WORKING PAPERS IN SPEECH RECOGNITION
- I -

R. Reddy, L. Erman, R. Neely, et al.

Computer Science Department
Carnegie -Mellon University
April 21, 1972

*This research was supported in part by the Advanced Research Projects Agency of the Department of Defense under contract no. F44620-70-C-0107 and monitored by Air Force Office of Scientific Research.

ABSTRACT

This report represents a collection of papers published in various conference proceedings that are not readily available for researchers working in the field of speech recognition. The papers reprinted are:

1. Reddy -- Speech Input Terminals (June 1970).
2. Reddy, Erman, and Neely -- The CMU Speech Recognition Project (October 1970).
3. Erman and Reddy -- Telephone Speech (August 1971).
4. Neely and Reddy -- Noise in Speech (August 1971).
5. Reddy -- Speech Recognition: Prospects (August 1971).
6. Reddy, Bell, and Wulf -- Speech Recognition in a Multiprocessor Environment (December 1971).
7. Reddy, Erman, and Neely -- A Mechanistic Model of Speech (April 1972).

The authors would like to thank Allen Newell who has read most of the papers appearing in this report and made several valuable comments on them. We would like to express our appreciation to Bunny Kostkas and Heather Shoub for their typing, editing, and modification of the manuscript using the PDP-10 XCRIBL Document generation system.

Reprinted from:

Reddy, D.R., Speech Input Terminals for Computers: Problems and Prospects, IEEE Computer Conf., Wash. (June 1970).

SPEECH INPUT TERMINALS FOR COMPUTERS: PROBLEMS AND PROSPECTS

D. R. Reddy

Computer Science Department
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

INTRODUCTION

It is not surprising that many scientists and engineers should have thought about building machines that could understand speech. What is surprising is how many of them grossly underestimated the effort required to build a non-trivial speech recognition system. In spite of many years of research, not only are we not in a position to provide speech input terminals for computers but we cannot even answer satisfactorily a few essential questions about them. What type of a speech input terminal can we expect to have within the next decade? Will it be usable, i.e., reliable, accurate, fast, etc.? Who would want to use it? What would it cost? The answer to these, and other similar questions, is 'we are not sure'. What is more, there is no clear plan at present to obtain reliable answers to these questions. All that can be done is to see what has been accomplished and what problems remain to be solved before we can begin to answer these questions. To this end, this paper presents an evaluation of the state of the art, describes the structure of a real time speech recognition system presently working in a time-sharing environment, and discusses several unsolved problems which must be solved, at least partially, before we can expect to have speech input terminals for computers.

Why do we need speech terminals? We seem to be doing fairly well with cards, tapes, keyboards, and CRTs (Pierce, 1969). Why waste our resources on this area, especially when it looks like no speech input terminal we can hope to build in the foreseeable future is likely to converse in spoken English with the facility of a native speaker in a noisy environment? Such a comparison would not be entirely relevant since we don't now use English, or noisy telephone lines, for man computer communication with the exisiting I-O devices.

A more appropriate question would be whether there exist situations where a speech input terminal is needed and where the presently available devices are not satisfactory*. There are several simple tasks which are worthwhile and can be done using a limited vocabularly word recognizer. The main problem here is that the people who are not in favor of speech recognition research claim, and rightfully so, that "anything" you can do with a limited vocabulary recognizer you can do with a specially designed box of function keys.

Clearly, if we had a computer system which can do half as decent a job of recognizing human speech as other human beings can, and do it economically, speech will eventually replace cards, paper tape and even keyboards for communication with computers. But we are not likely to perform speech recognition economically for some time to come. Thus it is necessary for us to look for tasks where the economics are only secondary to the problem of getting the task done. It seems to be rather difficult to come up with a task domain in which speech recognition systems can play a useful role and where the cost incurred is justifiable.

We suggest a few task domains that come to mind.

a. Applications that need human control of large numbers of devices where their hands and feet alone are not sufficient, e.g., aircraft and spacecraft control.

b. Applications where one can only afford an inexpensive input device like a telephone for communication with the computer, e.g., computer conducted polls and referendums.

c. Applications where the sophisticated control provided by the computer is necessary, but the human being in the loop is not able to key-in the necessary data fast enough to keep up with the rapidly changing situation, e.g., air traffic control problems.

----

* See Lea (1968) for an optimistic viewpoint on the value of voice communications with computers.

d. Scientific problems such as automatic protocol analyses which are used to model human problem solving behavior. Here we have a limited task domain in which free-flowing English is used by the human being to describe his problem solving behavior permitting us to construct the necessary semantic model, namely the problem behavior graph, which can then be used to predict what the speaker is likely to say next.

Speech terminals are not likely to replace other input-output terminals in the foreseeable future but are likely to be invaluable in a few specialized application areas just as the graphics terminal have become in computer-aided design problems.

The analogy with graphics terminal is worth pursuing. Although CRTs had been available even before the emergence of the computer, it was not until their use in computer-aided design they had captured the imagination of the computer industry. Since then the use of video-graphic terminals continues to increase each year in many new and unanticipated directions. Once an easily correctable, if not highly accurate, speech terminal becomes available it appears possible that it will be used in many presently unanticipatable ways. This seems inevitable, if for no other reason than that speech is the universal and natural mode of communication. On the other hand, the ultimate acceptance of speech input terminals in day-to-day use is likely to be limited by the cost.

The cost per bit is likely to be much higher for speech input terminals than for discrete devices like a keyboard for the following obvious reasons. Firstly, some processing will be required to discretize the continuous speech signal. Secondly, since the resulting input string (of pseudophones) can never be guaranteed to be error-free, the string interpreters in the monitors and compilers will have to have better facilities for error detection and correction. The cost of performing these functions will require non-trivial amounts of processor time in contrast to present day I-O devices. However, if the present trend continues and the processors become less and less expensive while the cost of mechanical I-O devices remains steady, the cost of a speech terminal may not be as exorbitant as it might seem at a first glance.

The complexity of present speech processing algorithms indicates that a speech terminal is likely to be another peripheral digital processor rather than a hard-wired device. The cost of this peripheral processor will depend on the performance expected of it. To be more specific, the cost per bit, while using a speech terminal, may be reduced by relaxing any of the following performance measures, e.g., accuracy, response time, size and structure of the vocabulary, and size and structure of the language as illustrated by the following remarks.

1. Cost can be reduced by lowering the expected accuracy. Computational effort appears to grow exponentially with the required accuracy. Our experience indicates that almost any approach to speech recognition can expect to achieve 80% accuracy. Twice as much computational and research effort seems to be required to increase the accuracy from 80% to 90%, twice as much again to go from 90% to 95%, and so on.

2. In applications where response time is not critical the cost can be reduced by using a less expensive processor.

3. Larger vocabularies will require more memory to store the lexicon of acoustic descriptions and correspondingly more time to search the lexicon.

4. Discrimination among phonetically similar words ("spit", "split", "slit") requires substantially more computational effort than between phonetically dissimilar words. Thus, the cost can be reduced by carefully choosing the vocabulary. This might occasionally require going to the extreme of coining new words in the language.

5. Phonemic ambiguity among words can often be resolved at a higher level if two similar words do not occur in the same syntactic or contextual position. Thus by suitably modifying the structure or the complexity of language one can reduce the cost.

These considerations indicate how systems can be tailored to suit the needs of any specific application and are also useful for evaluating the effectiveness of many different approaches to speech recognition.

The overriding factor governing the cost, usability, and availability of a speech terminal will be the progress we make in research over the next decade. If we attempt to extrapolate from our experience of the last two decades we find that the future is very bleak indeed. When one looks for the reasons for the slow rate of progress of the last two decades in speech recognition research, it becomes obvious that investigators have in general grossly underestimated the complexity of the problem. In the face of unexpected difficulties, many left the field after having traced the same uncertain ground without building on each other's results. Others chose to work on some peripheral undemanding problem where the criterion for success or failure is not as well defined. Some knowledgeable scientists, who might well have made the difference, chose to ignore the problem reasoning that recognition of spoken English with equal facility as a native speaker in a noisy environment seems far away. Lacking the long term intensive problem oriented research that any complex problem needs, progress has naturally been slow.

The slow progress in speech recognition has also been due to inadequate models. Attempts to force speech recognition into the simplistic mold of a "feature extraction-classification" paradigm of classical pattern recognition have met with partial success in the recognition of digits and other very small vocabularies (Talbert et al, 1963; King and Tunis, 1966). But with large vocabularies and connected speech this paradigm is either unusable or tends to become a brute-force technique. At the other extreme, models such as 'Analysis-by-Synthesis' (Stevens and Halle, 1964), have not progressed much beyond the proposal stage for a number of reasons, not the least of which is that synthesis has proved to be no easier than analysis.

Inadequate technology has also been responsible, in part, for the slow progress. Before the availability of appropriate computer systems, attempts by Fry and Denes (1959) and Sakai and Doshita (1963) to build speech recognition machines were abandoned after limited success. The main reason appears to be that hardware modification and checkout of a new idea often requires many man-months of effort and at the end one may have to un-modify the system since the attempt did not succeed. Even now, most speech research groups are limited to the use of small dedicated computers which make it difficult to experiment with complex models. When larger computer systems were used, (Bobrow and Klatt, 1968), the inability of the monitors to handle large data rate real-time requests has forced researchers to use limited, pre-recorded data sets thereby making it difficult to measure the performance of the system in a realistic situation.

The HEAR (Highly Efficient Audio Recognition) system being developed by the author and his colleagues at Carnegie-Mellon University does not suffer from some of the above disadvantages. This system uses a large time-shared PDP-10 computer with real-time facilites and is based on an "analysis-by-learning" model. The acoustic features required for comparison are abstracted from actual utterances and stored in a lexicon, thereby eliminating the need for the specification of an a priori model of speech production required by the "analysis-by-synthesis" approach. A presently working program, based on this model, was written by Vicens (1969) as part of his doctoral dissertation and is capable of close-to-realtime interaction.

THE HEAR SYSTEM

The main aim of the HEAR system is the recognition of connected speech of languages of about the same complexity as the present day computer languages, with an efficiency approaching the human perception of speech. This goal was chosen because it separates the problem of connected speech recognition from the problem of dealing with the idiosyncrasies of the English language and because this appears to be the most difficult subproblem in speech recognition which can be undertaken with some hope of achieving the goal in less than ten years. The requirement of connected speech was imposed because the system would be of very limited use if the speaker had to pause between words. It is assumed that languages like Fortran and Algol would be awkward for speaking to computers and that each user would specify his own language to suit the needs of his problem. The system is expected to handle vocabularies of around a thousand words without too much difficulty.

A critical requirement of the HEAR system is that it should equal human performance in at least a limited language situation. The time for recognition should be no more than the time for saying the utterance. Furthermore, most of the analysis is expected to take place concurrently while the command is being uttered so that the task requested may be performed immediately following the utterance. While a large number of approaches to speech recognition have been suggested, most of them seem to ignore the question of efficient recognition. The literature abounds with brute-force methods of questionable value.*

---

* The reader can verify the validity of this statement by applying the performance measures discussed earlier to the long list of references given in Pierce (1969).

The requirement of highly efficient recognition of connected speech of non-trivial vocabulary makes our approach significantly different in many ways from various short term attempts at speech recognition, as will be demonstrated in the rest of this section. Classical methods of pattern recognition, such as the use of a metric in a multidimensional space paritioned by hyperplanes, are not easily extendable for analysis of complex sequence of sounds which may be part of a spoken message. The structure of the message and the interrelationships among the sounds of the message are important factors. Even in those parts of the analysis where classification is required, such as the comparison of part of the utterance with the entries in a lexicon, what seems to be more important than classification is the selection of a few relevant candidates from the lexicon by heuristic and associative addressing techniques. Similar comments apply to analysis-by-synthesis approach. Clearly one does not want to synthesize acoustic representations of many different utterances each time in the hope that one of them will match the incoming signal.

Analysis by Learning

Since we have no satisfactory model of human speech perception, we have found it necessary to let the machine formulate its own model from a training set of words and sentences of the language to be recognized. This implies that we must provide the system with the necessary data structures and processes which are able to abstract the relevant information from the training set. In our present system many of the thresholds and heuristics which can conceivably be abstracted from the training set are 'built-in' as is the syntactic and contextual information about the language to be recognized. We expect that, at some later stage, the system will be able to modify its thresholds, heuristics, syntax, and contextual information based on past experience.

Learning in the present system is restricted to the construction of a lexicon of acoustic descriptions associated with concepts (or print names). Learning, in our limited context, is defined to be the process of modification of the data structure of the lexicon by a previously unknown acoustic structure. Recognition, then, is the comparison of the incoming acoustic description with the various entries in the lexicon. By organizing the lexicon in terms of the gross sequential structure of the utterance, comparison can be limited to only those entries in the lexicon that have similar structure. The similarity between the parameters of corresponding segments of the incoming utterance and an entry in the lexicon is measured in terms of the similarity score with the range of 0 to 100 (very dissimilar to identical). Recognition is defined to be the discovery of that entry in the lexicon which has the highest score exceeding a given threshold.

The analysis performed is the same whether the system is learning or recognizing. In both cases the system searches the lexicon to see if there exists an acoustic description in the lexicon that corresponds to the presently analyzed part of the incoming utterance. If the search fails or if the result is wrong, a new entry is made into the lexicon. Thus learning is treated as a special case of recognition. Of course it is also possible to direct the system to 'always learn' or 'only recognize'.

The HEAR system does not attempt to model human learning of speech. The basic structure of the lexicon is much more analogous to rote-learning. However, by making it possible to associate several names to various parts of the same acoustic description and vice versa, a much richer, though complex, memory structure is obtained than by a one-to-one mapping of names and acoustic descriptions.

Note that the learning mechanism proposed here is very different from the learning implied by the use of perception type of devices. We do not propose to connect a parameter extractor to a learning net and expect it to adapt itself. The emphasis here is on the development of a sophisticated data structure capable of acting as an associative net. Thus, once certain gross characteristics of the incoming utterance are known, they can be used to localize the search to some small subpart of the associative net. Also note that the input to the system is the labeled pseudophonemic segments and not the raw acoustic signal or the output of a bank of filters.

The HEAR system has at least five different phases: parameter extraction, segmentation, sound classification, sentence analysis and word boundary determination, and word recognition. These operations on the incoming utterance are not normally intended to be performed in sequence. We expect that they will act as a set of coroutines with feed back from higher levels guiding the search at lower levels. At each stage the system has to use phonological, syntactic and contextual constraints to reduce the search. The rest of this section discusses the problems and functional characteristics of various phases of the system. Those who wish to find the implementation details of the systems are referred to Reddy (1967), Reddy and Vicens (1968), and Vicens (1969).

Parameter Extraction

One as-yet-unresolved problem that has attracted more than its fair share of attention is the search for the so-called "acoustically invariant" parameters of speech (Lindgren, 1965). Although certain dominant features like formants were discovered, it was found that most of these dominant features could not be counted on to be present for every speaker, or even the same speaker in a slightly different context. It appears that, if we consider phones to be made up of bundles of features, the presence of a majority of these features is sufficient for the perception of the phone. So much so it sometimes happens that two completely disjoint bundles of features are perceived as the same phone by a human listener. Researchers who hope to discover the features relevant for analysis by synthesizing speech should beware. Just because they have succeeded in synthesizing, say, a single phoneme /k/ they should not expect to find the same set of features in every phoneme /k/. These considerations have led us to abandon the search for acoustically invariant features of speech and build a system that is not critically dependent on the presence or absence of any single feature.

The other main as-yet-unresolved problem is "what type of analysis should we use to extract the parameters?". After many years of experiments with zero-crossing analysis, spectral analysis, formant analysis, polynomial fitting, etc., our somewhat surprising conjecture is that "it does not matter". The main problem is that in day-to-day speech the acoustic signal provides only part of the information. The rest is provided by the listener's own contextual and extralinguistic abilities. No amount of sophisticated analysis can discover features that are not present in the original signal. So much so it seems irrelevant to fret about determining a frequency very accurately when that component might very well be absent the next time. It is our conjecture that, for most recognition tasks, it does not matter what type of analysis is used as long as the results of the analysis are consistent. The syntactic and contextual information is usually sufficient to resolve ambiguities arising from the variability in parameters. When the language to be recognized becomes more complex, such as two phonemically ambiguous words occurring in the same syntactic position, a careful look at the acoustic signal might be needed. Useful but less dependable features are extracted in our system only when they are absolutely required.

At present we use as parameters zero crossing and amplitudes in six frequency bands in the audible frequency range sampled every 10 miliseconds. We have found this to be a reasonable compromise between high data rate 40 channel filter bank and low data rate amplitude and zero crossing measurements of the original signal.

Segmentation

Figure 1 illustrates the machine segmentation of the utterance "How now brown cow". Note that the diphthong /au/ takes on different shapes in different contexts, illustrating one of the reasons why consistent segmentation is difficult to achieve.

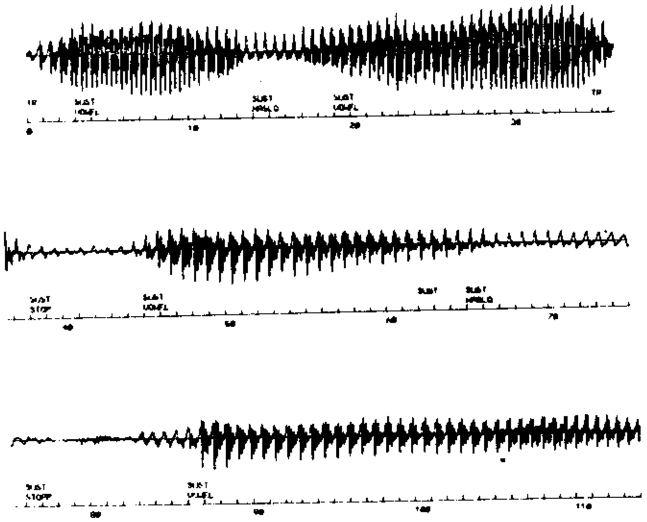


Figure 1.
The many faces of /au/ in "How now brown cow".

