Editor's Note: *This is the fifty-first in a series of review and tutorial papers on various aspects of acoustics.*

Review of the ARPA Speech Understanding Project

Dennis H. Klatt

Massachusetts Institute of Technology, Cambridge, Massachusetts 02139
(Received 10 May 1977; revised 1 September 1977)

In September of 1976, four speech understanding systems were demonstrated, signifying the end of a five-year program of research and development sponsored by the Advanced Research Projects Agency (ARPA). The best performance was displayed by the Harpy system developed at Carnegie-Mellon University. Harpy satisfied a set of design goals that were specified at the beginning of the program, including the goal of understanding over 90% of a set of naturally spoken sentences composed from a 1000-word lexicon. After defining the nature of the speech understanding problem, the four systems are described and critically evaluated. Based on this review, a structure for a next-generation speech understanding system is proposed and parts of it are considered as a possible model of the early stages of speech perception. The perceptual model addresses the issue of lexical access and includes a decoding network composed of expected spectral sequences for all word strings of English.

PACS numbers: 43.10.Ln, 43.70.Sc, 43.70.Dn

INTRODUCTION

In November of 1971, the Information Processing Technology Office of the Advanced Research Projects Agency of the Department of Defense (ARPA) initiated a five-year research and development program with the objective of obtaining a breakthrough in speech understanding capability that could then be used toward the development of practical man—machine communication systems (Newell *et al.*, 1973). The specific goals set forth by an ARPA study group[1] are outlined in Table I. The objectives were to develop several speech understanding systems that accept continuous speech from many cooperative speakers of a General American dialect. Recordings were to be made in a quiet room using a good-quality microphone. Slight tuning of the system would be allowed to handle new speakers, but the users could be required to make only natural adaptations to the system. The language definition should include a slightly selected vocabulary of at least 1000 words and an artificial syntax appropriate to the limited task situation (e.g., a data management task). Less than 10% semantic error would be tolerated and the system would have to run in a few times real time using the next generation of computers [(i.e., machines capable of executing 100 million machine instructions per second (MIPS)]. These goals were to be achieved by November 1976.

Significantly (and deliberately) absent from the specifications were requirements that the demonstration tasks be relevant to real-world problems, that the languages be habitable, and that the systems be cost effective. These omissions helped to get the project focused on scientific and computational issues, but they have resulted in questions concerning the work remaining to develop future practical systems.

The study group emphasized the concept of speech understanding as opposed to speech recognition. They believed that the hope for the program lay in analyzing speech within the context of specific tasks that employed strong grammatical constraints, as well as strong semantic and dialogue constraints, so that many sources of knowledge could be brought to bear to attain successful understanding of what was said or intended by the speaker. Accuracy was to be measured by the correctness of the response and not by whether all of the words were correctly recognized.

There were two possible ways to meet the ARPA goals: (1) simplify the general speech recognition problem by finding ways to apply syntactic and semantic constraints and (2) improve upon previous speech recognition capabilities. As noted above, the steering committee emphasized the first alternative and recommended that funding be given to research groups that were composed mainly of computer scientists, not speech scientists. It turned out that the various research groups tried different combinations of the two strategies, but the only clearly successful speech understanding system, Harpy, relied heavily on the first technique. In fact, if the ARPA project were to be judged on its contributions to speech recognition and the speech sciences, rather than judging it against its stated goals, a more negative appraisal might have to be given.

The second column of Table I characterizes the performance of the Harpy speech understanding system, which was developed at Carnegie—Mellon University (Lowerre, 1976; Reddy *et al.*, 1977). Harpy essentially meets or exceeds each of the specifications. Given this set of criteria, Harpy performed the best of all the systems that were demonstrated at the end of the project.

A general overview of what has been accomplished during the past five years has been published by the

TABLE I. The ARPA five-year goals are compared with the performance of Harpy.

GOAL (Nov., 1971)	Harpy (Nov., 1976)
ACCEPT CONNECTED SPEECH	YES
FROM MANY	5 (3 MALE, 2 FEMALE)
COOPERATIVE SPEAKERS	YES
IN A QUIET ROOM	COMPUTER TERMINAL ROOM
USING A GOOD MICROPHONE	CLOSE-TALKING MICROPHONE
WITH SLIGHT TUNING/SPEAKER	20 TRAINING SENTENCES/TALKER
ACCEPTING 1000 WORDS	1011
USING AN ARTIFICIAL SYNTAX	AVG. BRANCHING FACTOR = 33
IN A CONSTRAINING TASK	DOCUMENT RETRIEVAL
YIELDING < 10% SEMANTIC ERROR	5%
IN A FEW TIMES REAL TIME	80 TIMES REAL TIME
ON A 100 MIPS MACHINE	ON A .4 MIPS PDP-KA10 USING 256K OF 36-BIT WORDS AND COSTING $5 PER SENTENCE PROCESSED

ARPA steering committee (Medress *et al.*, 1977). The primary concern of this review is to compare and evaluate the structures and components of four speech understanding systems that were developed.[2] Only brief mention will be made of other activities that were carried out in support of the system development efforts. The remainder of the paper is divided into an initial section that sets forth the scientific problems to be solved, a section describing the four speech understanding systems, and a section concerned with an overall scientific evaluation of the program, a proposal for a second-generation speech understanding system, and a discussion of the implications of this research for models of the speech perception process.

The ARPA project, while large in funding terms, is only one of many past and present efforts to recognize spoken utterances. The reader is referred to other sources for a more complete picture. For example, there are reviews such as have been published by Lindgren (1965), Pierce (1969), Fant (1970), Hyde (1972), Wolf (1976), and especially Reddy (1976); conference proceedings such as have been edited by Erman (1974), Reddy (1975), Fant (1975), Teacher (1976), and Silverman (1977); and descriptions of other recent speech understanding systems such as have been published by Bahl *et al.* (1976), Jelenek (1976), De Mori *et al.* (1975), Sakai and Nakagawa (1975), Haton and Pierrel (1976), and Medress *et al.* (1977).

I. THE PROBLEM

At the beginning of the ARPA project, isolated word recognition by pattern matching techniques was enjoying some initial success. However, it was realized that many words appearing in sentence contexts varied dramatically in acoustic characteristics depending on the surrounding phonetic environment and depending on certain phonological processes of English (Stevens and Klatt, 1973; Oshika *et al.*, 1975), so a simple-minded pattern-matching word identification strategy could not be applied to the sentence understanding problem. Therefore it seemed necessary to follow a more traditional approach, the first step of which was to process the acoustic input to recover a phonetic transcription of what had been said. A phonetic transcription is a discrete representation of articulatory activity in terms of a sequence of configurational goals or states called phonetic segments.

The second step in the hypothetical understanding strategy would be to take the (probably errorful) phonetic transcription of an unknown utterance and try to find candidate words and word sequences that might be present. Consider the phonetic transcription:

[d ɪ ǰ ə h ɪɾɾ ɪ t ə t a m] (1)

No word boundaries are indicated in (1) because acoustic cues to word boundary locations are rarely present. The lexical search problem [to find the sequence of words corresponding to (1)] is extremely difficult because of the combinatorics of possible word boundary locations, because the phonetic transcription may contain substitution errors, omissions, and extra segments, and because the talker uses a system of phonological rules to modify and simplify the pronunciation of individual words in some sentence environments. For example, the normal way to say "Did you" is [d ɪ j ə], i.e., "Dija" but "you" is pronounced differently in "are you." The "t" in "hit" usually is realized as a very brief tongue flap [ɾ] in "hit it," but not in "hit some." The two adjacent "t"s of "it to" reduce to a single [t], resulting in (1) as the normal way to pronounce "Did you hit it to Tom?"

Each of the simplifications in (1) can be described by general phonological rules that presuppose an underlying basic representation for the word (called the phonemic representation). The phonemic string that would be stored in the lexicon for "you" might be /yu/. A phonological rule [d # y]→[ǰ] transforms the /y/ into [ǰ] if the previous word ends in a [d]. The application of inverse phonological rules for sentence decoding is complicated by the fact that there is no unique inverse rule in most cases. A [ǰ] that is observed could be the first or last sound of a word like "judge," or it could be the surface manifestation of /d/—/y/ in a word pair like "did you." Similarly an observed flap [ɾ] may indicate a word containing a /t/, a /d/, or possibly even an /n/. Almost any segment could be simultaneously the manifestation of the last phoneme of one word and the first phoneme of the next word.

All of these phonological phenomena result in lexical ambiguity so that even the best lexical hypothesis routines will propose many words that are not in the original sentence, simply due to fortuitous matches. The third step in the process would therefore be to use syntactic–semantic modules to weed out the false lexical hypotheses and put together a word string that represents what was spoken.

The block diagram shown in Fig. 1 summarizes what we have just said. Speech understanding systems may be thought to consist of two main components, a "bottom end" that converts acoustic data into lexical hypotheses and a "top end" that accepts lexical hypotheses and tries to find the most likely sentence that could have been spoken.

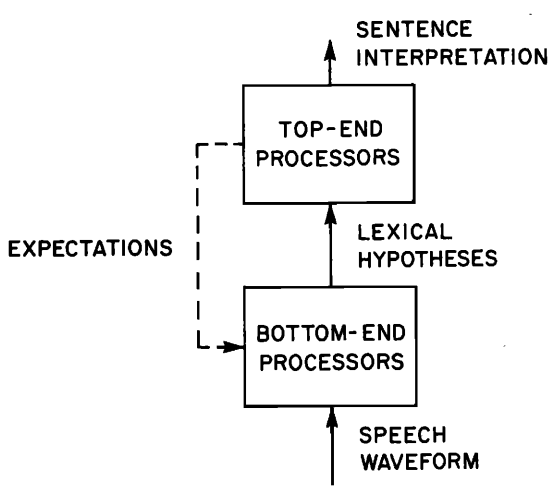


FIG. 1. Simplified overview of the speech understanding problem.

An important point to make concerning Fig. 1 is that the top end can provide the bottom end processor with constraints concerning what might be expected next. The relative success of the four speech understanding systems to be described is more highly correlated with the type of constraint provided by the top end than with any other variable. The most successful system, Harpy, exhaustively lists those and only those acoustic segment sequences that form acceptable input sentences, and the Harpy grammar severely constrains the acoustic alternatives much of the time. The advantage of applying strong constraints at the acoustic level is that one can avoid having to perform generalized phonetic recognition or generalized lexical hypothesization that would otherwise generate a large number of spurious hypotheses that have to be rejected later by the top end (a computationally costly and often difficult undertaking).

When the ARPA program began, it was believed that the scientific problems associated with top end design would be concerned with how to combine lexical hypotheses into larger and larger sentence fragments that are (1) syntactically acceptable, (2) semantically acceptable, (3) and plausible given what the user has said previously and some notion of what he/she wants to do. Syntactic analyzers used earlier in text processing applications would have to be modified to function in the face of errorful input, to consider and score multiple alternatives, and to include semantic knowledge before sufficient constraints could be applied effectively in the speech understanding context. The algorithms would have to be fast enough to permit evaluation of many word combinations and they would have to include sophisticated scoring algorithms to select among those alternatives that are grammatically acceptable. Progress in each of these areas is summarized in Sec. III.

The scientific problems associated with bottom end design when the ARPA program began included (1) selecting an acoustic representation, (2) improving segmentation and phonetic labeling strategies that had been developed previously, and (3) recognizing words that have undergone phonetic modifications at word boundaries and/or phonological recoding. Unanswered questions were: What kinds of improvements could be made to existing phonetic recognition strategies? How good does phonetic recognition have to be? Does one have to normalize for speaking rate? Can routines be made to work for any talker? How can one take advantage of prosodic cues (the pattern of voicing fundamental frequency, segmental durations, and intensity fluctuations), which indicate syllable stress and the syntactic structure of a spoken sentence? Progress in these areas and an interesting change in viewpoint are discussed in Sec. III.

TABLE II. Performance of the speech understanding systems as of November 1976. Statistics are based on more than 100 sentences spoken by several talkers, except for CMU Hearsay-II whose preliminary evaluation employed a smaller data set.

System	Sentences understood	Average branching factor
CMU Harpy	95%	33
CMU Hearsay-2	91, 74	33, 46
BBN Hwim	44	195
SDC	24	105

TABLE III. Task domains of the four systems and an example of an acceptable input sentence.

Group	Task	Sample sentence
SDC	FACTS ABOUT SHIPS	"How fast is the Theodore Roosevelt?"
BBN Hwim	TRAVEL BUDGET MANAGEMENT	"What is the plane fare to Ottawa?"
CMU Harpy	DOCUMENT RETRIEVAL	"How many articles on psychology are there?"
Hearsay-II	DOCUMENT RETRIEVAL	"How many articles on psychology are there?"

The block diagram of Fig. 1 describes a *system*. Some issues of speech understanding system design are obscured if one simply discusses component performance requirements. The system design problems extant at the onset of the ARPA program included (1) how to coordinate the effort to bring up and debug effectively a very large system, (2) how to define communication links between system components, (3) how to schedule activity among components, and (4) how to combine conflicting scores from different knowledge sources.

II. THE SPEECH UNDERSTANDING SYSTEMS

The performance of the final four speech understanding systems, when processing sentences composed from a 1000 word lexicon, is summarized in Table II. Also presented is one measure of the constraint provided by the syntactic and semantic knowledge. The average branching factor is defined here to be the average number of words that would have to be considered at each point along the correct left-to-right path through the syntactic production rules during the processing of a typical utterance. Branching factor has been shown to be a better measure of task difficulty than vocabulary size *per se*, although other aspects of the grammar and inherent confusibility of lexical items contribute to task complexity. Some systems do not process an utterance in a strictly left-to-right manner, but the estimated branching factors are roughly comparable.

Taking account of the range of task difficulties implied in part by the different branching factors, it is unclear whether there are large differences in ability among the top three systems. However, only Carnegie–Mellon University (CMU) was able to meet the ARPA goals. In judging the performance figures given in Table II, it should also be noted that System Development Corporation (SDC) was handicapped by the loss of one of their computers, which prevented them from making use of components being developed jointly with Stanford Research Institute.

