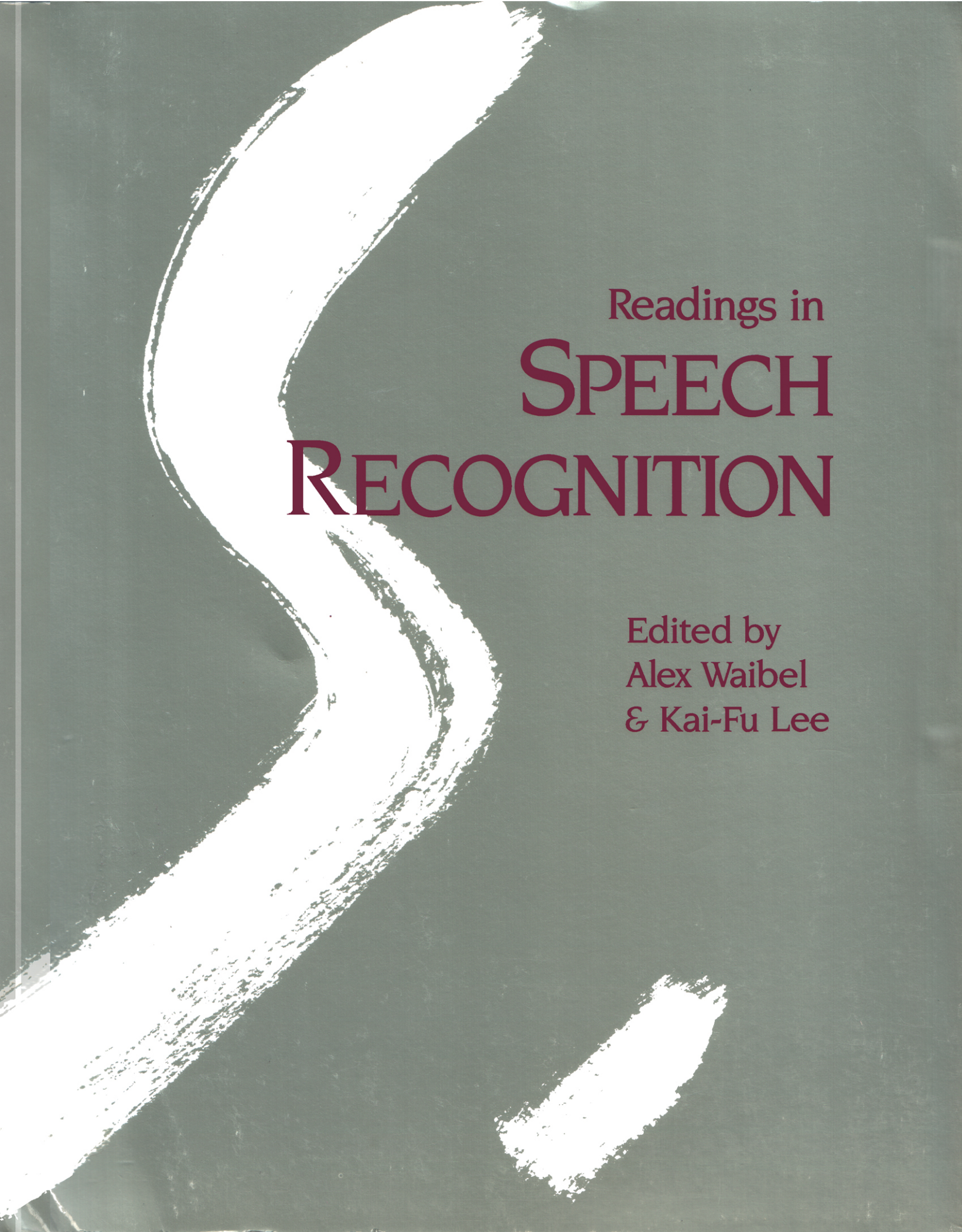
Readings in
SPEECH
RECOGNITION
Edited by
Alex Waibel
& Kai-Fu Lee

Readings in SPEECH RECOGNITION

This page intentionally left blank

Readings in SPEECH RECOGNITION

Edited by
Alex Waibel
& Kai-Fu Lee

Morgan Kaufmann Publishers, Inc.
San Mateo, California

Editor *Michael B. Morgan*
Production Manager *Shirley Jowell*
Copy Editor *Paul Medoff*
Cover Designer *Andrea Hendrick*
Typesetter *Technically Speaking Publications*

Library of Congress Cataloging-in-Publication Data

Readings in speech recognition/edited by Alexander Waibel
and Kai-Fu Lee.
p. cm.
ISBN 1-55860-124-4
1. Automatic speech recognition. 2. Speech processing systems.
I. Waibel, Alex. II. Lee, Kai-Fu.
TK7882.S65R42 1990
006.4'54--dc20

89-71329
CIP

MORGAN KAUFMANN PUBLISHERS, INC.
Editorial Office:
2929 Campus Drive
San Mateo, California
Order from:
P.O. Box 50490
Palo Alto, CA 94303-9953

Preface

Despite several decades of research activity, speech recognition still retains its appeal as an exciting and growing field of scientific inquiry. Many advances have been made during these past decades; but every new technique and every solved puzzle opens a host of new questions and points us in new directions. Indeed, speech is such an intimate expression of our humanity—of our thoughts and emotions—that speech recognition is likely to remain an intellectual frontier as long as we search for a deeper understanding of ourselves in general, and intelligent behavior in particular. The recent decade has not rested on the laurels of past achievements; the field has grown substantially. A wealth of new ideas has been proposed, painful and sweet lessons learned and relearned, new ground broken, and victories won.

In the midst of our excitement and eagerness to expand our horizons, we conceived this book to fill a real need. We were motivated in part by the desire to tell the casual observer what speech recognition is all about. More importantly though, we found ourselves much too often at the copier, copying what we felt was important background reading for our colleagues and students, whom we have the good fortune to work with or supervise. To be sure, there are several good textbooks that introduce speech processing in general or describe speech recognition in the context of a particular approach or technique. Yet, because the field has grown so rapidly, none of these books covers the more recent developments and insights.

The present *Readings in Speech Recognition* is intended to fill this need by compiling a collection of seminal papers and key ideas in the field that we feel a serious student of speech recognition should know about. Rather than presenting the material in predigested form, we believe that readers should be exposed to the original papers and learn about the ideas themselves. There is no better way than to learn directly from the field's pioneering efforts—the motivations and inspirations, the points of view and controversies, the partial attempts, difficulties and failures, as well as the victories and breakthroughs. In a field as dynamic as speech recognition, learning about the problems and being exposed to the creative process of solving them is just as important as learning about the current methods themselves. In order to make this book timely, we have purposely included not only classic papers but also a number of important recent developments, thus providing an up-to-date overview of the latest state of the art.

Beyond collecting some key papers, we have attempted to organize the major schools of thought into individual chapters and to give the reader perspective in the form of book and chapter introductions. The introductions highlight for each chapter some of the major insights, points of view, differences, similarities, strengths, and weaknesses of each technique presented. It is our hope that the casual reader will find these introductions useful as a quick guided tour and as an entry for selective reading to satisfy his or her curiosity about aspects of the field. For the serious student or system

developer, we hope that the introductions help to pass on some of the hard-learned lessons of research in decades past, provide pointers to important detail, and put any one particular technique into an overall perspective.

In editing this book, we have profited immensely from our colleagues, students and friends. In particular, the detailed comments and suggestions by several known leaders in the field who have reviewed our initial outline have added considerable balance and quality to this book—we gratefully acknowledge their contributions. We would like to thank Fred Jelinek and Erik McDermott for providing us with two original contributions for this volume. We are particularly indebted to Prof. Raj Reddy, one of the founders and pioneers of speech recognition. We were both fortunate to have grown into this field under his supervision during our own student years. Special thanks are also due to Mike Morgan and Shirley Jowell for their persistent reminders to keep moving ahead on draft revisions and organization and their tireless efforts to get this book to press in a timely fashion. We would also like to thank IEEE for giving us permission to reproduce their publications. Last, but not least, we would like to thank our wives, Naomi and Shen-Ling, for their patience during the preparation of this book.

Contents

Chapter 1

Why Study Speech Recognition?

1. Introduction

The goal of automatic speech recognition is to develop techniques and systems that enable computers to accept speech input. The problem of speech recognition has been actively studied since the 1950's, and it is natural to ask why one should continue studying speech recognition. Does it have practical utility? Is it interesting in the first place? What lessons will we learn from exploring the questions in speech recognition? In light of past activity, what aspects of the problem have already been solved? What are the challenges for future research? Do the rewards warrant our continued efforts?

We firmly believe that automatic speech recognition is a very rich field for both practical and intellectual reasons. Practically, speech recognition will solve problems, improve productivity, and change the way we run our lives. Intellectually, speech recognition holds considerable promise as well as challenges in the years to come for scientists and product developers alike.

1.1 Practical Utility

The performance of speech recognizers has improved dramatically due to recent advances in speech science and computer technology. With continually improving algorithms and faster computers, it appears that man–machine communication by voice will be a reality within our lifetime.

Even in the short term, many speech recognition applications will be possible. Information retrieval is a major component of these applications. For example, simple inquiries about bank balance, movie schedules, and phone call transfers can already be handled by small-to-medium sized vocabulary, speaker-independent, telephone-speech recognizers. While information retrieval is often telephone based, another application, data entry, has the luxury of using high-quality speech. Voice-activated data entry is particularly useful in applications such as medical and darkroom applications, where hands and eyes are unavailable as normal input medium, or in hands-busy or eyes-busy command-and-control applications. Speech could be used to provide more accessibility for the handicapped (wheelchairs, robotic aids, etc.) and to create high-tech amenities (intelligent houses, cars, etc.).

