole utterance (as in a simple hierarchic model). The "coroutine" model, in which each process passes control to the next level when a "chunk" is perceived and regains control when a new chunk is needed, would be satisfactory. But this organization can lead to irrevocable loss of data if a higher level process does not return control in time to process new chunks of incoming speech. Thus, there must be at least two parallel processes, one of which is continuously monitoring the input speech and the other proceeding with recognition. This, in addition to requirements 1) and 2), suggests a model with parallel processes.

An important aspect of the model is the nature of cooperation between processes. The implication is that, while each of the processes is independently capable of decoding the incoming utterance, they are also able to cooperate with each other to help recognize the utterance faster and with greater accuracy. Process "*A*" can guide and/or reduce the hypothesis generation phase of process "*B*" by temporarily restricting the parts of the lexicon that can be accessed by *B*, or by restricting the syntax available to process *B*, and so on. This assumes that process *A* has additional information that it can effectively use to provide such a restriction. For example, in a given syntactic or semantic situation only a small subset of all the words of a language may appear.

The need for a hypothesize-and-test paradigm arises from 4). The "errorful" nature of speech processing at every stage implies that every source of knowledge has to be brought to bear to resolve ambiguities and errors at every stage of processing. This implies rich connectivity among various processes and involves both feedforward and feedback. The hypothesize-and-test paradigm represents an elegant way of obtaining this cooperation in a uniform manner.

The notion of hypothesize-and-test is not new. It has been used in several artificial intelligence programs (Newell [25]). It is equivalent to analysis-by-synthesis (Halle and Stevens [10]) if the "test" consists of matching the incoming utterance with a synthesized version of the hypothesis generated. In most cases, however, the test is of a much simpler form; for example, it is not necessary to generate the whole formant trajectory when a simpler test of the slope can provide the desired verification. This not only has the effect of reducing the computational effort but also increases the differentiability between phonemically ambiguous words.

Extendability and generalizability of the model is mainly an issue of implementation. It requires that representation of sources of knowledge be separate from and independent of mechanisms that operate on them. One way of achieving this is to represent the knowledge in a form most suitable for modification by the user and have a set of preprocessors that then transform the knowledge into the representation required by the system.

### HEARSAY System

HEARSAY is a speech-recognition system that incorporates many of the ideas presented in the previous section and is presently under development at Carnegie-Mellon University. It is not restricted to any particular recognition task. Given the syntax and the vocabulary of a language and the semantics of a task, HEARSAY will attempt recognition of utterances in that language.

Fig. 1 gives an overview of the HEARSAY system. The EAR module accepts speech input, extracts parameters, and performs some preliminary segmentation, feature extraction, and labeling, generating a "partial symbolic utterance description." The recognition overlord (ROVER) controls the recognition process and coordinates the hypothesis generation and verification phases of various cooperating parallel processes. The TASK provides the interface between the task being performed and the speech recognition and generation (SPEAK-EASY) parts of the system. The system overload (SOL) provides the overall control for the system. A more detailed, but earlier, description of the goals and various components of this system are given in Reddy *et al.* [33] and Reddy [32].

Here we will describe the operation of the HEARSAY system by considering a specific task: voice-chess. The task is to recognize a spoken move in a given board position. In any given situation there are generally 20–30 legal moves and several thousand different ways of expressing these moves. The syntax, semantics, and vocabulary of the task are restricted, but the system is designed to be easily generalizable to larger tasks, which was not the case for our earlier systems. Larger syntax (e.g., a subset of English) and vocabularies (1000–5000 words) for a more complex semantic task will make HEARSAY slower and less accurate but are not likely to be crippling.

Fig. 2 shows the recognition process in greater detail. At present, it contains three independent processes: acoustic, syntactic, and semantic. We will give a short description of how these processes cooperate in recognizing "king bishop pawn moves to bishop four." Let us assume that this is a legal move (otherwise, at some stage of processing, the system will reject it as semantically inconsistent).

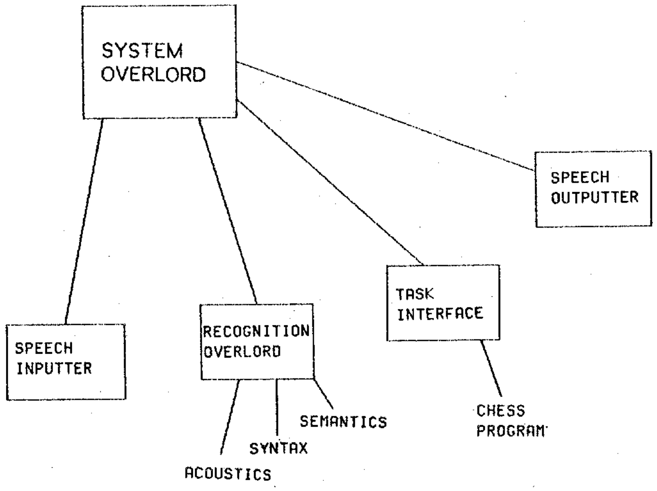

Fig. 1. Overview of the HEARSAY system.