The tasks employed by the three system builders are summarized in Table III. Also included is an example of a sentence accepted by each grammar. Each task involves data management of one sort or another. While only questions are given as examples in Table III, each

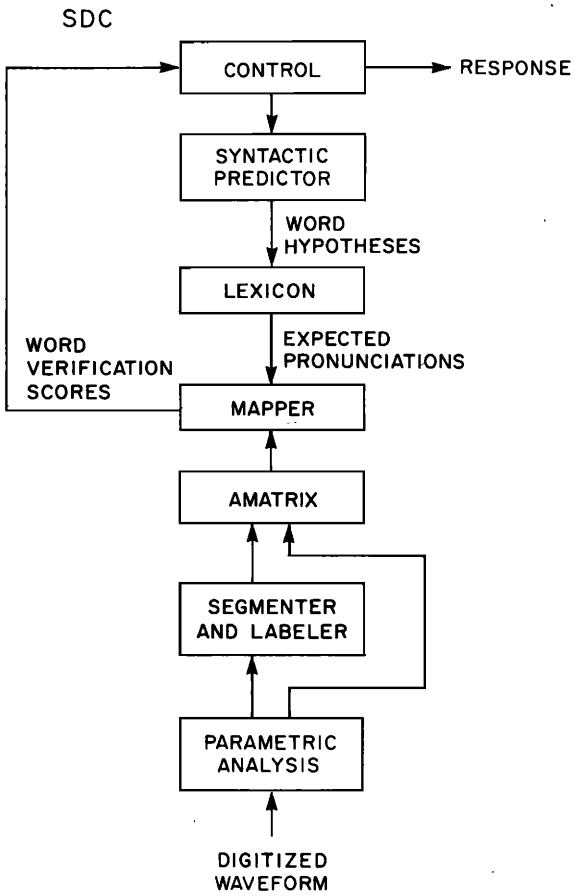


FIG. 2. Block diagram of the SDC system organization.

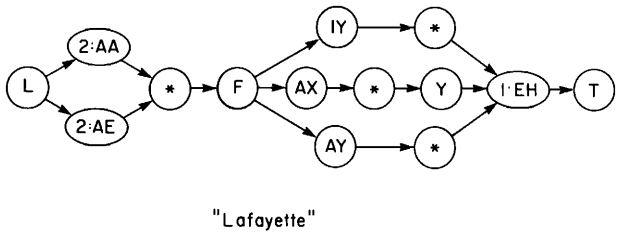


FIG. 3. Lexical representation in the SDC system for the word "Lafayette." Branches in the string indicate acceptable alternative pronunciations. The "*" is a syllable boundary symbol, and the "1" and "2" indicate relative lexical stress levels. Other phonemic symbols have the obvious interpretation. The advantages of a network representation for alternative phonetic pronunciations of words include compactness of form and efficiency of search compared with a simple list of alternatives.

of the systems was also capable of understanding commands and statements of various types.

A. Systems development corporation

The structure of the final SDC speech understanding system is shown in Fig. 2 (Ritea, 1975; Bernstein, 1976). Formant frequencies and other parameters are first extracted from the input waveform. A phonetic transcription is obtained, including several alternative labels for each phonetic segment, and all of this information is placed in a data array called the A-matrix for later examination by top-end routines.

The utterance is processed from left to right by first generating a list of all possible sentence-initial words. The control box then retrieves an abstract phonemic representation from the lexicon for each lexical hypothesis and computes expected phonetic variants, resulting in a phonetic graph representation such as is shown in Fig. 3. The phonetic graphs are sent, one at a time, to the mapper to see how good an acoustic match is obtained with the current position in the unknown utterance. The mapper is organized according to the syllable structure of a word and it examines the A-matrix in order to determine if the expected vowels and proper allophones of adjacent consonants are present. Since an exact match is unlikely, the mapper includes techniques for estimating the probability that the expected word is present given the phonetic and acoustic data. Performance of the mapper is indicated in Table IV.

On the basis of mapper scores, the control box decides which word or partial sentence hypothesis to pursue next, and generates a list of all words that can follow this sentence fragment. A similar "best-first" control strategy was used earlier in the Hearsay I speech understanding system (Reddy, Erman, and Neely, 1973)

TABLE IV. Performance statistics for three work verification components—the SDC mapper, the BBN verifier, and the CMU Hearsay-II verifier. The last row indicates that the SDC verifier is presented with lexical hypotheses from a syntactic module, whereas the BBN and CMU verifiers are preceded by lexical hypothesizers that screen out all but the best acoustic candidates.

	VERIFICATION DECISION					
	SDC		BBN		CMU	
LEXICAL PROPOSAL	ACCEPT	REJECT	ACCEPT	REJECT	ACCEPT	REJECT
CORRECT WORD	65	6	101	19	312	20
PERCENT	92%	8%	84%	16%	94%	6%
INCORRECT WORD	372	11,253	367	713	6462	6591
PERCENT	3%	97%	34%	66%	49%	51%
WORDS HYPOTH. CORRECT WORD	165		10		40	
ACOUST. SIMILARITY	RANDOM		BEST 5%		BEST 14%	

and in a system developed at Lincoln Laboratories (Klovstad and Mondshein, 1975). A more detailed description of the SDC system is presented in Appendix A.

Discussion of SDC

The mapper constitutes a verification strategy based on syllables, which is a theoretically attractive design for embedding context-dependent rules for expected manifestations of phonetic segments. The mapper is capable of rejecting a large fraction of the word hypotheses not in the sentence, but at a cost of rejecting about 10% of the words actually present. Fatal absolute rejections of correct words occurred either because the mapper lost track of which syllable was being processed or because a phonetic confusion occurred that had not been seen during a prior statistics-gathering run.

Unfortunately, the mapper performance is not good enough for a top-end system organization in which there is no mechanism for recovering from a single bad lexical matching score. It is unfortunate that SDC had so little time to design a more powerful top end after being prevented from using an SRI module, because a system can only perform as well as its weakest link.

The main criticism that can be made of the SDC effort is that their system failed its objectives in such a way that it is difficult to say what more restricted goals could be met by a modified system design. Is it simply a matter of shaking the bugs out of the system, or must one place further restrictions on the vocabulary and/or syntax? Or is it that the simple control strategy employed is essentially incapable of performing at an acceptable understanding rate in any moderate sized task because of the nature of speech and the inherent inaccuracies to be expected in any kind of mapper? Answers to these and other questions might have come from a year of system performance evaluation that was planned by the steering committee, but not funded by ARPA.

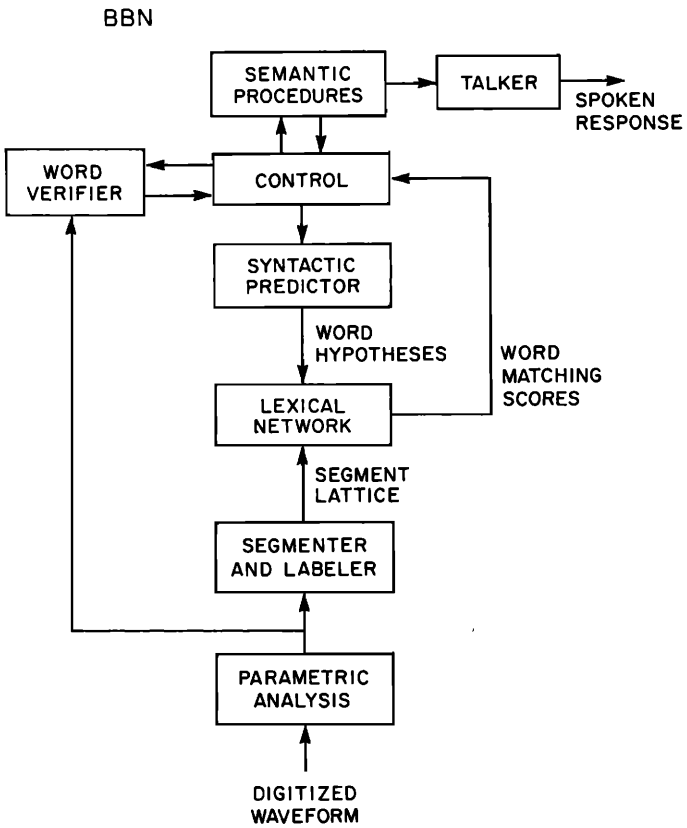


FIG. 4. Block diagram of the BBN Hwim system organization.

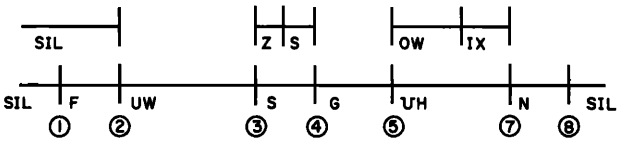


FIG. 5. A BBN segment lattice for the utterance "Who's going?" Alternative segmentations are displaced vertically, while each label represents the top choice among a set of 71 phonetic categories. Similarity scores are computed for all possible labels at each alternative segmentation point. For example, the segment starting at boundary (1) is labeled "F," but the correct phonetic segment "H" was the fourth best label choice.

An interesting aspect of the SDC system emerged in comparing its performance with an earlier version that did better on an easier task. The earlier system understood 65% of a set of test sentences formed from a 200-word lexicon and a more rigid syntax that was devoid of function words. Function words are usually acoustically reduced and difficult to identify. One might speculate that one reason for the poor performance of the more ambitious system was the dependence on function word recognition. Creation of a syntax that perhaps allowed some function words, but in no way depended on their identification to choose a path in the grammar, might be a better strategy for the realization of limited systems. (It is interesting to note that the Harpy grammar is essentially of this form.)

B. Bolt Beranek and Newman Inc. Hwim

The general organization of the BBN Hwim (Hear what I mean) system is shown in Fig. 4 (Woods *et al.*, 1976). As a first step in the processing of an unknown utterance, formant frequencies and other parameters are extracted from the digitized waveform. This information is used to derive a set of phonetic transcription alternatives that are arranged in a "segment lattice," as shown in Fig. 5. The advantage claimed for the lattice structure is that it can represent segmentation ambiguity in those cases where decisions are most difficult.

The identification process begins by searching through the segmental representation of the utterance for good matching words (anywhere in the utterance) that can be used as "seeds" for building up longer partial sentence hypotheses. The best-scoring initial word match is sent to a word verification component which returns to the parametric data to get a quasi-independent measure of the quality of the match. The method of verification is analysis by synthesis (Klatt, 1975).[3] The verification score is combined with the lexical matching score, and if the combined score is high, the word hypothesis is then sent to a syntactic predictor component which pro-

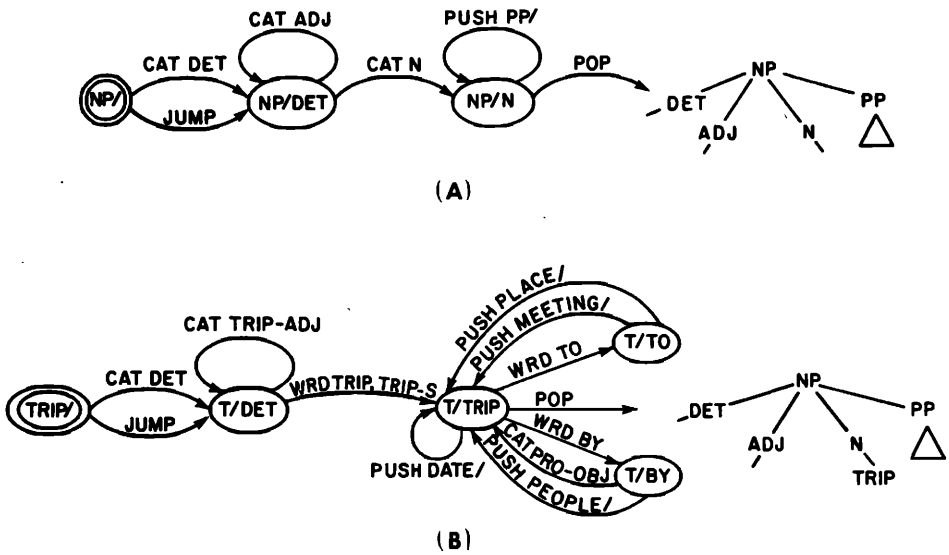


FIG. 6. A portion of the augmented transition network grammar. The network of part (a) defines acceptable noun phrases as consisting of the categories determiner, optional adjective string, noun, and optional prepositional phrases. A way to embed semantic constraints on the relations between words of an acceptable noun phrase is shown in part (b). Only one type of noun phrase concerning a trip is accepted by this fragment of the ATN grammar.

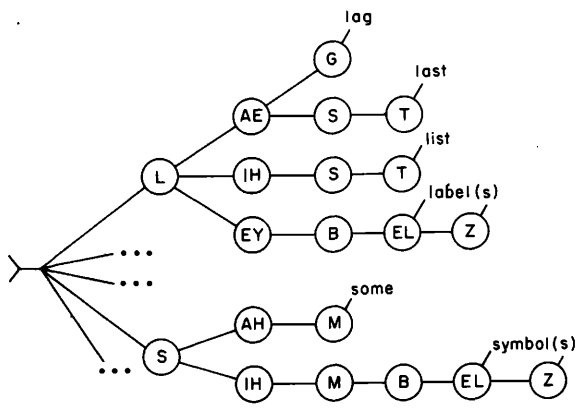
poses words that can appear to the left and to the right of the seed word, given the grammatical constraints. An augmented transition network grammar (Woods, 1970) is used to characterize syntactic and semantic constraints, in a manner that is illustrated in Fig. 6.