Whereas these short-term applications will increase productivity and convenience, more evolved prototypes could in the long-run profoundly change our society. A futuristic application is the *dictation machine* that accurately transcribes arbitrary speech. Such a device can further be extended to an *automatic ear* that "hears" for the deaf. An even more ambitious application is the *translating telephone* [Kurematsu88] that allows interlingual communication. The translating telephone requires not only speech *recognition*, but also speech *synthesis*, *language understanding* and *translation*. Finally, the ultimate *conversational computer* has all of these capabilities, as well as the ability of thought. Computers that listen and talk are the ultimate application of speech recognition.

1.2 The Intellectual Challenge and Opportunity

Like many frontiers of artificial intelligence, speech recognition is also still in its infancy. Speech and language are perhaps the most evident expression of human thought and intelligence—the creation of machines that fully emulate this ability poses challenges that reach far beyond the present state of the art.

The study of speech recognition and understanding holds intellectual challenges that branch off into a large spectrum of diverse scientific disciplines. Time and again the field has fruitfully and productively benefited from sciences as diverse as computer science, electrical engineering, biology, psychology, linguistics, statistics, philosophy, physics and mathematics. Among the more influencial activities within these disciplines are work in signal processing, pattern recognition, artificial intelligence, information theory, probability theory, computer algorithms, physiology, phonetics, syntactic theory, and acoustics. Speech-recognition research continues to be influenced and driven by scientists with different backgrounds and training, who have contributed a variety of important intuitions and who, in turn, have motivated aspects of ongoing research in their own fields.

The questions raised range from philosophical questions on the nature of mind to practical design consideration and implementational issues. Motivated by the desire to understand human intelligence, speech recognition can provide a good testing ground for an otherwise introspective and potentially subjective undertaking. Engineering design, in turn, is always evaluated against its progress toward the ultimate goal—unrestricted, free communication between man and machine—in a changing and uncertain world. This interplay between different intellectual concerns, scientific approaches, and models, and its potential impact in society make speech recognition one of the most challenging, stimulating, and exciting fields today.

2. Dimensions of Difficulty in Speech Recognition

Considering the immense amount of research over the last three decades, one may wonder why speech recognition is still considered an unsolved problem. As early as 1950s, simple recognizers have been built, yielding credible performance. But it was soon found that the techniques used in these systems were not easily extensible to more sophisticated systems. In particular, several dimensions emerged that introduce serious design difficulties or significantly degrade recognition performance. Most notably, these dimensions include

- Isolated, connected, and continuous speech
- Vocabulary size
- Task and language constraints
- Speaker dependence or independence
- Acoustic ambiguity, confusability
- Environmental noise.

We will now explain the difficulty involved in each of these areas.

The first question one should ask about a recognizer or a task is: *is the speech connected or spoken one word at a time?* Continuous-speech recognition (CSR) is considerably more difficult than isolated word recognition (IWR). First, word boundaries are typically not detectable in continuous speech. This results in additional confusable words and phrases (for example; "youth in Asia" and "Euthenasia"), as well as an exponentially larger search space. The second problem is that there is much greater variability in continuous speech due to stronger coarticulation (or inter-phoneme effects) and poorer articulation ("did you" becomes "didja").

A second dimension of difficulty is the size of the vocabulary. The vocabulary size varies inversely with the system accuracy and efficiency—more words introduce more confusion and require more time to process. Exhaustive search in very large vocabularies is typically unmanageable. The collection of sufficient training data becomes practically more difficult. Finally, word templates (or models) are untrainable and wasteful. Instead, one must turn to smaller subword units (phonemes, syllables), which may be more ambiguous and harder to detect and recognize. In order to realize a large vocabulary system, research in compact representation, search reduction, and generalizable subword units is essential.

Vocabulary size alone is an inadequate measure of a task's difficulty, because in many applications not all words are legal (or active) at a given time. For example, a sentence like "*Sleep roses dangerously young colorless*" need not be searched because of its illegal syntactic construction. Similarly, a sentence like "*Colorless yellow ideas sleep furiously*" is syntactically sound, but semantically absurd. A system with a semantic component may

eliminate such sentences from consideration. Finally, a sentence like *"I look forward to seeing you."* is much more likely to occur than *"Abductive mechanism is used in model generative reasoning,"* although both are meaningful sentences. A system with a probabilistic language model can effectively use this knowledge to rank sentences. All of the above examples require use of higher-level knowledge to constrain or rank acoustic matches. These knowledge sources, or language models can reduce an impossible task to a trivial one, but in so doing, severely limit the input style. The challenge in language modeling is to derive a language model that provides maximum constraint while allowing maximum freedom of input. Like the vocabulary size, the constraining power of a language model can be measured by *perplexity*[1], roughly the average number of words that can occur at any decision point.

In addition to vocabulary and linguistic constraints, there are a number of other constraints that can affect accuracy and robustness. The most prominent issue is that of *speaker dependence* as opposed to *speaker independence*. A speaker-dependent system uses speech from the target speaker to learn its model parameters. This strategy leads to good accuracy, but requires an inconvenient period for each new speaker. On the other hand, a speaker-independent system is trained once and for all, and must model a variety of speakers' voices. Due to their increased variability, speaker-independent systems are typically less accurate than speaker-dependent systems. In practice, some applications can be speaker dependent, while others require speaker independence. Both types of systems have been built and studied extensively.

Speech-recognition-system performance is also significantly affected by the acoustic confusability or ambiguity of the vocabulary to be recognized. While some recognizers may achieve respectable performance over relatively unambiguous words (e.g., "zero," though "nine"), such systems may not necessarily deliver acceptable recognition rates for confusable vocabularies (e.g., the words for the alphabetic letters, B, D, E, P, T, C, Z, V, G). A confusable vocabulary requires detailed high-performance acoustic pattern analysis.

Another source of recognition-system performance degradation can be described as variability and noise. Some examples include environmental noises (e.g., factory floor, cockpit, door slams), crosstalk (several people may be talking simultaneously), differing microphone characteristics (headset microphone, telephone receiver, table microphones), speaker induced noise (lipsmacks, pops, clicks, caughing, sneezing), speaking rate, and speaker stress (emotional, physiological).

With so many dimensions of difficulty, speech recognizers naturally have a wide range of accuracies. For example, for recognition of high-quality, read, legal credit-card numbers, a sentence accuracy of over 99.9% can be reached. On the end of the spectrum, recognition of noisy conversational speech with infinite vocabulary and no grammar far exceeds the capabilities of any system to date. While the ultimate goal of truly unrestricted, spontaneous, speech understanding may require decades of further research, many useful applications are achievable today, since most applications can impose restrictions along some of the dimensions outlined here. Credit-card numbers, telephone numbers, and zip codes, for example, require only a small vocabulary. Similarly, dictation may be limited, in some cases, to a "master's voice;" or follow a typical limited grammar, style, or vocabulary.

3. The Chapters of this Book

This book is intended to cover background material on speech recognition. We try to provide a cross section of today's most promising ideas and follow the evolution of speech recognition research in the past 20 years.

Chapter 2 includes two papers on the *background* of the speech recognition problem. They describe in greater detail the motivation, the difficulty, and the missing science in speech.

Chapter 3 describes the *front end* of speech recognizers, or *speech analysis*. Four papers here describe the most promising and popular digital representations of speech as used in most speech-recognition systems today.

Chapter 4 begins a four-part "schools of thought in speech recognition." This chapter describes the *template-based approach*, where units of speech (usually words) are represented by templates in the same form as the speech input itself. Distance metrics are used to compare templates to find the best match, and dynamic programming is used to resolve the problem of temporal variability. Template-based approaches have been successful, particularly for simple applications requiring minimal overhead.

1. See Paper 8.1 for a precise definition.

One criticism of template-based techniques was that they do not facilitate the use of human speech knowledge. Chapter 5 describes the *knowledge-based approach*, proposed in the 1970s and early 1980s. The pure knowledge-based approach emulates human speech knowledge using expert systems. Rule-based systems have had only limited success. A more successful approach segregates knowledge from algorithms and integrates knowledge into other mathematically sound approaches. The addition of knowledge was found to improve other approaches substantially.

Another weakness of the template-based approach is its limited ability to generalize. Chapter 6 describes the *stochastic approach*, which is somewhat similar to the template-based approach. One major difference is that probabilistic models (typically hidden Markov models, or HMMs) are used. HMMs are based on a sound probabilistic framework, which can model the uncertainty inherent in speech recognition. HMMs have an integrated framework for simultaneously solving the segmentation and the classification problem, which makes them particularly suitable for continuous-speech recognition. Most successful large-scale systems today use a stochastic approach.

One characteristic of HMMs is that they make certain assumptions about the structure of speech recognition, and then estimate system parameters as though the structures were correct. This has the advantage of reducing the learning problem, but the disadvantage of relying on often-incorrect assumptions. Chapter 7 describes the *connectionist approach*, which differs from HMMs in that many of these assumptions need not be made. Connectionist approaches use distributed representations of many simple nodes, whose connections are trained to recognize speech. Connectionist approaches is a most recent development in speech recognition. While no fully integrated large-scale connectionist systems have been demonstrated yet, recent research efforts have shown considerable promise. Some of the problems that remain to be overcome include reducing training time and better modeling of sequential constraints.

Spoken sentences always contain ambiguities that cannot be resolved by pure word-level acoustic-phonetic recognition. Successful sentence recognition must therefore incorporate constraints that transcend this level, including syntactic, semantic, and prosodic constraints. Chapter 8 on language processing reviews papers addressing this concern. Indeed, most of the best current large-scale recognition systems succeed by taking advantage of powerful language models.

Chapter 9, finally, gives a selection of papers that represents some of the seminal speech-recognition systems developed in the past two decades. The papers presented here are by no means a complete list of existing systems. Rather, we attempt to give a sample of some of the more-successful systems that have extended recognition capabilities along the dimensions discussed above while maintaining high recognition accuracy.