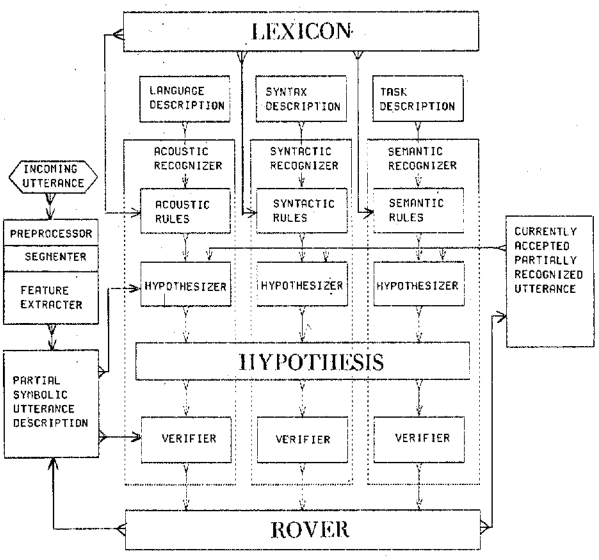

Fig. 2. Detail of the recognition process.

### Parametric Level Analysis

The speech from the input device (microphone, telephone, or tape recorder) is passed through five octave bandpass filters (spanning the range 200–6400 Hz) and an unfiltered band. Within each band the maximum intensity and the number of zero crossings are measured for every 10-ms interval.

This results in a vector of 12 parameters every 10 ms. These parameters are smoothed and log transformed and a subset of the parameters is chosen for further processing. Fig. 3 gives the parameters used, at present, for part of the utterance "king bishop pawn . . . ." Each column represents a 10-ms time unit. Rows $P1$, $P2$, $P3$, and $AU$ represent the log-amplitude parameters in the frequency bands 200–400, 400–800, 800–1600 Hz, and the unfiltered band, respectively. The amplitudes are quantized to 32 levels and represented as a single character (blank, 0–9, $A$-$U$, and *, which represents a value greater than 31). Rows $P4$ and $P5$ represent values that are functions of both amplitude and zero crossing in bands 1600–3200 and 3200–6400. Details of various operations on these parameters are given in Erman [6].

This vector of parameters ($P1$–$P5$ and $AU$) are compared with a standard set of parameter vectors to obtain a minimum distance classification for each time unit using a highly modified version of a procedure proposed by Astrahan [1]. The row labeled $PP$ gives the classification for each 10-ms unit. The standard

```
       k       I          ŋ      b     I       ʃ       ə        p           ɔ           n
        .    :    .    :    .    :    .    :    .    :    .    :    .    :    .    :    .    :
P1       BMLJJKKKJJJJJJJKKKKJCBEPSSTTTRPMDDA73200 LNOMG96311       8GGHIJJLLLMNNNNMMLKKKKKKF62
P2       6FHHFEEEDDDDDDDCB97622  DLLLLLJIEC72220   2DJHD51     00 BKOSRRRSRRQQRRRQQPPNHHFDC9731
P3             122232223200      13331              1650          AFJKLMNOQQQPQRRRRQMHA6631
P4    8HFHNT******RKLLMMNMKIHC     J******PMLNPPOMMJ7DPOL20          9CKLMMQSUUUUUTPPOONMMLD22
P5    9IJJC 2 2   0                       269FGEGGFE6                                21
AU    20025LNONPPPMLLMMLLLKJIHGF61BLQSSSSQLIEDEDCB9 2BLNID600       CNQQRRRSSSSTTUTSSQOMLLLKJF84
PP   -FCC$$eIIIeeeINNNNNINNNNYMMDdVNIeeeeeIIIII$$$$SF+INIUD+++++--++-daaaaaa@@@@@@@@@@AA@@&NN&UWdd+
SP   -CCC$$IIIIIIIIINNNNNNNNNNNNMMdddIIeeeeeIIIII$$$$+++NNN+++++++--++--aaaaaa@@@@@@@@@@@@@@NNNNWWdd+
VF   .fffffvvvvvvvvvvvvvvvvvvvvvvvvvv-vvvvvvvvvvvvvvffff...vvv..............vvvvvvvvvvvvvvvvvvvvvvvvvvvvv.
```

Fig. 3. Parameters and segmentation for "king bishop pawn · · · ." *P*1-*P*5 and *AU* (amplitude) are the input parameters. *PP* is the phone-like name given to the segment. *SP* is the locally smoothed *PP*. *VF* is a segmentation based on the *SP*'s: · unvoiced, nonfricated; *f* unvoiced, fricated; *v* voiced, nonfricated; and *z* voiced, fricated.

set of parameters is obtained by selecting cluster centers from a training set of utterances containing various phonemes in neutral contexts. When a phoneme is represented by several articulatory gestures, more than one cluster center may be added to the standard set. Speaker characteristics and the noise characteristics of the environment or the transducer may be reflected in the standard set of clusters by recording the training set in that environment. Fig. 4 gives cluster centers for several representative sounds. A complete list of clusters used and the details of the speaker normalization program are given in Erman [6].

*Remark 1:* The labels in row *PP* of Fig. 3 are not to be confused with phonetic transcription. Accurate phonetic transcription, where possible, would require modifying the labels taking into account segment and sentence level context.

*Remark 2:* If one wanted to use formant frequencies and amplitudes (assuming they can be determined without mislabeling) one would reanalyze the training set for this parametric representation to determine the new cluster centers. Representing the parameters as a vector with a weighted distance metric defined on the vector space is all that is needed to use a new parametric representation in the HEARSAY system. There are several disadvantages to this approach, e.g., errors in labels, inability to take advantage of special features of a parametric representation, etc. However, this approach provides a convenient way of obtaining the best first approximation to the phonetic representation.