Matching scores are obtained for all of these word proposals, using a lexical decoding network (Klovstad, 1977). The lexical decoding network contains a representation of the expected phonetic realizations of each word in all possible phonetic contexts. To derive this network, a set of phonological rules (Woods and Zue, 1976) first transforms a phonemic lexicon into phonetic alternatives arranged in a tree structure, as shown in Fig. 7(a). Then a second set of word-boundary phonological rules attaches terminal nodes back to selected initial nodes of the tree, creating a network of permissible phonetic strings for all possible word sequences from the 1000-word lexicon. For example, the word "list" may be pronounced as [lis] in "list some" due to an optional word boundary phonological rule OPT {ST# S} → {S}, and this fact is captured in the network structure of Fig. 7(b). However, if "list" is to be recognized without the {t}, a word beginning with [s] must follow.

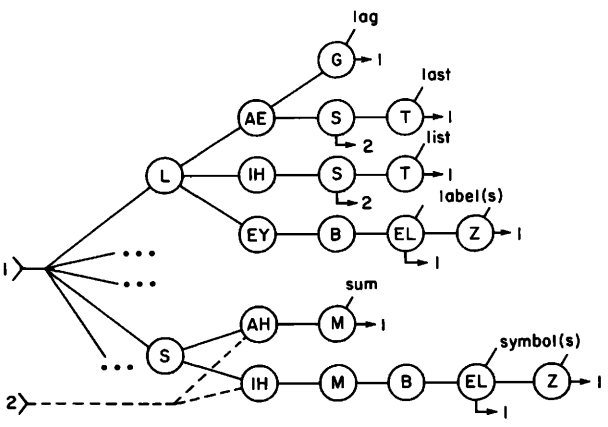
Each word receiving a good score from the lexical decoding network is combined with the seed word to produce a two-word hypothesis, a verification score is derived for the new two-word hypothesis and the hypothesis is then placed in an "event queue." The best scoring partial sentence hypothesis is always extended next. When a complete sentence is found, a deep structure representation of the word string can be sent to the semantic procedures component in order to compute an appropriate response. The response is spoken over a loudspeaker, using a speech synthesis by rule program. A more detailed description of the system is given in Appendix B.

Discussion of BBN

The BBN speech understanding system has a task domain with a more general syntax than the other systems, so it is difficult to judge how much better or worse the system design and individual components are. The same criticism applies to BBN that was leveled at SDC: The way in which the demonstration system failed to meet the ARPA goals makes it impossible to determine what more



(A) INITIAL TREE STRUCTURE



(B) FINAL DECODING NETWORK

FIG. 7. The partial lexical tree for the phonetic representation of several English words shown in part (a) is transformed into a lexical decoding network for the recognition of sequences of words by application of a set of word-boundary phonological rules (Klovstad, 1977).

limited task domain might have resulted in acceptable performance. It would be interesting to know, for example, how much of an improvement in certain critical components is needed to achieve acceptable performance, or how much improvement would be gained by restricting the language definition in various ways.

The most interesting ideas to come out of the BBN project were a lexical decoding network incorporating sophisticated phonological rules, the technique of representing segmentation ambiguity by a lattice of alternatives, and the concept of word verification at the parametric level. However the performance of these components individually and as a total system did not seem to live up to their theoretical potential. Because of the slowness of the system, there was apparently not enough effort devoted to debugging and optimizing individual components in a system context. Specific problems that were never resolved were (1) how to ensure that the segment lattice was in fact providing more information than a linear string of best guesses, (2) how to normalize for talker differences, (3) whether sufficient data were analyzed to rely on the probability estimates of various phonetic confusions, extra segments, and missing segments, and (4) whether the system would perform significantly better if it were fast enough to evaluate many more partial sentence fragments.

C. Carnegie-Mellon University Hearsay-II

The CMU Hearsay-II system organization is shown in Fig. 8 (Lesser *et al.*, 1975; Lesser and Erman, 1977; Reddy *et al.*, 1977). The recognition process is similar in some respects to that employed in BBN Hwim, although the block diagrams and organizational philosophies are disparate. The CMU system configuration consists of a set of parallel asynchronous processes that simulate each of the component knowledge sources of a speech understanding system. Knowledge sources communicate via a global "blackboard" data base. When activated by the appearance of certain types of new information on the blackboard, a knowledge source tries to extend the analysis.

The information on the blackboard is divided into several major categories: sequences of segment labels, syllables, lexical items proposed, accepted words, and partial phrase theories. A knowledge source accepts information at one level and attempts to provide new information at a higher level (bottom-up analysis) or lower level (top-down prediction and verification).

Initially, amplitude and zero-crossing parameters are used to divide an utterance into segments that are categorized by manner-of-articulation features (Goldberg and Reddy, 1976). Good performance is obtained by avoiding the more difficult place-of-articulation decisions in the preliminary analysis.

A word hypothesizer lists all words having a syllable structure compatible with the partial phonetic representation. For example, there might be ten lexical items that are consistent with a fricative—stop—vowel—stop pattern, three items consistent with a fricative—stop—vowel subpattern, and five more items consistent with a stop—vowel subpattern. The performance of the lexical hypothesizer is such that only 70 percent of the correct words are detected (Smith, 1976), but others are found by top-down prediction at a later stage.

A word verification component scores each lexical hypothesis by comparing an expected sequence of spectra with observed linear-prediction spectra. The lexicon used for verification is adapted from Harpy and thus is defined in terms of expected spectral patterns instead of expected phonetic patterns. Coarticulation across word boundaries is a problem using this approach, but some word-boundary acoustic rules are included. Performance of the verification component is indicated in Table IV.

High-scoring words activate a syntactic component which tries to put words together into partial sentence theories. Grammatically acceptable adjacent words are also predicted since the word hypothesizer is not expected to get all of the words of the sentence. The control strategy is similar to that used by BBN in that best-scoring words or sentence-fragment pieces are sought anywhere in the utterance and extended to the left and/or to the right. CMU obtained significantly better performance with an island-driven strategy than BBN, but it is argued below that the Harpy left-to-right control strategy has advantages over any middle-out strategy. Once a complete sentence has been found, a response could be computed by accessing a data base. A more detailed description of the system is presented in Appendix C.

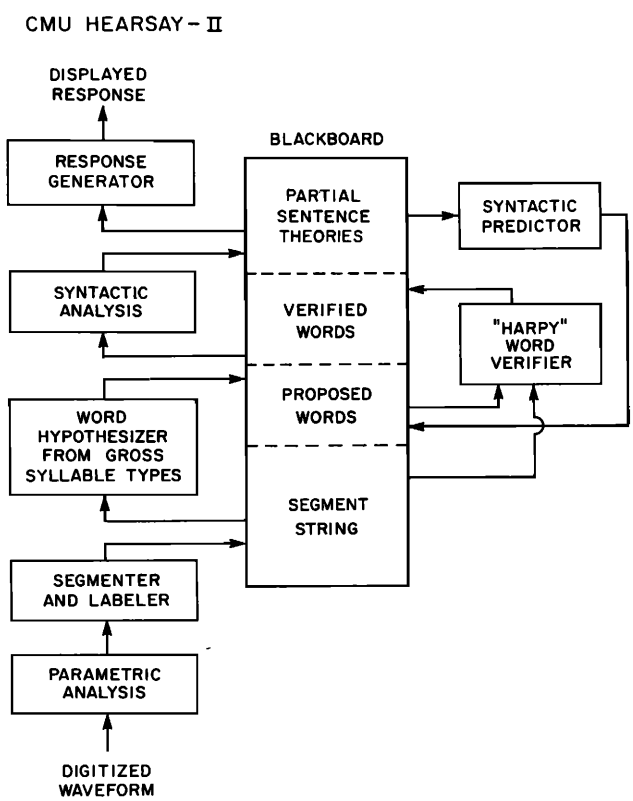


FIG. 8. A block diagram of the CMU Hearsay-II system organization.

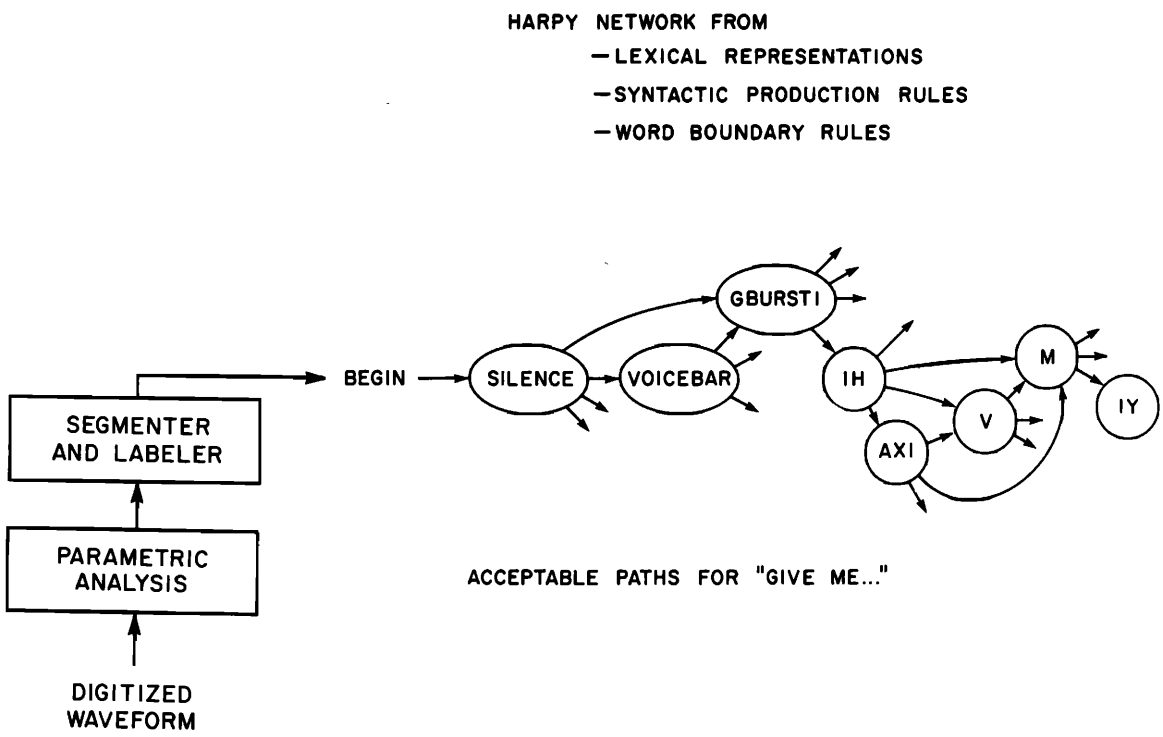


FIG. 9. A block diagram of the CMU Harpy system organization. Shown is a small (hypothetical) fragment of the Harpy state transition network, including paths accepted for sentences beginning with "Give me." Each node is named by the expected linear-prediction spectrum pattern. In general, many paths leave each node, corresponding to other possible sentence-initial words. A finite set of about 10**8 sentences (of length up to 8 words drawn from a 1011 word lexicon) can be recognized by the 15 000 state Harpy network.

Discussion of CMU Hearsay-II

Hearsay-II exhibited the best performance of the systems other than Harpy. Since it is not at all clear that Hearsay-II used components having better absolute performance, it is of interest to speculate on those aspects of the overall system design that account for its superior behavior. There are three essential reasons in my view: (1) as in the BBN system, absolute decisions (e.g., to reject a word hypothesis) were avoided by assigning graded scores so that component errors were not necessarily fatal, (2) computational efficiency issues were always of primary concern so that more alternatives could be considered, and (3) syntactic complexity (i.e., the average number of words to be considered to the right of any correct word) was directly controlled and reduced to a point where the system performance was acceptable. The use of strong syntactic and semantic constraints was encouraged by the ARPA goal structure, and should be utilized in future practical systems to improve performance.

D. Carnegie-Mellon University Harpy

The Harpy system, as implemented by Lowerre (1976), is shown in Fig. 9. The system includes a network of 15 000 states. Embedded in the state transition network are (1) all possible paths through the finite state grammar (i.e., a graph representation of all possible sentences), (2) alternate representations of all lexical items in terms of acoustic segments, and (3) a set of rules describing expected changes to acoustic segment sequences across word boundaries. The set of word-boundary rules, lexical representations, and grammar equations are automatically compiled into the efficient network representation shown in the figure.

The input utterance is divided into brief roughly stationary acoustic segments. Each segment is compared with 98 talker-specific linear-prediction spectral templates to obtain a set of 98 spectral distances, using the minimum residual error metric (Itakura, 1975). Template selection for a new talker is automatic, but requires that the user read about 20 selected sentences.

Each state in the network has an associated spectral template. The decoding strategy is to try to find the best scoring path through the state transition network by comparing the distance between the observed spectra and template sequences given in the network. Generally a state can accept a sequence of several sufficiently similar input segments, although some states are constrained so as to grab a specified minimum or maximum duration of the input.

Harpy is an extension of a Markov model of sentence decoding originally employed by Baker (1974) in a sentence recognition system called Dragon. In Dragon, a "breadth-first" dynamic programming strategy was used to find the optimal path through a network, but in the Harpy implementation a "beam-search" technique is used in which a restricted beam of near-miss alternatives around the best-scoring path are considered, thus reducing the search time significantly. Dragon also

used *a priori* probabilities in choosing the most likely path through the network, while Harpy considers only spectral distance. A more detailed description of the CMU Harpy system is presented in Appendix D.

Discussion of CMU Harpy

Harpy and its predecessor Dragon represent a significant breakthrough in the application of simple structured models to speech recognition. It might seem to someone versed in the intricacies of phonology and the acoustic–phonetic characteristics of speech that a search of a graph of expected acoustic segments is a naive and foolish technique to use to decode a sentence. In fact such a graph and search strategy (and probably a number of other simple models) can be constructed and made to work very well indeed if the proper acoustic–phonetic details are embodied in the structure. The keys to success seem to me to be (1) the way that important structural aspects of language and speech can be folded into an initial network structure, (2) the possibility of optimizing of the network and the spectral templates using a very large body of training utterances, and (3) imposition of strong syntactic constraints.