4. Further Study

The primary sources of information on speech recognition in the U.S. are *IEEE Transactions on Acoustics, Speech,* and *Signal Processing (ASSP), Computer Speech and Language*, and *Journal of the Acoustical Society of America (ASA). IEEE Transactions on ASSP* is an engineering journal that covers topics beyond speech recognition and has the widest readership. *Computer Speech and Language*, published by Academic Press, is a newer quarterly journal devoted to the processing of speech and language. By comparison, the *Journal of the ASA* contains fewer speech-recognition-system articles and emphasizes human-speech production, perception, and processing papers. The *IEEE ASSP Magazine* and the *Proceedings of the IEEE* also have special issues devoted to speech and speech recognition. Speech articles have also appeared in other IEEE Transactions, such as *Pattern Analysis* and *Machine Intelligence, Computer, Information Theory, Systems, Man, and Cybernetics*, and *Communication*. In Europe, the largest publication is the multinational *Speech Communications* published by the European Association of Signal Processing. Another publication is the journal of the British *IEE*. In Japan, major activity is reported in the *Journal of the Acoustical Society of Japan (ASJ)* and the journal of the *IECE*.

Major conferences which report research in speech recognition include: the IEEE; the International Conference on Acoustics, Speech, and Signal Processing (ICASSP); Acoustical Society of America; Speech Tech.; Eurospeech (in Europe); and International Conference on Spoken Language Processing (in Japan). Numerous other conference organized in Europe and Japan (such as the ASJ and the IECE) are held in their own languages.

Because speech is a relatively young science, most of the recent research is found in the journals

and conferences described above. There are a few books that cover all aspects of speech in general, including speech communication, processing, synthesis and coding, such as [Flanagan 72], [Rabiner 78], [Oshaughnessy 87], and [Furui 89]. In addition, there are a number of earlier anthologies similar to this one available for further study. Four of these, edited by Reddy [Reddy 75], Lea [Lea 80], Dixon and Silverman [Dixon 79], and Cole [Cole 80] focus on speech recognition. Another anthology edited by Fallside and Woods [Fallside 83] contains papers that contributed to a speech course. Finally, two books edited by Perkell and Klatt [Perkell 86] and Furui and Sondhi [Furui 90] have somewhat different emphases, but both contain substantial papers on speech recognition.

References

[Cole 80] Cole, R.A. *Perception and Production of Speech.* Lawrence Erlbaum Associates, Hillsdale, N.J., 1980.

[Dixon 79] Dixon, N.R. and Martin, T.B. *Automatic Speech and Speaker Recognition.* IEEE Press, New York, 1979.

[Fallside 83] Fallside, F., Woods, W.A. *Computer Speech Processing.* Prentice-Hall International, Englewood Cliffs, N.J., 1983.

[Flanagan 72] Flanagan, J.L. *Speech Analysis; Synthesis and Perception.* Springer-Verlag, Berlin, 1972.

[Furui 89] Furui, S. *Digital Speech Processing.* Marcel Dekker, Inc., New York, 1989.

[Furui 90] Furui, S., Sondhi, M. *Recent Progress in Speech Signal Processing.* Marcel Dekker, Inc., N.J., 1990.

[Kurematsu 88] Kurematsu, A. A Perspective of Automatic Interpreting Telephony. *Journal of the Inst. of Electronics*, Information and Communication Engineering, August, 1988.

[Lea 80] Lea, W.A. *Trends in Speech Recognition.* Speech Science Publishers, Apple Valley, Minn. 1980.

[Oshaughnessy 87] O'Shaughnessy, D. *Speech Communication; Human and Machine.* Addison Wesley, Reading, Mass., 1987.

[Perkell 86] Perkell, J.S., Klatt, D.M. *Variability and Invariance in Speech Processes.* Lawrence Erlbaum Associates, Hillsdale, N.J., 1986.

[Rabiner 78] Rabiner, L.R., Shafer, R.W. *Digital Processing of Speech Signals.* Prentice-Hall International, London, 1978.

[Reddy 75] Reddy, D. R. (editor). *Speech Recognition.* Academic Press, New York, 1975.

This page intentionally left blank

Chapter 2

Problems and Opportunities

Introduction

Why is speech recognition so difficult and still a subject of so much study? Human beings grow up learning to speak with no apparent instruction of programming and communicate with each other via speech with remarkable ease. Fast, efficient, reliable speech is a critical part of intelligent behavior and of human self-expression. So much is speech a central part of our humanity that the complexities of speech understanding have always been vastly underestimated, despite several decades of research. We begin this book with two papers that give a general introduction to the problem of speech recognition, its difficulties, and its potential. *Speech Recognition by Machine: A Review*, by Raj Reddy is a classic that, many years later, still holds fundamental insights and lessons in the field. *The Value of Speech Recognition Systems* by Wayne Lea discusses the value and potential of machines capable of recognizing and conversing with humans by way of speech.

Speech Recognition by Machine: A Review

D. RAJ REDDY

Abstract—This paper provides a review of recent developments in speech recognition research. The concept of sources of knowledge is introduced and the use of knowledge to generate and verify hypotheses is discussed. The difficulties that arise in the construction of different types of speech recognition systems are discussed and the structure and performance of several such systems is presented. Aspects of component subsystems at the acoustic, phonetic, syntactic, and semantic levels are presented. System organizations that are required for effective interaction and use of various component subsystems in the presence of error and ambiguity are discussed.

I. Introduction

THE OBJECT of this paper is to review recent developments in speech recognition. The Advanced Research Projects Agency's support of speech understanding research has led to a significantly increased level of activity in this area since 1971. Several connected speech recognition systems have been developed and demonstrated. The role and use of knowledge such as acoustic-phonetics, syntax, semantics, and context are more clearly understood. Computer programs for speech recognition seem to deal with ambiguity, error, and nongrammaticality of input in a graceful and effective manner that is uncommon to most other computer programs. Yet there is still a long way to go. We can handle relatively restricted task domains requiring simple grammatical structure and a few hundred words of vocabulary for single trained speakers in controlled environments, but we are very far from being able to handle relatively unrestricted dialogs from a large population of speakers in uncontrolled environments. Many more years of intensive research seem necessary to achieve such a goal.

Sources of Information: The primary sources of information in this area are the *IEEE Transactions on Acoustics, Speech, and Signal Processing* (pertinent special issues: vol. 21, June 1973; vol. 23, Feb. 1975) and the *Journal of the Acoustical Society of America* (in particular, Semiannual Conference Abstracts which appear with January and July issues each year; recently they have been appearing as spring and fall supplements). Other relevant journals are *IEEE Transactions* (Computer; Information Theory; and Systems, Man, and

Manuscript received September 1, 1975; revised November 19, 1975. This work was supported in part by the Advanced Research Projects Agency and in part by the John Simon Guggenheim Memorial Foundation.

The author is with the Computer Science Department, Carnegie-Mellon University, Pittsburgh, PA 15213.

Originally appeared in IEEE Proceedings 64(4):502–531, April, 1976.

Cybernetics), *Communications of ACM, International Journal of Man-Machine Studies, Artificial Intelligence,* and *Pattern Recognition.*

The books by Flanagan [44], Fant [40], and Lehiste [84] provide extensive coverage of speech, acoustics, and phonetics, and form the necessary background for speech recognition research. Collections of papers, in the books edited by David and Denes [25], Lehiste [83], Reddy [121], and Wathen-Dunn [158], and in conference proceedings edited by Erman [34] and Fant [41], provide a rich source of relevant material. The articles by Lindgren [88], Hyde [66], Fant [39], Zagoruiko [171], Derkach [27], Hill [63], and Otten [113] cover the research progress in speech recognition prior to 1970 and proposals for the future. The papers by Klatt [74] and Wolf [163] provide other points of view of recent advances.

Other useful sources of information are research reports published by various research groups active in this area (and can be obtained by writing to one of the principal researchers given in parentheses): Bell Telephone Laboratories (Denes, Flanagan, Fujimura, Rabiner); Bolt Beranek and Newman, Inc. (Makhoul, Wolf, Woods); Carnegie-Mellon University (Erman, Newell, Reddy); Department of Speech Communication, KTH, Stockholm (Fant); Haskins Laboratories (Cooper, Mermelstein); IBM Research Laboratories (Bahl, Dixon, Jelinek); M.I.T. Lincoln Laboratories (Forgie, Weinstein); Research Laboratory of Electronics, M.I.T. (Klatt); Stanford Research Institute (Walker); Speech Communication Research Laboratory (Broad, Markel, Shoup); System Development Corporation (Barnett, Ritea); Sperry Univac (Lea, Medress); University of California, Berkeley (O'Malley); Xerox Palo Alto Research Center (White); and Threshold Technology (Martin). In addition there are several groups in Japan and Europe who publish reports in national languages and English. Complete addresses for most of these groups can be obtained by referring to author addresses in the *IEEE Trans. Acoust., Speech, Signal Processing*, June 1973 and Feb. 1975. For background and introductory information on various aspects of speech recognition we recommend the tutorial-review papers on "Speech understanding systems" by Newell, "Parametric representations of Speech" by Schafer and Rabiner, "Linear prediction in automatic speech recognition" by Makhoul, "Concepts for Acoustic-Phonetic recognition" by Broad and Shoup, "Syntax, Semantics and Speech" by Woods, and "System organization for speech understanding" by Reddy and Erman, all appearing in *Speech Recognition: Invited Papers of the IEEE Symposium* [121].

Scope of the Paper: This paper is intended as a review and not as an exhaustive survey of all research in speech recognition. It is hoped that, upon reading this paper, the reader will know what a speech recognition system consists of, what makes speech recognition a difficult problem, and what aspects of the problem remain unsolved. To this end we will study the structure and performance of some typical systems, component subsystems that are needed, and system organization that permits effective interaction and use of the components. We do not attempt to give detailed descriptions of systems or mathematical formulations, as these are available in published literature. Rather, we will mainly present distinctive and novel features of selected systems and their relative advantages.

Many of the comments of an editorial nature that appear in this paper represent one point of view and are not necessarily shared by all the researchers in the field. Two other papers appearing in this issue, Jelinek's on statistical approaches and Martin's on applications, augment and complement this paper. Papers by Flanagan and others, also appearing in this issue, look at the total problem of man-machine communication by voice.