Another instance of "How now brown cow" might result in a different number of segments even when it is uttered by the same speaker in the same environment. These difficulties in obtaining consistent segmentation have led many investigators to look for approaches to recognition which do not require segmentation of the acoustic continuum into discrete parts. The analysis-by-synthesis approach is one such. Segmentation-free recognition has so far proved to be usable only in very small limited vocabulary situations. It is usually time-consuming because it does not lend itself to techniques for reduction of the search space.

That we need segmentation for the analysis of connected speech is obvious. The question is what type of segmentation should we use: phonemic, syllabic or morphemic segmentation? Present linguistic definitions are usually subjective and almost impossible to specify in an algorithmic form suitable for use by a computer. Many investigators just locate well defined boundaries such as unvoiced fricatives and stops (Hughes and Hemdal, 1965; Denes and von Keller, 1968). The main disadvantage with this approach is that sentences such as 'How are you' would then have to be treated as a single segment, thereby complicating subsequent analyses. In our system we find we need all the three concepts of phoneme, syllable, and morpheme. Hence we have defined the concepts of pseudo-phoneme (a collection of adjacent 10 ms. segments with similar acoustic characteristics) and pseudo-syllable (a collection of phonemic segments containing one and only one local maximum of amplitude) to be suitable for machine segmentation. A hierarchical segmentation procedure for obtaining pseudo-phonemic segmentation is given by Reddy and Vicens (1968).

Segment Classification

Classification of segments into phoneme-like categories, while unreliable because of the variability of parameters discussed earlier, is often useful for the generation of an ordered candidate list by associative addressing techniques. The difficulties in obtaining reliable phonemic classification has led us to generate segment descriptions in terms of 10 to 15 supra-phonemic categories. Such classification, while not in itself complete, is useful in describing the gross characteristics of a segment. This description can be used in minimizing the search space in word boundary analysis and word recognition.

Word Boundary Analysis

Determination of word boundaries in connected speech is by far the least understood of all the problems. The apparent preoccupation of most investigators with the acoustically invariant parameters of speech has been responsible for the lack of progress in the subsequent problem areas. Our own limited investigations show that this is likly to be the main bottleneck in the analysis of connected speech. Two main sources of the difficulties are the substantial modification of acoustic characteristics of a word in various contexts and the word boundary ambiguity problem, e.g., "ice cream", vs. "I scream". We are presently using a temporary expedient which requires careful selection of the syntax and the vocabulary of the language so as to minimze these difficulties.

Word Recognition

A main problem in word recognition is the correction of segmentation errors. When two utterances of the same phrase by the same speaker results in a different number of segments, the question arises as to which segment corresponds with which, so that proper comparison can take place. This is known as the segment synchronization problem in speech recognition (Reddy, 1969). It is similar to sequential decoding in the presence of errors in communication theory. In our system a mapping procedure determines the correspondences between segmental descriptions. Dominant segments corresponding to vowels and fricatives are mapped first. The remaining segments are then mapped on the basis of similarity of parameters.

Another problem in word recognition is the formulation and specification of various heuristics to reduce the time per recognition. In various word recognition experiments we found that as the vocabulary increases from 50 to 500 words, the time spent in searching the lexicon increases from 50% to 90% of the total time for recognition even though the search procedure uses several heuristics for the reduction of search space. This fact reiterates ourecordedearlier comment that there exist many other important problems in speech recognition research besides feature extraction and classification. Word recognition, then, requires the development of efficient procedures for the search of the lexicon, and these become critically important as the vocabulary increases. The following heuristics, used in our system, illustrate the type of devices that are helpful in reducing the search.

1. The data representation in the lexicon is arranged so that only those entries of the lexicon which contain the same number of syllables and unvoiced fricatives as the incoming utterance are considered first.

2. The search is terminated when a candidate obtains a high enough score, say 95% similarily.

3. If a candidate has a different global structure, i.e., if the sequential similarity constraint is not satisfied, the candidate is rejected without any further processing.

4. The candidates are ordered so that candidates with similar vowel structure are considered first.

5. If the stressed vowel is significantly different, then the candidate is rejected.

6. If a candidate obtains a high score but not enough to terminate further search, the candidate list is re-ordered so that all other phonemically similar candidates are considered first.

7. If no candidate in the initial list obtains a high enough score, say > 80%, then an attempt is made to transform the incoming description to correct for possible errors.

The word recognition system developed by Vicens (1969) obtains 92% to 98% accuracy for a single speaker depending on the noise and amount of learning. For multiple speakers the accuracy is around 80% to 85%. For a given accuracy the recognition time is a function of the size of the vocabulary and varies from 6 times real time for a 50 word vocabulary to 30 times real time for a 500 word vocabulary on a PDP-10 system. We seem to be a factor of 50 away from our goal of real time recognition for a 1000 word vocabulary.

UNSOLVED PROBLEMS

Of the main unsolved scientific and engineering problems in speech research I shall restrict myself here to those problems that are likely to be critical once the speech input terminals leave the laboratory environment and the tender loving care and protection of their creators. These concern the variables governing the characteristics of the speakers, the terminal, distance between the speaker and receiver, noise, etc.

In speech, characteristics of an utterance vary not only from speaker to speaker depending on the sex, age and physical condition, but even for the same speaker depending on his emotional state at the time of utterance. In our experiments we found that utterances of a speaker irritated by the mistakes of the machines are anything but normal. Speaker variability is at present minimized in our system by training for individual speakers and by requiring the speaker to be cooperative. The main limitation of this constraint is that every speaker must train the system before he can use it reliably. Mr. Erman of our group is attempting to formulate techniques for speaker normalization. Determination of differences and similarities of the characteristics of various speakers is one of the unsolved problems that is likely to require many man years of sustained effort.

The human ear can detect sounds between 50 to 20,000 Hz of sound intensities within the 0 to 110 decibel range with the smallest detectable sound resolution of 10 to 20 ms (/l/ as in "slit" as opposed to "sit"). Most voice input systems to computers have a 100 to 5000 Hz frequency response with a 40 decibel range (approximately 8 bits of resolution) with a sound resolution of 10 ms. The lower frequency response results in an occasional confusion between fricatives /f/, /θ/, and /s/. While it is within the capabilities of the engineering technology to build a receiver of any desirable characteristics, there has not been much effort to determine the optimal characteristics of a receiver for speech terminals satisfying the conflicting requirements of low bit rate and wide dynamic range.

The distance between the source and the receiver is also likely to be a problem for speech terminals. A microphone held too close to the lips also records the expiration after the utterance giving the illusion of an extra sound, and when held too far results in the loss of resolution with an additional problem of lower signal to noise ratio. Reliable discrimination between speech and expiration is proving to be difficult (note that the /h/ sound in English is a special case of expiration). Development of the mini-microphones which can be attached to a collar might minimize this variability.

By far the most difficult problem for speech input terminals is likely to be the discrimination between the speech source and various noise sources. While some of the attempts at noise reduction (such as the design of directional microphones) are acceptable, others (such as putting the speaker in a noise-free acoustic chamber) would not be acceptable for use with speech terminals. Some of the sources of external noise for speech terminals are: air conditioning and fan noise, printer noise, other speakers in the room, 'Hmms' and 'Hahs' and clearing of the throat of the speaker himself, etc. There is no simple unique solution to all these problems. Software solutions to these problems are likely to be difficult and time consuming, and are not compatible with the less than real time recognition requirement for speech input terminals. Social solutions, such as that no one may sneeze or cough in the speech terminal room are not likely to work either. Thus it is imperative that speech terminal designers design their system so that an occasional error cannot be catastrophic. Further, it should be possible to correct the system with minimum of effort. One possible solution is to couple the speech input terminal with a CRT for error detection and correction. In a real-time environment the commands would appear on the CRT as they are uttered by the speaker permitting him to immediately verify and correct the command in case of an error.

Unlike other I/O devices, the initiation and termination of I/O for speech terminals is data dependent. There are some devices in the market for the detection of the presence or absence of speech. However, since these are amplitude activated, they are noise sensitive and cannot yet be activated by low amplitude sounds, such as stops and fricatives. Development of a more sophisticated device will be necessary to minimize unnecessary interrupt processing by the computer.

CONCLUDING REMARKS

In this paper I have tried to outline a number of factors affecting the cost, utility, structure, and engineering of speech input terminals for computers. In particular, it is not enough to just measure the accuracy of a proposed algorithm, but one must consider all the relevant factors, e.g., accuracy, response time, vocabulary size, complexity of words, and complexity of language. All of these will affect the cost, utility and structure of a speech terminal.

Seymour Pappert of M.I.T once said, while commenting on the disappointing rate of progress in robotics research, that if we were to think that building a robot requires any less effort than putting a man on Mars, we would be sadly mistaken. Since we can produce people much less expensively it is very unlikely, at this time of shifting national priorities, that the billions of dollars in funding required for the research, development, and engineering of a robot will be forthcoming. Speech perception, being a difficult part of robotics research, is likely to fare no better. In view of the limited resources available for this type of research, it is essential that we avoid duplication of research, choose research goals that are likely to be of lasting value, avoid working on inconsequential peripheral problems, and develop a close cooperation between various interested research groups.

ACKNOWLEDGEMENTS

This speech research project was initiated by the author at Stanford University in 1964, following the lead and inspiration provided by Dr. John McCarthy. Since then, several graduate students have made significant contributions to the project, both at Stanford and Carnegie-Mellon Universities. Particular mention should be made of Pierre J. Vicens who wrote our first working real time work recognition program and Dr. Morton Astrahan, Lee Erman, Gary Goodman and Richard Neely who are developing interesting techniques for machine learning, speaker normalization, language design and conversational systems, respectvely. The author is indebted to Dr. Allen Newell for many thoughtful comments on a preliminary version of this paper.

REFERENCES

[1] Bobrow, D.G. and D.H.Klatt (1968), "A Limited Speech Recognition System", Proc. FJCC '68, 305-318.

[2] Denes, P.B. and T.G.von Keller (1968), "Articulatory Segmentation for Automatic Recognition of Speech", Proc. International Congress on Acoustics, Tokyo, 11, B143-B146.

[3] Fry, D.B. and P.B.Denes (1959), "The Design and Operation of a Mechanical Speech Recognizer", J. British IRE, 19, 211-229.

[4] Hughes, G.W. and J.F.Hemdal (1965), "Speech Analysis", Tech. Rept AFCRL-65-681, (P137552), Purdue University.

[5] King, J.H. and C.J.Tunis (1966), "Some Experiments in Spoken Word Recognition", IBM J. of R. and D., 10, 1, 65-79.

[6] Lea, W.A. (1968), "Establishing the Value of Voice Communication with Computers", IEEE Transactions on Audio and Electroacoustics, A-U-16, 2, 184-197.

[7] Lindgren, N. (1965), "Automatic Speech Recognition", IEEE Spectrum, 2, 3, 114-136.

[8] Pierce, J.R. (1969), "Whither Speech Recognition", J. Acoust, Soc. Am., 46, 4, 1049-1051.

[9] Reddy, D.R. (1967), "Computer Recognition of Connected Speech", J. Acoust. Soc. Am., 42. 2, 329-347.

[10] Reddy, D.R. and P.J.Vicens (1968), "A Procedure for Segmentation of Connected Speech", J. Audio Engr. Soc., 16, 4, 404-412.

[11] Reddy, D.R. (1969), "Segment Synchronization Problem in Speech Recognition", J. Acoust. Soc. Am., 46, 1, 1, 89 (abstract).

[12] Sakai, T. and S.Doshita (1963), "The Automatic Speech Recognition System for Conversational Sound", IEEE Trans., EC-12, 835-846.

[13] Stevens, K.N. and M.Halle (1964), "Remarks on Analysis by Synthesis and Distinctive Features", Proc. of Symposium on Models for the Perception of Speech and Visual Form, AFCRL, 1964, Ed. by W. Wathen-Dunn, MIT Press, 88-102.

[14] Talbert, L.R. et al. (1963), "A Real-Time Adaptive Speech Recognition System", Tech. Rept. No. 670-1 (ASD-TDR-63-660),

(P133441), Stanford Electronics Lab.

[15] Vicens, P.J. (1969), "Aspects of Speech Recognition by a Computer", Ph.D. Thesis, AI Memo No. 85, Computer Science Department, Stanford University.

Reprinted from:

Reddy, D.R., L.D.Erman, and R.B.Neely, The C-MU Speech Recognition Project, Proc. IEEE System Sciences and Cybernetics Conf., Pittsburgh (Dec. 1970).

THE CMU SPEECH RECOGNITION PROJECT

D. R. Reddy, L. D. Erman, R. B. Neely
Computer Science Department
Carnegie-Mellon University
Pittsburgh, Pa.

INTRODUCTION

Efforts at speech recognition in the past have ranged from recognition of a few isolated words to attempts at the recognition of spoken English in a noisy environment with the facility of a native speaker. While word recognition has been moderately successful, systems capable of understanding spoken English have never gotten past the model formulation stage. This is in part due to speech-independent linguistic problems, e.g., connected speech, multiple speakers, syntax analysis in the presence of errors, and so on. Most speech-dependent unsolved problems can be solved through the study of restricted spoken languages. This paper describes the CMU speech recognition system which is designed to be the main research tool for the study of these unsolved problems.

The term "speech recognition," not unlike the term "pattern recognition," has in the past been used to cover a wide range of problems, varying from the trivial problem of a yes/no recognizer to the presently unsolvable problem of recognition of spoken English. Even the presently accepted measure of speech recognition systems in terms of number of words and speakers that the system can handle can often be meaningless. A system capable of recognizing the ten words "ore, core, tore, pour, gore, door, bore, four, Thor, and more" would have to be much more sophisticated than a system for recognizing the digits. Accuracy figures can also be meaningless. A system which gives 90% accuracy in real time may in fact be superior to a system which gives 98% accuracy but takes 100 times longer to do so. In this paper, we use the term "speech recognition" to denote a system capable of recognizing connected speech utterances of an English-like language with restricted syntax and semantics, for a number of speakers with limited amount of training.

At present, there are no systems that are capable of understanding such restricted languages, let alone English. However, restricted language recognition permits one to bypass many as yet unresolved linguistic aspects of English so that one may concentrate on speech-related problems. This problem appears solvable within the next few years and seems to be a necessary intermediate step which will help us to study many unsolved problems in speech. Our current speech recognition project at Carnegie-Mellon University is devoted to building restricted language recognition systems.

This project is a continuation of our earlier work at Stanford University which resulted in a phonemic transcription system, a large vocabulary (500 words) isolated word recognition system, and a small vocabulary(16 words) highly restricted syntax connected speech system. Earlier attempts by Fry and Denes (1959), Sakai and Doshita (1963), Martin et al. (1964), Hughes and Hemdal (1965), Gold (1966), and Bobrow and Klatt (1968) are representative of some of the more significant achievements in speech recognition over the last two decades.

Why is speech recognition of interest? There is, of course, the desirability of developing another mode of man-machine communication, a mode which is natural, has a relatively high data rate, and does not require use of hands or feet. However, the main scientific reason for speech recognition research is that it provides a problem domain in which one can measure the effectiveness of various models, methodologies, and algorithms in a number of different areas. Models of speech production and perception are but some of these. In computer science, speech recognition research permits the study of techniques for reduction of search space, classification techniques, models for machine learning, associative addressing, computer structures, real-time systems and so on. In linguistics, competence and performance models can only be validated by studying their effectiveness in speech recognition.

A system capable of recognition of limited languages appears to be feasible at this time but is dependent on the satisfactory solution of several unsolved problems. An on-line system, in which the user can immediately verify success or failure of a recognition attempt, permits evaluation of the adequacy of the solutions to these unsolved problems. The CMU speech system described in a later section provides convenient facilities for such evaluation in a time-shared environment.

SOME UNSOLVED PROBLEMS

The Connected Speech Problem

The acoustic characteristics of phones and words exhibit great variability in different contexts. This variability is caused by differing anticipatory movements of the vocal tract in different contexts. This connected speech problem is well-known to speech scientists, but they do not know what to do about it. Most previous speech recognition attempts have ignored this problem by accepting only single words or short phrases in isolation and treating each of these utterances as a single unit.

The only successful attempt at connected speech recognition so far has been Vicens and Reddy's system for the analysis of commands for a computer controlled hand. Considering the difficulties they had, even with a restricted syntax and a 16 word vocabulary, in reliably detecting word boundaries (which in turn required constant tinkering with the vocabulary), this is likely to be a major obstacle in the way of a general speech recognition system.

There are very few cues in the acoustic data to indicate where word boundaries occur; therefore it would seem that they would have to be hypothesized in a feedback from higher-level parts of the recognition system. In order to test these hypotheses, then, a set of phonologically based synthesis rules could be used to operate on two (or more) entries in the lexicon and predict what the result would be if the lexicon entries were to occur adjacent to each other in speech.

The connected speech problem is further complicated by prosodic features which can have effects on the acoustic signal for time periods considerably longer than one or two words. The addition of prosodic features both adds supra-segmental variability and also contains information which is often necessary for correct understanding of the utterance. The primary prosodic features of amplitude, duration, and pitch have been used somewhat in recogntion systems. It is not yet known if there is other significant variability caused by prosodic features.

The Multiple Speaker Problem

Present solutions to the problems of recognizing speech emitted by several speakers require either multiple acoustic descriptions of the same word, the acceptance of lower accuracy, or often both, resulting in a 10-20% degradation in performance (e.g., accuracy going from 95% to 85% and computation time and lexicon size tripling). This performance is incompatible with our goals.

An ideal solution to this problem would have a new speaker initially utter a few sentences under the direction of the recognition system. From these controlled samples the system would abstract whatever parameters are needed to tune the recognition process for this particular speaker. After that, only these parameters, which describe this speaker's characteristics, would have to be remembered by the system.

Earlier research indicates that it is possible to define fairly simple speaker-dependent normalizations in the case of manually measured parameters for at least some aspects of speech. We have been so far unsuccessful in attempts to apply these techniques to a particular recognition system, but we believe these failures are caused by shortcomings in the recognition system. The errors in automatic segmentation and feature extraction make it difficult to identify and compensate for speaker variability; advances in these areas are necessary before more sophisticated speaker normalization can occur. Further, we believe that the multi-speaker problem -- the inter-speaker variations -- occur along the same dimensions as the intra-speaker variations; they are just greater.