Harpy is essentially a verification strategy. All alternative sentences are specified by the network, and the task is to verify which sequence of spectral states (path through the network) corresponds to the input sequence of spectra. It has been argued elsewhere (Klatt and Stevens, 1973) that verification of expected acoustic patterns for words is an easier task than phonetic analysis due to the inherent ambiguity of acoustic–phonetic decoding rules. To the extent that phonologists are better able to write generative than analytic rules to describe speech, the advantages of verification strategies will remain.

Due to syntactic constraints, the Harpy network is not particularly dense, so that minimal acoustic–phonetic distinctions are rarely required to distinguish between utterances. The present spectral sequence network may be capable of distinguishing minimal pairs of words quite well, but there is no direct evidence that it can, and some reason to doubt its detailed phonetic abilities given only 98 templates and limited word-boundary acoustic–phonetic rules. Even when syntactic constraints have been applied and the correct sentence has been identified, only 40% of the time does the top-scoring template match the expected template for each state in the best-scoring path through the network. If the templates were analogous to phonetic segments (which is roughly true for consonants in the current implementation, but not as true for vowels) this statistic would imply a less than 40% phonetic transcription performance in the absence of syntactic constraints, which is worse than in any of the other systems. However, there seems to me to be no fundamental limit to the ultimate transcription performance ability of Harpy-like networks if the lexical representations and word-boundary rules are sufficiently detailed.

The grammar on which Harpy and Hearsay-II were demonstrated was actually a member of a set of related grammars manifesting different branching factors and thus a range of task difficulties. One of the reasons for CMU's success was the ability to manipulate branching factor and observe changes to performance. This was a significant achievement given the ARPA objectives. Unfortunately, within the set of branching factors investigated, excellent performance was achieved only by using a rather low branching factor grammar, i.e., one that constrained acceptable sentences so that just two large syntactic classes were allowed: topics, and authors. Test sentences were always constrained so that at least one of these two classes appeared in each test sentence.

There clearly exist tasks for which a Harpy-like network would appear to be applicable [e.g., connected digit recognition or even perhaps air traffic control (Connolly, 1975)], but the languages for such applications will have to be fairly artificial and not a so-called "habitable subset of English" (Watt, 1968). Still the job of creating a Harpy system for a new task domain is not simple; it took careful analysis of 747 sentences to achieve the present level of Harpy performance on this particular 1000-word lexicon.

E. Other ARPA-funded speech understanding research

The ARPA project included a number of supporting efforts that were important to the task of creating the four large speech understanding systems just described. In this review, we have emphasized the systems, but a brief mention of the activities of the other contractors is provided in the paragraphs below.

1. Lincoln Laboratory

Researchers at MIT Lincoln Laboratory spent considerable effort on the development of phonetic recognition strategies (Weinstein *et al.*, 1975). Techniques included formant tracking and the use of formant transition information for stop place-of-articulation categorization. The performance and documentation of these strategies was probably the best of the initial system builders at the time when funds were re-allocated from five to the three research groups showing promise of putting together the best total systems. Lincoln staff also developed a lexical network representation (Klovstad and Mondshein, 1975) that later evolved into the BBN lexical decoding network.

2. Stanford Research Institute

When a planned joint SDC/SRI system development program was no longer possible, Stanford Research Institute staff were forced to carry out their development and testing of system components and strategies using a simple simulation of the behavior of a SDC mapper for word verification. Simulation proved to be a valuable technique for optimizing several system design choices concerning speed/accuracy tradeoffs, without the added run-time cost of using the actual mapper (Paxton, 1976; 1977). The results of the simulations were used, for example, to specify the performance required from the mapper for a given vocabulary size in order to obtain 90% sentence understanding, using a language definition that allows fairly general syntactic constructions,

an independent semantic component, and capabilities for anaphoric references and ellipsis in processing sequential items in a dialogue (Walker, 1976).

3. Special contractors

As part of the overall research and development plan, funds were allocated to several research groups to provide support in the area of acoustic—phonetic analysis. The research contributions of the special smaller contractors[2] have not been discussed in this review. However, significant work was performed toward the development of phonological/phonetic rules for the description of spoken English sentences (Oshika *et al.*, 1975), prosodic decoding rules (Lea *et al.*, 1975), acoustic—phonetic recognition strategies (Mermelstein, 1975a; 1975b), and evaluation of the complexity of the grammars employed in the four systems (O'Malley, unpublished).

III. DISCUSSION AND CONCLUSIONS

Is there a need for speech understanding systems? Ochsman and Chapanis (1974) present evidence that man—man communication via speech is more natural and efficient than other modes of communication such as typing. This would presumably also be true of man—machine communication, especially for unskilled persons interacting with a system for either data input or information retrieval (Lea, 1968). It seems that the need for speech understanding systems is already present, and this need will grow as our dependence on computerized stores of information increases.

One answer to the demand for automated man—machine communication by voice might be a system that recognizes sentences formed by speaking a series of words separated by pauses (Herscher and Cox, 1976). It has been claimed that users can readily learn to insert short pauses between words, transforming the sentence recognition problem into an easier isolated word recognition problem. Coarticulation between words is minimized in this way and word identification by pattern matching is possible. Syntactic and semantic constraints can be applied to limit the set of acceptable words at each sentence location, and thus perform with a recognition rate significantly better than *P-to-the-N*th, where *P* is the probability of single word recognition, and *N* is the number of words in the pseudosentence.

Isolated word concatenation appears to have practical applications in many limited task domains (Martin, 1976). However, it is no more that a compromise solution to the attainment of fast natural communication with computers. The procedures have yet to be generalized to handle large vocabulary tasks, and it has not been shown that users are able to stay in the "pause-between-words" mode in a more complex task environment. There was and is a pressing practical need to study and develop procedures for simulating normal sentence understanding.

Given the demand for speech understanding by computers, was the ARPA project a good thing? The list of scientific achievements in the next section indicates that significant advances have been made in the speech understanding field. Yet if one spends three million dollars a year for five years and the best system turns out to be a one-man-year Ph.D. thesis, not all is well. The other projects sought to build more powerful general systems, but all failed to meet the ARPA goals.

It was potentially beneficial to shake-up the field with a large funding effort, and much good can still come from the ARPA project. On the other hand, it is disruptive to send funding oscillations through the basic research community and to subject science to fads and anti-fads. The danger now is that funds will be less available for the basic science that must be done in the speech analysis area before real further progress is made.

A. Scientific achievements of the ARPA program

The following paragraphs list a number of good ideas drawn from the four speech understanding systems and elsewhere.[4] In addition to identifying several scientific achievements of the ARPA program, this section is intended to summarize the state of the art and to suggest guidelines for the development of future speech understanding systems.

1. System organization

The structures of Harpy and Dragon represent a significant improvement in the realization of sentence verification procedures. System organization is immensely simplified by precompiling disparate knowledge into a uniform network representation at the spectrum level. A second new organizational concept comes from Hearsay-II and involves creation of a set of parallel asynchronous processes that communicate via a blackboard. As a conceptual model, the approach may be applicable in other problem solving domains.

2. Grammar design

The ability to manipulate grammatical complexity and observe changes to system performance as the task is simplified was an important factor in the success of CMU. The shift of attention from size of the lexicon to effective grammatical branching factor is an important advance in the quantification of task difficulty from the original ARPA goal of a 1000-word lexicon. It means that a difficult problem can be made easier by reducing the apparent size of the lexicon. What is needed now are techniques to reduce grammatical complexity while maintaining task objectives and retaining language habitability.

3. Control strategy

Control strategies that work from the middle out, starting with a good-matching content word utilize less syntactic constraint and have been found to cost a great deal more in complexity and computation time than strategies based on strict left-to-right processing through an utterance. If phonological rules handle function word variability well, then a strict left-to-right

strategy with a breadth-of-search capability, as in Harpy, seems to be the best choice.

4. Semantics and context

Most semantic constraints employed by these systems are realized within the syntactic production rules. BBN Hwim contained a separate semantics module, but it was not used very much during sentence recognition. None of the four systems were able to use prior discourse information to reject a sentence such as "What is their registration fee?" because there was no assignable referent for "their." However, earlier, Hearsay-I (Reddy *et al.*, 1973) contained a chess-playing program that checked requested moves for plausibility.

5. Syntax

It is likely that almost any parser structure will do for simple speech understanding tasks in which all that is required is an enumeration of the possible lexical items following a given sentence fragment. In fact, the best solution for a finite grammar is very likely to precompile a list of the permissible word sequences into a network, as is done in Harpy.

The speech understanding project has benefitted from prior work on the automatic parsing (syntactic analysis) of written sentences. Powerful mechanisms such as an augmented transition network grammar and parser (Woods, 1970) have already been developed for processing word strings from left to right. While many grammar formulations could be considered, the augmented transition network grammar has the advantages of permitting semantic constraints to be written into the grammar and allowing many alternative parses to be computed efficiently in parallel (Woods, 1970). The grammar also includes simple methods of searching most-likely structures first and can produce structural representations that are ideal input for semantic processing routines involved in response generation.

6. Word identification/verification

Each word or morpheme of the lexicon has been specified at a fairly abstract phonemic level in several of the systems. This makes lexical development and augmentation much easier than if all possible detailed phonetic or acoustic forms must be listed. Phonological rules that operate within words and/or across word boundaries are used to expand the lexicon into multiple representations. Phonology seems to have come of age over the past few years in that formal rules of considerable predictive power have been developed. As a starting point the morphological expansions and phonological rules of Zue (Woods *et al.*, 1976, Vol. 3, pp. 57—72) might be used. Additional more general rules are to be found in the work of Cohen and Mercer (1975) and Oshika *et al.* (1975).

The potential role of the syllable in lexical verification was elaborated by SDC, who suggested that allophonic variations can be predicted in a relatively straightforward way if one begins verification at a syllable peak and then looks for acoustic evidence of adjacent consonants that are expected. The advantages of the syllable as a recognition unit are less clear. It might be argued that the dyad (Peterson, Wang, and Sivertsen, 1958), an interval from the middle of one phonetic segment to the middle of the next segment, is a unit having about the same theoretical advantages. There are far fewer dyad types than syllable types in English. Silverman and Dixon (1976) have employed the dyad as a recognition unit with good success for a single talker. The diad has been termed a "diphone" when used as a building block for speech synthesis (Dixon and Maxey, 1968).

Word identification in sentence contexts is possible only if the effects of phonetic/phonological recoding at word boundaries can be decoded. This requires that the phonological encoding rules be known, and that computational procedures be available for applying the inverse rules rapidly and selectively. To take the example of the rule [s t # s] → [s] as in "list some," it would be costly to test every [s] in an utterance for a possible underlying [s t # s], especially considering the number of word-boundary rules that would have to be treated in this way. The solution that comes from the BBN system is to incorporate word boundary phonology into the stored lexical representations by first constructing a lexical tree of expected phonetic sequences, and then transforming the tree into a phonetic-sequence network of the type shown in Fig. 7.

7. Acoustic-phonetic processing

An advance in the area of acoustic—phonetic processing has been the realization that *phonetic* segmentation and labeling is not necessary to word identification in connected speech. The Harpy philosophy of representing words by sequences of spectral templates in a network that takes into account word boundary phonology shows great promise.

The actual spectral representation (linear prediction spectral analysis) used in Harpy and the spectral distance measure (the minimum residual error) used are computationally very efficient, but probably not optimal and not related very closely to perceptual distance. For example, the metric does not incorporate overall spectral intensity and may therefore confuse a silence spectrum with some speech sound having a similar spectral shape. Also Harpy used only 98 templates to represent the entire inventory of spectral variations in speech. The excellent performance of Harpy may mean that the details of spectral representations and distance measures are not critical. On the other hand, perhaps even better performance in harder task domains is possible within the Harpy framework by using improved metrics.

Comparison of the general performance of the phonetic analysis components of the SDC, BBN, and CMU Hearsay-II systems in phonetic labeling with for example, Silverman and Dixon (1976), Weinstein *et al.*, 1975, and earlier work suggests that the contribution of the ARPA project to improved phonetic recognition strategies is not in proportion to the level of effort expended. Schwartz and Cook (1977) have recently attained 67%

correct phonetic transcription capabilities using 71 phonetic categories in a phonetic vocoder application, but their system is not well documented. There is a clear need for continued work in this area.

Shockey and Reddy (1975) discovered that linguists are actually not very consistent at phonetic transcription if the language is unfamiliar. From their data, they speculate that machines should not be expected to do better than 60%—70% correct phonetic transcription performance. We believe that their results may be a reasonable test of the current status of a universal phonetic theory, but they do not measure transcription abilities of listeners who are permitted to make use of the phonetic and phonological constraints of English. Recent experiments by Mark Liberman and Lloyd Nakatani (personal communication) suggest that listeners can transcribe English nonsense names embedded in sentences (and obeying the phonological constraints of English) with better than 90% phonemic accuracy. It is likely that machine performance must approach this figure before very powerful speech understanding systems are realized. Alternatively, perhaps the best bet is not to do phonetic labeling at all, as in Harpy.

Prosodic cues (fundamental frequency, segmental duration, and the intensity contour) suggest a stress pattern for the incoming syllable string, and thus could assist in lexical hypothesization. Prosodic cues also indicate clause boundaries, phrase boundaries, and, to a minor extent, word boundaries. While relatively little use was made of prosodic information in the four speech understanding systems, some ideas for prosodic analysis were proposed (Lea, Medress, and Skinner, 1975).

8. Use of statistics

Jelenek (1976) argues for the use of decision strategies that are based on the collection of an appropriate set of probabilities determined experimentally. Several of the speech understanding systems used estimates of the probability of a phonetic or lexical decision given the acoustic data in scoring the goodness of a theory, and each seems to have gotten into trouble by so doing. The problem is to analyze enough data to be sure of the probability of infrequent confusions. This is nearly impossible if one wants to take into consideration factors such as phonetic environment.