A. The Nature of the Speech Recognition Problem

The main goal of this area of research is to develop techniques and systems for speech input to machines. In earlier attempts, it was hoped that learning how to build simple recognition systems would lead in a natural way to more sophisticated systems. Systems were built in the 1950's for vowel recognition and digit recognition, producing creditable performance. But these techniques and results could not be extended and extrapolated toward larger and more sophisticated systems. This had led to the appreciation that linguistic and contextual cues must be brought to bear on the recognition strategy if we are to achieve significant progress. The many dimensions that affect the feasibility and performance of a speech recognition system are clearly stated in Newell [108].

Fig. 1 characterizes several different types of speech recognition systems ordered according to their intrinsic difficulty. There are already several commercially available isolated word recognition systems today. A few research systems have been developed for restricted connected speech recognition and speech understanding. There is hope among some researchers that, in the not too distant future, we may be able to develop interactive systems for taking dictation using a restricted vocabulary. Unlimited vocabulary speech understanding and connected speech recognition systems seem feasible to some, but are likely to require many years of directed research.

The main feature that is used to characterize the complexity of a speech recognition task is whether the speech is connected or is spoken one word at a time. In connected speech, it is difficult to determine where one word ends and another begins, and the characteristic acoustic patterns of words exhibit much greater variability depending on the context. *Isolated word recognition systems* do not have these problems since words are separated by pauses.

The second feature that affects the complexity of system is the vocabulary size. As the size or the confusability of a vocabulary increases, simple brute-force methods of representation and matching become too expensive and unacceptable. Techniques for compact representation of acoustic patterns of words, and techniques for reducing search by constraining the number of possible words that can occur at a given point, assume added importance.

Just as vocabulary is restricted to make a speech recognition problem more tractable, there are several other aspects of the problem which can be used to constrain the speech recognition task so that what might otherwise be an unsolvable problem becomes solvable. The rest of the features in Fig. 1, i.e., task-specific knowledge, language of communication, number and cooperativeness of speakers, and quietness of environment, represent some of the commonly used constraints in speech recognition systems.

One way to reduce the problems of error and ambiguity resulting from the use of connected speech and large vocabularies is to use all the available task-specific information to reduce search. The *restricted speech understanding systems* (Fig. 1, line 3) assume that the speech signal does not have all the necessary information to uniquely decode the message and

	Mode of Speech	Vocabulary Size	Task Specific Information	Language	Speaker	Environment
Word recognition-isolated (WR)	isolated words	10-300	limited use	—	cooperative	—
Connected speech recognition-restricted (CSR)	connected speech	30-500	limited use	restricted command language	cooperative	quiet room
Speech understanding-restricted (SU)	connected speech	100-2000	full use	English-like	not uncooperative	—
Dictation machine-restricted (DM)	connected speech	1000-10000	limited use	English-like	cooperative	quiet room
Unrestricted speech understanding (USU)	connected speech	unlimited	full use	English	not uncooperative	—
Unrestricted connected speech recognition (UCSR)	connected speech	unlimited	none	English	not uncooperative	quiet room

Fig. 1. Different types of speech recognition systems ordered according to their intrinsic difficulty, and the dimensions along which they are usually constrained. Vocabulary sizes given are for some typical systems and can vary from system to system. It is assumed that a cooperative speaker would speak clearly and would be willing to repeat or spell a word. A not uncooperative speaker does not try to confuse the system but does not want to go out of his way to help it either. In particular, the system would have to handle "uhms" and "ahs" and other speech-like noise. The "—" indicates an "unspecified" entry variable from system to system.

that, to be successful, one must use all the available sources of knowledge to infer (or deduce) the intent of the message [107]. The performance criterion is somewhat relaxed in that, as long as the message is understood, it is not important to recognize each and every phoneme and/or word correctly. The requirement of using all the sources of knowledge, and the representation of the *task*, *conversational context*, *understanding*, and *response generation*, all add to the difficulty and overall complexity of speech understanding systems.

The *restricted connected speech recognition systems* (Fig. 1, line 2) keep their program structure simple by using only some task-specific knowledge, such as restricted vocabulary and syntax, and by requiring that the speaker speak clearly and use a quiet room. The simpler program structure of these systems provides an economical solution in a restricted class of connected speech recognition tasks. Further, by not being task-specific, they can be used in a wider variety of applications without modification.

The restricted speech understanding systems have the advantage that by making effective use of all the available knowledge, including semantics, conversational context, and speaker preferences, they can provide a more flexible and hopefully higher performance system. For example, they usually permit an English-like grammatical structure, do not require the speaker to speak clearly, and permit some nongrammaticality (including babble, mumble, and cough). Further, by paying careful attention to the task, many aspects of error detection and correction can be handled naturally, thus providing a graceful interaction with the user.

The *(restricted) dictation machine* problem (Fig. 1, line 4) requires larger vocabularies (1000 to 10 000 words). It is assumed that the user would be willing to spell any word that is unknown to the system. The task requires an English-like syntax, but can assume a cooperative speaker speaking clearly in a quiet room.

The *unrestricted speech understanding* problem requires unlimited vocabulary connected speech recognition, but permits the use of all the available task-specific information. The most difficult of all recognition tasks is the *unrestricted connected speech recognition* problem which requires unlimited vocabulary, but does not assume the availability of any task-specific information.

We do not have anything interesting to say about the last three tasks, except perhaps speculatively. In Section II, we will study the structure and performance of several systems of the first three types (Fig. 1), i.e., isolated word recognition systems, restricted connected speech recognition systems, and restricted speech understanding systems.

In general, for a given system and task, performance depends on the size and speed of the computer and on the accuracy of the algorithm used. Accuracy is often task dependent. (We shall see in Section II that a system which gives 99-percent accuracy on a 200-word vocabulary might give only 89-percent accuracy on a 36-word vocabulary.) Accuracy versus response time tradeoff is also possible, i.e., it is often possible to tune a system and adjust thresholds so as to improve the response time while reducing accuracy and vice versa.

Sources of Knowledge: Many of us are aware that a native speaker uses, subconsciously, his knowledge of the language, the environment, and the context in understanding a sentence. These sources of knowledge (KS's) include the characteristics of speech sounds (*phonetics*), variability in pronunciations (*phonology*), the stress and intonation patterns of speech (*prosodics*), the sound patterns of words (*lexicon*), the grammatical structure of language (*syntax*), the meaning of words and sentences (*semantics*), and the context of conversation (*pragmatics*). Fig. 2 shows the many dimensions of variability of these KS's; it is but a slight reorganization (to correspond to the sections of this paper) of a similar figure appearing in [108].

1.	Performance	Nature of input	Isolated words? connected speech?
		Response time	Real time? close to real-time? no hurry?
		Accuracy	Error-free (>99.9%)? almost error-free (>99%)? occasional error (>90%)?
2.	Source characteristics (acoustic knowledge)	Acoustic analysis	Airconditioning noise? computer room? reverberation?
		Noise sources	Dialect? sex? age? cooperative?
		Speaker characteristics	High quality microphone? telephone?
			Spectrum? formants? zerocrossings? LPC?
3.	Language characteristics (phonetic knowledge)	Features	Voiced? energy? stress? intonation?
		Phones	Number? distinguishability?
		Phonology	Phone realization rules? junction rules?
		Word realization	Insertion, deletion and change rules?
			Word hypothesis? word verification?
4.	Problem characteristics (task specific knowledge)	Size of vocabulary	10? 100? 1,000? 10,000?
		Confusability of vocabulary	High? what equivalent vocabulary?
		Syntactic support	Artificial language? free English?
		Semantic and contextual support	Constrained task? open semantics?
5.	System characteristics	Organization	Strategy? representation?
		Interaction	Graceful interaction with user? graceful error recovery?

Fig. 2. Factors affecting feasibility and performance of speech recognition systems. (Adapted from Newell *et al.* [108].)

To illustrate the effect of some of these KS's, consider the following sentences.

1) Colorless paper packages crackle loudly.
2) Colorless yellow ideas sleep furiously.
3) Sleep roses dangerously young colorless.
4) Ben burada ne yaptigimi bilmiyorum.

The first sentence, though grammatical and meaningful, is pragmatically implausible. The second is syntactically correct but meaningless. The third is both syntactically and semantically unacceptable. The fourth (a sentence in Turkish) is completely unintelligible to most of us. One would expect a listener to have more difficulty in recognizing a sentence if it is inconsistent with one or more KS's. Miller and Isard [101] show that this is indeed the case.

If the knowledge is incomplete or inaccurate, people will tend to make erroneous hypotheses. This can be illustrated by a simple experiment. Subjects were asked to listen to two sentences and write down what they heard. The sentences were "In mud eels are, in clay none are" and "In pine tar is, in oak none is." The responses of four subjects are given below.

In mud eels are,	*In clay none are*
in muddies sar	in clay nanar
in my deals are	en clainanar
in my ders	en clain
in model sar	in claynanar
In pine tar is,	*In oak none is*
in pine tarrar	in oak ? es
in pyntar es	in oak nonnus
in pine tar is	in ocnonin
en pine tar is	in oak is

The responses show that the listener forces his own interpretation of what he hears, and not necessarily what may have been intended by the speaker. Because the subjects do not have the contextual framework to expect the words "mud eels" together, they write more likely sounding combinations such as "my deals" or "models." We find the same problem with words such as "oak none is." Notice that they failed to detect where one word ends and another begins. It is not uncommon for machine recognition systems to have similar problems with word segmentation. To approach human performance, a machine must also use all the available KS's effectively.

Reddy and Newell [124] show that knowledge at various levels can be further decomposed into sublevels (Fig. 3) based on whether it is task-dependent *a priori* knowledge, conversation-dependent knowledge, speaker-dependent knowledge, or analysis-dependent knowledge. One can further decompose each of these sublevels into sets of rules relating to specific topics. Many of the present systems have only a small subset of all the KS's shown in Fig. 3. This is because much of this knowledge is yet to be identified and codified in ways that can be conveniently used in a speech understanding system. Sections III through V review the recent progress in representation and use of various sources of knowledge.