Real-Time Performance

It is often said that artificial intelligence has an existence proof in the human being. For robotics this has an extra twist. In tasks such as chess and theorem proving the human has sufficient trouble himself so as to make reasonably crude programs of interest. But humans seem to perform effortlessly (and with only modest error) in visual or speech perception tasks that give machines the hiccups. This carries an implication: If and when we build speech-understanding systems, the human who uses these systems will be very demanding in terms of performance. Whether he will use a speech understanding system or not will be a function of the cost, accuracy, response-time, size and structure of the vocabulary and the size and structure of the language provided by the system. We believe that for a general system to be above threshold the following are appropriate requirements:

a. The system should cost no more than $1,000 per month.

b. The accuracy should not be less than 95%.

c. The system should usually be ready to respond to the speaker by the time he finishes saying the sentence.

d. The system should have at least a 10,000 word vocabulary.

e. The system should be capable of dealing with a non-trivial subset of English language.

If we build a system with the presently existing pieces the cost will be 20-100 times higher; the accuracy will be around 80-95% depending on all the other variables; the response time will be 10-20 times slower; size and structure of vocabulary and language are likely to be severely restricted by the space and speed limitations of the existing machines.

One thing is clear: We will have to re-engineer the existing pieces to achieve the required 10 to 100 times improvement in performance. Such improvements are not likely to be realized simply by speeding up the existing algorithms, but by developing more powerful heuristics to solve these problems. Since we do not know what these powerful heuristics are going to be, it is hard to predict when we might have a handle on the real-time performance problem.

Self-Analysis

One of the features of existing speech recognition systems, and probably of future ones as well, is the existence of error at every level of analysis and the consequent proliferation of heuristic devices throughout the system to control such error and permit recycling with improved definitions of the situation. Almost entirely missing from the literature, not only of speech recognition, but elsewhere in artificial intelligence as well, are techniques for evaluating performance characteristics of proposed algorithms and heuristics. By techniques we mean both suitable instrumentation and experimental design to measure accuracy, response time, cost, etc. in relation to vocabulary, language, and context. Until such techniques are developed and applied to existing components of a speech-understanding system, these components should be considered of questionable value in an applied system.

Speech Independent Linguistic Problems

Put bluntly, no one understands yet what it means to understand mechanistically. Thus we are not sure what the understanding component of a speech-understanding program should be. The models we have, i.e., existing programs that understand in some sense, are too partial and too lean to hang much confidence on. Certainly, it is true that as we gradually restrict the task domain to a narrower and narrower set of questions, we gradually re-enter the domain of specialized representations with particularistic programs written to generate answers. It would seem we could find tasks to be handled by such programs and representations, but it is not clear what we would gain from it.

Why should one have a system that combines speech and understanding? From a selective view it is conceivable that contributions could flow in either (or both) directions. However, until now there has been almost no work in how characteristics of speech (e.g., stress, intonation, paralinguistic aspects) might aid semantics. In the converse direction belief is certainly strong: whenever limits to recognition systems occur there is a tendency to see it as revealing the requirement for increasingly wider realms of context. Thus, semantics is to contribute directly to recognition. Although there are certainly plenty of good examples of higher context being applied to help recognition tasks, there is very little work that has been done for semantic context in this respect.

A particular difficulty that stands in the way of using semantics to help with speech recognition is the lack of grammaticality and general well-formedness in free speech. Although one may legislate against some of the difficulties in written language, it is harder to do so in spoken language. Not only do people "humm" and "hah", and clear their throats, they utter fragments: "Now the...th'...oh well...they are plying flames--I mean flying planes." We believe a whole set of new language analysis tools will have to be developed before we can expect to have sophisticated cooperation between speech and understanding components of a single system.

IMPLEMENTATION OF THE SPEECH SYSTEM

The CMU speech system is being implemented on a Digital Equipment Corporation (DEC) PDP-10 computer. This 36-bit machine, which has 112K of 1 and 1.8 micro-sec. core, 10 million words of disk file space, and 330K of swapping drum space, runs under the DEC 10/50 time sharing monitor and can support up to 30 or more users. More core and additional processors are planned for the future.

The Audio Machine

At CMU we are adding hardware and software support to the PDP-10 to handle several real-time audio input/output devices. We refer to these devices and their support as the audio machine. Research in an ill-understood area such as speech requires a great deal of experimentation; much work in the past has been painful or unattempted because of the difficulties involved with this experimentation. A major goal of the audio machine is to relieve the user of many of the real-time problems associated with speech input and output.

The most important hardware components of the audio machine are the analog to digital (and digital to analog) devices. The A/D converter produces 9-bit digital values of the audio signal at selected sampling frequencies from 200Hz to 20KHz (180,000 bits/sec., which is required for high quality speech input).

Our principle input device for speech recognition is a preprocessor which filters the audio signal into 6 bands and produces for each band a count of the zero-crossings and the log of the maximum peak-to-peak amplitude in each consecutive 10 msec. sampling period. (Fig. 1) Thus it produces 12 nine-bit numbers every 10 msec. (10,800 bits/sec.). Previous research [2,4,10] indicates that this type of data is sufficient for a wide range of recognition tasks. This preprocessor is an economical means (in terms of hardware cost versus computing effort) of doing a large portion of the data reduction which is a major aspect of speech recognition.

Audio response from the computer is provided by a D/A converter and also by a hardware speech expander which expands time-domain compressed speech.

These devices are interfaced with the computer via the I/O bus of the PDP-10 (Fig. 2). They are connected to microphones, telephones, speakers, tape recorders, etc. through an audio multiplexing system (AMS) which has four pairs of input and output channels. There can be as many as eight each of A/D type and D/A type devices. There are up to 16 input devices (microphones, etc.) and 16 output devices (speakers, etc.). On each AMS input channel, one of the inputs is used to monitor the audio of its corresponding output channel, and vice versa. E.g., the audio produced by the speech expander can be fed back in through an input channel and re-digitized by the A/D, or the audio coming in from a microphone can be recorded on an audio tape deck. These monitoring facilities provide excellent means for having the audio machine monitor and test itself.

The AMS, in addition to providing connecting and mixing facilites, also allows for functions such as automatic gain control, selective frequency enhancement, and amplification.

The entire operation of these devices (the AMS, the A/D devices, tape decks, etc.) is controlled by commands from programs running on the PDP-10. The "audio machine", then, is made up of these devices and the software support on the PDP-10 which interprets and executes the commands.

Real-time, interactive recognition (and synthesis), which is the goal of the speech system, requires real-time I/O handling. This means that the audio machine must be fast and responsive enough so that no data is lost. It also implies that the speech I/O must continue concurrently and asynchronously with the rest of the speech system, at the same time supplying it with input and accepting output when requested.

While real-time performance is a major goal of the system, it is not explicitly constrained to operate and respond within any given time period. Thus, the audio machine must be able to accept real-time data and supply it to the speech system at any rate which the speech system requests it; in this sense, the audio machine can be viewed as a buffer or "de-timer" which protects the speech system from the pressures of real world timing and allows it to operate as slow as it must and as fast as it can.

The audio machine accomplishes this "de-timing" control by separating its activities into two functions: that of transmitting data in real-time between the A/D device and buffering files on the disk file storage and that of transmitting data upon request of the speech system between the files and the speech system. This separation of functions also allows for a simple method of building a "library" of digitized speech which can be used many times. This works on input to allow the same data to be fed to different recognition algorithms for controlled evaluation experiments and on output as a means of "canning" responses which can then be re-synthesized by the D/A converter or the speech expander.

Besides the buffering function, the audio machine also provides for low level smoothing, silence detection, and preliminary segmentation of the digitized input. These functions are critical to recognition and are an area of continued investigation. The audio machine structure is designed for convenient modification of these algorithms.

Programming Implementation

A real-time interactive speech system is a complex systems programming task with several people usually working on various parts. Our approach is to construct the system as a set of cooperating parallel processes, each of which is a job on the PDP-10. This modular approach allows for easier modification or replacement of some section of the system because it forces clean interfaces between the various modules.

Parallel processes are implemented through the use of two primitive capabilities on the PDP-10 system:

1. The "pseudo-teletype" construct allows one process (job) to initiate and control other processes and to go into a wait state contingent on another process.

2. Several jobs can have a section of core storage in common; this allows the jobs to communicate very efficiently among themselves.

Most of the programming (95%) is done in SAIL, an ALGOL-like language with string processing, an imbedded associative language, powerful I/O capabilities, and facilities for inserting machine language instructions within the source code.

Concluding Remarks

During the last seven years we have built several recognition systems of increasing complexity. The system described here is a natural outgrowth of these earlier systems. It eliminates many of the short-comings of the previous systems and is expected to be an adequate tool for speech recognition research over the next five years.

ACKNOWLEDGMENT

We are grateful to Professor Allen Newell for helpful suggestions in the preparation of this paper.

REFERENCES

[1] Reddy, D.R., "Computer Recognition of Connected Speech", J. Acoustical Soc. Amer., V. 42, 329:347 (August 1967).

[2] Vicens, P., "Aspects of Speech Recognition by Computer", Computer Science Department Report CS127, Stanford University (April 1969).

[3] Fry, D.B. and P.Denes, "Experiments in Mechanical Speech Recognition," Information Theory (Butterworth and Co. Ltd., London) 206-212 (1955).

[4] Sakai, T. and S.Doshita, "The Automatic Speech Recognition System for Conversational Sound", IEEE Trans. Electronic Computers, EC-12, 835-846 (December 1963).

[5] Martin, T.B., A.L.Nelson, and H.J.Zadell, "Speech Recognition by Feature Abstraction Techniques", Wright-Patterson AFB AF Avionics Labs., Rept. AL-TDR 64-176 (1964).

[6] Hughes, F.W. and J.F.Hemdal, "Speech Analysis", Purdue Res. Foundation, Lafayette, Ind., TR-EE 65-9 (1965).

[7] Gold, B., "Word Recognition Computer Program", Tech. Report 452, Lincoln Labs., MIT (1966).

[8] Bobrow, D.G. and D.H.Klatt, "A Limited Speech Recognition System", Proc. AFIPS Fall Joint Computer Conference (Thompson, Washington, D. C.) V. 33, 305-318 (1968).

[9] Gerstman, L.J., "Classification of Self-Normalizing Vowels", IEEE Trans. Audio and Electronacoustics, AU-16, 78-80 (March 1968).

[10] Scarr, R.W.A., "Zero Crossings as a Means of Obtaining Spectral Information in Speech Analysis", IEEE Trans. Audio and Electroacoustics, AU-16, 247-255 (June 1968).

[11] Swinehart, D. and R.Sproull, SAIL, Stanford Artificial Intelligence Project, Operating Note 57.1 (April 1970).

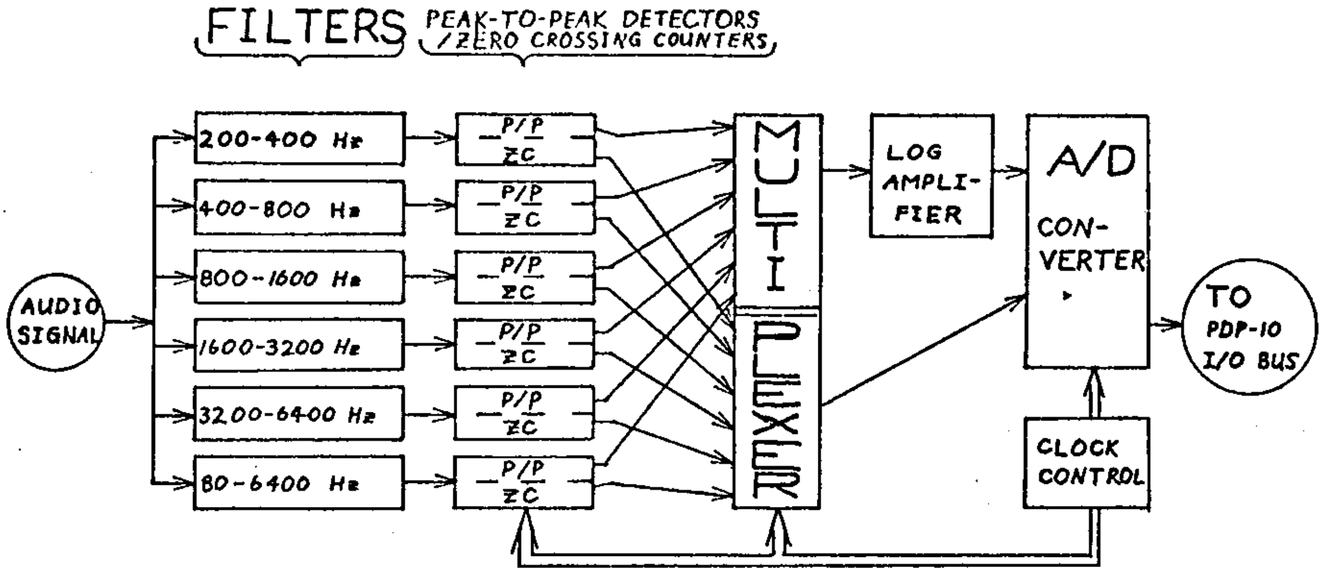


FIG. 1. SPEECH PREPROCESSOR

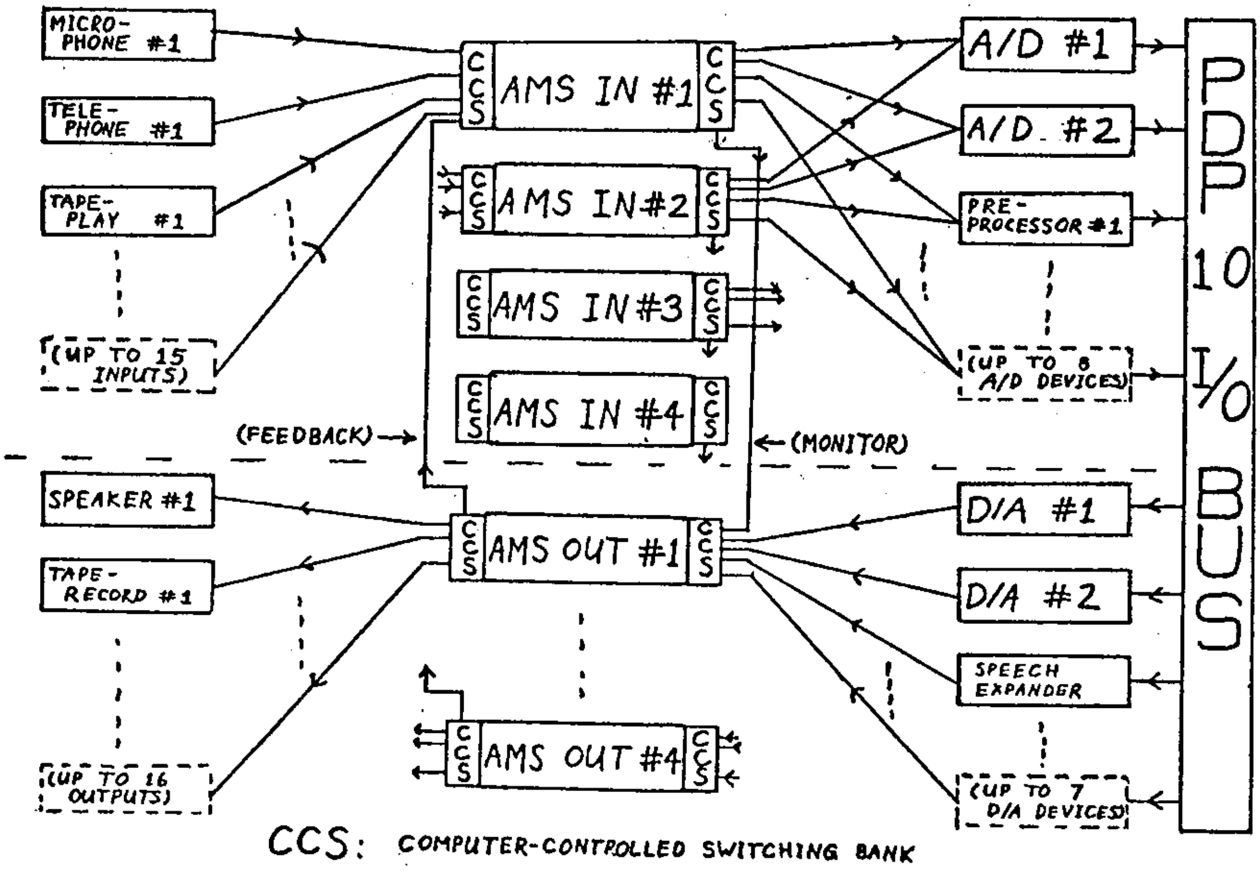


FIG 2. AUDIO MACHINE HARDWARE

Reprinted from:
Erman, L.D. and D.R.Reddy, Implications of Telephone Input for Automatic Speech Recognition, Proc. 7th Inter. Congress on Acoustics, Budapest, V.3, 85-88 (August 1971).

IMPLICATIONS OF TELEPHONE INPUT FOR AUTOMATIC SPEECH RECOGNITION

L. D. Erman, D. R. Reddy
Carnegie-Mellon University
Pittsburgh, Pa. 15213

INTRODUCTION

The telephone, because of its low cost and wide availability, is an attractive device to consider for input to automatic speech recognition systems. This attraction, however, is tempered by the distortion in the speech signal which the telephone introduces. A great deal is known about the kinds of distortions which occur over the telephone (Alexander, Gryb and Nast, 1960; Andrews and Hatch, 1970; Inglis and Tuffnell, 1951) and their effect on human perception of speech (Flanagan, 1965), but nothing is known about their effects on machine perception of speech. We present here some experiments using telephone input to a particular speech recognition system which was designed with no thought of telephone input.

Telephone induced distortions include:

1. Bandwidth limitation--300-3200Hz as opposed to 150-7000Hz for speech.

2. Attenuation distortion--relatively flat response from 300Hz to 1100 Hz but a linear fall of about 6db per octave outside of that range.

3. Envelope delay distortion--phase delays at the high and low cut-off frequencies are as much as 1 msec. relative to those in mid-band.