*Remark 3:* The tendency is to blame every error on inadequate parametric representations. We have gone from one set of amplitude and zero crossing parameters to three sets and now to five. Others divide the frequency range into 12, 17, 24, 32, and 48 regions or the full resolution given by FFT. The increase in noisiness of the parameters with increasing resolution makes it imperative that one transform the high resolution data to a smaller number of robust parameters such as the efforts by Li *et al.* [16] and Pols [28] in dimensionality reduction of spectra.

*Remark 4:* The parameters we use represent a crude spectrum. A mixed strategy in which finer analysis is performed only when necessary (Reddy [30]) seems more appropriate for an efficient realization of the system than obtaining every possible parameter at the start.

| PP | P1 | P2 | P3 | P4 | P5 | AU |
|---|---|---|---|---|---|---|
| d | 22 | 14 | 5 | 8 | 0 | 18 |
| s | 0 | 0 | 0 | 47 | 39 | 9 |
| m | 30 | 10 | 2 | 2 | 0 | 33 |
| u | 43 | 30 | 11 | 7 | 0 | 39 |
| a | 37 | 62 | 44 | 38 | 0 | 59 |

Fig. 4. Several typical *PP*-cluster centers.

*Remark 5:* Spectral representation appears to be more robust than formant representation because of the likelihood of mislabeling a formant.

*Remark 6:* Parcor parameter representation (Itakura and Saito [14]) has also been used successfully (Nakano *et al.* [21]) and may have efficient machine realizations within the framework of the HEARSAY system.

*Remark 7:* Zero-crossing measurements and formant frequency measurements are more prone to error than energy measurements in a noisy environment. It appears more difficult to devise noise subtraction algorithms for frequency than for amplitude (Neely and Reddy [24]).

### Segmentation

The purpose of segmentation is to divide the continuous parameter sequence into discrete phone-size chunks. This is usually based on an acoustic similarity measure (Reddy and Vicens [34]). Labeling every 10-ms unit by a phone-like cluster name permits the segmentation to be divided in terms of these labels. Fig. 3 shows two levels of segmentation for "king bishop pawn . . . ." The first level is derived by doing a local "smoothing" of the *PP* names assigned to each of the 10-ms segments; this is displayed on the row labeled *SP*. A segment is defined to be a contiguous run of a single *PP*, flanked by *PP*'s not the same as those in the run. This segmentation is approximately at the phoneme level but is, by itself, very unreliable.

A second level of segmentation is derived by associating a voiced/unvoiced decision and a fricated/nonfricated decision with each *PP*. These binary decisions, when applied to the *SP*'s (and modified with a few simple rules for smoothing and breaking of long segments according to significant local amplitude peaks), segment the signal very reliably. The row in Fig. 3 labeled *VF* indicates this segmentation for the sample.

*Remark 1:* It is now commonly agreed among all researchers that some form of segmentation of acoustic signals is necessary for connected speech recognition (see Fant and Lindblom [8], Reddy [29], Denes and von Keller [4], Broad [2], Medress [19], Dixon and Tappert [5], Klatt and Stevens [15], Stalhammar and Karlsson [35], Hemami and Lehiste [11]). No systematic evaluation has been made of these and other methods of segmentation that have been proposed or implemented. Our present view is that almost any of the schemes, given enough careful tuning, will work in a large majority of the cases; the more important question is then not how to segment, but rather how to use the segmentation without being crippled by the inevitable errors.

*Remark 2:* This use of segmentation represents a trend away from segmentation-free recognition schemes (Halle and Stevens [10]). However, segmentation-free recognition still seems to be a useful concept if one is mainly interested in isolated word recognition (Hill [12], White [39]).

### Acoustic Recognizer

The role of the acoustic recognizer is to predict and verify syllables and words based on the features present in the incoming utterance, the present context, and the lexicon. The structure and phonetic description of syllables and words in the lexicon is prespecified. An entry for a word in the lexicon contains the phonemic spelling(s) of the word and annotations that are used to describe expected anomalies that cannot be predicted by rule from the phonemic spelling. A more detailed description of the lexicon and the preprocessing is given in Erman [6].

The acoustic recognizer has three sources of knowledge available for the generation and verification of hypotheses: acoustic, phonological, and vocabulary restrictions. The acoustic knowledge appears in the form of expected parameters (or features) for a phoneme in a neutral context. The phonological knowledge appears in the form of a coarticulation model that modifies the expected features based on context. The between-word coarticulation effects have to be determined wherever applicable through the use of the "currently accepted partially recognized utterance" (Fig. 2), which provides the boundary phonemes. The vocabulary restriction appears in the form of a valid subset of words in the lexicon that contain a given sequence of features.

The acoustic recognizer uses these sources of knowledge in two stages: the hypothesis and the verification. The acoustic hypothesizer does not have any knowledge of the syntax or semantics of the situation, but can use the gross features (such as /ʃ/ of "bishop") in the "partial symbolic utterance description" (Fig. 2) to retrieve those words of the lexicon that are consistent within the features present.