In Section III, we consider aspects of signal processing for speech recognition. There is a great deal of research and many publications in this area, but very few of them are addressed to questions that arise in building speech recognition systems. It is not uncommon for a speech recognition system to show a catastrophic drop in performance when the microphone is changed or moved to a slightly noisy room. Many parametric representations of speech have been proposed but there are few comparative studies. In Section III, we shall review the techniques that are presently used in speech signal and analysis and noise normalization, and examine their limitations.

There are several KS's which are common to most connected speech recognition systems and independent of the task. These can be broadly grouped together as task-independent aspects of a speech recognition system. Topics such as feature extraction, phonetic labeling, phonological rules, (bottom-up) word hypothesis, and word verification fall into this category. In Section IV, we will review the techniques used and the present state of accomplishment in these areas.

Given a task that is to be performed using a speech recognition system, one is usually able to specify the vocabulary, the grammatical structure of sentences, and the semantic and contextual constraints provided by the task. In Section V, we will discuss the nature, representation, and use of these KS's in a recognition (or understanding) system.

Control Structure and System Organization: How is a given source of knowledge used in recognition? The Shannon [140] experiment gives a clue. In this experiment, human subjects demonstrate their ability to predict (and correct) what will appear next, given a portion of a sentence.

Just as in the above experiment, many recognition systems use the KS's to generate hypotheses about what word might

Type of knowledge	Task-dependent knowledge	Conversation-dependent knowledge	Speaker-dependent knowledge	Analysis-dependent knowledge
Pragmatic and Semantic	A priori semantic knowledge about the task domain	Concept subselection based on conversation	Psychological model of the user	Concept subselection based on partial sentence recognition
Syntactic	Grammar for the language	Grammar subselection based on topic	Grammar subselection based on speaker	Grammar subselection based on partial phrase recognition
Lexical	Size and confusability of the vocabulary	Vocabulary subselection based on topic	Vocabulary subselection and ordering based on speaker preference	Vocabulary subselection based on segmental features
Phonemic and phonetic	Characteristics of phones and phonemes of the language	Contextual variability in phonemic characteristics	Dialectal variations of the speaker	Phonemic subselection based on segmental features
Parametric and acoustic	A priori knowledge about the transducer characteristics	Adaptive noise normalization	Variations resulting from the size and shape of vocal tract	Parameter tracking based on previous parameters

Fig. 3. Sources of knowledge (KS). (From Reddy and Newell [124].)

(1)	Speed of Communication	Speech is about 4 times faster than standard manual input for continuous text.
(2)	Total System Response Time	Direct data entry from remote source, which avoids relayed entry via intermediate human transducers, speeds up communication substantially.
(3)	Total System Reliability	Direct data entry from remote source with immediate feedback, avoiding relayed entry via intermediate human transducers, increases reliability substantially.
(4)	Parallel Channel	Provides an independent communication channel in hands-busy operational situations.
(5)	Freedom of Movement	Within small physical regions speech can be used while moving about freely doing a task.
(6)	Untrained Users	No training in basic physical skill required for use (as opposed to acquisition of typing or keying skills); speech is natural for users at all general skill levels (clerical to executive).
(7)	Unplanned Communication	Speech is to be used immediately by users to communicate unplanned information, in a way not true of manual input.
(8)	Identification of Speaker	Speakers are recognizable by their voice characteristics.
(9)	Long Term Reliability	Performance of speech reception and processing tasks which require monotonous vigilant operation can be done more reliably by computer than by humans.
(10)	Low Cost Operation	Speech can provide cost savings where it eliminates substantial numbers of people.

Fig. 4. Task demands providing comparative advantages for speech. (From Newell *et al.* [109].)

appear in a given context, or to reject a guess. When one of these systems makes errors, it is usually because the present state of its knowledge is incomplete and possibly inaccurate. In Section VI, we shall review aspects of system organization such as control strategies, error handling, real-time system design, and knowledge acquisition.

B. The Uses of Speech Recognition

Until recently there has been little experience in the use of speech recognition systems in real applications. Most of the systems developed in the 1960's were laboratory systems, which were expensive and had an unacceptable error rate for real life situations. Recently, however, there have been commercially available systems for isolated word recognition, costing from $10 000 to $100 000, with less than 1-percent error rate in noisy environments. The paper by Martin in this issue illustrates a variety of applications where these systems have been found to be useful and cost-effective.

As long as speech recognition systems continue to cost around $10 000 to $100 000, the range of applications for which they will be used will be limited. As the research under way at present comes to fruition over the next few years, and as connected speech recognition systems costing under $10 000 begin to become available, one can expect a significant increase in the number of applications. Fig. 4, adapted from Newell *et al.* [109], summarizes and extends the views expressed by several authors earlier [63], [78], [87], and [89] on the desirability and usefulness of speech—it provides a list of task situation characteristics that are likely to benefit from speech input. Beek *et al.* [17] provide an assessment of the potential military applications of automatic speech recognition.

As computers get cheaper and more powerful, it is estimated that 60–80 percent of the cost of running a business computer installation will be spent on data collection, preparation, and entry (unpublished proprietary studies; should be considered

speculative for the present). Given speech recognition systems that are flexible enough to change speakers or task definitions with a few days of effort, speech will begin to be used as an alternate medium of input to computers. Speech is likely to be used not so much for program entry, but rather primarily in data entry situations [33]. This increased usage should in turn lead to increased versatility and reduced cost in speech input systems.

There was some earlier skepticism as to whether speech input was necessary or even desirable as an input medium for computers [116]. The present attitude among the researchers in the field appears to be just the opposite, i.e., if speech input systems of reasonable cost and reliability were available, they would be the preferred mode of communication even though the relative cost is higher than other types of input [109]. Recent human factors studies in cooperative problem solving [23], [110] seem to support the view that speech is *the* preferred mode of communication. If it is indeed preferred, it seems safe to assume that the user would be willing to pay somewhat higher prices to be able to talk to computers. This prospect of being able to talk to computers is what drives the field, not just the development of a few systems for highly specialized applications.

II. Systems

This section provides an overview of the structure of different types of speech recognition systems. To accomplish this, one needs to answer questions such as: what are the important concepts and principles associated with each of these systems, what are their distinguishing features, how well do they perform, and so on. It is not always possible to answer these questions. Very few comparative results based on common test data are available. In many cases all the returns are not yet in. There are so many possible design choices that most systems are not strictly comparable with each other. Therefore, it will be necessary to restrict our discussion to somewhat superficial comparisons based on accuracy, response time, size of vocabulary, etc.

In this section, we will examine the structure and performance of the first three classes of systems shown in Fig. 1: *isolated word recognition systems, restricted connected speech recognition systems,* and *restricted speech understanding systems*. We will illustrate the principles of design and performance by picking a few systems which are representative of the state of the art in each category. For the sake of brevity, we will leave out the words "isolated" and "restricted" for the rest of this paper. Unless otherwise indicated, it is to be assumed that we are always talking about isolated word recognition systems, restricted connected speech recognition systems, and restricted speech understanding systems.

A. Word Recognition Systems (WRS)

Here we will look at the structure and performance of three systems by Itakura [70], Martin [96], and White [161]. Given a known vocabulary (of about 30 to 200 words) and a known speaker, these systems can recognize a word spoken in isolation with accuracies around 99 percent. The vocabulary and/or speaker can be changed but this usually requires a training session. These systems, though similar in some respects, have several interesting and distinguishing features. Fig. 5 summarizes some of the features of these systems that affect cost and performance. Researchers desirous of working in the field of word recognition would also benefit from studying the structure and features of several earlier (and somewhat lower performance) systems by Gold [51], Shearme and Leach [141], Bobrow and Klatt [18], Vicens [152], Medress [98], Valichiko and Zagoruiko [151], Vysotsky *et al.* [153], Pols [117], Von Keller [154], Itahashi, Makino, and Kido [67], and Sambur and Rabiner [135].

All three systems use the classical pattern recognition paradigm as their recognition strategy. The general paradigm involves comparing the parameter or feature representation of the incoming utterance with the prototype reference patterns of each of the words in the vocabulary. Fig. 6 presents the flow chart of a typical word recognition system. The main decisions to be made in the design of a word recognition system are: how to normalize for variations in speech; what is the parametric representation; how does the system adapt to a new speaker or new vocabulary; how does one measure the similarity of two utterances; and how to speed up the matching process.

Normalization: Even when the speaker and the microphone are not changed, variations in speech occur as a result of free variation from trial to trial, as well as the emotional state of the speaker and the ambient noise level of the environment. These result in changes in amplitude, duration, and signal-to-noise ratio for a given utterance. Before a match with stored templates can take place, some form of normalization is necessary to minimize the variability. Itakura [70] uses a second-order inverse filter based on the entire utterance to achieve noise and amplitude normalization. Martin [96] identifies several types of noise related problems: room noise, breath noise, intraword stop gaps, and operator-originated babble. Some of these result in incorrect detection of beginning and end of the utterance. Most of the noise problems can be overcome by careful attention to detail, such as close-speaking microphones, looking for spectra that are typical of breath noise, rejecting utterances that do not get a close match to any of the words, and so on. White, before measuring distances, normalizes all filter samples by dividing by the total energy.

Parametric Representations: Itakura uses linear predictive coding (LPC) coefficients, Martin uses hardware feature detectors based on bandpass filters, while White uses a 1/3-octave filter bank (see Section III). White [161] has studied the effect of using different parametric representations. Results of this experiment are given in Fig. 7. It shows that the 1/3-octave filter bank and LPC yield about similar results, and using a 6-channel-octave filter bank increases the error rate from 2 to 4 percent while doubling the speed of recognition. Transforming the parametric data into a pseudophonemic label prior to match can lead to significant reduction of storage but the error rate increases sharply to 9 percent. Reference pattern storage requirement is also affected by the choice of parametric representation. Assuming an average duration of 600 ms per word, White requires from 2160 to 7200 bits of storage (depending on parametric representation) per reference pattern and Itakura requires 4480 bits, while Martin requires only 512 bits per pattern.