4. Crossmodulation--introduction of an extraneous speech signal can occur "randomly."

5. Discretization noise--digital pulse-coded modulation currently uses a 7-bit encoding for long distance transmission.

6. Random noise--random noise always occurs with transmission; a major independent variable in its generation is the particular circuit switching path of the connection.

THE SYSTEM

The automatic speech recognition system used for this study is a version of the EARS system developed by Vicens and Reddy (Vicens, 1969) and modified by the authors. This version recognizes isolated utterances (words or phrases) of up to several seconds duration after previous training on a different set of the same utterances. The input to the system consists of 10 msec. samples of speech parameters. The parameters are obtained by filtering the speech signal into five wide bands (200-400Hz, 400-800, 800-1600, 1600-3200, 3200-6400) and, for each 10 msec. sampling period, producing the number of zero-crossing and the maximum peak-to-peak amplitude for each band. (The filtering, zero-crossing counting, and amplitude detection are done by analog hardware.) Thus, 10 parameters of 7 bits are produced every 10 msec. for a rate of 7000 bits per second.

The system combines the 10 msec. samples into phoneme-like segments and makes a rough classification into 7 groups -- nasal, fricative, burst, stop, transition, vowel, and "other". The vowels are further subclassified on the basis of their parameters. The fricative and vowel information is used to select from the previously learned "dictionary" the most likely candidates. Each candidate is matched to the unknown input and results in a similarity score; the candidate with the highest score, if above a threshold, is taken to be the "answer"; if none is above the threshold, then the system responds with no result. The system has many heuristic algorithms both for efficiency and for correcting errors made during the segmentation and candidate selection. Learning (training) occurs when an utterance is placed in the dictionary along with its name.

The system operates in 3-15 times real-time (depending mostly on dictionary size) on a time-sharing PDP-10 computer with a basic cycle time of about 2 micro-sec. Results obtained by Vicens (1969) include 98% to 100% correct for a 54-word list after 4 training lists by a single speaker, 91% on a 561-word list after 3 trainings, and 85% on 54 words after training on 7 lists by 7 other speakers.

THE EXPERIMENT

The list of 54 words used for this experiment was originally used by Gold (1966). The data was recorded by Dr. Ken Stevens on high quality audio tape over a good microphone in a quiet room (S/N > 35db). These recordings have been used by Bobrow and Klatt (1968), and Vicens (1969).

To produce the telephone input, the following procedure was carried out: The two versions of the 54-word list were played on a Sony TC104 tape deck with the tone control set at its mid-point. The mouthpiece of a standard telephone was placed about 7 cm. in front of the Sony's speaker and a connection was made over the telephone through a local switchboard into the public telephone network and received by a recorder coupler (Voice Connecting Arrangement CD6 -- Bell Tel. Co.). The output from the recorder coupler was recorded on a Scully tape unit.

For digitization, the audio tapes (both the original and the telephone transmission) were played on the Scully and input through the Audio Multiplexing System (Reddy, et al, 1970) into the hardware preprocessor, all under the control of the computer. In addition, a third set of data was obtained by digitization of the telephone recording with an Advent Frequency Balance Control connected in the audio circuit. This device has ten individual octave filters from 20 to 20,480 Hz and was used to enhance the high and low ranges of the speech signal in an attempt to compensate for attenuation distortion. The frequency enhancement of the setting used is shown in figure 1.

Each run of this experiment consisted of having the recognition system learn the first 54-word version and then attempt to recognize the words in the second version.

Word	High Quality	Tele-phone	Enhanced Phone	Modified Enhanced	Word	High Quality	Tele-Phone	Enhanced phone	Modified Enhanced
1 INSERT	.	.	.	.	28 NAME	.	EXCHANGE	.	.
2 DELETE	??	.	.	.	29 END	.	.	.	.
3 REPLACE	.	.	.	.	30 SCALE	.	NAME	.	.
4 MOVE	.	.	.	.	31 CYCLE	.	.	.	.
5 READ	.	.	.	.	32 SKIP	.	SIX	.	.
6 BINARY	.	.	.	.	33 JUMP	.	POINT	.	.
7 SAVE	.	.	.	.	34 ADDRESS	.	.	.	.
8 CORE	.	FOUR	FOUR	FOUR	35 OVERFLOW	.	.	.	.
9 DIRECTIVE	.	OCTAL	OUTPUT	OUTPUT	36 POINT	.	ONE	.	.
10 LIST	.	.	??	.	37 CONTROL	.	.	COMPARE	COMPARE
11 LOAD	.	.	.	.	38 REGISTER	.	.	.	.
12 STORE	.	WHOLE	.	.	39 WORD	.	.	.	.
13 ADD	.	.	.	.	40 EXCHANGE	.	.	.	.
14 SUBTRACT	.	.	.	.	41 INPUT	.	.	.	.
15 ZERO	.	NAME	.	.	42 OUTPUT	.	.	.	.
16 ONE	.	BYTE	.	.	43 MAKE	.	.	.	.
17 TWO	.	MOVE	.	.	44 INTERSECT	.	.	.	.
18 THREE	READ	.	READ	READ	45 COMPARE	.	.	.	.
19 FOUR	CORE	WHOLE	.	.	46 ACCUMULATE	.	.	.	.
20 FIVE	.	.	.	.	47 MEMORY	.	END	END	END
21 SIX	.	.	SKIP	??	48 BYTE	.	JUMP	.	.
22 SEVEN	.	.	.	.	49 QUARTER	.	.	.	.
23 EIGHT	.	.	.	.	50 HALF	.	ONE	.	.
24 NINE	.	.	.	.	51 WHOLE	.	.	.	.
25 MULTIPLY	.	.	.	.	52 UNITE	.	.	.	.
26 DIVIDE	.	.	.	.	53 DECIMAL	.	.	.	.
27 NUMBER	.	.	.	.	54 OCTAL	.	.	.	.

	High Quality	Tele-Phone	Enhanced phone	Modified Enhanced
Correctly recognized	51 94.4%	39 72.2%	47 87.0%	48 88.9%
Rejected	1 1.9%	0 0	1 1.9%	1 1.9%
Incorrect	2 3.7%	15 27.8%	6 11.1%	5 9.3%
Total errors	3 5.6%	15 27.8%	7 13.0%	6 11.1%
Mean computation time	1.5 sec	2.1 sec	2.1 sec	2.0 sec

. indicates correct answer.
?? indicates word rejected.
word indicate the incorrect answer given.

TABLE 1: Results of the recognition system runs.

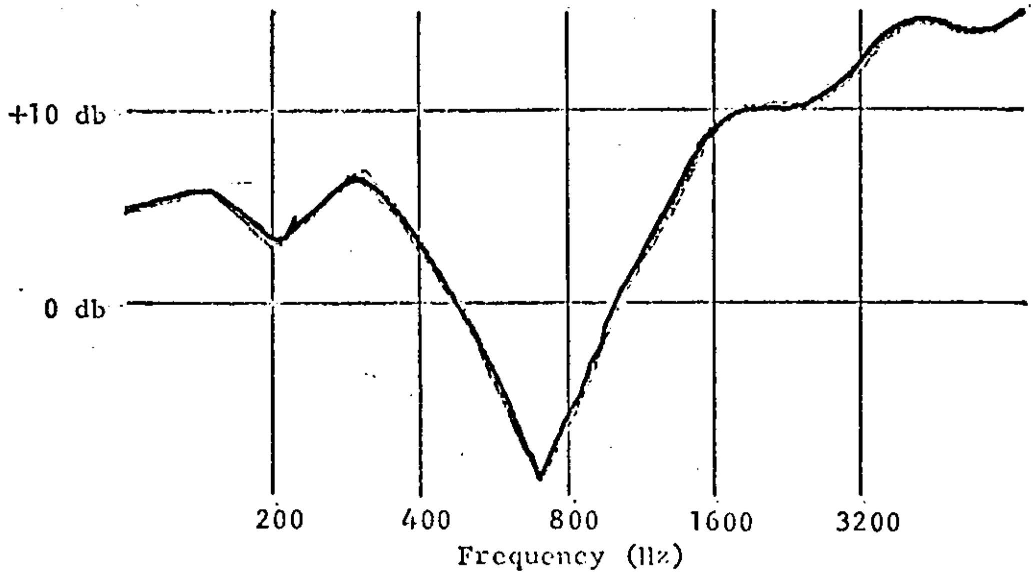


Figure 1: Response curve of Frequency Balance Control.

RESULTS

The results of the runs are shown in Table 1. The column labeled "word" contains the words actually uttered. The other columns contain the recognition system's answers for each of the runs.

The first run, labeled "high quality," was done with the original data. The second run, called "Telephone," was of the unmodified telephone signal. The third run was made with the high and low frequency enhanced telephone signal. Investigation of the printouts of the errors made on the enhanced signal led to a change of several thresholds used in the classification system; the same enhanced digital data was then run again and produced the results called "modified enhanced."

At the end of the table, statistics on the runs are presented. The computation times shown are the average amount of central processor time used per utterance.

DISCUSSION

The /s/- and /z/-like fricatives are used extensively by the recognition system as primary clues because of their reliability and ease of detection. The telephone input contained no segments which were classified as fricatives; this is caused by the high frequency attenuation which masks the major features of these fricatives. The frequency enhancement was used in an attempt to boost the high frequencies at the expense of the band from about 450Hz to 1000Hz, where the greatest speech energy is. The result of this enhancement was the third run which had fricative classification at about the level of the non-telephone data and, which had 13% errors as opposed to the 28% of the raw telephone input. The modifications of the last run lowered two of the fricative thresholds in an attempt to improve the fricative classification further.

The enhancement was also designed to improve recognition of nasals by boosting the response in the range below 350Hz, where the nasals' first formants lie.

The errors in the last run serve as good examples of the problems yet to be faced. "Core" and "Four" are very difficult to differentiate under any conditions and, in some sense, are "honest" mistakes for any recognition system with no semantic or syntactic support. The errors in "Directive" and "Memory" were caused by difficulty in vowel segmentation in the stressed part of the words (spoken as dRECtive" and "MEMRY"). This difficulty is probably caused by the envelope delay which introduces distortion in the amplitude detector of the hardware pre-processor.

The two versions of "Three" had considerably different vowel amplitudes and represent a speaker-induced variability. (It is somewhat curious that the non-telephone run also had this error but the unenhanced telephone run did not.)

The errors on "Six" and "Control" represent fricative detection problems. The final fricative in the second version of "Six" was not detected and the plosive /t/ in the first version of "Control" was misclassified as a fricative. Further tuning of the system (or a change in the enhancement filter settings) might correct these errors.

CONCLUSIONS

For the system studied, the simple analog frequency enhancement of the telephone signal resulted in an error rate of 13% versus 6% for high quality data; in addition, the required computation increased by 33%. This degradation does not seem very high for a system which has not been modified in any other way to handle telephone input; it is expected that this degradation could be reduced by at least half by a moderate amount of "tuning" threshold parameters without making any changes to the basic organization or algorithms used.

The results must be tempered by the facts that only one particular system was investigated, and only a small amount of data from one speaker over one local telephone connection was used.

The results indicate that telephone input need not have a crippling effect on automatic speech recognition systems; the authors believe that the degradation of machine speech recognition over the telephone, relative to high-quality input, may be on the order of, and probably will be less than, the degradation of human speech perception under the same conditions.

REFERENCES

Alexander, A.A., R.M.Gryb, and D.W.Nast (1960), "Capabilities of the Telephone Network for Data Transmission," Bell System Technical Journal, 39, pp. 431-476.

Andrews, F.T. and R.W.Hatch (1970), "National Telephone Network Transmission Planning in the American Telephone and Telegraph Company," International Seminar on National Telephone Transmission Planning: Melbourne, Australia, Feb. 27-Mar. 2; and IEEE 1970 International Conference on Communications, San Francisco, June 9.

Bobrow, D.G. and D.H.Klatt (1968), "A Limited Speech Recognition System," Proc. AFIPS Fall Joint Computer Conference, Thompson, Wash., D.C., 33, pp. 305-318.

Flanagan, J.L. (1965), "Speech Analysis, Synthesis and Perception," N.Y.: Academic Press.

Gold, B. (1966), "Word Recognition Computer Program," Tech. Report 456, Lincoln Labs, MIT.

Inglis, A.H. and W.L.Tuffnell (1951), "An Improved Telephone Set," Bell Sys. Tech. J., 30, pp. 239-270.

Reddy, D.R., L.D.Erman, and R.B.Neely (1970), "The CMU Speech Recognition Project," IEEE System Science and Cybernetics Conference.

Vicens, Pierre (1969), "Aspects of Speech Recognition by Computer," Report CS-127, Computer Science Department, Stanford University.

Reprinted from:

Neely, R.B. and D.R.Reddy, Speech Recognition in the Presence of Noise, Proc. 7th Inter. Congress on Acoustics, Budapest, V.3, 177-180 (1971).

SPEECH RECOGNITION IN THE PRESENCE OF NOISE

R. B. Neely, D. R. Reddy
Carnegie-Mellon University
Pittsburgh, Pa.

INTRODUCTION

There have been studies which evaluate the effect of noise on human perception of speech (Miller and Nicely, 1955). It has been difficult to evaluate the effect of noise on machine perception of speech because of the paucity of working speech recognition systems. It is important that we have adequate means of evaluating the effect of noisy environments with, e.g., computer noise, air conditioning, or teletype noise. This paper presents the effect of three different types of noise at different signal/noise ratios on a particular speech recognition system and discusses possible transformations on the speech to reduce the degradation in recognition caused by noise.

THE SYSTEM

The basic speech recognition system is that developed by Vicens and Reddy (Vicens, 1969) and extended by Erman. This system is described in the paper on telephone speech by Erman and Reddy in these proceedings. The source data and the vocabulary are also the same. The reader is referred to the above paper for details.

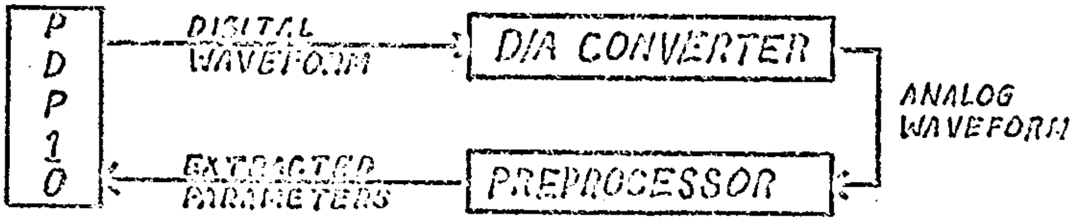
Parameter extraction from speech is performed by a special hardware preprocessor interfaced to a PDP-10 computer. The parameters are obtained by filtering the speech signal into five bands (200-400 Hz, 400-800, 800-1600, 1600-3200, and 3200-6400 Hz) and the maximum peak-to-peak amplitude and number of zero crossings are determined for every 10 ms. interval of speech. The recognition system extracts, from these parameters, crude estimates of three formant frequencies and their amplitudes. We shall see later that the introduction of noise causes wide variability in the formant estimates, making them unusable. Only the amplitudes, after suitable normalization for noise, were used in the final experiment.

THE EXPERIMENT

Data. The speech data for this investigation consisted of two versions of a 54 word list. The first list was used to train the system and the second list was used for recognition. Results from this recognition experiment served as the control for this investigation. Three types of noise -- teletype idling, teletype typing, and machine room (fans and air conditioners) -- were recorded using an omnidirectional microphone. Each kind of noise was mixed with each of the utterances contained in the two lists at two different signal/noise ratios: 15 and 25. This yielded 6 pairs of lists in addition to the control pair, thus making 7 pairs for the entire investigation.

Production of Noise-mixed Speech. For each control list, the entire list was first digitized at 10 kHz sampling rate. The beginning and end of each utterance in the list was then detected and each kind of noise was separately mixed with each of the utterances in the list at 15 and 25 db signal/noise ratio. For each utterance, the average power was determined. Then each type of noise was added individually to the utterance, after appropriate scaling, to yield the desired signal/noise ratios. The noise mixing could have been accomplished by analog means rather than digital; however, it would have been difficult to control the accuracy of mixing of signal and noise. In addition, if analog mixing were used, the detection of speech boundaries would have to be done on already noisy data. While the problem of detection of speech boundaries in the presence of noise is important, it was considered to be of a secondary nature.

Parameter Extraction. The amplitude and zero crossing parameters of each noise-mixed utterance were obtained by the use of the hardware preprocessor described above. This was accomplished by setting up the preprocessor and the D/A converter (both of which are devices attached to a PDP-10 computer) as in the following diagram:



The noise-mixed speech in digital form was converted to an analog signal by a D/A converter which was then used by the preprocessor to generate the parameters. Care was taken to see that the D/A converter and the preprocessor were run synchronously within the time-sharing system.

Recognition Process. As with the original speech data, the first list of each of the 6 pairs of noise-mixed lists was used to train the system, and recognition was done on the second list of each pair. The initial recognition of noise-mixed speech (without any attempt at subtracting the noise from the extracted parameters) resulted in very high error rates -- only 2 percent of the words were correctly identified.

This made it necessary for us to consider a way of removing the noise. The average values of noise amplitude parameters were determined (for each of the six noises) using the preprocessor and these were subtracted from the corresponding values of the parameter vectors of the noise-mixed speech. The zero crossing parameters were left unaltered except for local smoothing. Even this transformation did not improve the results appreciably. Analysis of the resulting parameters revealed that noise that was over the average value still caused significant variability in the parameters causing erroneous recognition.

The next attempt at noise removal consisted of subtracting twice the average amplitudes (which corresponds roughly with the maximum noise levels) from the noisy speech parameters. The system was also modified to ignore the zero crossing parameters. This drastic step did significantly increase the recognition of noisy speech, but it was still considerably inferior to the recognition of noise-free speech.

To accurately estimate the effect of noise, it became necessary to perform similar transformations on the parameters of the noise-free speech, i.e., ignoring the zero crossing parameters. This resulted in a lower recognition accuracy for the noise-free speech. This, in addition to the degradation caused by the 10 kHz sampling rate and digitization noise, resulted in a drop of the accuracy for the control speech from 94 percent to 76 percent.