The task of a verifier is to determine whether a given hypothesis is consistent with the context presently available to it. For example, let us assume that alternative hypotheses of the words "king's," "pawn," "bishop," "queen's," and "knight" have been made in the context "king --- pawn · · ·" (where "---" represents the hypothesized words) and that the word actually spoken was "bishop." Detailed verification, by the acoustic verifier, of every phoneme of every option word is not necessary. All that is needed, in this example, are some simple tests that notice that there is a strong fricative indicated near the middle of the area of interest, which causes "pawn" and "knight" to be rejected, and some other simple tests on the vowel portion, e.g., duration, high/low, and front/back, which would indicate that both "queen's" and "king's" are unlikely, whereas "bishop" is highly likely.

A more detailed matching of features and the use of coarticulation rules at the word boundaries may, of course, be needed for other cases. Detailed matching often implies generation of a test. For example, if the verification to be made is among "sit," "spit," and "split," the presence of /s/, /I/, /t/ and the transitions between /I/ and /t/ are irrelevant. What is needed is the test for the presence or absence of a stopgap and for the presence of /l/-like formant structure following the stopgap.

*Remark:* That some form of hypothesization and verification is needed seems to be recognized by many researchers at this point. Halle and Stevens [10] proposed synthesis and match as a means of verification in their analysis-by-synthesis model. Hypothesis and verification for isolated word recognition was used in the Vicens–Reddy system (Vicens [38]). More recently, similar techniques have also been used by Klatt and Stevens [15], Lindblom and Svensson [18], Tappert *et al.* [36], and Itahashi *et al.* [13].

### Syntactic Recognizer

The role of the syntactic recognizer is to predict phrases based on the syntactic structure of the language to be recognized and the context. The predicted phrases induce (specify) words that might appear in that context. The grammar for the voice-chess language is context free. The voice-chess grammar, specified as a set of BNF productions, is given in Fig. 5. For example, in this grammar, "<move>" is defined to be either "<move1>" followed by "<checkword>" or "<move1>." The total number of different utterances permitted by this grammar is about five million.

The role of the syntax hypothesizer is to use the syntactic source of knowledge to predict words. In

```
1.  <move>          ::= <move1> <check-word> | <move1>

2.  <move1>         ::= <regular-move> | <capture> | <castle>

3.  <castle>        ::= <castle-word> ON <uniroyal> SIDE
                        | <castle-word> <uniroyal> SIDE
                        | <castle-word>

4.  <regular-move> ::= <man-loc> <move-word> <square>

5.  <capture>       ::= <man-loc> <capture-word> PAWN EN-PASSENT
                        | <man-loc> <capture-word> <man-loc>

6.  <castle-word>   ::= CASTLE | CASTLES

7.  <move-word>     ::= TO | MOVES-TO | GOES-TO

8.  <capture-word> ::= TAKES | CAPTURES

9.  <check-word>    ::= CHECK MATE | CHECK

10. <man-loc>       ::= <man-spec> ON <square> | <man-spec>

11. <man-spec>      ::= <uniroyal> <unipiece> PAWN
                        | <uniroyal> <piece> | <uniroyal> pawn
                        | <unipiece> pawn | <man>

12. <square>        ::= <uniroyal> <piece> <rank> | <nopawn> <rank>

13. <man>           ::= KING | QUEEN | BISHOP | KNIGHT | ROOK | PAWN

14. <uniroyal>      ::= KING | QUEEN | KING'S | QUEEN'S

15. <unipiece>      ::= BISHOP | KNIGHT | ROOK
                        | BISHOP'S | KNIGHT'S | ROOK'S

16. <nopawn>        ::= KING | QUEEN | BISHOP | KNIGHT | ROOK

17. <piece>         ::= BISHOP | KNIGHT | ROOK

18. <rank>          ::= ONE | TWO | THREE | FOUR
                        | FIVE | SIX | SEVEN | EIGHT
```

Fig. 5. Voice-chess syntax.

hypothesization the syntax recognizer uses only very local context to predict words. Predictions may be made either to the right or the left of already existing words. For example, if "--- moves-to ---" is given, then words may be hypothesized to the left of "moves-to" or to the right of "moves-to." Hypothesization uses only inexpensive methods, and often generates words that would not fit in the complete context of the sentence.

Traditional parsing schemes are not very useful in generating hypotheses. Further, the syntax recognizer must be capable of processing errorful strings containing spurious words and repetition of words. This implies that it must be capable of working both forwards and backwards. This is achieved in HEARSAY by the use of antiproductions.

Antiproductions act as a concordance for the grammar giving all the contexts for every symbol appearing in the grammar. They are used to predict words that are likely to occur following or preceding a word using only limited context. Fig. 6 gives antiproductions for productions 1-6 of the grammar of Fig. 5. These are produced automatically by a preprocessing program. In this figure, the symbols in the column labeled CENTER are the entries in the concordance. Each symbol in the subset of the grammar appears in this column once for each occurrence of it in the subset. The entries in the LEFT and RIGHT columns denote symbols that can appear to the left and right of the entry in the center column. When an ↑ appears in the LEFT or RIGHT column, it indicates that the original production did not have an entry to the left or right of that symbol.