Training: Change of speaker or vocabulary is accomplished in all three systems by training the system to produce a new set of reference patterns. Both Itakura and White use a single reference pattern per word. A single reference pattern cannot capture all variations in pronunciations for even a single speaker. Thus when a word exhibits higher than acceptable error rate it is desirable to store additional patterns. But this

	ITAKURA	MARTIN	WHITE
1. Transducer	Telephone	Close speaking microphone	Close speaking microphone
2. Noise level	68 dB (A)	90 dB (A)	65 dB (A)
3. Parametric representation	LPC	Hardware feature extractor	1/3 octave filter bank
4. No. of templates per word	Single template	Average of multiple templates	Single template
5. Space required per reference pattern	4480 bits	512 bits	7200 bits
6. Computer system	DDP-516	Nova, PDP/11 or Microcomputers	SIGMA-3

Fig. 5. Distinctive aspects of three word recognition systems. (Compiled from Itakura [70], Martin [95], [96], and White and Neely [161].)

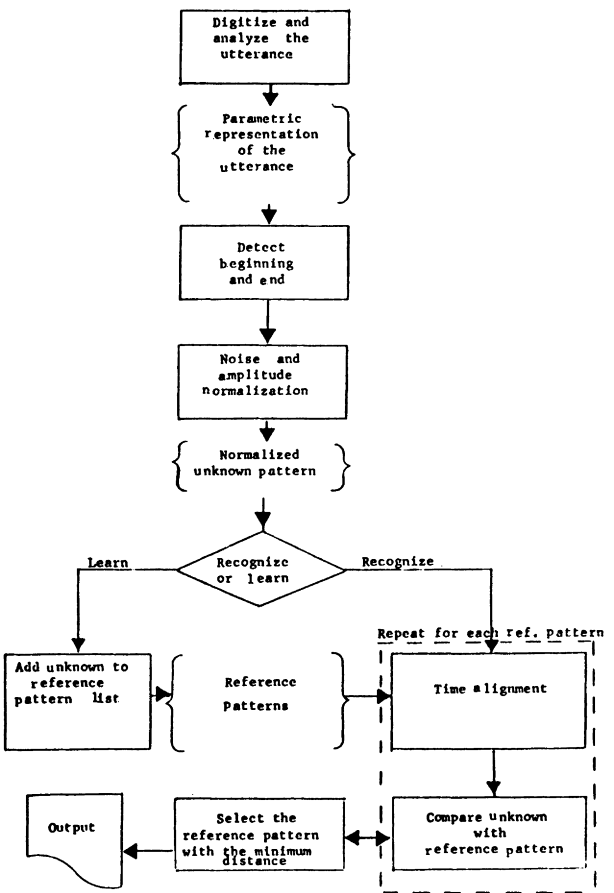

Fig. 6. Flow chart of a typical word recognition system.

Preprocessing method	Alpha-Digit vocabulary % correct	Recognition time per utterance	Data Rate bits per sec approximate
20 channel (1/3 octave filters)	98%	30 sec	12,000
LPC	97%	20 sec	4,200
6 channel (octave filters)	96%	15 sec	3,600
Phone code	91%	2 sec	500

Fig. 7. Effect of parametric representation on accuracy and response time of a system. Preprocessing produces four different parametric representations arranged in order of increasing data compression (lower bit rate). Recognition accuracy goes down as compression goes up. Phone code attempts to give a single pseudophonetic label for each 10-ms unit of speech.

requires additional storage and computation. Martin attempts abstraction of reference patterns by generating an average template from multiple samples.

Matching and Classification: Given an unknown input word, it is compared with every reference pattern to determine which pattern is most similar, i.e., has the minimum distance or maximum correlation to the unknown. This similarity measure is usually established by summing distances (or log probabilities as the case may be) between parameter vectors of the unknown and the reference. There are many design choices that affect the performance at this level, e.g., the choice of the basic time unit of sampling, the choice of the distance metric, differential weighting of parameters, and the choice of the time normalization function.

Itakura and White use dynamic programming for time normalization, while Martin divides the utterance into 16 equal time units. Itakura measures the distance between the unknown and the reference by summing the log probability based on residual prediction error every 15 ms. White measures the distance by summing the absolute values of the differences (Chebyshev norm) between the parameter vectors every 10 ms. Martin uses a weighted correlation metric to measure similarity every 40 ms or so (actually 1/16 of the duration of the utterance).

White shows that the nonlinear time warping based on dynamic programming is better than linear time scaling methods. He also shows Itakura's distance measure based on LPC linear prediction error yields about the same accuracy as other conventional methods. It is generally felt (based on speech bandwidth compression experiments) that significant loss of information results when speech is sampled at intervals exceeding 20 ms. However, note that Martin extracts averaged features based on longer time intervals and is not just sampling the signal parameters.

System	Vocabulary	Size	Noise	Microphone	Nunber of speakers	Accuracy (includes rejects if any)	Resp. time in times real time
Martin	Digits	10	—	Close speaking microphone (CSM)	10	99.79	Almost real-time
Martin	Aircraft ops.	11x12	—	CSM	10	99.32	Almost real-time
Martin	1 thru 34	34	90dB	CSM	12	98.5	Almost real-time
White	Alpha-digit	36	65	CSM	1	98.0	30
White	North Am. states	91	65	CSM	1	99.6	—
Itakura	Alpha-digit	36	68	Telephone	1	88.6	—
Itakura	Japanese geographical names	200	68	Telephone	1	98.95	22

Fig. 8. Performance characteristics of three word recognition systems. (Compiled from Itakura [70], Martin [95], [96], and White and Neely [161].)

Heuristics for Speedup: If a system compares the unknown with every one of the reference patterns in a brute-force manner, the response time increases linearly with the size of the vocabulary. Given the present speeds of minicomputers which can execute 0.2 to 0.5 million instructions per second (mips), the increase in response time is not noticeable for small vocabularies of 10 to 30. But when the size of vocabulary increases to a few hundred words it becomes essential to use techniques that reduce computation time. Itakura uses a sequential decision technique and rejects a reference pattern if its distance exceeds a variable threshold at any time during the match operation. This results in a speedup of the matching process by a factor of almost 10. White uses the duration, amplitude contour, and partial match distance of the first 200 ms as three independent measurements to eliminate the most unlikely candidates from the search list. Others have used gross segmental features [152] and pronouncing dictionary with phonological rules [67] in reducing search. But these require a more complex program organization.

Performance: Fig. 8 gives the published performance statistics for the three systems. It is important to remember that accuracy and response time are meaningful only when considered in the context of all the variables that affect the performance. Although recognition performance scores have been quoted only for systems ranging from 10 to 34 words, Martin's system has been used with vocabularies as high as 144 words. It is the only system that has been shown to work in very high noise (>90 dB) environments and with multiple speakers (using reference patterns which represent all the speakers). The accuracy of Itakura's system drops to 88.6 percent on the alpha-digit word list (aye, bee, cee, $\cdots$, zero, one, $\cdots$, nine). But note that it is the only system that uses a telephone as the transducer. In addition to restricting the frequency response to about 300 to 3000 Hz, the telephone introduces burst noise, distortion, echo, crosstalk, frequency translation, envelope delay, and clipping to list a few. In addition, the alpha-digit vocabulary is highly ambiguous. The fact that the system achieves about 1-percent error rate (and 1.65-percent rejection rate) on a less ambiguous 200-word vocabulary is indicative of its true power. White's system not only achieves high accuracies but also is notable for its system organization which permits it to use different parameters, different time normalization strategies, and different search reduction heuristics with ease. The important thing to remember is that each of these systems seems capable of working with less than 2-percent error rate in noisy environments given vocabularies in the range of 30 to 200. It seems reasonable to assume that accuracy will not degrade substantially with larger vocabularies. A useful indicator of this is the early system by Vicens [152] which achieved 91.4-percent with a 561-word vocabulary.

Future Directions: As long as the cost/performance requirements do not demand an order of magnitude improvement, the present systems approach will continue to be practical and viable. The improvements in computer technology have already brought the cost of such systems to around $10 000. However, if it becomes necessary to reduce the cost to the $1000 range, significant improvement to the basic algorithms will be necessary. The principal avenues for improvement are in the reference pattern representation and search strategies. Rather than storing a vector of parameters every 10 ms, it may be necessary to go to a segmentation and labeling scheme (see Section IV) as has been attempted by some earlier investigators [67], [152]. Rather than storing multiple reference patterns for multiple speakers, it will be necessary to find techniques for abstraction. It may also be necessary to use mixed search strategies in which a simpler parametric representation is used to eliminate unlikely candidates before using a more expensive matching technique. Since many of these techniques are essential for connected speech recognition, it is reasonable to assume that progress in that area will gradually lead to low-cost/high-performance word recognition systems.

B. Connected Speech Recognition (CSR)

In this section we will look at the structure and performance of four different connected speech recognition (CSR) systems: Hearsay-I and Dragon developed at Carnegie-Mellon University [7], [123]; the Lincoln system developed at M.I.T. Lincoln Laboratories [47], [48], [56], [97], [159], [162]; and the IBM system developed at IBM, T. J. Watson Research Center [10], [30], [31], [71], [72], [149], [150], [172], [173]. Hearsay-I was actually designed as a speech understanding sys-

tem, but the semantic and task modules can be deactivated so as to permit it to run like a connected speech recognition system. Both the Dragon and Lincoln systems were designed to add task-specific constraints later, but in their present form can be looked upon as connected speech recognition systems. These systems have achieved from 55- to 97-percent word accuracies. Since a sentence is considered to be incorrect even if only one word in the utterance is incorrect, the sentence accuracies tend to be much lower (around 30 to 81 percent). With tuning and algorithm improvement currently in progress, some of these systems are expected to show significant improvement in accuracy. Researchers interested in CSR systems might also wish to look at the papers in [26], [28], [95], [120], [148], and [152].

Why Is Connected Speech Recognition Difficult? When isolated word recognition systems are getting over 99-percent accuracies, why is it that CSR systems are straining to get similar accuracy? The answers are not difficult to find. In connected speech it is difficult to determine where one word ends and another begins. In addition, acoustic characteristics of sounds and words exhibit much greater variability in connected speech, depending on the context, compared with words spoken in isolation.