THE RESULTS

The following table presents the recognition results:

		Accuracy	Average Time
High-quality speech		94%	1.5 sec.
Original speech after 10 kHz digitation		83%	2.1 sec.
10 kHz speech with only amplitude parameters		76%	2.8 sec.
Idling noise	25 db	67%	3.3 sec.
	15 db	22%	3.3 sec.
Typing noise	25 db	43%	3.7 sec.
	15 db	---- *	---- *
Machine room noise	25 db	54%	3.3 sec.
	15 db	43%	2.8 sec.

* In the case of typing noise of 15 db, there was so much variation in the amplitude parameters that even after noise subtraction, the data was useless for the recognizer.

The recognition system minimizes the time for recognition by the use of two heuristics: the ordering heuristic and high score termination heuristic. The ordering heuristic attempts to order the candidates for comparison so that the candidates most likely to succeed appear toward the beginning of the list. This ordering is based on quickly attainable similarity characteristics of the vowels within the utterance. With the introduction of noise there is greater variability in the observed parameters resulting in unreliable ordering of the candidates. This, in turn, affects the time for recognition. The high score termination heuristic suffers a similar fate. This heuristic terminates any further comparison with possible candidates when one of the candidates attains a very high score, say 95 percent similar. In noisy speech, such high scores are seldom attained, which again results in greater computation time. Thus, as we see from the actual results above, recognition time for noisy speech was usually more than a factor of two longer than that for high-quality speech. The teletype idling and typing noise (particularly the latter) cause the greatest degradation in recognition, since they contain many high-frequency components. In addition, the typing noise also contains greater impulse-type variability that is hard to correct for. The machine room noise, on the other hand, is more constant and has mainly low-frequency components.

Many of the preliminary error analyses were performed on the speech mixed with teletype idling noise. This resulted in the setting of some thresholds which are probably more tuned towards that type of noise. We suspect that it may be possible to obtain similar accuracies for the other types of noises with similar tuning.

CONCLUSIONS

In this investigation we have ignored the problem of detection of beginning and ending of speech in the presence of noise. This problem is very crucial, since without this detection, segmentation and matching would become impossible.

We will now attempt to draw a comparison between the results presented here and the results generated by Miller and Nicely (1955). Before this can be done, certain facts concerning the comparability of these two sets of results need to be considered. First, in the experiments of Miller and Nicely, recognition of isolated phonemes in nonsense syllables was done; whereas, this present investigation consisted of word recognition, a somewhat easier task. Second, Miller and Nicely used only randomly-generated noise, which none of the experiments presented here did. Third, because of the methods used in this investigation, the control data (10 kHz digitized speech) is already somewhat degraded. Fourth, it is clear that the human recognition system is well adapted to operating in a noisy environment. Therefore, comparison of data on an absolute scale would be meaningless. We present instead comparisons of degradation in recognition caused by a drop in signal/noise ratio in terms of percent degradation per db:

	Result 1	Result 2
Human recognition	72% at 0 db	27% at -12 db
Machine recogniton:		
Idling noise	67% at 25 db	22% at 15 db
Machine noise	54% at 25 db	43% at 15 db

	% Reduction	% per db
Human recognition	(72-27)/72=63%	63/(0-(-12))=5.25%
Machine recognition:		
Idling noise	(67-22)/67=67%	67/(25-15)=6.7%
Machine noise	(54-43)/54=20%	20/(25-15)=2%

Each result in the table is a particular percent of correct recognition for some signal/noise ratio for one of the experiments. The percent reduction is a relative measure of the degradation in recognition between Result 1 and Result 2. The percent per db is just the percent reduction normalized for the drops in signal/noise ratio in the different experiments. The human recognition degradation is approximately comparable to the idling-noise degradation in this investigation. The degradation for machine-room noise, however, is much less. Miller and Nicely used only randomly-generated noise, and its spectrum is more like the idling noise spectrum than the machine-room noise spectrum. The latter contains fewer high-frequency components and so does not degrade recognition as much at high levels.

The results contained here are clearly preliminary. More complex noise subtraction procedures have not yet been investigated and should reduce the error rates. One possibility would be subtracting the overall spectrum of the noise from the spectrum of the speech as it varies in time. Also, it might be possible to do formant tracking on the subtracted spectrum. This would not be possible without the spectrum subtraction because of the variability of the noise spectra. In addition, modification of various thresholds and the procedures in the recognition system to anticipate and correct for noise should improve the results. Although the results of recognition with noise are appreciably inferior to recognition without noise, the improvements made with even simple transformations are encouraging.

REFERENCES

Miller, G.A. and P.E.Nicely (1955), "An Analysis of Perceptual Confusions Among Some English Constants," Journal of the Acoustical Society of America, 27, pp. 338-352, March 1955.

Vicens, P.J. (1969), "Aspects of Speech Recognition by Computer," Report CS-127, Computer Science Department, Stanford University.

Reprinted from:

Reddy, D.R., Speech Recognition: Prospects for the Seventies, Proc. IFIP 1971, Ljubljana, Yugoslavia, Invited paper section, pp. I-5 to I-13.

SPEECH RECOGNITION: PROSPECTS FOR THE SEVENTIES

D. R. REDDY

Computer Science Department
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

> "...lead us to believe that performance will continue to be very limited unless the recognizing device understands what is being said with something of a facility of a native speaker (that is, better than a foreigner who is fluent in the language). If this is so, should people continue work toward speech recognition by machine?" (Pierce, 1969)

INTRODUCTION

It is clear that we won't have a speech recognition system which understands English with the facility of a native speaker for a long time to come. However, it seems possible that systems capable of performing as well as a native speaker in limited task domains using a restricted (but not very limited) English-like language can be built before the end of this decade. In this paper, we will outline the nature of the restrictions on the language and task domain, discuss models for recognition of this class of systems, and describe the structure of the Hear-Say system which is potentially capable of natural conversation in limited task environments.

Can a system whose performance is less than that of a native speaker of English be of any use to anyone? Indeed, there appear to be several tasks which can operate adequately in a restricted language environment and would benefit from voice input. The main characteristic of these systems is that they gather information or provide information in response to a vocal request from the user about a restricted and prespecified task. Voice communication is preferred in these tasks because of many factors: higher data rate, extra motor process when hands and feet are already engaged, ease of use, or the ready availability of inexpensive telephone terminals.

A recent study committee (Newell et al. [11]) on speech understanding systems considered several tasks that could benefit by voice input. These included: querying a data management system, data acquisition of formatted information, querying the operational status of a computer, consulting on the use of a computer, airline guide information service, air traffic control, medical history taking, physical inventory taking and so on. Of these, the first four were studied by the committee in greater detail to isolate and identify the problem areas.

Fig. 1 (reproduced from Newell et al.) shows the many different dimensions along which speech recognition systems can vary. This figure illustrates the multitude of trade-offs that are possible in designing a speech recognition system. The column on the left shows the dimensions and the column on the right gives the possible choices available to a system designer along each of the dimensions. It should be clear that there are literally hundreds of intermediate systems that should be experimented with before one can even begin to seriously consider a system for understanding English with the facility of a native speaker. To seriously suggest that we stop working on speech recognition systems because we can't build English understanding systems is like saying that we should stop building rockets because they can't fly at the speed of light.

In spite of two decades of research, progress in the field of speech recognition has been very limited. When one looks for reasons for this slow and unsteady progress one finds that over-optimism, inadequate technology, and incorrect models have been the prime causes. The net result has been a large number of paper designs and very few working systems. As is the case with much of the artificial intelligence research, it has proved to be difficult to build on each others' research in this field.

Lindgren [10] and Hyde [7] provide excellent surveys of the state of the art up to 1968. Here we will limit ourselves to the discussion of the more recent results. Recently several systems capable of recognizing 50 to 500 word vocabularies of isolated words and phrases have been developed (Gold [6]; Bobrow and Klatt [2]; Vicens [19]; Zagoruiko [21]; Vysotskiy et al. [20]). These systems claim to achieve 85% to 99% accuracy on recognition tasks. However, it has been difficult to evaluate the relative merits of the systems. It is not enough to merely report the accuracy of a proposed

1. What sort of speech? (The *continuous speech* problem)	Isolated words? Continuous speech?
2. How many speakers? (The *multiple speaker* problem)	One? Small set? Open population?
3. What sort of speakers? (The *dialect* problem)	Cooperative? Casual? Playful? Male? Female? Child? All three?
4. What sort of auditory environment? (The *environmental noise* problem)	Quiet room? Computer room? Public place?
5. Over what sort of communication system? (The *transducer* problem)	High quality microphone? Telephone?
6. How much training of the system? (The *tuneability* problem)	Few sentences? Paragraphs? Full vocabulary?
7. How much training of the users? (The *user training* problem)	Natural adaptation? Elaborate?
8. How large and free a vocabulary? (The *vocabulary* problem)	50? 200? 1,000? 10,000? Preselected? Selective rejection? Free?
9. What sort of language? (The *syntactic support* problem)	Fixed phrases? Artificial language? Free English? Adaptable to user?
10. What task is to be performed? (The *semantic support* problem)	Fixed response for each total utterance (e.g., table look up)? Highly constrained task (e.g., simple retrieval)? Focussed task domain (e.g., numerical algorithms)? Open semantics (e.g., dictation)?
11. What is known psychologically about the user? (The *user model* problem)	Nothing? Interests? Current knowledge? Psychological model for responding?
12. How sophisticated is the conversational dialogue? (The *interaction* problem)	Task response only? Ask for repetitions? Explain language? Discuss communication?
13. What kinds of error can be tolerated? (Measured, say, in % error in final semantic interpretation) (The *reliability* problem)	Essentially none (<.1%). Not inconvenience user (<10%). High rates tolerable (>20%).
14. How soon must the interpretation be available? (The *real time* problem)	No hurry (non real time). Proportional to utterance (about real time). Equal to utterance with no delay (real time).
15. How much processing is available? (Measured, say, in millions of instructions per second of speech)	1 mips? 10 mips? 100 mips? 1000 mips?
16. How large a memory is available? (Measured, say, in millions of bits accessible many times per second of speech)	1 megabit? 10 megabits? 100 megabits? 1000 megabits?
17. How sophisticated is the organization? (The *systems organization* problem)	Simple program? Discrete levels? Multiprocessing? Parallel processing? Undirectional processing? Feedback? Feed forward? Backtrack? Planning?
18. What should be the cost? (Measured, say, in dollars per second of speech) (The *cost* problem)	.001 $/s? .01 $/s? .10 $/s? 1.00 $/s?
19. When should the system be operational?	1971? 1973? 1976? 1980?

Fig. 1. Considerations for a speech-understanding system.

algorithm, but one must consider all the relevant factors, e.g., accuracy, response time, vocabulary size, complexity of the words, and complexity of language. For example, a system capable of recognizing the ten words "ore, core, tore, pour, gore, door, bore, four, Thor, more" would have to be much more sophisticated than a system for recognizing the digits. A system which gives 90% accuracy in real-time may in fact be superior to a system which give 98% accuracy but takes a 100 times longer to do so. As a benchmark we will give the performance characteristics of the Vicens-Reddy system. This system (Vicens [19]) can recognize about 500 slightly selected words and phrases commonly occuring in the English language, spoken by cooperative speakers who have been trained on the system, in about 10 times real-time, with 95% accuracy, on a PDP-10 (approximately 500,000 instructions per second) and requiring about 2000 bits of memory for each word in the lexicon.

In spite of several attempts, there has been no significant breakthrough in the recognition of connected speech of a large population of speakers. Most of the difficulties arise from the lack of adequate rules to account for the wide variability of the observed acoustic parameters of a phoneme from context to context. Attempts at phonetic transcription systems (Sakai and Doshita [16]; Reddy [15]; Tappert et al. [18]) appear to be of limited value since they cannot adequately account for the variability without the knowledge of syntax, semantics, task environment, and speaker characteristics.

It is expected that most of the contextual variability can be accounted for through a better understanding of the acoustic-phonetic rules governing the speech production and perception process. Some of the rules are known and many others remain to be discovered, and those rules that are known are not readily accessible. Interested computer scientists are referred to the works of leading researchers in this field for useful pointers into the literature (Fant [4]; Flanagan [5]; Kozhevnikov and Chistovich [8]; Lehiste [9]; Chomsky and Halle [3]).

MODELS FOR SPEECH RECOGNITION

Most earlier attempts at connected speech recognition have failed because of their inability to account for the effect of phonetic, syntactic, and semantic context on the parametric variability among various allophones of phonemes of English. In this section, we will outline the features of three models which appear to be promising. A more detailed discussion of these models can be found in Newell et al. [12] and in the associated references.

1. What sort of speech? (The *continuous speech* problem)	Isolated words? Continuous speech?
2. How many speakers? (The *multiple speaker* problem)	One? Small set? Open population?
3. What sort of speakers? (The *dialect* problem)	Cooperative? Casual? Playful? Male? Female? Child? All three?
4. What sort of auditory environment? (The *environmental noise* problem)	Quiet room? Computer room? Public place?
5. Over what sort of communication system? (The *transducer* problem)	High quality microphone? Telephone?
6. How much training of the system? (The *tuneability* problem)	Few sentences? Paragraphs? Full vocabulary?
7. How much training of the users? (The *user training* problem)	Natural adaptation? Elaborate?
8. How large and free a vocabulary? (The *vocabulary* problem)	50? 200? 1,000? 10,000? Preselected? Selective rejection? Free?
9. What sort of language? (The *syntactic support* problem)	Fixed phrases? Artificial language? Free English? Adaptable to user?
10. What task is to be performed? (The *semantic support* problem)	Fixed response for each total utterance (e.g., table look up)? Highly constrained task (e.g., simple retrieval)? Focussed task domain (e.g., numerical algorithms)? Open semantics (e.g., dictation)?
11. What is known psychologically about the user? (The *user model* problem)	Nothing? Interests? Current knowledge? Psychological model for responding?
12. How sophisticated is the conversational dialogue? (The *interaction* problem)	Task response only? Ask for repetitions? Explain language? Discuss communication?
13. What kinds of error can be tolerated? (Measured, say, in % error in final semantic interpretation) (The *reliability* problem)	Essentially none (<.1%). Not inconvenience user (<10%). High rates tolerable (>20%).
14. How soon must the interpretation be available? (The *real time* problem)	No hurry (non real time). Proportional to utterance (about real time). Equal to utterance with no delay (real time).
15. How much processing is available? (Measured, say, in millions of instructions per second of speech)	1 mips? 10 mips? 100 mips? 1000 mips?
16. How large a memory is available? (Measured, say, in millions of bits accessible many times per second of speech)	1 megabit? 10 megabits? 100 megabits? 1000 megabits?
17. How sophisticated is the organization? (The *systems organization* problem)	Simple program? Discrete levels? Multiprocessing? Parallel processing? Unidirectional processing? Feedback? Feed forward? Backtrack? Planning?
18. What should be the cost? (Measured, say, in dollars per second of speech) (The *cost* problem)	.001 $/s? .01 $/s? .10 $/s? 1.00 $/s?
19. When should the system be operational?	1971? 1973? 1976? 1980?

Fig. 1. Considerations for a speech-understanding system.

algorithm, but one must consider all the relevant factors, e.g., accuracy, response time, vocabulary size, complexity of the words, and complexity of language. For example, a system capable of recognizing the ten words "ore, core, tore, pour, gore, door, bore, four, Thor, more" would have to be much more sophisticated than a system for recognizing the digits. A system which gives 90% accuracy in real-time may in fact be superior to a system which give 98% accuracy but takes a 100 times longer to do so. As a benchmark we will give the performance characteristics of the Vicens-Reddy system. This system (Vicens [19]) can recognize about 500 slightly selected words and phrases commonly occuring in the English language, spoken by cooperative speakers who have been trained on the system, in about 10 times real-time, with 95% accuracy, on a PDP-10 (approximately 500,000 instructions per second) and requiring about 2000 bits of memory for each word in the lexicon.

In spite of several attempts, there has been no significant breakthrough in the recognition of connected speech of a large population of speakers. Most of the difficulties arise from the lack of adequate rules to account for the wide variability of the observed acoustic parameters of a phoneme from context to context. Attempts at phonetic transcription systems (Sakai and Doshita [16]; Reddy [15]; Tappert et al. [18]) appear to be of limited value since they cannot adequately account for the variability without the knowledge of syntax, semantics, task environment, and speaker characteristics.

It is expected that most of the contextual variability can be accounted for through a better understanding of the acoustic-phonetic rules governing the speech production and perception process. Some of the rules are known and many others remain to be discovered, and those rules that are known are not readily accessible. Interested computer scientists are referred to the works of leading researchers in this field for useful pointers into the literature (Fant [4]; Flanagan [5]; Kozhevnikov and Chistovich [8]; Lehiste [9]; Chomsky and Halle [3]).

MODELS FOR SPEECH RECOGNITION

Most earlier attempts at connected speech recognition have failed because of their inability to account for the effect of phonetic, syntactic, and semantic context on the parametric variability among various allophones of phonemes of English. In this section, we will outline the features of three models which appear to be promising. A more detailed discussion of these models can be found in Newell et al. [12] and in the associated references.

Analysis-by-Synthesis

This model involves a comparison of the input spectrum with some internally generated spectra, with an error signal fed back to the generator for the next stage of analysis-by-synthesis. This model is one of the leading candidates because most rules that predict contextual variability are available only in generative form and the best way to use them is by synthesis and comparison. This method is time-consuming and many of the rules for synthesis are yet to be discovered.

Hypothesis-and-Test

If most generative rules can also be expressed in an analytic form, then the computationally more economical "hypothesize-and-test" might be more desirable. This technique involves hypothesizing the presence of a phonemic sequence and formulating or selecting a test that would verify the hypothesis (Newell [11]).

Analysis-by-Learning

This method involves the abstraction of useful information about contextual variability from several exemplars. Thus, if the phonetic realization of a given sequence of phonemes is not known in theory, then the computer attempts to extract the appropriate tests by examining the parameters of several utterances containing that phonemic sequence. The overall structure of the test would be preprogrammed from known linguistic knowledge and the specific details of the test would be filled in by the computer from the examination of data.