| CENTER | LEFT | RIGHT | HEAD |
|---|---|---|---|
| CASTLE | ↑ | ↑ | <castle-word> |
| CASTLES | ↑ | ↑ | <castle-word> |
| EN-PASSENT | PAWN | ↑ | <capture> |
| ON | <castle-word> | <uniroyal> | <castle> |
| PAWN | <capture-word> | EN-PASSENT | <capture> |
| SIDE | <uniroyal> | ↑ | <castle> |
| SIDE | <uniroyal> | ↑ | <castle> |
| <move1> | ↑ | <check-word> | <move> |
| <move1> | ↑ | ↑ | <move> |
| <check-word> | <move1> | ↑ | <move> |
| <regular-move> | ↑ | ↑ | <move1> |
| <capture> | ↑ | ↑ | <move1> |
| <castle> | ↑ | ↑ | <move1> |
| <castle-word> | ↑ | ON | <castle> |
| <castle-word> | ↑ | <uniroyal> | <castle> |
| <castle-word> | ↑ | ↑ | <castle> |
| <uniroyal> | ON | SIDE | <castle> |
| <uniroyal> | <castle-word> | SIDE | <castle> |
| <man-loc> | ↑ | <move-word> | <regular-move> |
| <man-loc> | ↑ | <capture-word> | <capture> |
| <man-loc> | ↑ | <capture-word> | <capture> |
| <man-loc> | <capture-word> | ↑ | <capture> |
| <move-word> | <man-loc> | <square> | <regular-move> |
| <square> | <move-word> | ↑ | <regular-move> |
| <capture-word> | <man-loc> | PAWN | <capture> |
| <capture-word> | <man-loc> | <man-loc> | <capture> |

Fig. 6. Antiproductions for a subset of the syntax of Fig. 5. (The subset consists of productions 1-6.)

When the LEFT (or RIGHT) context given in an antiproduction is satisfied, then the RIGHT (or LEFT) context is hypothesized for recognition. If the hypothesized symbol happens to be a nonterminal, then all the possible terminal symbols that can appear at the left of this nonterminal are hypothesized. Detailed descriptions of the structure and use of antiproductions will be given in Neely [23].

The role of the syntactic verifier is to accept or discard hypotheses using syntactic consistency checks. This is usually a more expensive process than hypothesization because it involves complete parsing of the partially recognized sentences. The verifier may work both on hypotheses that the syntactic hypothesizer has generated, as well as those generated by other hypothesizers.

### Semantic Recognizer

The role of the semantic recognizer is to predict concepts based on the semantics of the task and semantics of the preceding utterance. A predicted concept (a legal move for voice-chess) is used in conjunction with the present context to predict a word that might appear in the utterance. The semantics of the task and the preceding utterances are captured for chess by the current board position. The board position for the utterance in discussion, "king bishop pawn moves to bishop four," is shown in Fig. 7.

HEARSAY has, as a subpart, a chess program (Gillogly [9]) that generates an ordered list of moves that are possible in that situation. A partial list of legal moves with numbers representing the likelihood of occurrence is given in Fig. 8.