Any attempt to extend the design philosophy of isolated word recognition systems and recognize the utterance as a whole becomes an exercise in futility. Note that even a 10-word vocabulary of digits requires the storage of 10-million reference patterns if one wanted to recognize all the possible 7-digit sequences. Some way must be found for the recognition of the whole by analysis of the parts. The technique needed becomes one of analysis and description rather than classification (moving away from pattern recognition paradigms toward hierarchical systems, i.e., systems in which component subparts are recognized and grouped together to form larger and larger units).

To analyze and describe a component part, i.e., a word within the sentence, one needs a description of what to expect when that word is spoken. Again, the reference pattern idea of word recognition systems becomes unsatisfactory. As the number of words in the vocabulary and the number of different contextual variations per word get large, the storage required to store all the reference pattern becomes enormous. For a 200-word vocabulary, such as the one used by Itakura [70], a CSR system might need 2000 reference patterns requiring about 8-million bits of memory, not to mention the time and labor associated with speaking them into the machine. What is needed is a more compact representation of the sound patterns of the words such as those used by linguists, i.e., representation of words as a sequence of phones, phonemes, or syllables. This change from signal space representation of the words to a symbol space representation requires segmenting the continuous speech signal into discrete acoustically invariant parts and labeling each segment with phonemic or feature labels. A phonemic dictionary of the words could then be used to match at a symbolic level and determine which word was spoken.

Since CSR systems do not have the advantage of word recognition systems, of knowing the beginning and ending of words, one usually proceeds left-to-right, thereby forcing at least the beginning to be specified prior to the match for a word. Given where the first (left-most) word of the utterance ends, one can begin matching for the second word from about that position. One must still find techniques for terminating the match when an optimal match is found.

However, the exact match cannot be quite determined until the ending context (the word that follows) is also known. For example, in the word sequence "some milk" all of the nasal /m/ might be matched with the end of "some" leaving only the "ilk" part for a subsequent match. This is a special case of the juncture problem (see Section IV). Techniques are needed which will back up somewhat when the word being matched indicates that it might be necessary in this context. (see also Section VIII of Jelinek [72].)

Finally, error and uncertainty in segmentation, labeling, and matching make it necessary that several alternative word matches be considered as alternative paths. If there were 5 words in an utterance and we considered 5 alternative paths after each word, we would have 3125 (5^5) word sequences, out of which we have to pick the one that is most plausible. Selection of the best word sequence requires a tree search algorithm and a carefully constructed similarity measure.

The preceding design choices are what make CSR systems substantially more complex than word recognition systems. We do not yet have good signal-to-symbol transformation techniques nor do we fully understand how to do word matching performance of CSR systems when compared with word recognition systems. However, researchers have been working seriously on CSR techniques only for the past few years, and significant improvements can be expected in the not too distant future. The following discussion reviews the design choices made by each of the four systems (Fig. 9).

Front End Processing: The purpose of the front end in a CSR system is to process the signal and transform it to a symbol string so that matching can take place. The first three design choices in Fig. 9 affect the nature of this signal-to-symbol transformation. The Dragon system uses the simplest front end of all the systems. It uses the 10-ms speech segment as a basic unit and attempts matching at that level. Given a vector of 12 amplitude and zero-crossing parameters every 10 ms, the system computes the probabilities for each of 33 possible phonemic symbols. To account for allophonic variations, it uses multiple reference patterns (vectors) to represent each phonemic symbol.

Hearsay-I uses amplitude and zero-crossing parameters to obtain a multilevel segmentation into syllable-size units and phoneme-size units. Every 10-ms unit is given a phonemic label based on a nearest neighbor classification using a predefined set of cluster centers. Contiguous 10-ms segments with the same label are grouped together to form a phoneme-size segment. A syllable-like segmentation is derived based on local maxima and minima in the overall amplitude function of the utterance. These larger segments are given gross feature labels such as silence, fricative, and voiced.

The Lincoln system is described in detail by Weinstein *et al.* [159]. The fast digital processor (FDP) computes LPC spectra, tracks formant frequencies [97], performs a preliminary segmentation, and labels the segments as one of vowel, dip (intervocalic voiced consonants characterized by a dip in amplitude), fricative, and stop categories. Formant frequencies, formant motions, formant amplitude, and other spectral measurements are used in further classifying the segments into phone-like acoustic-phonetic elements (APEL) labels.

The IBM system front end is based on the approach developed by Dixon and Silverman [31], [32], for pattern

	Hearsay-I	Dragon	Lincoln	IBM
Parametric representation	Amplitude + zero crossings in 5 octave bands	Amplitude and zero crossings in 5 octave bands	LPC Spectra formants	Spectrum
Segmentation	Heuristic multilevel (syllabic + phonetic)	None	Heuristic	Heuristic
Labeling	Two level prot. matching	Prototype matching	Feature based	Prototype matching
Word matching	Heuristic	Stochastic	Heuristic	Stochastic
Phonological rules	Ad Hoc	None (can be added)	Yes	Yes
Word representation	Phonemic base form	Network	Phonemic base form	Network
Syntax	Productions anti-productions	Finite state network	Productions anti-productions	Finite state network
Search	Left to right search best first	Left to right search all paths	Left to right search best first	Left to right search using sequential decoding (similar to best first)

Fig. 9. Design choices of the four connected speech recognition (CSR) systems. (Compiled from Reddy *et al.* [123], Baker [8], Forgie *et al.* [48], Baker and Bahl [10], and other related publications.)

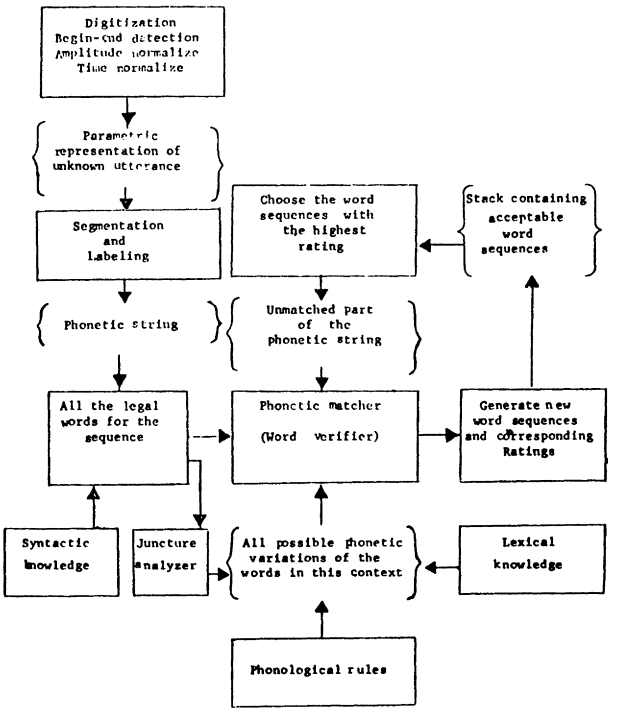

Fig. 10. Flow chart of a typical CSR system.

recognition using complex decision-making rules and dynamic segmentation [148]. The segmentation and labeling procedure uses energy, spectra, spectral change, an ordered list of five "most similar" classes, and their similarity values. The labeling is done by prototype matching as in the case of Hearsay-I and Dragon but using about 62 label classes.

Knowledge Representation: There are three types of knowledge that are usually required in a CSR system: phonological rules, lexicon, and syntax. The Dragon system has the most elegant representation of knowledge of the four systems [7]. All the knowledge is represented as a unified finite-state network representing a hierarchy of probabilistic functions of the Markov processes.

Hearsay-I organizes knowledge into independent and cooperating knowledge processes, which makes it easy to add or remove a knowledge source. The representation of knowledge within each process is somewha arbitrary and depends on the needs of that process. Syntax is represented as a set of productions (generative rewriting rules) and antiproductions (analytic prediction rules). The lexicon contains only the phonemic base forms. Phonological information is embedded in various acoustic analysis procedures.

In the Lincoln system, syntactic constraints are represented by a set of production rules. Phonological and front-end-dependent rules are used to construct a lexicon from a base form dictionary [55], [56]. Other such rules are also applied during a heuristic word and matching process.

The IBM system uses a finite-state grammar and a directed graph representation of each lexical element [24]. Phonological rules are compiled into the lexicon. To account for the rules that do involve word boundaries, the graphs have multiple starting nodes labeled with conditions that must be met by preceding or following words. An important component of the IBM system is the extensive use of statistical information to provide transition probabilities within the finite-state networks representing task-dependent information. (See also Jelinek [72].)

Although both the Dragon and IBM systems use network representations and stochastic matching, they differ in several respects. Dragon uses an integrated representation of all the knowledge, whereas the IBM system has independent representations of the language, phonology, and acoustic components. Dragon evaluates the likelihood of all possible paths, while the IBM system uses sequential decoding to constrain the search to the most likely path.

Matching and Control: Fig. 10 is a flow chart of the recognition process of a typical CSR system. All the systems except Dragon use a stack (or a set) containing a list of alternative word sequences (or state sequences) arranged in descending order of their likelihoods (or scores) to represent the partial sentence analysis so far. Given the word sequence with the highest likelihood, the task-specific knowledge generates all the words that can follow that sequence. Each of these words is matched against the unmatched symbol (phonemic) string to estimate conditional likelihoods of occurrence. These are used to generate a new list of acceptable word sequences and their likelihoods. This process is repeated until the whole utterance is analyzed and an acceptable word sequence is determined. The Dragon system, rather than extending the best word sequence, extends all the sequences in parallel. The Markovian assumption permits it to collapse many alternative sequences into a single state, thus avoiding exponential growth.

The four systems differ significantly in the way in which insertion, deletion, and substitution errors are handled in the matching process, and the way in which likelihoods are estimated. Hearsay-I and Lincoln systems use heuristic techniques, while Dragon and IBM systems use the principles of stochastic modeling [72], [7] to estimate likelihoods. In Section IV, we will discuss techniques for word matching and verification in greater detail.