THE HEAR-SAY SYSTEM

In this section we will illustrate the structure and organization of speech-recognition systems by considering a specific example of a system being developed by the author and his colleagues at Carnegie-Mellon University. This system, called the Hear-Say System, is an attempt to build a task-independent kernel system within which several different tasks of varying degrees of complexity can be explored. While a task-specific system such as querying the operational status of a computer can probably be developed much more easily, it seems undesirable at this point in time. Task-specific systems not only necessitate reprogramming of the system for every new task but also make it difficult to conduct a systematic study of the many unsolved problems. The Hear-Say System attempts to provide facilities common to all speech-recognition systems by separating and identifying task-specific factors within such systems.

Fig. 2 gives a functional flowchart of various subprocesses within the system. The rectangles represent processes operating on the data (within braces) to produce the next level of representation. The system provides speech and graphic output facilities and a question-answering system. However, the main emphasis is on the recognition subsystem. In the remainder of this section we will describe the functional characteristics of various subprocesses within the system.

Speech analyzer

The purpose of this process is to extract a sequence of parameters from the speech signal. The speech from the input device (microphone, telephone, or tape recorder) is passed through 5 band-pass filters (200-400 Hz, 400-800 Hz, 800-1600 Hz, 1600-3200 Hz, and 3200-6400 Hz) and within each band the intensity and the number of zero-crossings are measured for each 10 ms interval.

Two main problems arise at this stage. First, speech, unlike other forms of input to the computer, requires continuous monitoring of the input device. The initiation and termination of input is a function of the incoming data itself. Further, if the subsequent stages of the recognition system are unable to use the data as it becomes available, the system has to preserve the data by storing it in the secondary storage (if necessary) and making it available on demand to subsequent stages.

The second problem is that the signal level and the signal-to-noise ratio vary from device to device, from room to room, and from person to person. The traditional automatic gain control distorts the signal in ways which make it difficult for subsequent processing. Thus it becomes necessary for the system to continuously keep track of the signal-to-noise ratio and warn the user if it cannot be corrected automatically and perhaps suggest a remedy, e.g., holding the microphone closer.

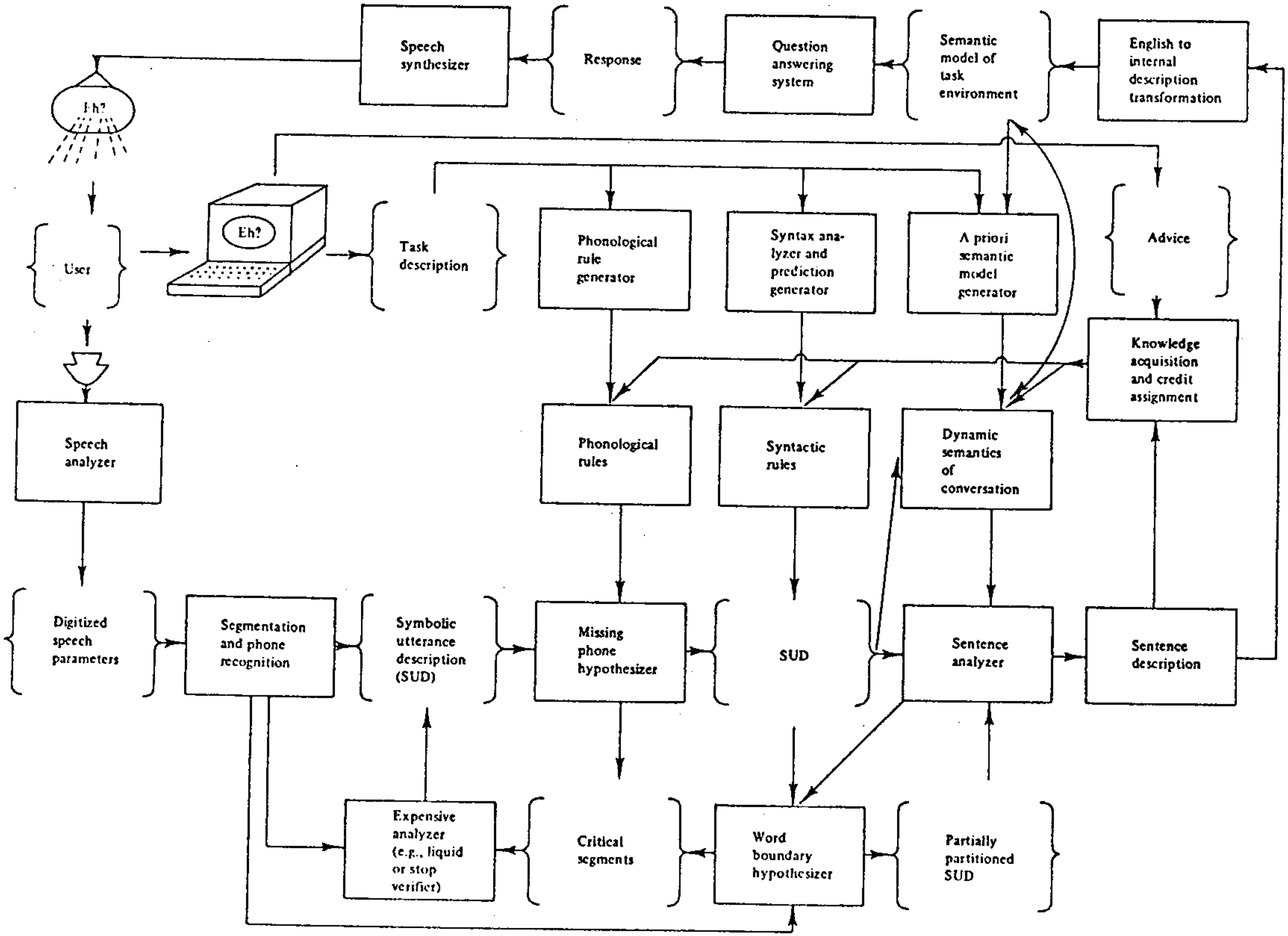

Speech synthesizer
Response
Question answering system
Semantic model of task environment
English to internal description transformation
Eh?
Eh?
User
Task description
Phonological rule generator
Syntax analyzer and prediction generator
A priori semantic model generator
Advice
Knowledge acquisition and credit assignment
Speech analyzer
Phonological rules
Syntactic rules
Dynamic semantics of conversation
Digitized speech parameters
Segmentation and phone recognition
Symbolic utterance description (SUD)
Missing phone hypothesizer
SUD
Sentence analyzer
Sentence description
Expensive analyzer (e.g., liquid or stop verifier)
Critical segments
Word boundary hypothesizer
Partially partitioned SUD

Fig. 2. The CMU Hear-Say System.

Segmentation and phone recognition

The purpose of this process is to divide the continuous parameter sequences into discrete phone-size chuncks. This is usually based on an acoustic-similarity measure (Reddy and Vicens [15]; Astrahan [1]) which is mainly suitable for the steady-state portions of the speech signal. These segments are then compared with known templates of sounds suitably normalized for speaker characteristics resulting in the assocation of phoneme-like labels with each segment. Diphthongs, liquids, and other gradually varying sounds result in quasi-random segmentation into steady-state subparts and such a subsegmentation dipthong is usually meaningless. In addition there may occasionally be a missing segment marker as in the case of "some milk" where the ending phoneme of the first word is the same as the beginning phoneme of the second. At this stage no attempt is made to locate or correct these errors since they can usually be handled much more easily at later stages.

Phonological rules

These rules usually deal with the theory of sound change in differing contexts in a natural language. We use the term in a much more restricted sense. Given that the vocabulary of the task is completely specified, it is possible to formulate a set of rules which completely account for the parametric variability for the lexicon and furthermore generate a structural representation of the lexicon giving different instances of similar structure. Thus all the words which are phonemically ambiguous with respect to each other are grouped together within the lexical data structure.

Missing phoneme hypothesizer

Given the preliminary segmentation and labeling (the symbolic utterance description), the missing phoneme hypothesizer consults the data structure produced by the phonological rule generator to locate all words with similar syllabic nucleus and similar segmental conntext. Given that part of the utterance description contained a fricative, /i/-like vowel, and a stop, we may get a set of possible candidates such as "fit", "sit", "split", "slit", and "splìt". By comparing the structural descriptions of these words one decides that it is important to see if there is a stop or a liquid between the fricative and the vowel. In particular, one may wish to test for a /p/-like stop or /l/-like liquid. Supposing the expensive analyzer (below) finds these are not present, one would still want to see if the fricative is an /s/ or an /f/.

Expensive analyzer

Once the missing-segment hypothesizer generates a plausible hypothesis, it is the function of the expensive analyzer to devise appropriate tests to verify the hypothesis. It does this by keeping a list of phoneme characteristics, a difference table for phoneme pairs, and by modifying these ideal characteristics by normalizing for the segmental context. Thus given that the fricative segment in the above example is either an /s/ or an /f/, one does not test for all the known characteristics of /s/ or all the known characteristics of /f/ but rather just those which differentiate the two phonemes, i.e., the amplitude of the signal which indicates if the segment is a strong fricative or a weak fricative.

Symbolic utterance description

The segmentation and phone recognition procedure produces a preliminary symbolic description of the utterance. Starting with a stressed syllabic nucleus and its context, one uses the constraints induced by the vocabulary of the task to make specific guesses at the possible word being uttered and resolving any local ambiguities using a hypothesize-and-test process. The resulting description consists of a sequence of labelled segments with a few of them grouped together to form words. This process works well only for stressed words with stops or fricatives at the word boundaries. When the juncture of two words has phonemes with the same manner of articulation, we get an ill-defined word boundary. Further, the coarticulation effects between words modify the characteristics of the initial and final phonemes of the word. Thus, it becomes necessary to hypothesize the word-level context so as to properly account for coarticulation across word boundaries.

Syntactic rules

The purpose of this procedure is to predict the most likely words that may occur before and after a given word. For example, the utterance description may indicate the presence of a noun (which happens to be the stressed word within that group) which is part of a noun phrase. Then the syntactic rules predict which words may precede it (the appropriate set of adjectives and articles) and which may follow it (verb phrase). Further, using the partial information available in the segmental description an ordering of these candidates is made to determine the most likely word-level context. Predicatability at this level is proportional to restrictiveness of the grammar.

Word boundary hypothesizer

The role of this procedure is to compute the expected parameters based on the word level context, examine the actual parameters present in the utterance description, and accept the most plausible word boundary context. Note that the context words themselves are modified by their context and this process extends until we reach a pause (breath group). The difficulty with word-level contextual variability is alleviated somewhat by the fact that when the stressed word boundary starts with a stop or fricative the effect of word-level coarticulation is minimal and can be easily accounted for. If at this point some expected segments are missing from the utterance description, these critical segments are isolated and a resegmentation is attempted using the expensive analyzer. Otherwise a sequence of word-boundary markers are introducted into the utterance description.

Dynamic semantics of conversation

The purpose of this procedure is to provide a great deal of selectivity in the location and identification of the words in the utterance string. There are three different sources of knowledge available at this point.

(1) Given that a stressed word in the utterance is recognized, then the semantics of the task can predict other words that may co-occur with this word.

(2) The semantic model of the task environment can provide selectivity on what may be expected.

(3) A model of the user, his beliefs and his needs can also provide direction.

These three sources of knowledge are interrelated, but they deal with three distinct aspects of dynamic semantics of conversation.

Task-dependent prediction generators

The sources of knowledge at each of the lexical, syntactic, and semantic levels are for the most part dependent on the task. The representation of this knowledge in a form suitable for recognition is presently prepared by the user for a given task. However, there appears to be no reason in theory why these cannot be generated by other procedures acting on the task description. The main obstacle at present is our lack of knowledge of the most appropriate representation for these predictors.

Knowledge acquisition

The purpose of this subprogram is to provide the system with the mechanisms which make it possible for it to learn new words, syntactic constructs, and semantic interpretations. These mechanisms would be activated while attempting to correct for errors on the basis of additional specification by the user. This is probably the least understood part of the Hear-Say System. The term "learning" in this context appears to mean addition and modification of the respective data structures and the automatic generation of new heuristics procedures which detect conditions under which this knowledge is to be activated. This form of language learning has not yet been successfully implemented on computers. Even simpler attempts at building extendable programming languages have not been very successful.

RELATED PROBLEMS OF INTEREST TO COMPUTER SCIENCE

Besides being of interest as one of the means of man-computer communication, speech recognition as a research area poses several problems whose solution is of general interest to artificial intelligence and computer science. In this section, we will discuss the problems of system organization, heuristic evaluation and credit assignment, and syntactic and semantic analyses of errorful strings.

System organization

Any speech recognition system of the complexity of Hear-Say, which attempts to include all the available sources of knowledge, will be large. Further, to equal human performance it must sometimes be able to answer questions even before they are completed. This means that the system may have to be segmented into subprograms which are paged-in or overlayed and every subprogram must do its part as soon as it is able to. To achieve this smoothly the system must provide facilities for inter-process communication and interruption. The co-routine mechanism can provide these facilities but only at pre-programmed points. This can sometimes lead to irrevocable loss of data if an appropriate program is not activated in time to process the incoming utterance.

A parallel program organization, in which independently scheduled programs perform their respective operations seems appropriate. The required working set can be paged-in at a given time with variable quantum times depending on the priority of the process. Presently available time-sharing systems can perform this process except that many of them do not provide facilities for several programs to work a single task. The systems organization problem is expected to be a major obstacle in the immediate realization of demonstrable speech understanding systems.

Heuristic evaluation and credit assignment

One of the features of existing speech recognition systems, and undoubtedly of future ones as well, is the existence of error at every level of analysis and the consequent proliferation of heuristic devices throughout the system to control such error and permit recycling with improved definitions of the situation. Almost entirely missing from the literature, not only of speech recognition but elsewhere in artificial intelligence as well, are techniques for evaluating performance characteristics of proposed algorithms and heuristics. By techniques, we include both suitable instrumentation and experimental design to measure accuracy, response time, cost, etc. in relation to vocabulary, language, and context. Until such techniques are developed and applied to existing components of a speech-understanding system, these components should be considered of questionable value in an applied system.

Syntactic and semantic analysis of errorful strings

A particular difficulty that stands in the way of using syntax and semantics to help with speech recognition is the lack of grammaticality and general well-formedness in free speech. Although one may legislate against some of the difficulties in written language, it is harder to do so in spoken language. Not only do people "humm" and "hah", and clear their throats, they utter fragments: "Now the ... th'... ...oh well..they are plying flames --- I mean flying planes". We belive a whole set of new language analysis tools will have to be developed before we can expect to have sophistacted cooperation between speech and understanding components of a single system.

CONCLUSION

In this paper, we have attempted to show that while recognition of spoken English seems distant, restricted language recognition systems of substantial utility can be built within this decade. These systems should be able to accept continuous speech, from many cooperative speakers of general American dialect, in a quiet room, over a good quality microphone, allowing slight tuning of the system per speaker, using a slightly selected vocabulary of 1000 words, with highly artificial syntax, in a task like the data management task with less than 10% semantic error and work in real-time. However, such a system will only materialize if we avoid duplication of research and begin working on the main research problems immediately.

ACKNOWLEDGEMENTS

The author would like to thank his colleagues, Allen Newell, Lee Erman and Richard Neely, for many interesting discussions on the subject.

REFERENCES

[1] Astrahan, M., "Speech Analysis by Clustering or the Hyperphoneme Method," AI Memo 124, Computer Science Department, Stanford University, Stanford, California (1970).

[2] Bobrow, D.G. and D.H. Klatt, "A Limited Speech Recognition System," Proc. FJCC (1968) 305-318.

[3] Chomsky, N. and M.Halle, "The Sound Pattern of English," Harper and Row, New York (1968).

[4] Fant, G., "Acoustic Theory of Speech Production," Mouton and Company: The Hague (1960).

[5] Flanagan, J.L., "Speech Analysis, Synthesis, and Perception," Academic Press: New York (1965).

[6] Gold, B., "Word Recognition Computer Program," RLE Report No. 452, MIT, Cambridge, Mass. (1966).

[7] Hyde, S.R., "Automatic Speech Recognition: Literature Survey and Discussion," RDR No. 45, Post Office Research Department, Dollis Hill, London N.W.2 (1968).

[8] Kozhevnikov, V.A. and L.A.Chistovich, "Speech: Articulation and Perception," Moscow-Leningrad (1965). Translated by Joint Publication Research Service, Washington, D. C.

[9] Lehiste, I., "Readings in Acoustic-Phonetics," MIT Press, Cambridge, Mass. (1967).

[10] Londgren, N., "Machine Recognition of Human Language," IEEE Spectrum 2, Nos. 3 and 4 (1965).

[11] Newell, A., "Heuristic Programming: Ill Structured Problems," in J. S. Avonofsky (ed.), Progress in Operations Research, Vol 3 (John Wiley and Sons) 363-415.

[12] Newell, A., J.Barnett, J.Forgie, C.Green, D.Klatt, J.C.R.Licklider, J.Munson, R.Reddy, and W.Woods, "Final Report of a Study Group on Speech Understanding Systems," Comp. Sci. Dept., Carnegie-Mellon Univ., Pittsburgh, (1971).

[13] Pierce, J.R., "Whither Speech Recognition," J. Acoust. Soc. Am. 46 (1966) 1049-1051.

[14] Reddy, D. R., "Computer Recognition of Connected Speech," J. Acoust. Soc. Am. 42 (1967) 329-347.

[15] Reddy, D.R. and P.J.Vicens, "A Procedure for Segmentation of Connected Speech," J. Audio Eng. Soc., 16,4 (1968) 404-412.

[16] Saki, T. and S.Doshita, "The Automatic Speech Recognition System for Conversational Sound," IEEE Trans. on Electronic Computers, 12 (1963) 835.

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

[18] Tappert, C.C., N.R.Dixon, D.H.Beetle, and W.D.Chapman, "The Use of Dynamic Segments in the Automatic Recognition of Continuous Speech," IBM Corp., RADC-TR-70-22, Rome Air Development Center, Rome, New York (1970).

[19] Vicens, P.J., "Aspects of Speech Recognition by Computer," Pn.D. Thesis, Computer Science Department (Report No. 127), Stanford University, California (1969).