9. Acoustic analysis

It is now known that the important information-bearing elements of the speech code are contained in the magnitude spectrum of speech, i.e., in a sequence of well-chosen short-term spectra. Linear prediction spectra have proven to be a robust spectral representation having the additional advantages of being a pleasing visual idealization of speech, of having an existing simple metric for spectral comparisons, and of permitting the estimation of formant frequencies. On the other hand, it appears that filter banks designed carefully to take into account critical bands and other psychophysical constraints are equally useful as spectral representations (Klatt, 1976b).

10. Talker normalization

A surprisingly powerful method of talker normalization is incorporated in the Harpy system. About twenty known sentences are processed to derive talker-specific spectral templates automatically. These templates are capable of capturing a wide range of talker characteristics including important differences between men and women. Other talker differences such as differences in dialect are best overcome for the present by restricting system usage to talkers of a single fairly uniform dialect.

11. Response generation

The BBN Hwim system and CMU Hearsay-II included a data base and response generator (although these components were not usually connected during a recognition demonstration). In this sense, a distinction between speech understanding (is the response correct?) and speech recognition (are all of the words correct?) was realized. Cases where a correct response would be generated in spite of lexical identification errors were fairly frequent in Hearsay-II. To that extent, the systems described here represent the beginnings of true machine understanding of spoken language.

To generate a proper response, one must solve an information retrieval problem, choose an appropriate frame sentence for a response, and synthesize an audio output. Some progress in general solutions to these problems was achieved at BBN (Woods *et al.*, 1976; Klatt, 1976a).

12. Contributions to speech science

Many of the scientific achievements listed above impact on the speech sciences. One might have expected more in the way of detailed algorithms for the recognition of phonetic categories or descriptions of acoustic—phonetic details of sentences spoken by different talkers, but speech scientists should be made aware of advances such as the observations that (1) linear prediction spectra are a useful representation of speech for spectral analysis or formant frequency analysis (2) progress has been made in describing the steps involved in predicting the phonetic characteristics of words in sentences from a phonemic representation, (3) strategies exist for automatic phonetic transcription with performance of about 60%—70% correct, (4) talker normalization by acquisition of talker-specific spectral templates works surprisingly well, (5) lexical hypothesization need not include a step in which a phonetic transcription is derived, and (6) some of the computational structures suggested for a speech understanding system may in fact constitute a good model of sentence perception.

13. A proposed future system

A possible structure for a future speech understanding system that incorporates many of these ideas is shown in Fig. 10. An acoustic-segment lexical decoding network is generated off-line from a phonemically organized task-specific lexicon. A set of phonological rules, a diphone dictionary, and a set of word-boundary phono-

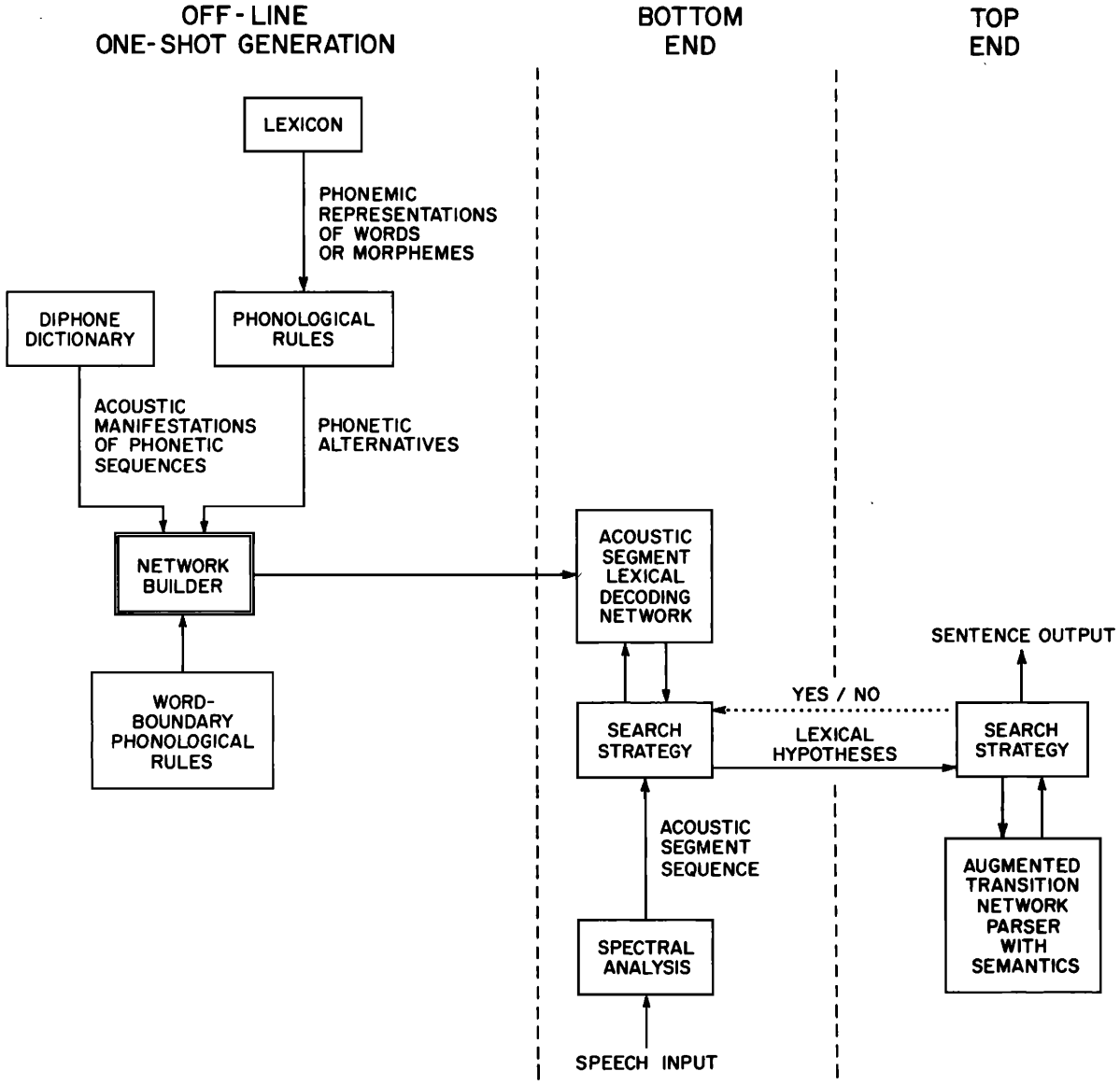


FIG. 10. Proposed structure for a future speech understanding or sentence recognition system.

logical rules convert the lexical representations into a network of expected acoustic segment sequences for all possible word sequences composed from the vocabulary. Each state of the network is represented by one of a moderate set of spectral templates.

A spectral template is envisioned to be something like a short-time spectrum, but the representation is refined to take into account critical bands, masking, loudness, and other limitations imposed by the peripheral auditory system. The comparison of spectral templates with input spectra is assumed to include sophisticated calculations depending on the types of spectra being compared. The decoding network includes a specification of duration limits for each template and an ability to normalize these durational constraints on the basis of estimates of local speaking rate or stress.

The bottom end is patterned after that of Harpy. An input sequence of acoustic segments is processed from left to right, always seeking the best path through the lexical decoding network. The resulting partial string of words is sent to the top end to see if the sequence is permitted by the grammar or if the next-best path through the network should be pursued instead. The Harpy talker normalization procedure of adaptive acquisition of talker-specific templates is assumed, although more than 98 spectral templates will probably be required to achieve optimum phonetic and lexical discrimination. There is no phonetic or phonemic level of representation at any stage in the bottom end recognition process, although important use is made of these linguistic concepts during network generation.

The top end uses an augmented transition network parser to monitor the word sequence proposed by the bottom end and to terminate immediately ungrammatical hypotheses. The first acceptable sentence that spans the input is sent to a semantics/response generation module (not shown) or is simply typed out if the system is being run in sentence recognition mode.

The keys to success of this proposed approach are the selection of a good set of spectral templates, development of a diphone dictionary that captures the acoustic characteristics of English phone sequences, and a set of phonological rules that can generate all possible phonetic manifestations of word sequences. Stated in this way, the task ahead is still formidable, but it is an interesting and well-defined challenge. The missing pieces must come from speech science: the computer sciences have done the job of providing a structure in which to embed speech knowledge for recognition purposes.

B. Relations to psychological models of speech perception

Previous psychological models of phonetic analysis include the motor theory of speech perception (Liberman *et al.*, 1962; 1967), analysis by synthesis (Halle and Stevens, 1962; Stevens, 1972b), multistage feature-based models (Pisoni, 1977), and quantal invariance models (Stevens, 1972a; Cole and Scott, 1974). An alternative psychological model of early stages in the speech perception process might be based on the organization shown in Fig. 10. This model makes four novel claims which, in terms of speech perception, must be looked upon as conjectures:

Conjecture 1: lexical decoding is usually direct, with no analysis-by-synthesis or other mechanism for rich information feedback from higher linguistic levels.

Conjecture 2: generative rules are used to precompile acoustic—phonetic and phonological knowledge into an appropriate decoding structure because analytic rules are too ambiguous and thus computationally inefficient.

Conjecture 3: there is no use made of a phonetic level of representation during sentence processing.

Conjecture 4: syntactic analysis is direct, using a precompiled network to characterize permitted sentence structures rather than a set of generative rules and an analysis-by-synthesis strategy.

These are independent claims. It is possible that any or all of them are true. The first claim has some support in the recent experimental work of Garrett (1977). The second claim is a theoretical argument that would be invalidated if, e.g., processing time for "list some" were greater than for "list one" due to the extra processing required to overcome [t] deletion in the one case. The third claim is superficially contradicted by numerous experiments concerned with the properties of phonetic feature detectors (see Studdert-Kennedy, 1975, for a review). However, if one looks carefully at these experiments, one can argue that none of them have invalidated the kind of template theory that we have in mind because the concept of phonetic similarity, as measured by the number of phonetic features that are shared, can be replaced directly by a concept of spectral similarity. The fourth claim already has its proponents (Wanner and Maratsos, 1978).

Research is needed to (1) investigate how pieces of the model might evolve during language acquisition, (2) determine how prosodic analysis is performed in such a model, (3) see how unfamiliar words are added to the lexicon, (4) determine how noise-corrupted speech is processed, and (5) make predictions that would distinguish the model from phonetic feature based models.

The model of Fig. 10 is probably too simple to be true. It is more likely that the central nervous system can make use of many strategies, including phonetic analysis and analysis-by-synthesis, in which case the conjecture made here is that, under normal circumstances, the primary way of initiating sentence analysis is indicated by the block diagram of Fig. 10.

C. Future research

The key to improved bottom-end performance (lexical hypothesization) in future speech understanding systems, it seems to me, is the transformation of a phonetic identification problem into a spectral state verification problem, as was done in Harpy. Phonetic representations for words of the lexicon are replaced by sequences of expected spectra and some durational constraints. This translation problem is nontrivial and required many trial-and-error iterations during Harpy development. The availability of a list of expected template sequences for all possible phonetic strings of English would considerably aid in bringing up new lexicons. The main objective of future research in this area should therefore be the accumulation of more detailed linguistic and acoustic—phonetic facts about English sentences.

Future research on top-end design might focus on the imposition of realistic semantic and task constraints and on computational efficiencies that will ultimately permit the use of more general grammars. Another worthwhile pursuit might be to interface a Harpy-like lexical recognition network to a top end with a nonfinite grammar, as in Fig. 10.

Additional research will be required to build a practical speech understanding system that has a complete human-engineered interactive capability. Problem areas in need of further work include the design of habitable languages, how to monitor the input to know when a signal is present (Rabiner and Sambur, 1975), including rejecting "umm's" and breath noise (Martin, 1975), how to know when an utterance is complete, whether it will be necessary for a user to see a display of the request and give a "yes"—"no" response, etc.

The steering committee prepared a follow-on research plan (Newell *et al.*, 1975) that took up a number of these issues and also proposed that the four speech understanding systems not be allowed to die. They argued that the systems required substantial additional debugging and tuning before reaching their full potential and that experiments were needed to better understand their capabilities and limitations. The committee proposed that the systems ought to be preserved as a resource for use by a wider community because considerable scientific insight might be gained from widespread experimentation with these systems. Based on the current funding picture, it appears that none of these recommendations will come to pass.

As a final note, we may ask "How hard is the sentence understanding problem in the limited contexts investigated during the ARPA project?" In 1970, when compared with isolated word recognition, the problems seemed immense. After the limited success of Harpy, one becomes more optimistic about the abilities of future systems. Still, the nature of the problem is elusive. Even the dimensions of task difficulty have yet to be adequately defined. Limited experience suggests that significant increases in difficulty are associated with increased grammar branching factor (caused by increased lexical size or increased syntactic freedom), the inherent acoustic ambiguity of words that must be distinguished, and the importance of unstressed function word recognition to sentence decoding. If these factors can be controlled in a particular task domain, speech understanding by machines is now a practical goal.

ACKNOWLEDGMENT

Preparation of this review was supported by the Advanced Research Projects Agency of the Department of Defense under Contract N00014-75-C-0533. I wish to thank everyone associated with the ARPA Speech Understanding Project who spent many hours helping me to understand these complex systems. Special thanks go to W. A. Lea, A. Newell, J. E. Shoup, D. R. Reddy, J. J. Wolf, and W. A. Woods for correcting some of my errors of judgment.

APPENDIX A. DETAILED DESCRIPTION OF THE SDC SYSTEM

A. Acoustic analysis

The speech waveform is digitized at 20 000 samples/sec with no prior frequency-domain preemphasis. The waveform is divided into 10-ms chunks, and the energy is computed for each chunk. Other parameters that are computed every 10 ms include zero crossing counts and the fundamental frequency contour. Fundamental frequency is extracted with a down-sampled center-clipped autocorrelation technique (Gillmann, 1975). A linear prediction spectral analysis is performed every 10 ms, using 24 coefficients and a 25.6-ms Hamming window. A formant tracker estimates which of the linear prediction poles are the four lowest formant frequencies over voiced intervals.