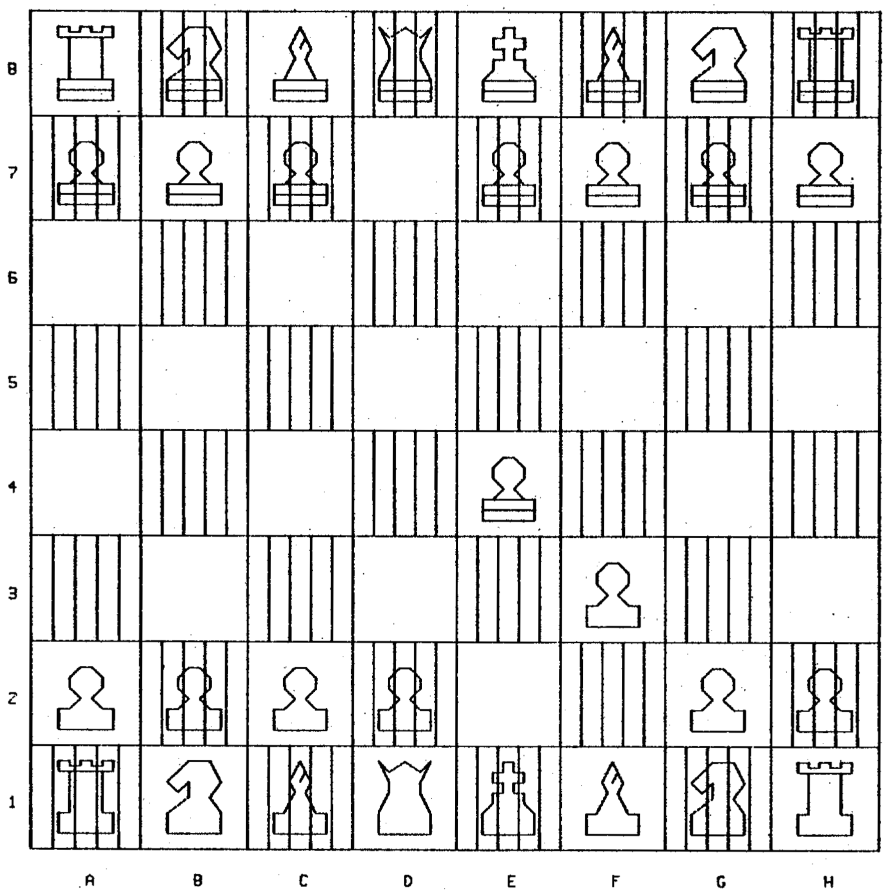

Fig. 7. Board position for utterance in discussion.

```
KBP/KB3XKP/K4    100
QP/Q2-Q4          50
QN/QN1-QB3        49
KB/KB1-QB4        48
KN/KN1-K2         47
QP/Q2-Q3          46
KB/KB1-K2         45
Q/Q1-K2           44
QBP/QB2-QB4       43
QBP/QB2-QB3       42
K/K1-KB2          41
K/K1-K2           40
KRP/KR2-KR4       39
KNP/KN2-KN4       38
QNP/QN2-QN4       37
QRP/QR2-QR4       36
KN/KN1-KR3        35
KNP/KN2-KN3       34
QNP/QN2-QN3       33
KRP/KR2-KR3       32
QRP/QR2-QR3       31
QN/QN1-QR3        30
KB/KB1-QN5CH      25
KBP/KB3-KB4       24
KB/KB1-QR6        12
KB/KB1-Q3          6
```

Fig. 8. Ordered list of legal moves supplied by the chess-playing program for the board position of Fig. 7.

The semantic hypothesizer uses the ordered list of moves for hypothesis generation. In our example the hypothesizer would concentrate only on the "noncapture" moves that start with the word "king." If there are none, then there is an inconsistency in the currently accepted partially recognized utterance. This may be due to an illegal statement or incorrect recognition. In the latter case, the partially recognized utterance is modified by replacing the weakest link by the second best choice for that position.

```
    bishop      62
    knight      62
    bishop's    44
    rook        41
    on          41
    knight's    38

ds hypothesized by semantic hy
```

Fig. 9. Words hypothesized by semantic hypothesizer.

Fig. 9 gives the words hypothesized by the semantic hypothesizer in the context of "king ---." Associated with each hypothesis is a rating (ranging from 1 to 100) indicating the semantic likelihood of the hypothesis. This likelihood is derived from the likelihoods of the projected legal moves from which the hypotheses are taken, and from intrasentence semantic clues. The semantic hypothesizer uses word- and phrase-level semantic consistency checks to restrict hypothesization. The structure and the mechanism used by the semantic hypothesizer are described in Neely [23].

### Control of the Processes

Since the different recognizers are independent, the recognition overlord needs to synchronize the hypothesis generation and verification phases of various processes. Synchronization ensures that hypotheses generated by one process will be verified by all the other processes in the subsequent time slice. Several strategies are available for deciding which subset of the processes generates the hypotheses and which verify. At present this is done by polling the processes to decide which process is most confident about generating the correct hypothesis. In voice-chess, where the semantic source of knowledge is dominant, that module usually generates the hypotheses. These are then verified by the syntactic and acoustic recognizers. However, when robust acoustic cues are present in the incoming utterance, the roles are reversed with the acoustic recognizer generating the hypotheses.

```
S: Actually spoken
R: Recognized by HEARSAY

1. S: PAWN TO KING  FOUR
   R: PAWN TO QUEEN FOUR

2. S: KNIGHT TO KING'S  BISHOP THREE
   R: PAWN   TO QUEEN'S BISHOP THREE

3. S: BISHOP TO KNIGHT FIVE
   R: PAWN TO QUEEN THREE

4. S: KNIGHT TO QUEEN BISHOP THREE
   R: KNIGHT TO QUEEN BISHOP THREE

5. S: PAWN TO QUEEN FOUR
   R: PAWN TO QUEEN FOUR

6. S: KNIGHT TAKES PAWN
   R: KNIGHT TAKES PAWN
```

Fig. 10. Some preliminary results from one run. (Approximately 4–7 times real-time processing on a PDP-10 computer.)

The verification process continues until a hypothesis is found that is acceptable to all the verifiers with a high enough level of confidence. All the unverified hypotheses are stored on a stack for the purpose of backtracking at a later stage. Given an acceptable hypothesis, ROVER updates the currently accepted partially recognized utterance and updates the partial symbolic utterance description with additional features that were discovered during the process of hypothesis generation and verification. If the utterance still has unrecognized portions of speech and if the interpretation of the utterance is still unclear, then all the active processes are reactivated to generate hypotheses in the new context. If there are no unrecognized portions of speech in the utterance and the sentence is uninterpretable, the knowledge acquisition part of the system (unimplemented in the present system and not shown in Fig. 2) is activated to update the lexicon and the acoustic, syntactic, and/or semantic rules.

### Preliminary Results

The system described in the preceeding sections has been operational since June 1972. We view HEARSAY as a continually evolving system that is expected to serve as a research tool for explorations in speech-recognition research at Carnegie-Mellon University. Fig. 10 gives some preliminary results of recognition by the system. More comprehensive results containing time, accuracy, and error analyses will be given in Erman [6] and Neely [23].

## Discussion

### Models of Speech Perception

This paper presents a model of speech perception that has been arrived at not so much by conducting experiments on how humans perceive speech but in the process of constructing several speech-recognition systems using computers. The emphasis has been on developing efficient recognition algorithms, with little attention to modeling of known human perceptual behavior. The general framework (for a model) that evolved is different from some previously proposed models by Liberman *et al.*, [17] and Halle and Stevens [10], which imply that perception takes place through the active mediation of motor centers associated with speech production. Our results tend to support "sensory" theories advanced by Fant [7], and others, in which speech decoding proceeds without the active mediation of speech motor centers.