[20] Vysotskiy, G.Y., B.N.Rudnyy, V.N.Trunin-Donskoy, and G.I.Tsemel', " Experiment in Voice Control of Computers," Isvestiya Akademii Nauk SSSR, Moscow, Teknicheskaya Kibernetika, No. 2 (1970) 134-143.

[21] Zagoruiko, N.G., "Automatic Recognition of 200 Oral Commands," Proc. of Computational Systems 37 (Novosibirsk, 1969) 73-76.

Reprinted from:
Reddy, D.R., C.G.Bell, and W.A.Wulf, Speech Recognition in a Multiprocessor Environment, Proc. IEEE Conf. on Automatic Control, Miami (Dec. 1971).

SPEECH RECOGNITION IN A MULTIPROCESSOR ENVIRONMENT

D. R. Reddy, C. G. Bell, and W. A. Wulf
Computer Science Department
Carnegie-Mellon University
Pittsburgh, Pa. 15213

INTRODUCTION

When a person plays chess or proves a theorem, most people seem to agree that he is exhibiting intelligent behavior. Answering a trivial question, watching a TV show, or driving a car do not seem to belong to this category. One appears to do these tasks without any conscious effort. This raises three questions:

1. Does a human being use different mechanisms for perceptual and intellectual activities?

2. Why is it that computers seem to have so much trouble performing such perceptual tasks?

3. What is the role of perception research in Artificial Intelligence?

There appears to be no simple answers to these questions. After presenting some results and problems that arise in speech recognition research in a multiprocessor environment, we will attempt to discuss these issues.

PRESENT STATE OF THE ART

Lindgren (1965), Hyde (1968), and Hill (1971) provide excellent surveys of the state of the art. We will illustrate the present state by considering the Vicens-Reddy speech recognition system (Vicens, 1969). The structure of this system is illustrated in Figure 1. A preprocessor extracts a set of parameters from the signal. A phone segmentation and recognition procedure divides this continuum of parameters into discrete parts and assigns labels such as vowel, fricative, stop, etc. This description is then used to select a list of likely candidates from a lexicon of acoustic descriptions of words. A sophisticated match procedure compares the parameters of the likely candidates to obtain a best match. If the match procedure finds at least one candidate with a high enough score, then it is chosen as the result of the recognition process. If no satisfactory match is found, the incoming utterance is entered into the lexicon along with the name of the utterance provided by the user.

This system can recognize up to 500 isolated words of a cooperative single speaker with less than 5% error rate in close to real time after three to four rounds of training. It can recognize multiple speakers (approximately 10) and highly restricted connected speech but only with significant deterioration in performance.

THE HEAR-SAY SYSTEM

HEAR-SAY is a speech recognition system currently under development at Carnegie-Mellon University (Reddy, Erman, and Neely, 1970). It represents an attempt to build a general purpose recognition system which will eventually be able to recognize connected speech of many different speakers in several restrictied task domains. Figure 2 gives a functional flow chart of various subprocesses within the system. The recognition part of the system is similar in some respects to Figure 1. However, many more sources of knowledge (lexical, phonological, syntactic, semantic) are brought to bear on the recognition process. There is extensive feed-back and feed-forward within the system. A more detailed description of various components of this system is given in Reddy (1971).

☆ Related issues are also discussed in Newell, et al. (1971). See Erman (1972) and Neely (1972) for details of implementation of parts of the HEAR-SAY system.

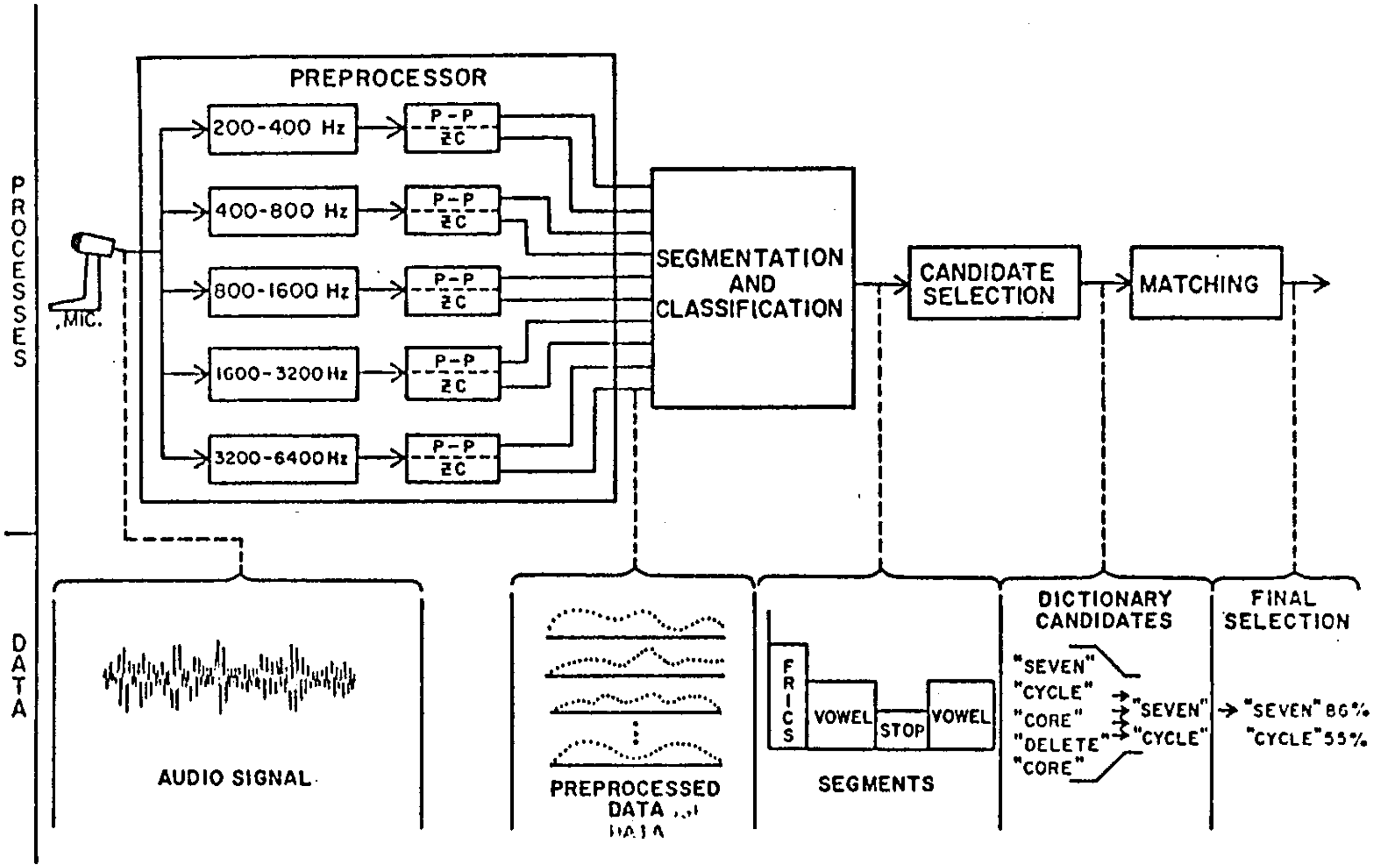


Figure 1. The Structure of the Vicens-Reddy Recognition System

Figure 2. The HEARSAY System
(See Page 5-7)

Here we will mainly address ourselves to the questions of systems organization. If the HEAR-SAY system is to equal human performance in these limited task domains, it must be able to answer trivial questions as soon as they are uttered (some times even before they are completed). This implies that various modules of the system should be able to operate on the incoming data as soon as they are able to do so, without waiting for the completion of the whole utterance. This suggests the use of co-routine structures. The feed-forward and feed-back mechanisms imply rich connectivity among these co-routines.

However, these co-routines must be able to interrupt their processing at unpreprogrammed points. It may become necessary for a routine to interrupt other routines in the midst of their computation for one of two reasons. First, if the preprocessing program is not activated in time to process the incoming utterance (at high data rates) it could lead to irrevocable loss of data. Second, since the main goal of the HEAR-SAY system is to recognize the utterance as soon as possible, it has to bring to bear the full power of every source of knowledge available to it. Suppose the semantic routine obtains some information which would make the current hypotheses of other routines invalid. It ought to be able to broadcast this information to other routines without their having explicitly to interrogate the system for this additional piece of knowledge. While this type of parallel program operation can be simulated with difficulty using a single processor (by each module checking the status of a global variable every few statements*), it seems to call for a computer organization in which several parallel processors can cooperate in solving a single problem. In some sense, this is the inverse of a time sharing system in which a single processor is used to solve several tasks at the same time. However, the kind of parallel organization required lies in a different direction from schemes such as ILLIAC-4 (Bell and Newell, 1971), since the processors are not in lock-step.

----

* If a time-sharing system is designed so that it will accept program-generated interrupts to other programs, and if programs are permitted to service their interrupts without the monitor providing mandatory interrupt handling service, then this problem would become somewhat simplified. It is interesting that few existing systems provide such facilities.

THE CMU MULTIMINIPROCESSOR SYSTEM

Various research needs, including the above need for cooperating parallel processes, have led to our present plans to construct a multiminiprocessor computer (C.mmp) with sharable global memory and with facilities for interprocess communication. Figure 3 illustrates the PMS (processor-memory-switch structure, Bell and Newell, 1971) of the proposed multiprocessor system. It consists of a 16 × 16 cross point switch which connects 16 PDP-11 processors to 16 high speed memory modules. The architecture of this system is discussed in Bell et al. (1971).

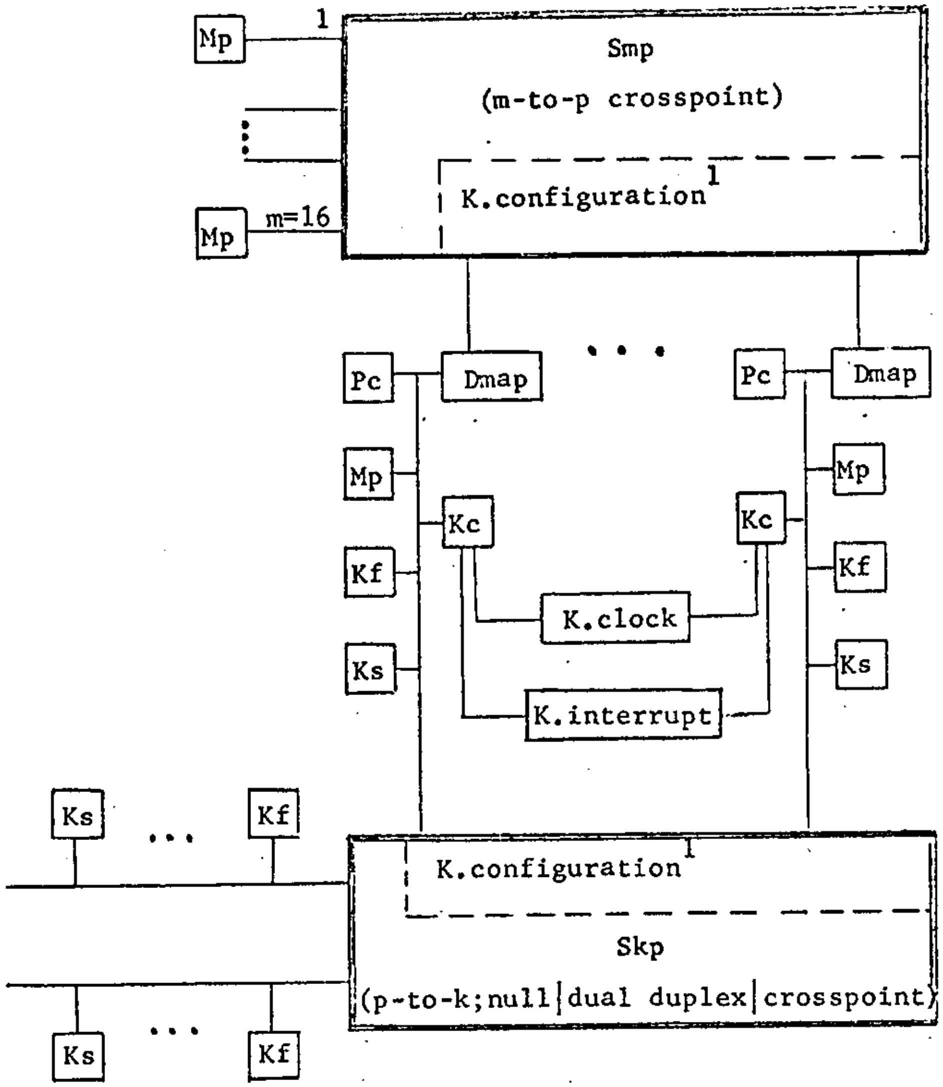
In addition to designing and constructing the cross point switch and the memory mapping device, one has to develop operating systems, languages and program debugging tools that are capable of operating in a multiprocessing environment. Here we will briefly discuss the problems to be solved in these areas.

Most existing operating systems are designed for operation with one or two processors. The allocation of resources among N processors solving K problems is the main problem facing a multiprocessor operating system designer. If several of these processors are attempting to solve the same problem, then one also has to solve the problems of memory sharing and facilities for interprocess communication, such as programmable interlocks, programmable interrupt handling, and program initiated interrupts.

Most higher level languages are inadequate for the specification of parallel algorithms. Languages and compilers must provide facilities through which several independent programs can refer to the same global data structures. Control statements for monitoring, interrupt processing, and processor and memory interlocks should be part of the language structure.

Program debugging in a multiprocessor environment, when several processors are cooperating to perform a task, also presents several new problems. A display-oriented diagnostic system showing what process is active in any given processor and what data structures are being modified at any given time seems a necessary and integral part of such a system.

Even when all the above systems have been developed, we still have the problem of specifying speech recognition algorithms in such a way that several cooperating processes can work on the incoming utterance at the same time. While the general structure of the system may be clear (see Figure 2), many of the details of interprocess communication have yet to be worked out.



where: Pc/central processor; Mp/primary memory; T/terminals;
Ks/slow device control (e.g., for Teletype);
Kf/fast device control (e.g., for disk);
Kc/control for clock, timer, interprocessor communication

[1] Both switches have static configuration control by manual and program control

Fig. 3. Proposed CMU multiminiprocessor computer/C.mmp.

DISCUSSION

We will now attempt to discuss the questions raised in the introduction based on our limited knowledge in trying to provide speech input to computers. The role of perception research in Artificial Intelligence seems clearer than the rest. Problems in perception are typified by high data rates, large masses of data and the availability of many sources of knowledge. Contrast this to many problem solving systems in which weaker and weaker methods are used to solve a problem using less and less information about the actual task. Computers have a great deal of trouble performing perceptual tasks becuase we do not yet know how to effectively bring to bear all the sources of knowledge in problem solution. We may need many different representations of the task domain with many different mechanisms to meet the performance requirements. Thus, the role of perception research in Artificial Intelligence is to address itself to the questions of task representations, data representations, and program organizations which will permit effective use of many sources of knowledge in solving problems involving high data rates and large masses of data in close to real time.

We do not at present know whether a human being uses different types of mechanisms for perceptual and intellectual tasks. Our conjecture is "no, he doesn't; it is just a matter of how effectively he is able to use the available mechanisms." For speech and vision, he probably has many different representations of the data (resulting from, say, many different observations of the same scene), and different mechanisms are able to access and process this data in parallel (not unlike our multiprocessor!). If this conjecture is true, then a person should be able to play master's level chess if he begins to learn to play chess at the age of two and continues to devote a major part of his waking life to playing chess for several years.

In conclusion, we can say that cooperating parallel processes which can effectively utilize all the available sources of knowledge appear to be promising. The eventual success of this effort will depend on our ability to solve the problems of system organization, algorithm specification, and error detection and correction in a parallel processing environment. If successful, this project may be a forerunner to computer control of complex processes with significant feed-forward, feed-back, and critical performance requirements.

ACKNOWLEDGEMENTS

We would like to thank Allen Newell whose criticisms and views on Artificial Intelligence, speech, and multiprocessor architecture have been very valuable; Lee Erman and Richard Neely who are mainly responsible for the present version of multiprogrammed speech recognition system; and Bill Broadley who is responsible for the design and interfacing to the PDP-10 of speech devices and the multiprocessor.

REFERENCES

[1] Bell, C.G. and A.Newell (1971), "Computer Structures," McGraw Hill, N. Y.

[2] Bell, C.G., et al. (1971), "C.mmp: The CMU Multiminiprocessor Computer," Technical Report, Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pa.

[3] Erman, L.D. (1972), Ph.D. Thesis (in preparation).

[4] Hill, D.R. (1971), "Man-Machine Interaction Using Speech," in F. L. Alt, M. Rubinoff, and M. C. Yovits (eds.), Advances in Computers, N.Y.: Academic Press, Vol. 11, pp. 165-230.

[5] Hyde, S.R. (1968), "Automatic Speech Recognition: Literature, Survey, and Discussion," Research Dept. Report No. 35, P.O. Research Dept., Dollis Hill, London, N.W. 2.

[6] Lindgren, N. (1965), "Machine Recognition of Human Language," IEEE Spectrum, 2, Nos. 3 and 4.

[7] Neely, R.B. (1972), Ph.D. Thesis (in preparation).

[8] Newell, A., et al. (1971), "Speech Understanding Systems: Final Report of a Study Group," copies available from Department of Computer Science, Carnegie-Mellon University, Pittsburgh, Pa.

[9] Reddy, D.R. (1971), "Speech Recogntion: Prospects for the Seventies," Proceedings of IFIP 71, Ljubljana, Yugoslavia, Invited paper section, pp. I-5 to I-13.

[10] Reddy, D.R., L.D.Erman, and R.B.Neely (1970), "The CMU Speech Recognition Project," Proceedings of IEEE Conf. System Sciences

and Cybernetics, Pittsburgh, Pa.

[11] Vicens, P.J. (1969), "Aspects of Speech Recognition by Computer," Ph.D. Thesis, Computer Science Department, Stanford University, Stanford, Ca.

Reprinted from:
Reddy, D.R., L.D.Erman, and R.B.Neely, A Mechanistic Model of Speech Perception, Proceedings of AFCRL-IEEE Conf. on Speech Communication and Processing, Boston (April 1972) pp. 334-337.