These parameters are used to apply gross phonetic labels to relatively stationary sequences of 10-ms spectra. Special routines then characterize vowel-like segments (Kameny, 1975; 1976) and frication spectra (Molho, 1976) in terms of phonetic labels with associated scores for the three best alternatives. A post-analysis pass smooths the label names through local continuity constraints. The resulting analysis is placed in an array called the A-matrix. Segmental identification is based on acoustic-phonetic recognition strategies having a good theoretical basis—formants for vowels, gross spectral shapes for fricatives and plosive burst onsets. However, the performance (first choice label correct about 50 percent of the time for 40 phonetic categories) suggests that there are not enough detailed facts about expected influences of coarticulation with adjacent phones in the algorithms.

B. Lexical representations

The format used to represent the word "Lafayette" in the lexicon was shown in Fig. 3. The pronunciation variants for this lexical item were generated automatically from a set of phonological rules (Barnett, 1974) and then stored in the form shown in Fig. 3 so as not to have to execute the rules each time a word is hypothesized by the top end. Similar lexical networks have been used in other speech understanding systems (Tappert, Dixon, and Rabinowitz, 1973; Tappert, 1975; Cohen and Mercer, 1975; Baker, 1975; Woods and Zue, 1976).

C. Word verification

The mapper forms the interface between the top end and the A-matrix. Each time a word is hypothesized, the mapper calls "phoneme sniffer" subroutines that give matching scores for each expected phoneme. The mapper is organized in terms of syllables, attempting to match a vowel nucleus first, then working outward to verify expected adjacent consonants. The best-scoring alternative of a lexical spelling lattice is found, and phoneme scores are combined by taking the geometric mean to get a total word score. Use of the geometric mean penalizes a single bad phonemic match. The control program converts this score into an estimate of the probability of correct verification (on the basis of a prior data collection experiment) and accepts the best word from a syntactic class if it exceeds a threshold. Other words from this syntactic class that exceed the threshold may be used later to form other sentence fragment hypotheses if the best word fails to generate an acceptable total sentence. Frequently there are unexplained temporal gaps and overlaps between the starting and ending times of adjacent words of a sentence hypothesis, so the control box also sends word pairs to the mapper for evaluation of word adjacency plausibility.

The phoneme sniffers first consult the list of phoneme labels in the A-matrix in their search for a match (Weeks, 1974). If there is not a direct match with the phoneme being sought, other parameters of the A-matrix may be examined. In other cases, a matching score is obtained by consulting a phoneme confusion probability matrix in which entries reflect the probability that phoneme x will be confused with phoneme y, as estimated from a limited data sample. Phonetic context, such as the fact that [t] and [k] are more likely to be confused before [i] than before other vowels, is not considered when using the confusion matrix.

One of the more difficult problems in structuring the mapper has been to keep track of the time position within the A matrix, and not to miss a syllable or detect extra syllables. Errors of this type are fatal if they reject a correct word, since the top end cannot overcome such a decision. Fatal errors (often of this type) occurred in 40% of the large set of utterances tested, indicating that the present mapper/A-matrix combination must be im-

proved, or the exclusively top-down system strategy must be changed.

The mapper has been evaluated by making over 11 000 verification requests using the 1000-word lexicon in the course of attempting to recognize several sentences. Relevant performance statistics are given in Table IV. Of 71 correct words hypothesized, 65 were verified and 6 were (fatally) rejected. Of 11 000 decoy word proposals, all but 372 were correctly rejected. However, the 372 false "yes" answers mean that for each of the 65 correct responses by the mapper, there are on the average 6 false words accepted that must be rejected by syntactic constraints imposed by the top end as attempts are made to extend these false partial sentence theories.

APPENDIX B. DETAILED DESCRIPTION OF THE BBN HWIM SYSTEM

A. Acoustic analysis

The speech waveform is digitized at 20 000 samples/sec and the first difference is computed to remove the dc component and tilt the spectrum up somewhat. A 13-pole selective linear prediction analysis (Makhoul, 1975) is performed over the 0—5-kHz range every 10-ms using a 20-ms Hamming window. (Essentially no use was ever made of the 5—10-kHz information.) Formant frequencies are estimated as the lowest bandwidth poles in the linear prediction analysis during voiced intervals. Other parameters used for segmentation and labeling include the energy as a function of time in several different frequency passbands, a zero-crossing count, and fundamental frequency.

Segmentation and phonetic labeling is accomplished in several passes across the data, with each new pass refining the decisions made on an earlier pass. Segmentation ambiguity is represented by a lattice of alternatives, as shown in Fig. 5. For each segment of the lattice, all possible phonetic labels are given a rating score based on acoustic similarity and/or experimentally determined confusion probability. As in the SDC system, probability matrix entries were computed independent of phonetic context.

The performance of the segmenter has been evaluated for 124 sentences read by three talkers, using the dictionary spelling for the words of an utterance as the correct answer. For 2850 dictionary segments, the segmenter found 5127 segments arranged in a lattice such that, if one followed the most-correct path through the lattice, only 1.6% of the expected segmentation points were missed, and only 1.8% of the time did the best path contain an extra segment (Woods *et al.*, 1976, Vol. 2, pp. 9—39).

The labeling performance of the system was evaluated by considering the best path through the lattice, as defined above. With 71 possible phonetic labels, the rank-order distribution of the correct label ranges from 52% correct first choice to being within the top five label choices 83% of the time. This performance might appear to be somewhat better than that of the other systems, but it is not certain that the lexical matcher will find the best path through the lattice. Some high-scoring false word matches may be generated by using the alternative incorrect paths. A measure of information content would be needed to compare objectively the segment labelers used in the three systems.

The lexicon contains phonemic spellings for each word or morpheme. A set of phonological rules are applied to generate alternative pronunciations (Woods and Zue, 1976). On the average, a word begins with a single phonemic spelling and ends up with about two acceptable phonetic realizations after application of a modest set of phonological rules. Additional rules of the BBN phonological rule expansion system optionally delete or change some segment labels that are frequently missed by the acoustic—phonetic labeler. For example, a rule optionally deletes the /n/ in poststressed /nt/ clusters. If possible, it would have been better to make improvements to the labeler so that it could detect nasalization in these cases, but it is nevertheless very important for the lexicon to predict what is actually observed by the acoustic processing routines, and these special rules are one possible way of achieving this goal.

The entire lexicon of alternative pronunciations is folded into a single state transition network that incorporates word-boundary phonological rules and that can be searched very efficiently for lexical matches (Klovstad, 1977; Klovstad and Mondshein, 1975; Woods *et al.*, 1976, Vol. 3, pp. 12—27). The lexical decoding network is of sufficient theoretical interest for the design of future speech understanding systems that its general character will be described in some detail here. The first step in the creation of a lexical decoding network is to combine the expected phonetic spellings for all lexical items into a tree structure, as shown in Fig. 7a. Initial parts of words are combined with other words having the same beginning phonetic segments. The termination of each word is identified by one or more nodes in the tree, and these word-terminal nodes are unique insofar as the words of the lexicon have disjoint phonetic representations.

The tree shown in Fig. 7(a) is transformed into a network by application of a set of word-boundary pronunciation rules. Consider the word "list" in the tree. The pronunciation of "list" may be [l IH s] in the environment "list some," so the derived network representation of "list" shown in Fig. 7(b) has a path going from the [s] node of "list" directly to the second phone of all words starting with [s], and a path from the [t] node of "list" to the first phone of all words. This network structure captures a general word-boundary phonological rule stating that words ending in [s t] can be pronounced (optionally) without the [t], but that this can only happen if the following word begins with [s]. The rule is stated in a natural notation: OPT{S T # S}→{S} before being compiled into various modifications to the lexical tree. In this example, it is possible to accept the word "list" without seeing a [t], but only words starting with [s] can be matched thereafter.

It is possible to embed within this network structure any of the types of rules in the literature on word boundary phonology. The advantages of a lexical decoding network go beyond its being a concise statement of allow-

able phonetic strings corresponding to all word sequences of connected-word utterances. The network also allows computational (search) and storage efficiencies that can become significant as vocabulary size increases. Klovstad has developed scoring procedures for input phonetic strings having a missing segment or an extra segment by referring to experimentally determined probabilities of these phenomena. In my opinion, this is an attractive structure for representing lexical information in a generalized speech understanding system, although there are problems in accumulating sufficient statistics to score missing/extra segments properly.

B. Word verification by parametric synthesis

A word verification component is included in the BBN system to overcome the inherent inaccuracies in performance of the segmentation and labeling scheme by returning to the parametric level. One can be more certain about what acoustic data to expect if a word and, if possible, its phonetic context are given (Klatt and Stevens, 1973). The strategy is of some theoretical interest since it is a concrete example of an analysis-by-synthesis model of speech perception (Halle and Stevens, 1962).

Verification of a word begins by sending the phonemic representation, obtained from the lexicon, and adjacent phonemes, if known, to a speech synthesis-by-rule program (Klatt, 1976a), which was modified to predict spectra instead of generating waveforms (Woods *et al.*, 1976, Vol. 2, pp. 40—57). Verification scores are obtained by comparing 10-ms frames of synthesized spectra with selected spectral frames of the unknown utterance, using a dynamic programming algorithm to find the best possible alignment (Woods *et al.*, 1976, Vol. 2, pp. 58—68).

The performance of the verification component has been tested during operation of the speech understanding system. The distribution of verification scores for words that the lexical matcher thought were good word candidates during sentence recognition can be compared with an arbitrary threshold to obtain the performance figures shown in Table IV (Cook and Schwartz, 1977), although the system used graded scores rather than an absolute accept/reject threshold of the type implied by the contents of Table IV. The performance evaluation included three talkers, and a few changes were made (by hand) to the synthesis rules for each talker. Superficially, performance may seem poorer than for the SDC verifier, but the lexical matcher has removed nearly 95% of the worst matching words from consideration by the BBN verification component, while the SDC verifier processes all words permitted by the syntax.

Analysis by synthesis has been a popular model of the perceptual process. However, considering the computational cost of analysis-by-synthesis techniques and the problems of normalization for different talkers, it is the author's opinion that a more promising verification alternative for a computer system is the spectral-sequence verification strategy implicit in the Harpy speech understanding system.

C. Syntactic/semantic component

The syntactic component consists of an augmented transition network (ATN) parser; a portion of the ATN grammar is shown in Fig. 6 (Woods, 1970). The network indicates several possible paths that can be taken after a determiner has been found. The label PP is an abbreviation for a sub-network that recognizes prepositional phrases. The network is augmented with registers that are used to develop a quantified predicate calculus "deep-structure" representation for the sentence, i.e., the surface effects of certain grammatical transformations are undone to arrive at a standard syntactic representation appropriate for semantic interpretation. For example, there is a subject register, a "negative-particle-found" register, etc. The contents of the registers are modified as the parser discovers e.g., that the sentence is in the passive mode. This type of parser is considered by some psycholinguists to be the best current model of how a listener decodes the underlying structure of a spoken sentence (Wanner and Maratsos, 1977).

Semantic constraints are applied in the BBN system by adding them to the parsing grammar (although a separate semantic network was used for response generation). As an example of how semantic constraints are added, the general noun phrase subnetwork shown in Fig. 6(a) is expanded into a number of specialized noun phrase recognition networks, one of which is shown in Fig. 6(b).

Augmented transition net parsers were originally designed to process a sentence from left-to-right. However, the BBN parser has been modified to work in either direction or middle-out for the speech understanding application (Woods *et al.*, 1976, Vol. 4; Bates, 1975). Since the grammar is fairly general and complex, computational efficiencies are achieved by precomputing the answers to certain frequently asked questions concerning what can appear to the left and right of word hypotheses belonging to selected syntactic categories.

D. Control strategies

A speech understanding system can be divided into a set of quasi-independent components (this is especially true of the Hearsay-II system), each of which provides some information toward the ultimate decoding of the utterance. The problem of combining information from the segment lattice, the lexical decoding network, and the verifier is handled in the BBN system by a philosophy of transforming all scores into log likelihood ratios that can be added to form an overall score for a sentence fragment hypothesis. The idea is that scores can be more easily combined if each is based on the probability of being correct, as estimated from prior performance analysis for each component.

A number of control strategies were investigated at BBN as means to find the correct sentence without considering too many alternatives. Quick convergence is needed because, as currently implemented, the system is very slow. One strategy estimated the best possible matching score as a function of position within an utter-

ance (using the lexical matcher to compare all lexical items with all possible segment starting positions). Partial sentence theories were then scored with respect to how far they fell short of reaching this maximum possible score. If a best-first search both to the left and right of a partial sentence theory were always pursued, BBN showed that the first sentence found had to be the best possible interpretation of the utterance. This optimal search theorem assumed that the initial scan gave an accurate estimate of the best possible score at any position in the utterance (not necessarily a good assumption since word boundary phonology was ignored in the initial scan).

Experience showed this strategy to be computationally costly whenever the segment lattice contained a phonetic error of a type not seen before in the probability-gathering stage. The system did not have time to recover from the resulting poor matching score for a correct word, so BBN modified the control strategy to work mostly from left to right, but to retain the theoretical advantage of initially jumping over the first few (less distinctly articulated) segments to find a seed word. The system then worked backward to find the word(s) to the left, and then worked strictly left-to-right. Strictly left-to-right sentence analysis is to be preferred in most speech understanding applications (and humans probably also process essentially left-to-right) because there are fewer local syntactic constraints that can be applied to a sentence fragment in the middle of an utterance.

The BBN system runs in about 1000 times real time on a PDP-KA10, which has a speed of about 0.4 MIPS (million instructions per second). This was a serious problem for system development and knowledge source debugging because it took a long time to accumulate performance data. BBN argued that a good deal can be learned about control issues by careful observation of system traces for only a few sentences. They claimed that it would be possible to implement the speech understanding system in real time on a large 100-MIPS machine by converting some of the code from INTERLISP to assembly language.