If one eliminates the synthesis part of analysis-by-synthesis, then our model is most similar to that of Halle and Stevens [10]. The important distinction to remember is that once a hypothesis is generated, say of the words "sit," "slit," and "split," one should never want to verify the hypotheses by generating formant trajectories for the word or phrase. That phonemes /s/, /I/, /t/ occur in the hypothesized words is no longer relevant. All that is needed is a verification of the presence of stopgap and the /l/-like formant transition preceding the vowel. Another limitation of synthesis and match is that the noise might swamp the finer distinction required, i.e., the variability in speaker performance of /s/, /I/, /t/ might overshadow the positive contributions of a /p/ or an /l/.

### Information-Processing Models

The model proposed in this paper raises several issues that may be of interest to speech scientists and cognitive psychologists interested in human speech perception. We would like to propose that, in addition to stimulus-response studies and neuro-physiological models, speech scientists should also make extensive use of information-processing models in the study of speech perception. The notion of an information-processing model reflects a current trend in cognitive psychology to view man as an information processor, i.e., that his behavior can be seen as the result of a system consisting of memories containing discrete symbols and symbolic expressions and processes that manipulate these symbols (Newell [26]). The main advantage of this approach to speech perception studies is that it permits a researcher to look at the total problem of speech perception at a higher functional and conceptual level than is possible with the other two approaches. (To attempt to study the total problem of speech perception by formulating a neurophysiological model would be like attempting to understand the workings of a TV set by looking at the flow of electrons through a transistor.)

One question that arises in this context is the nature of serial and parallel processing mechanisms used by humans. It is known that, at a higher problem-solving level, a human being behaves essentially as a serial information processor (Newell and Simon [27]). It is also known that parallel processing occurs at the preprocessing levels of vision and speech. What is not known is whether there are several independent processes or a single sophisticated process at the perceptual level that can use effectively all the available sources of knowledge.

The second question is how various sources of knowledge cooperate with each other. There are experiments (Miller and Isard [20], Collins and Quillian [3]) that can be interpreted to show that perception is faster or more intelligible depending on the number of available sources of knowledge. Any model of speech perception must deal with the nature and structure of the interaction between various sources of knowledge. Earlier models tend to ignore this question.

## Summary and Conclusions

A casual reader of this paper would probably only notice the superficial aspects of the system: that it accepts voice commands to play chess, uses crude parameters, and is not very smart at using the acoustic-phonetic and other sources of knowledge. That is beside the point. The main contribution of this research is to provide a model and a framework in which the role of phonology, syntax, semantics, and other sources of knowledge can be systematically studied and evaluated. It is no longer necessary for us to be content with vacuous statements about the importance of syntax or semantics.

We chose voice-chess as a task not because it is important to play chess with a computer over telephone, but because chess provides a good area to evaluate our ideas about the role of various sources of knowledge in speech perception. Chess plays the role in our system that the fruit fly plays in genetics. Just as the genetics of *drosophila* are studied not to breed better flies, but to learn the laws of heredity, so we choose chess as a task because the syntax, semantics, and vocabulary of discourse are well defined and are amenable to systematic study.

Similarly, the acoustic parameters and phonological, syntactic, and semantic rules currently used by the HEARSAY system are not particularly important or interesting. What is important to note is that while each module is "stupid," the system still works and does do a creditable job in spite of its weaknesses. The interesting features are the interaction and cooperation among various modules and the correction of errors by various sources of knowledge.

The system described in this paper was demonstrated in June 1972, at a workshop on speech recognition. It represents the first system to demonstrate live, connected speech recognition using nontrivial syntax and semantics. We expect to actively modify the system to greatly increase its performance, as well as use it as an experimental tool for studying speech understanding, recognition, and perception.

## Acknowledgment

The authors wish to thank A. Newell whose intellectual and moral support has been essential to sustain a long-term research effort of this kind in the midst of pessimistic predictions. We wish to acknowledge the important contributions of R. Fennell in the design and programming of ROVER and the global data structure. P. Karlton and B. Lowerre also helped in programming parts of the system. B. Broadley, P. Newbury, and J. Teter interfaced the speech devices to the PDP-10. G. Robertson and H. Wactlar made the necessary modifications to the PDP-10 monitor.

## References

[1] M. Astrahan, "Speech analysis by clustering or the hyperphoneme method," Dep. Comput. Sci., Stanford Univ., Stanford, Calif., AI Memo 124, 1970.

[2] D. J. Broad, "Formants in automatic speech recognition," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 295-298.

[3] A. M. Collins and M. R. Quillan, "Retrieval time from semantic memory," *J. Verbal Learn. Behav.*, vol. 8, 1969, pp. 204-267.

[4] P. B. Denes and T. G. von Keller, "Articulatory segmentation for automatic recognition of speech," in *Proc. 6th Int. Congr. Acoust.*, vol. B, 1968, pp. 143-146.

[5] N. R. Dixon and C. C. Tappert, "Derivation of phonetic representation by combining steady-state and transemic classification in automatic recognition of continuous speech," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 319-321.

[6] L. D. Erman, Ph.D. dissertation, in preparation.

[7] G. Fant, "Auditory patterns of speech," in *Models for the Perception of Speech and Visual Form*, W. Wathen-Dunn, Ed. Cambridge, Mass.: M.I.T. Press, 1964.