A MECHANISTIC MODEL OF SPEECH PERCEPTION

D. R. Reddy, L. D. Erman, and R. B. Neely

Carnegie-Mellon University
Pittsburgh, Pa. 15213

SUMMARY

This paper proposes an alternative to motor theory and analysis-by-synthesis models of speech perception with emphasis on efficient machine realization of the model. Our model can be characterized as a "hypothesize-and-test" model of perception. It consists of a small set of cooperating parallel processes, each of which is independently capable of decoding the incoming utterance. Each of these parallel processes has heuristics for generation and verification of hypotheses based on a semantic, syntactic or lexical representation of the language to be perceived. These processes are able to guide and/or reduce the search space of each other as various subparts of the utterance are recognized. Details of a recognition system which incorporates these ideas is presented.

THE MODEL

This paper presents a model of speech perception which has been arrived at not so much by conducting experiments on how humans perceive speech but in the process of constructing several speech recognition systems using computers. The emphasis has been on developing efficient recognition algorithms, and little on modeling of known human perceptual behavior. The general framework (for a model) that evolved is different from some previously proposed models by Liberman et. al. (1962) and Halle and Stevens (1962) which imply that perception takes place through the active mediation of motor centers associated with speech production. Our results tend to support "sensory" theories advanced by Fant (1964) and others, in which speech decoding proceeds without the active mediation of speech motor centers. Our present model consists of a set of cooperating parallel processes each of which is capable of generating hypotheses for decoding the utterance; the task of recognition is then reduced to one of verification of the hypotheses.

It is not our intention to propose yet another speculative model of speech perception. The main purpose of this paper is to propose that, in addition to stimulus-response studies and neuro-physiological modeling, speech scientists should also make extensive use of information processing models in the study of speech perception. The notion of an information processing model reflects a current trend in cognitive psychology to view man as an information processor i.e., that his behavior can be seen as the result of a system consisting of memories containing discrete symbols and symbolic expressions, and processes which manipulate these symbols (Newell, 1970). The main advantage of this approach to speech perception studies is that it permits a researcher to look at the total problem of speech perception at a higher functional and conceptual level than is possible with the other two approaches. (To attempt to study the total problem of speech perception by formulating a neuro-physiological model would be like attempting to understand the workings of a TV set by looking at the flow of electrons through a transistor.) After presenting the basic ideas in the model, we will present the details of a recognition system which incorporates these ideas and discuss the implications of the model.

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

The notion of a set of independent parallel processes, each of which is capable of generating hypotheses for verification, appears to be new. The need for a set of independent parallel processes arises in our model because of the requirement that the absence of one or more sources of knowledge should not have a crippling effect on the performance of the model. That semantic context should not be essential for perception is illustrated by overheard conversations among strangers. That syntactic or phonological context should not be essential is illustrated by conversations among children. That lexical representation is not essential is illustrated by our recognition of new words and nonsense syllables. In our model, the absence of one or more sources of knowledge has the effect of deactivating those processes, and recognition proceeds (albeit more slowly and with lower accuracy) using the hypotheses generated by the remaining processes.

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

The notion of hypothesize-and-test is not new. It has been used in several artificial intelligence programs (Newell, 1969). It is equivalent to analysis-by-synthesis if the "test" consists of matching the incoming utterance with a synthesized version of the hypothesis generated. In most cases, however, the "test" is of a much simpler form; for example, it is not necessary to generate the whole formant trajectory when a simpler test of the slope can provide the desired verification. This not only has the effect of reducing the computational effort but also increases the differentiability between phonemically ambiguous words.

Acquisition and representation of various sources of knowledge of the model are currently programmed into the system. There have been several studies on language acquisition (most of which are S-R theories), but again our feeling is that an information processing model would permit a better understanding of the issues concerning the organization of long term memory and additions, deletions and modifications of various sources of knowledge. There are several proposals for organization of memory (Quillian, 1966; Norman, 1968; Winograd, 1970). Their implications for speech perception are yet to be studied by speech scientists.

A REALIZATION OF THE MODEL

HEARSAY is a speech recognition system which incorporates many of the ideas presented in the previous section, and is presently under development at Carnegie-Mellon University. It is not restricted to any particular recognition task. Given the syntax and the vocabulary of the language and the semantics of the task, HEARSAY will attempt recognition of utterances in this language. Figure 1 gives a functional flowchart of a part of the HEARSAY system. A more detailed, but earlier, description of the goals and various components of this system are given in Reddy, Erman and Neely (1970) and Reddy (1971).

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

Figure 1 shows three independent processes; acoustic, syntactic and semantic. We will give a short description of how these processes cooperate in recognizing "king bishop pawn captures knight on king four". Let us assume that this is a legal move (otherwise, at some stage of processing, the system will reject it as semantically inconsistant). The incoming utterance is preprocessed to extract parameters, segmented (based on acoustic similarity), and segmental features are determined. The exact nature of parameters, segments or features is not important as long as it is consistant with (or can be equivalenced to) the phonetic descriptions in the lexicon. Suppose for the purpose of this description the system has already recognized "King ------- captures -------" and this is stored as the "currently accepted partially recognized utterance" (see Fig. 1).

Hypothesis Generation

The three independent processes are now in a position to generate hypotheses about the unrecognized portion of the utterance. The acoustic hypothesizer does not have any knowledge of the syntax or semantics of the situation, but can use the gross features in the "partial symbolic utterance description" (such as /ʃ/ of "bishop") to

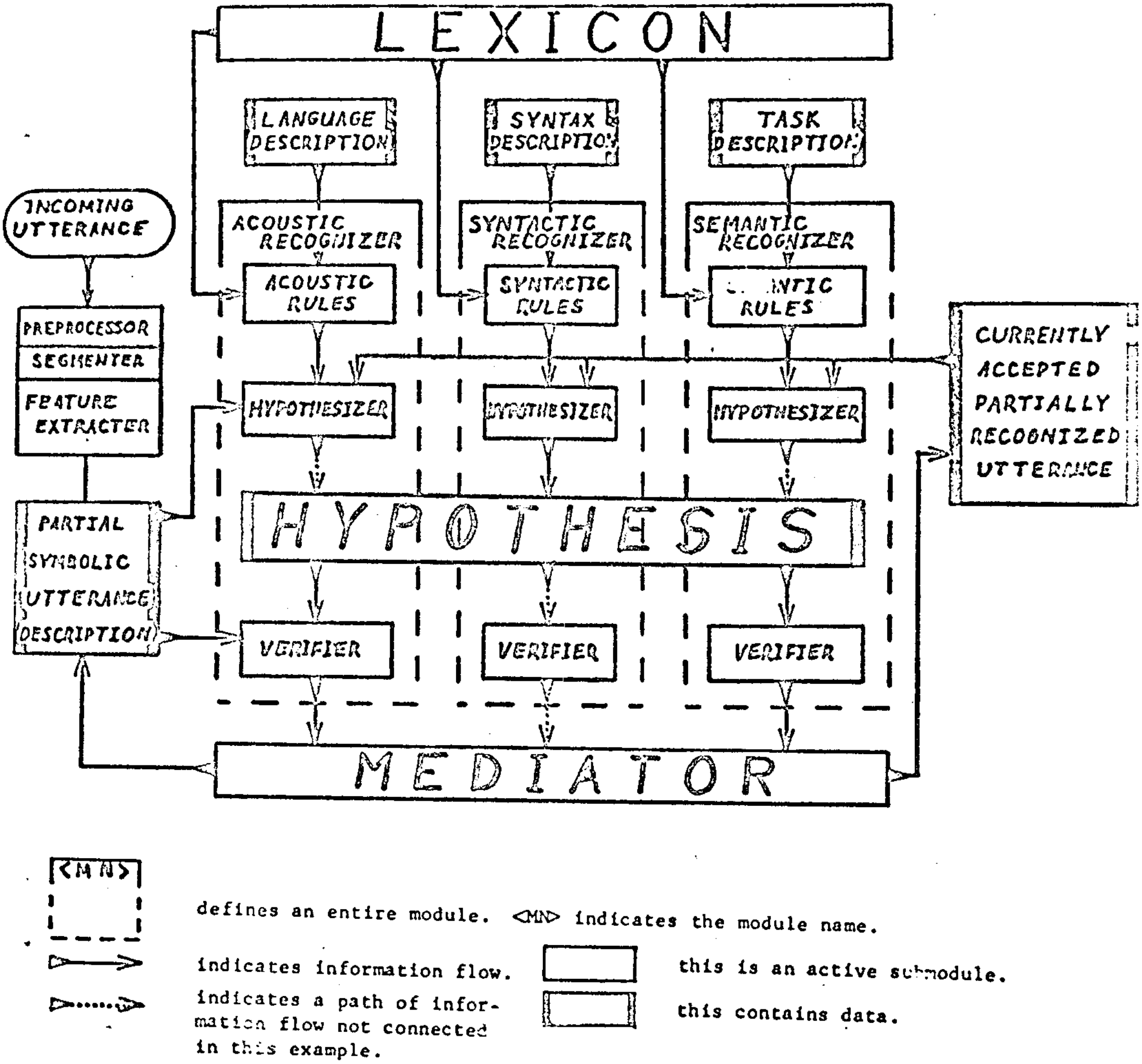


Figure 1. A Functional Model of the HEARSAY System

retrieve those words of the lexicon that are consistent. Within-the-word feature variations resulting from co-articulation are (presently) encoded into the lexical description. Between-word co-articulation effects are determined wherever applicable through the use of the "currently accepted partially recognized utterance" which provides the boundary phonemes.

The syntactic hypothesizer generates a partial parse-tree based on the partially recognized utterance which it then uses to predict words that can follow in that syntactic situation. In our present example it would have two partial parse trees, one based on "king" and the other based on "captures". It then selects the hypothesis which would result in the least number of words to be verified.

The semantic hypothesizer contains, as a subpart, a chess program (Gillogly, 1971) which generates an ordered list of moves that are possible in a given situation. In our example, the hypothesizer then concentrates on only the "capture" moves that start with the word "king". If there are none, then there is an inconsistancy in the "currently accepted partially recognized utterance". This may be due to an illegal statement or incorrect recognition. In the latter case the partially recognized utterance is modified by replacing the weakest link by the second best choice for that position.

There are several strategies for using independent hypothesis generators. One is the notion of most plausible hypothesis. In this case, each hypothesizer associates a confidence number to each hypothesis. Of all the hypotheses, the most plausible one is selected for verification. A computationally more effective procedure (in case there is only a single processor on the computer) is to select that process which has in the past generated most effective hypotheses. In the case of chess, the semantic hypothesizer is substantially more efficient. But there are many low context tasks where the semantic situation provides the least help.

Hypothesis Verification

The task of a verifier is to determine whether a given hypothesis is consistant with the context presently available to it. Consider the case in which only a single process is active, say, a task which has no syntax or semantics. Then the role of the verifier is to further restrict and/or validate the hypothesis. In the present example, an acoustic hypothesizer might select all the words that contain a sibilant, e.g., "bishop", "kings", "queens", "takes", "captures". A more detailed matching of features and the use of co-articulation rules at the word boundary between "king" and the hypothesized words would permit elimination of most of the possibilities. Detailed matching often implies generation of a test. For example, if the verification to be made is between "sit", "spit", and "split", the presence of /s/, /ɪ/, /t/ and the transitions between /ɪ/ and /t/ are irrelevant. The verifier generates tests for the presence or absence of a stopgap and for the presence of /l/-like formant structure following the stop-gap.

The role of syntactic and semantic verifiers in the case of a single active process is much more limited. They can attempt more sophisticated heuristics for better use of the "currently accepted partially recognized utterance". The nature of these heuristics is unclear at present. If more than one process is active then syntactic and semantic verifiers can play a significant role by attempting to eliminate those hypotheses (generated by other processes) that are either syntactically or semantically inconsistent.

Verifiers can be activated independently to validate the hypothesis, or sequentially to consider only those hypotheses considered valid by the preceeding verifiers.

Control of the Processes

The verification process continues until a hypothesis is found which is acceptable to all the verifiers with a high enough level of confidence. All the unverified hypotheses are stored on a stack for the purpose of back-tracking at a later stage. Given an acceptable hypothesis, the mediator updates the "currently accepted partially recognized utterance" and updates the "partial symbolic utterance description" with additional features that were discovered during the process of hypothesis generation and verification. If the utterance still has unrecognized portions of speech and if the interpretation of the utterance is still unclear, then all the active processes are reactivated to generate hypotheses in the new context. If there are no unrecognized portions of speech in the utterance and the sentence is uninterpretable, the knowledge acquisition part of the system (presently manual and not shown in Figure 1) is activated to update the lexicon and the acoustic, syntactic and/or semantic rules.

DISCUSSION

The main ideas present in the model are independent parallel processes, nature of cooperation, and nature of perception (sensory vs. motor models). Several questions arise in this context that are of interest to speech scientists and cognitive psychologists interested in human speech perception. As we stated earlier, our main interest continues to be efficient machine realizable models for speech recognition. However, since the human is the most effective speech perceiver to date, it is of interest whether he uses similar mechanisms.

It is known that, at a higher problem solving level, a human being behaves essentially as a serial information processor (Newell and Simon, 1972). It is also known that parallel processing occurs at the preprocessing levels of vision and speech. What is not known is whether there are several independent processes or a single sophisticated process at the perceptual level which can effectively use all the available sources of knowledge.

The second question is how these sources of knowledge cooperate with each other. There are experiments (Miller and Isard, 1963; Collins and Quillian, 1969) which can be interpreted to show that perception is faster or more intelligible depending on the number of available sources of knowledge. Any model of speech perception must deal with the nature and structure of the interaction between various sources of knowledge. Earlier models tend to ignore this question.

The question of whether humans use a sensory model or a motor model is probably not important but the implications for machine recognition are clear. There are several other questions that arise such as "what is the effect of an increase in vocabulary for human perception", "do human beings parse sentences from left to right" and so on. We believe that experiments can be designed within the framework of information processing models which will provide answers to many of these questions.

This paper illustrates our present model for machine perception of speech and provides a framework for human speech perception experiments. General models of perception, however limited or incomplete, have in the past played an important role in stimulating research. Our model differs from earlier models in that it provides specific structures for data and control processes that are useful in speech perception. A main advanage of this model is that one can now design experiments in which the same material is presented to both man and machine, observe the similarities and differences, and revise the model.

ACKNOWLEDGEMENT

We would like to thank Allen Newell, who read this manuscript and made several helpful comments in its preparation.

REFERENCES

[1] Collins, A.M. and M.R.Quillian (1969), "Retrieval Time from Semantic Memory," J. Verb. Learn. and Verb. Behavior, 8, 204-247.

[2] Fant, G. (1964), "Auditory Patterns of Speech," in W. Wathen-Dunn (ed.), Models for the Perception of Speech and Visual Form, MIT Press.

[3] Gillogly, J., (1971). "The Technology Chess Program", Tech Report, Computer Science Dept., Carnegie-Mellon University.

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

[5] Liberman, A.M., F.S.Cooper, K.S.Harris, and P.F.MacNeilage (1962), "A Motor Theory of Speech Perception," Proc. of the speech communication seminar, 2, KTH, Stockholm.

[6] Miller, G.A. and S.Isard, "Some Perceptual Consequences of Linguistic Rules," J. Verb. Learn. and Verb. Behavior, 2, 217-228.

[7] Newell, A. (1969), "Heuristic Programming: Ill-Structured Problems," in J. S. Aronofsky (ed.), Progress in Operations Research, 3, Wiley, 363-415.

[8] Newell, A. (1970), "Remarks on the Relationship between Artificial Intelligence and Cognitive Psychology," in R. Banerji and M. D. Mesarovic (eds.), Non-Numerical Problem Solving, 363-400, Springer-Verlag.

[9] Newell, A. and H.A.Simon (1972), Human Problem Solving, Prentice-Hall.

[10] Norman, D. (1968), Memory and Attention: An Introcuction to Human Information Processing, Wiley.

[11] Quillian, M. R. (1966), "Semantic Memory," Ph. D. Thesis, Carnegie-Mellon University; reprinted in M.Minsky (ed.) (1968), Semantic Information Processing, MIT Press.

[12] Reddy, D. R. (1971), "Speech Recognition: Prospects for the Seventies," Proc. of IFIP 71, I-5 to I-13.

[13] Reddy, D.R., L.D.Erman, and R.B.Neely (1970), "The CMU Speech Recognition Project," Proc. of IEEE Conf. Sys. Sci. and Cybernetics, Pgh.

[14] Winograd, T. (1970), "Procedures as a Representation of Knowledge in a Computer Program for Understanding Natural Language," Ph.D Thesis, MIT.

Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY *(Corporate author)* Computer Science Department Carnegie-Mellon University Pittsburgh, Pa. 15213	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

WORKING PAPERS IN SPEECH RECOGNITION - I

4. DESCRIPTIVE NOTES *(Type of report and inclusive dates)*

5. AUTHOR(S) *(First name, middle initial, last name)*

R. Reddy, L. Erman, R. Neely, et al.

6. REPORT DATE April 21, 1972	7a. TOTAL NO. OF PAGES 79	7b. NO. OF REFS 87

8a. CONTRACT OR GRANT NO. F44620-70-C-0107	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO. 9769	
c. 61102F	9b. OTHER REPORT NO(S) *(Any other numbers that may be assigned this report)*
d. 681304	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES TECH OTHER	12. SPONSORING MILITARY ACTIVITY Air Force Office of Scientific Research(NM) 1400 Wilson Blvd. Arlington, Va. 22209

13. ABSTRACT

This report represents a collection of papers published in various conference proceddings that are not readily available for researchers working in the field of speech recognition. The papers reprinted are:

1. Reddy -- Speech Input Terminals (June 1970).
2. Reddy, Erman, and Neely -- The CMU Speech Recognition Project (October 1970).
3. Erman and Reddy -- Telephone Speech (August 1971).
4. Neely and Reddy -- Noise in Speech (August 1971).
5. Reddy -- Speech Recognition: Prospects (August 1971).
6. Reddy, Bell, and Wulf -- Speech Recognition in a Multiprocessor Environment (December 1971).
7. Reddy, Erman, and Neely -- A Mechanistic Model of Speech (April 1972).