E. Response generation

The BBN system included a set of semantic procedures for accessing a data base to compute the response to an input utterance. An appropriate syntactic frame was selected to verbalize the response, and phonemic representations for the words of the response were sent to a speech synthesis by rule program (Klatt, 1976 a). The fully automatic generation of a spoken response to a correctly recognized sentence was demonstrated in February 1976.

APPENDIX C. DETAILED DESCRIPTION OF THE CMU HEARSAY-II SYSTEM

A. Acoustic analysis

The speech waveform is low-pass filtered to 5 kHz, digitized at 10 000 samples/sec, and filtered using a first difference to remove dc and tilt the spectrum up somewhat. The resulting waveform and a second waveform obtained from it using a low-pass smoothing filter are then divided into 10-ms chunks and characterized by a peak-to-peak amplitude and a zero-crossing measure. The two amplitude and zero-crossing functions are used to segment the utterance into intervals of relatively little acoustic change. Then each interval is assigned to one of several broad manner-of-articulation classes using a decision tree of logical threshold tests on the four parametric functions. The performance of the segmenter was compared with that of a hand segmentation. It was found that the segmenter missed 2% of the segments, hypothesized 20% extra segments, and gave a correct manner classification 90% of the time (Goldberg and Reddy, 1976).

A label was assigned to each quasi-phonetic segment (or several labels if they received about the same score) based on the distance between the linear prediction spectrum of the middle 10-ms frame of the segment and a set of 98 templates. The performance of the labeler was also evaluated. It was found that the correct label was the first choice 42% of the time and it was one of the five top choices 75% of the time, using 98 templates to represent 98 phonetic segment types and the minimum prediction residual distance measure to rank-order phonetic choices.

B. Word hypothesization

A word hypothesizer component takes as input the segment label string and tries to find word matches anywhere within the string, using the lexicon of stored representations for words. The hypothesizer is organized around the syllable structure of a word. All lexical items having the same syllable structure as a portion of the input (e.g., fricative—vowel followed by plosive—vowel) are proposed by the lexical hypothesizer. The technique is based on the assumption that manner-of-articulation decisions are more reliable than place-of-articulation decisions.

Performance evaluation of the word hypothesis component indicates that for a typical sentence, 50 words out of the 1000-word lexicon are hypothesized per word position, and the correct word is among the 50 about 70% of the time. Words can also be hypothesized top-down, using a syntactic predictor, so correct words missed by the word hypothesizer need not cause a fatal error in overall system performance.

C. Word verification

A word verification component is used to reduce the number of lexical hypotheses to be considered by the higher-level modules, and to rank order the merits of each accepted hypothesis. This component has the same structure as the Harpy system described below, i.e., a network consisting of sequences of expected spectral template alternatives for each word. Performance evaluation of the word verification component (Table IV) shows that, depending on the threshold employed, up to half the hypothesized words can be rejected while falsely rejecting only 6% of the correct words. The ratings of

the verification component are such as to place the correct word in about fourth position, on the average, of the 50 or so words hypothesized to start at that position. This type of verification structure has great potential, but the Hearsay-II implementation does not presently account for very many phonetic and phonological interactions at word boundaries.

The observed scoring behavior of the verification component is not good enough to be used as an absolute ranking criterion when deciding, for example, whether to pursue one of two non-overlapping sentence fragments. Instead, scores at each position in a sentence are normalized with respect to the best matching word observed so far at that position (a strategy similar to that employed at BBN).

A second type of verification procedure is used to see if two words proposed to be adjacent are acoustically compatible with the segment label string. A special set of rules is employed to consider only the critical time interval, taking into account phonological processes occurring across word boundaries. During one run of sentence understanding, 7100 two-word hypotheses were processed by this component. One hundred and ninety five (i.e., 95% of the correct word pairs) were correctly accepted and 59% of the many decoy word pairs were correctly rejected.

D. Syntax, semantics, and control strategies

The syntax and semantics module uses the Cocke algorithm (Aho and Ullman, 1972) to list all possible words to the left or right of a high-scoring seed word. Verification scores are obtained for some or all of these word hypotheses, and the best-scoring word hypothesis is combined with the seed word to form a partial phrase. Many partial phrase theories are considered in parallel, and the parser is capable of saying whether adjacent partial phrases can be combined syntactically. All semantic knowledge contained in the system is presently embedded in the syntactic rules.

The control strategy is to expand larger sentence fragments first (depth first) because such fragments are not expected to occur randomly. However, since the correct word rarely has the best verification score, it is necessary to pursue many paths before a complete sentence is obtained. The first sentence found is not necessarily the best scoring possible sentence, so the recognition process continues until all remaining sentence fragments have a lower score.

The Hearsay-II system design with its blackboard concept permits a good deal of potential parallelism. For example, the components of the system might be implemented on a set of smaller computers connected in parallel (Bell *et al.*, 1973). Also, since each knowledge source is independent, taking its input from the blackboard and placing its output on the blackboard, knowledge sources can be changed fairly easily. The demonstrated advantage of this system design approach is that six months before the end of the project, the performance of Hearsay-II was such that an intensive effort was made to redesign nearly every component in the system, and even eliminate some of the components that provided little help. This resulted in a significant improvement in performance.

The Hearsay-II system ran in about 250 times real time on a PDP-KA10 (0.4 MIPS) computer. The final performance of Hearsay-II (Table II) is encouraging, although the evaluation was restricted to a small number of sentences from a single talker. Of interest is the fact that only 77% of the sentences were correctly *recognized* (all words correct) while 91% were *understood* correctly. It appears that the CMU grammar contains some desirable characteristics for the realization of computer understanding.

APPENDIX D. DETAILED DESCRIPTION OF THE CMU HARPY SYSTEM

A. Acoustic analysis

The preliminary acoustic analysis performed in Harpy is of interest because it obviates the need for sophisticated acoustic-phonetic decoding rules. A speech waveform is low-pass filtered at 5 kHz, digitized at 10 000 samples per second, and 14 linear prediction coefficients are computed every 10 msec. 10-ms frames are then grouped together into "acoustic segments" if sufficiently similar. Working from left to right, the control component takes the parametric representation for a frame and compares it with the first frame and the middle frame of the current segment. A new segment is proposed if either distance exceeds a threshold.

The linear prediction coefficients used to represent the combined segment are obtained from the sum of the autocorrelation coefficients of all the frames in the segment. On the average, segments consist of about three 10-ms frames, and there are thus two to three acoustic segments for every phonetic segment in a word. The process of grouping together 10-ms frames reduces the number of matches with the network, decreases the recognition time, and is intended to smooth out potential noise in the label sequences that are obtained from the linear prediction spectra. Each segment is characterized by a 14-pole spectrum, and the distances between this spectrum and a set of 98 spectral templates are computed using a minimum residual error measure (Itakura, 1975).

The system is sensitive to missing segments (being constrained not to skip a state in the network), but redundant sequences of labels are easily handled by including an implicit output path that returns to the same state for each state in the network. Therefore the segmenter is biased to produce extra segments rather than to miss an expected segment. The advantage of this simple strategy is that time normalization of the input stream is not required in the template matching process. Little phonetic constraint seems to be lost by allowing each state to grab as many acoustic segments as it can (subject to certain minimum and maximum times for remaining in particular states).

B. Template selection

The spectral template inventory was developed by starting with a single template for each phoneme (two

for plosives, affricates, and diphthongs) and expanding the number of templates for cases where this simple approximation lead to recognition errors. Experimenter judgment was required to decide when to change a lexical representation using other available templates, when to add a word boundary rule, and when it was absolutely necessary to add to the template inventory. After processing over 700 sentences from a single talker, the number of templates reached 98, of which almost all of the additions helped describe voiced portions of consonant-vowel transitions.

C. Network generation

The Harpy grammar consists of a set of BNF statements that generate a finite list of acceptable sentences, where most sentences contain a word from a large class of "authors" and/or a large class of "topics." The lexical representation for each word consists of a state transition network of expected alternative sequences of spectral states, with duration limits given in parentheses for some of the states. A modest number of word-boundary rewrite rules are specified, but the coarticulatory phenomena covered are surprisingly few in number, and the restricted notation could not handle, for example, the {S T #S} phonological deletion example given earlier (unless the [s t] spectral sequence were replaced by a special dummy symbol). All this information is transformed into a 15 000 state network by a set of procedures that collapse common representations where possible, while assuring an ability to backtrack through the network once a terminal node is reached in order to determine what word sequence was spoken.

D. System tuning and talker normalization

The templates used in the segment labeler are tuned to each new talker, but the lexical representations stay the same for all talkers of a common dialect (it is acknowledged that a few phonetic alternatives may have to be added to improve performance for some individuals). Templates for a new talker are obtained by attempting to recognize a set of about 20 training sentences using someone else's template patterns. The system automatically generates a new set of template patterns from the label sequences used in a forced correct path through the network. For a 1000 word lexicon, it takes about a half hour to record and process the training sentences. This is perhaps more than intended by the ARPA steering committee specifications, but there is no doubt that the procedure contributes significantly to the very good performance of the system. Recent changes permit the system to adapt to a new talker dynamically by starting with a talker-independent set of templates and modifying their properties based on correctly understood input utterances (Lowerre, 1977) in a manner similar to that used in Hearsay I (Reddy, Erman, and Neely, 1973).

The Harpy system uses about 30 MIPSS (millions of instructions executed per second of speech). This is less than the computing time in the other systems, but it is far from real-time operation on the computer used (0.4 MIPS PDP-KA10). However, it is well within the target specifications of real-time response on a 100-MIPS future machine. Harpy also requires a large program space to store and manipulate its 15 000 state network representation of speech knowledge. The present program barely fits into 200K of 36-bit words, and it would probably not fit if the average branching factor of the grammar were increased.

[1]Members of the ARPA steering committee were F. S. Cooper, J. W. Forgie, C. C. Green, D. H. Klatt, J. C. R. Licklider (Ex-chairman), M. F. Medress (Acting Chairman), E. P. Newburg, A. Newell (ex-chairman), M. H. O'Malley, D. R. Reddy, B. Ritea, J. E. Shoup, D. E. Walker, and W. A. Woods.

[2]The original 5 system builders were a group at Bolt Beranek and Newman Inc. (BBN) headed by W. A. Woods, a group at Carnegie-Mellon University (CMU) headed by D. R. Reddy, a group at Lincoln Laboratories (LL) headed by J. Forgie, a group at Stanford Research Institute (SRI) headed by D. Walker, and a group at System Development Corporation (SDC) headed by B. Ritea. After two years, funding was concentrated on three main system builders. Smaller supporting research efforts were funded at Haskins Laboratories (F. S. Cooper), at Speech Communications Research Laboratory (J. E. Shoup), at Univac (M. F. Medress), and at Univ. California at Berkeley (M. H. O'Malley).

[3]In addition to serving on the ARPA advisory steering committee, the author also worked as a consultant during the development of the BBN system.

[4]No claim is made for the strict originality of these techniques. Even in the context of speech understanding systems, some of the ideas were preceded by or developed in parallel with research taking place elsewhere. For example, a major speech recognition effort was initiated at IBM about the same time as the ARPA project began (Bahl *et al.*, 1976; Jelenek, 1976). See Reddy (1976) for an excellent comparative review of progress in the field up to early 1976.

Aho, A. V., and Ullman, J. D. **(1972)**. *Theory of Parsing, Translation and Compiling* (Prentice-Hall, Englewood Cliffs, NJ).

Bahl, L., Baker, J., Cohen, P., Dixon, N., Jelenek, F., Mercer, R., and Silverman, H. **(1976)**. "Preliminary results on the Performance of a System for the Automatic Recognition of Continuous Speech," pp. 425–429 in Teacher (1976).

Baker, J. **(1975)**. "The Dragon System—An Overview," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 24–29.

Barnett, J. A. **(1974)**. "A Phonological Rule Compiler," pp. 188–192 in Erman (1974).

Bates, M. **(1975)**. "The Use of Syntax in a Speech Understanding System," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 112–117.

Bell, C., Chen, R., Fuller, S., Grason, J., Rege, S., and Siewiorek, D. **(1973)**. "The Architecture and Application of Computer Modules: A Set of Components for Digital Systems Design," Compcon 73, San Francisco, CA, pp. 177–180.

Bernstein, M. I. **(1976)**. "Interactive Systems Research: Final Report to the Director, Advanced Research Projects Agency," System Development Corporation, Santa Monica, CA, Report No. TM-5243/006/00.

Chomsky, N., and Halle, M. **(1968)**. *The Sound Pattern of English* (Harper and Row, New York).

Cohen, P. S., and Mercer, R. L. **(1975)**. "The Phonological Component of an Automatic Speech Recognition System," pp. 275–320 in Reddy (1975).

Cole, R. A., and Scott, B. **(1974)**. "Toward a Theory of Speech Perception," Psychol. Rev. **81**, 348–374.

Connolly, D. **(1975)**. "Minutes of the Speech Understanding Workshop," 81–82, Science Applications Inc., Arlington, VA.

Cook, C. C., and Schwartz, R. M. **(1977)**. "Advanced Acoustic Techniques in Automatic Speech Understanding," pp. 663–666 in Silverman (1977).

De Mori, R., Rivoira, S., and Serra, A. (1975). "A Speech Understanding System with Learning Capability," in *Proceedings of the 4th International Joint Conference on Artificial Intelligence* (Tbilisi, USSR).

Dixon, N. R., and Maxey, H. D. (1968). "Terminal Analog Speech Synthesis of Continuous Speech Using the Diphone Method of Segment Assembly," IEEE Trans. Aud. Electroacoust. AU-16, 40–50.

Erman, L. D., Ed. (1974). *Contributed Papers of the IEEE Symposium on Speech Recognition* (IEEE Catalog No. 74CH0878-9 AE).

Fant, G. (1970). "Automatic Recognition and Speech Research," Q. Prog. and Status Rep. QPSR-1, Speech Transmission Laboratories, KTH, Stockholm, Sweden, 16–31.

Fant, G., Ed. (1975). *Proceedings of the Stockholm Speech Communications Seminar* (Almqvist and Wiksell, Stockholm, and Wiley, New York).