[8] C. G. M. Fant and B. Lindblom, "Studies of minimal speech sound units," Speech Transmission Lab., Quarterly Prog. Stat. Rep., vol. 2, pp. 1-11, 1961.

[9] J. J. Gillogly, "The TECHNOLOGY chess program," *Artif. Intel.*, vol. 3, pp. 145-163, 1972.

[10] M. Halle and K. Stevens, "Speech recognition: A model and a program for research," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 155-159, Feb. 1962.

[11] H. Hemani and I. Lehiste, "Interactive automatic speech segmentation," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 291-294.

[12] D. R. Hill, "Man-machine interaction using speech," in *Advances in Computers*, vol. 11, F. L. Ait *et al.*, Ed. New York: Academic, 1971, pp. 165-230.

[13] S. Itahashi, S. Makino, and K. Kido, "Automatic recognition of spoken words utilizing dictionary and phonological rule," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 327-330.

[14] F. Itakura and S. Saito, "Speech analysis-synthesis system based on the partial autocorrelation coefficient," presented at the 1969 Acoust. Soc. Jap. Meeting (see also "On the optimum quantization of feature parameters in the parcor speech synthesizer," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 434-437).

[15] D. H. Klatt and K. N. Stevens, "Sentence recognition form visual examination of spectrograms and machine-aided lexical searching," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 315-318.

[16] K.-P. Li, G. W. Hughes, and A. S. House, "Correlation characteristics and dimensionality of speech spectra," *J. Acoust. Soc. Amer.*, vol. 46, pp. 1019-1025, 1969.

[17] A. M. Liberman, F. S. Cooper, K. S. Harris, and P. F. MacNeilage, "A motor theory of speech perception," in *Proc. Speech Commun. Seminar*, vol. 2, 1962.

[18] B. Lindblom and S.-G. Svensson, "Interaction between segmental and non-segmental factors in speech recognition," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 331-333.

[19] M. Medress, "A procedure for the machine recognition of speech," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 113-116.

[20] G. A. Miller and S. Isard, "Some perceptual consequences of linguistic rules," *J. Verbal Learn. Behav.*, vol. 2, pp. 217-228, 1963.

[21] Y. Nakano, A. Ichikawa, and K. Nakata, "Evaluation of various parameters in spoken digits recognition," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 101-104.

[22] R. B. Neely, "Experimental conversational computer system," *J. Acoust. Soc. Amer.*, vol. 46, p. 89(A), 1969.

[23] ——, Ph.D. dissertation, in preparation.

[24] R. B. Neely and R. D. Reddy, "Speech recognition in the presence of noise," in *Proc. 7th Int. Congr. Acoust.* (Budapest, Hungary), vol. 3, 1971, pp. 177-180.

[25] A. Newell, "Heuristic programming: Ill-structured problems," in *Progress in Operations Research*, vol. 3, J. S. Aronofsky, Ed. New York: Wiley, 1971.

[26] ——, "Remarks on the relationship between artificial intelligence and cognitive psychology," in *Non-Numerical Problem Solving*, R. Banerji and M. D. Mesarovic, Ed. Berlin, W. Germany: Springer-Verlag, 1970, pp. 363-400.

[27] A. Newell and H. A. Simon, *Human Problem Solving.* Englewood Cliffs, N.J.: Prentice-Hall, 1972.

[28] L. C. W. Pols, "Dimensional representation of speech spectra," in *Proc. 7th Int. Congr. Acoust.* (Budapest, Hungary), vol. 3, 1971, pp. 281-284.

[29] D. R. Reddy, "Segmentation of speech sounds," *J. Acoust. Soc. Amer.*, vol. 40, pp. 307-312, 1966.

[30] ——, "Computer recognition of connected speech," *J. Acoust. Soc. Amer.*, vol. 42, pp. 329-347, 1966.

[31] ——, "Segment-synchronization problem in speech recognition," *J. Acoust. Soc. Amer.*, vol. 46, p. 89(A), 1969.

[32] ——, "Speech recognition: Prospects for the seventies," in *Proc. IFIP*, vol. 71, 1971, pp. I-5-I-3.

[33] D. R. Reddy, L. D. Erman, and R. B. Neely, "The C-MU speech recognition project," in *Proc. IEEE Syst. Sci. Cybern. Conf.*, 1970.

[34] D. R. Reddy and P. J. Vicens, "A procedure for segmentation of connected speech," *J. Audio Eng. Soc.*, vol. 16, pp. 404-412, 1968.

[35] U. Stalhammar and I. Karlsson, "A phonetic approach to ASR," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 125-128.

[36] C. C. Tappert, N. R. Dixon, and A. S. Rabinowitz, "Application of sequential decoding for converting phonetic to graphemic representation in automatic recognition of continuous speech," in *Proc. Int. Conf. Speech Commun. Processing*, 1972, pp. 322-326.

[37] P. J. Vicens, "Use of syntax in the analysis of connected speech," *J. Acoust. Soc. Amer.*, vol. 46, p. 89(A), 1969.

[38] ——, "Aspects of speech recognition by computer," Dep. Comput. Sci., Stanford Univ., Stanford, Calif., AI Memo 85, Ph.D. dissertation, 1969.

[39] G. White, private communication, Xerox Palo Alto Res. Cen., Palo Alto, Calif., 1972.