Forgie, J. W., *et al.* (1974). *Speech Understanding Systems—Semiannual Technical Summary Report* (MIT Lincoln Laboratories, Lexington, MA).

Garrett, M. F. (1977). "Word and Sentence Processing," in *Handbook of Sensory Physiology, Volume* 8: *Perception*, edited by R. Held and X. Leibowitz (Springer-Verlag, Heidelberg).

Gillmann, R. A. (1975). "A Fast Frequency-Domain Pitch Algorithm," J. Acoust. Soc. Am. 58, S63(A).

Goldberg, H. G., and Reddy, R. (1976). "Feature Extraction, Segmentation and Labeling in the Harpy and Hearsay-II Systems," J. Acoust. Soc. Am. 60, S11(A).

Halle, M., and Stevens, K. N. (1962). "Speech Recognition: A Model and a Program for Research," IRE Trans. Inf. Theory IT-8, 155–159.

Haton, J.-P., and Pierrel, J.-M. (1976). "Organization and Operation of a Connected Speech Understanding System at Lexical, Syntactic and Semantic Levels," pp. 430–433 in Teacher (1976).

Hayes-Roth, F., and Lesser, V. R. (1976). "Focus of Attention in a Distributed-Logic Speech Understanding System," pp. 416–420 in Teacher (1976).

Herscher, M. B., and Cox, R. B. (1976). "Source Entry Using Voice Input," pp. 190–193 in Teacher (1976).

Hyde, S. R. (1972). "Automatic Speech Recognition: A Critical Survey and Discussion of the Literature," in *Human Communication: A Unified View*, edited by E. E. David and P. B. Denes (McGraw–Hill, New York).

Itakura, F. (1975). "Minimum Prediction Residual Principle Applied to Speech Recognition," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 67–72.

Jelenek, F. (1976). "Continuous Speech Recognition by Statistical Methods," Proc. IEEE 64, 532–556.

Kameny, I. (1975). "Comparison of Formant Spaces of Retroflexed and Nonretroflexed Vowels," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 38–49.

Kameny, I. (1976). "Automatic Acoustic-Phonetic Analysis of Vowels and Sonorants," pp. 166–169 in Teacher (1976).

Klatt, D. H. (1975). "Word Verification in a Speech Understanding System," pp. 321–341 in Reddy (1975).

Klatt, D. H. (1976a). "Structure of a Phonological Rule Component for a Synthesis-by-Rule Program," IEEE Trans. Acoust. Speech Signal Process. ASSP-24, 391–398.

Klatt, D. H. (1976b). "A Digital Filter Bank for Spectral Matching," pp. 537–540 in Teacher (1976).

Klatt, D. H. (1976). "Speech Perception: A Spectral-Sequence Decoding Network as a Model of Lexical Access" (in preparation).

Klatt, D. H., and Stevens, K. N. (1973). "On the Automatic Recognition of Continuous Speech: Implications of a Spectrogram-Reading Experiment," IEEE Trans. Audio Electroacoust AU-21, 210–217.

Klovstad, J. W. (1977). "Computer-Automated Speech Perception System," Ph.D. thesis (MIT).

Klovstad, J. W., and Mondshein, L. F. (1975). "The CASPERS Linguistic Analysis System," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 18–123.

Lea, W. A. (1968). "Establishing the Value of Voice Communication with Computers," IEEE Trans. Audio Electroacoust. AU-16, 184–197.

Lea, W. A., Medress, M. F., and Skinner, T. E. (1975). "A Prosodically Guided Speech Understanding System," IEEF Trans. Acoust. Speech Signal Process. ASSP-23, 30–38.

Lesser, V. R., Fennell, R. D., Erman, L. D., and Reddy, D. R. (1975). "Organization of the Hearsay-II Speech Understanding System," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 11–23.

Lesser, V. R., and Erman, L. D. (1977). "A Retrospective View of the Hearsay-II Architecture," IJCAI-77 (in press).

Liberman, A. M., Cooper, F. S., Harris, K. S., and MacNeilage, P. F. (1962). "A Motor Theory of Speech Perception," *Proceedings of the Speech Communication Seminar*, (Royal Institute of Technology, Stockholm), Paper D3.

Liberman, A. M., Cooper, F. S., Shankweiler, D. S., and Studdert-Kennedy, M. (1967). "Perception of the Speech Code," Psychol. Rev. 74, 431–461.

Lindgren, N. (1965). "Machine Recognition of Human Language," IEEE Spectrum 2, March, April, May.

Lowerre, B. T. (1976). "The Harpy Speech Recognition System," Ph.D. thesis (Department of Computer Science, Carnegie–Mellon University, Pittsburgh, PA.)

Lowerre, B. T. (1977). "Dynamic Speaker Adaptation in the Harpy Speech Recognition System," pp. 788–790 in Silverman (1977).

Martin, T. B. (1975). "Applications of Limited Vocabulary Recognition Systems," pp. 55–71 in Reddy (1975).

Martin, T. B. (1976). "Practical Applications of Voice Input to Machines," Proc. IEEE 64, 487–500.

Makhoul, J. (1975). "Linear Prediction: A Tutorial Review," Proc. IEEE 63, 561–480.

Medress, M. F., Cooper, F. S., Forgie, J. W., Green, C. C., Klatt, D. H., O'Malley, M. H., Newburg, E. P., Newell, A., Reddy, D. R., Ritia, B., Shoup-Hummel, J. E., Walker, D. E., and Woods, W. A. (1977). "Speech Understanding Systems: Report of a Steering Committee," Sigart Newsletter 62, 4–7 (April 1977); IEEE Trans. Prof. Commun. (1977) (in press).

Medress, M. F., Skinner, T. E., Kloker, D. R., Diller, T. C., and Lea, W. A. (1977). "A System for Recognition of Spoken Connected Word Sequences," pp. 468–473 in Silverman (1977).

Mermelstein, P. (1975a). "A Phonetic-Context Controlled Strategy for Segmentation and Phonetic Labeling of Speech," IEEE Trans. Acoust. Speech Signal Process. ASSP-23, 79–82.

Mermelstein, P. (1975b). "Automatic Segmentation of Speech into Syllable Units," J. Acoust. Soc. Am. 58, 880–883.

Molho, L. M. (1976). "Automatic Acoustic-Phonetic Analysis of Fricatives and Plosives," pp. 182–185 in Teacher (1976).

Newell, A., Barnett, J., Forgie, J. W., Green, C. C., Klatt, D. H., Licklider, J. C. R., Munson, J., Reddy, D. R., and Woods, W. A. (1973). *Speech Understanding Systems: Final Report of a Study Group* (North-Holland/American Elsevier, Amsterdam).

Newell, A., Cooper, F. S., Forgie, J. W., Green, C. C., Klatt, D. H., Medress, M. F., Newburg, E. P., O'Malley, M. H., Reddy, D. R., Ritea, B., Shoup-Hummel, J. E., Walker, D. E., and Woods, W. A. (1975). *Considerations for a Follow-on ARPA Research Program for Speech Understanding Systems* (August 1975). Available from Computer Science Department, Carnegie–Mellon University, Pittsburgh, PA 15213.

O'Malley, M. H. (1975). "The Children of Lunar: An Exercise in Comparative Grammar" (unpublished).

Ochsman, R. B., and Chapanis, A. (1974). "The Effects of

10 Communication Modes in the Behavior of Teams during Cooperative Problem Solving," Int. J. Man–Machine Stud. **6**, 579–619.

Oshika, B., Zue, V. W., Weeks, R. V., Nue, H., and Aurbach. J. **(1975)**. "The Role of Phonological Rules in Speech Understanding Research," IEEE Trans. Acoust. Speech Signal Process. ASSP-**23**, 104–112.

Paxton, W. H. **(1977)**. *A Framework for Speech Understanding*, Ph. D. dissertation (Stanford University).

Paxton, W. H. **(1976)**. "Experiments in Speech Understanding System Control," Artificial Intelligence Center Technical Note 134, Stanford Research Institute Project 4762, Menlo Park, CA.

Peterson, G., Wang, W., and Silvertsen, E. **(1958)**. "Segmentation Techniques in Speech Synthesis," J. Acoust. Soc. Am. **30**, 739–742.

Pierce, J. R. **(1969)**. "Whither Speech Recognition," J. Acoust. Soc. Am. **46**, 1049–1051.

Pisoni, D. **(1977)**. "Speech Perception," *Handbook of Learning and Cognitive Processes, Vol.* 5, edited by W. K. Estes (Erlbaum, New Jersey).

Rabiner, L. R., and Sambur, M. R. **(1975)**. "An Algorithm for Determining the Endpoints of Isolated Utterances," Bell Syst. Tech. J. **54**, 297–315.

Reddy, D. R. **(1975)**. *Speech Recognition: Invited Papers Presented at the* 1974 *IEEE Symposium* (Academic, New York).

Reddy, D. R. **(1976)**. "Speech Recognition by Machine: A Review," Proc. IEEE **64**, 501–531.

Reddy, D. R., Erman, L. D., and Neely, R. B. **(1973)**. "A Model and a System for Machine Recognition of Speech," IEEE Trans. **AU-21**, 229–238.

Reddy, D. R., *et al.* **(1977)**. *Speech Understanding Systems Final Report* (Computer Science Department, Carnegie–Mellon University).

Ritea, B. **(1975)**. "Automatic Speech Understanding Systems," *Proceedings of the* 11*th IEEE Computer Society Conference*, Washington, DC, pp. 319–322.

Sakai, T., and Nakagawa, S. **(1975)**. "Continuous Speech Understanding System LITHAN," Technical Report, Department of Information Science, Kyoto University, Kyoto, Japan.

Schwartz, R. M., and Cook, C. C. **(1977)**. "Advanced Acoustic Techniques in Automatic Speech Understanding," (IEEE Catalog No. 663.666).

Shockey, L., and Reddy, D. R. **(1975)**. "Quantitative Analysis of Speech Perception," in Fant (1975).

Silverman, H. F. (Chairman) **(1977)**. "Conference Record of the 1977 IEEE International Conference on Acoustics, Speech and Signal Processing," Hartford, 9–11 May (IEEE Catalog No. 77CH1197-3 ASSP).

Silverman, H. F., and Dixon, N. R. **(1976)**. "The 1976 Modular Acoustic Processor (MAP): Diadic Segment Classification and Final Phoneme String Estimation," pp. 15–20 in Teacher (1976).

Smith, A. R. **(1976)**. "Word Hypothesization in the Hearsay-II Speech Understanding System," pp. 549–552 in Teacher (1976).

Stevens, K. N. **(1972a)**. "The Quantal Nature of Speech: Evidence from Articulatory-Acoustic Data," in *Human Communication: A Unified View*, edited by E. E. David and P. B. Denes.

Stevens, K. N. **(1972b)**. "Segments, Features, and Analysis by Synthesis," in *Language by Eye and by Ear*, edited by J. F. Kavenaugh and I. G. Mattingly (MIT, Cambridge, MA), pp. 47–52.

Studdert-Kennedy, M. (1975). "Speech Perception," in *Contempory Issues in Experimental Phonetics*, edited by N. J. Lass.

Tappert, C. C. **(1975)**. "Experiments with a Tree-Search Method for Converting Noisy Phonetic Representation into Standard Orthography," IEEE Trans. Acoust. Speech Signal Process. ASSP-**23**, 129–135.

Tappert, C. C., Dixon, N. R., and Rabinowitz, A. S. **(1973)**. "Application of Sequential Decoding for Converting Phonetic to Graphic Representation in Automatic Speech Recognition of Continuous Speech (ARCS)," IEEE Trans. Audio Electroacoust. **AU-21**, 225–228.

Teacher, C. (Chairman) **(1976)**. *Conference Record of the* 1976 *IEEE International Conference on Acoustics Speech and Signal Processing*, Philadelphia, PA, 12–14 April (IEEE Catalog No. 76CH1067-8 ASSP).

Walker, D. E. **(1975)**. "The SRI Speech Understanding System," IEEE Transl. Acoust. Speech Signal Process. ASSP-**23**, 397–416.

Walker, D. E. (Ed.) **(1976)**. "Speech Understanding Research: Final Technical Report," Stanford Research Institute, Menlo, Park, CA.

Wanner, E., and Maratsos, M. **(1977)**. "An ATN Approach to Comprehension," in *Linguistic Theory and Psychological Reality*, edited by J. Bresnan and M. Halle (MIT, Cambridge, MA).

Watt, W. C. **(1968)**. "Habitability," Am. Doc. **19**, 338–351.

Weeks, R. V. **(1974)**. "Predictive Syllable Mapping in a Continuous Speech Understanding System," pp. 154–158 in Erman (1974).

Weinstein, C. J., McCandless, S. S., Mondshein, L. F., and Zue, V. W. **(1975)**. "A System for Acoustic-Phonetic Analysis of Continuous Speech," IEEE Trans. Acoust. Speech Signal Process. ASSP-**23**, 54–67.

Weisen, R. A., and Forgie, J. W. **(1974)**. "An Evaluation of the Lincoln Laboratory Speech Recognition System," J. Acoust. Soc. Am. **56**, S27(A).

Wolf, J. J. **(1976)**. "Speech Recognition and Understanding," in *Digital Pattern Recognition*, edited by K. S. Fu (Springer-Verlag, Berlin).

Woods, W. A. **(1970)**. "Transition Network Grammars for Natural Language Analysis," Commun. Assoc. Comput. Mach. **13**, 591–602.

Woods, W. A., and Zue, V. **(1976)**. "Dictionary Expansion via Phonological Rules for a Speech Understanding System," 561–564 in Teacher (1976).

Woods, W., Bates, M., Brown, G., Bruce, B., Cook, C., Klovstad, J., Makhoul, J., Nash-Webber, B., Schwartz, R., Wolf, J., and Zue, V. **(1976)**. "Speech Understanding Systems: Final Technical Progress Report," Bolt Beranek and Newman, Inc. Report No. 3438, Cambridge, MA (in 5 volumes).