Performance: Fig. 11 gives some performance statistics for the four systems. The systems are not strictly comparable because of the number of variables involved. However, some

	Hearsay-I	Dragon	Lincoln	IBM
No. of Sentences	102	102	275	363
No. of Word tokens	578	578	-	-
No. of Speakers	4	1	6	1
No. of Tasks	5	5	1	2
Sentence Accuracy	31%	49%	49%	81%
Word Accuracy	55%	83%	-	97%
Response Time (x real-time)	9-44	48-174	15-25	25
Environment	Terminal room	Terminal room	Computer room	Sound booth
Transducer	CSM and telephone	CSM	CSM	HQM
Size of Vocabulary	24-194	24-194	237	250
Live Input	Yes	No	Yes	No
Date Operational	1972	1974	1974	1975
Computer	PDP-10	PDP-10	TX-2/FDP	360/91 and 370/168
Average No. of instructions executed per second of speech in million	3-15	15-60	45-75	30

Fig. 11. Performance statistics for four CSR systems. (From sources given for Fig. 9.)

general comparisons can be made. The IBM system has the best performance of the four, but one should bear in mind the fact that most of their results to date are based on relatively noise-free high-quality data for a single speaker. It is also the only system being improved actively at present. This tuning of the system should lead to even higher accuracies.

Hearsay-I and Dragon were run on the same data sets to permit strict comparison. Dragon yields significantly higher accuracy, though it is slower by a factor of 4 to 5. Hearsay-I yields much higher accuracies on tasks and speakers with which it is carefully trained (see Fig. 15). It was tested on several speakers and several tasks. As the vocabulary increases, its relatively weaker acoustic-phonetic module tends to make more errors in the absence of careful tuning. It was one of the first systems to be built and still is one of the very few that can be demonstrated live.

The Dragon system performance demonstrates that simple and mathematically tractable CSR systems can be built without sacrificing accuracy. Although searching all possible alternative paths becomes unfeasible for very large vocabularies, for restricted tasks with a few hundred word vocabulary, Dragon with its simpler program structure represents an attractive alternative.

The Lincoln system is the only one of the four that works for several speakers without significant tuning for the speaker. The 49-percent sentence accuracy represents the composite accuracy for all the speakers taken together. It was also tested with a 411-percent word vocabulary, yielding about 28-percent sentence accuracy over the same set of six speakers.

Future Directions: How can CSR systems achieve significantly higher performance and cost under $20 000? Better search, better matching, and better segmentation and labeling are all essential if the systems are to achieve higher accuracies. The best-first search strategy used by Hearsay-I and other systems leads to termination of search when it exceeds a given time limit. When this happens, it is usually because errors in evaluation have led to a wrong part of the search space, and the system is exploring a large number of incorrect paths. In most systems, this accounts for 20–30 percent of the sentence errors.

Dragon does not have the problem of thrashing since it searches all the possible extensions of a word (state) sequence. An intermediate strategy in which several promising alternative paths are considered in parallel (best few without backtracking), rather than all or the best-first strategies of the present systems, seems desirable. Lowerre [90] has implemented one such strategy in the Harpy system currently under development at Carnegie-Mellon University and has reduced the computation requirement by about a factor of 5 over Dragon without any loss of accuracy. The number of alternative paths to be considered is usually a function of the goodness of the parametric representation (and accuracy of the segmental labels). Continued research into this class of systems should lead to the development of low-cost CSR.

Accuracies in word matching and verification approaching those of word recognition systems, i.e., greater than 99 percent, are essential for the success of CSR. Since words exhibit greater variability in connected speech, this becomes a much more difficult task. Klatt [75] proposes the use of analysis-by-synthesis techniques as the principal solution to this problem. Near-term solutions include learning the transition probabilities of a word network using training data, as is being done by IBM, or learning the lexical descriptions themselves from examples, as is being attempted at Carnegie-Mellon University. There has been very little work on comparative evaluation of segmentation and labeling schemes. Further studies are needed to determine which techniques work well, especially in environments representative of real life situations.

C. Speech Understanding Systems (SUS)

In this section, we will study approaches to speech understanding systems (SUS's) design by discussing three systems, viz., Hearsay-II [36], [86], SPEECHLIS [166], and VDMS [127], [156], currently being developed, respectively, at Carnegie-Mellon University, Bolt Beranek and Newman, and jointly by System Development Corporation and Stanford Research Institute. We cannot give performance statistics for these systems as they are not working well enough yet. However, at least one earlier system, Hearsay-I, illustrates the potential importance and usefulness of semantic and conversation-dependent knowledge. Experiments on this system show that 25–30-percent improvement in sentence accuracies (e.g., from about 52 to 80 percent on one task) were achieved using chess-dependent semantic knowledge in the voice-chess task. Researchers interested in other attempts at speech understanding systems should look at [13], [20], [35], [47], [64], [123], [134], [152], [157], and [160].

What Makes Speech Understanding Difficult? In addition to the problems of having to recognize connected speech, SUS's tend to have the additional requirement that they must do so even when the utterance is not quite grammatical or well formed, and in the presence of speech-like noise (e.g., babble, mumble, and cough). The requirement is somewhat relaxed

by the concession that what matters in the end is not the recognition of each and every word in the utterance but rather the intent of the message. The systems are also required to keep track of the context of the conversation so far and use it to resolve any ambiguities that might arise within the present sentence. Clearly, one can attempt to build CSR systems with all the preceding characteristics and yet not use any task-specific information. Here we will restrict ourselves to the apparent differences in approach between the CSR and the SU systems of the current generation.

How do the above requirements translate into specific problems to be solved? We still have the problem of determining when a word begins and ends in connected speech, and the problem of wide variability in the acoustic characteristics of the words. But the solutions adopted in CSR systems to solve these problems do not quite carry over to SUS. One can no longer proceed left-to-right in the analysis of an utterance because of the possibility of error or unknown babble in the middle of the utterance. Thus the useful technique of keeping an ordered list of word sequences which are extended to the right after each iteration has to be modified significantly.

Another design choice of CSR systems that leads to difficulties is the notion that there is a bottom-up acoustic analyzer (the *front end*) which generates a phonemic (or some such) symbol string, and a top-down language model (the *back end*) predicting possible word candidates at that choice point, which are then compared by a matching procedure. As the vocabularies get larger, often the roles have to be reversed. One cannot afford to match 1000 possible nouns just because the grammar predicts that the next word might be a noun. In such cases, the phonemic string may be used to generate plausible word hypotheses, while the language model is used to verify such hypotheses for compatibility and consistency. In general, one wants systems in which the role of knowledge sources is somewhat symmetric. They may be required to predict or verify depending on the context. The representations of knowledge required to perform these different roles will usually be different.

In CSR we have seen that at a given time one of several words might be possible given the acoustic evidence. This is what leads to the nondeterministic search, i.e., consideration and evaluation of an ordered list of alternate word sequences in the flow chart given in Fig. 10. This nondeterministic (and errorful) nature of decisions permeates all the levels of the speech decoding process, i.e., segmental, phonetic, phonemic, syllabic, word, phrase, and conceptual, and not just the word level. There is no such thing as error-free segmentation, error-free labels, and so on up the levels. This requires the representation of alternate sequences at all levels, not just the word level as in the case of CSR systems. Fig. 12 (from Reddy and Erman [122]) illustrates the consequences of this nondeterminism.

At the bottom of Fig. 12, we see the speech waveform for part of an utterance: '. . . all about . . . '. The "true" locations of phoneme and word boundaries are given below the waveform. In a recognition system, the choices of segment boundaries and labels to be associated with each of the segments are not as clear cut. (In fact, even getting trained phoneticians to agree on the "true" locations is often difficult.) A segmentation and labeling program might produce segment boundaries as indicated by the dotted lines connecting the waveform to the segment level. Given the segmental features, the phoneme represented by the first segment might be /aw/,

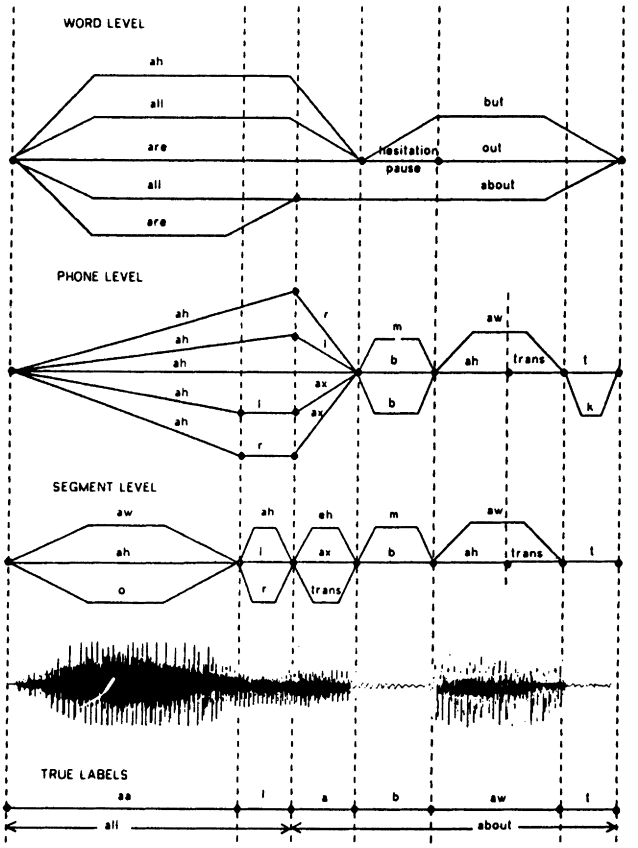

Fig. 12. Example of network representation of alternative choices at various levels. (From Reddy and Erman [122].)

/ah/, or /ow/. Similarly, several different labels can be given to each of the other segments. Given the necessary acoustic-phonetic rules, it is possible to combine, regroup, and delete segments, forming larger phoneme-size units, as shown in the figure. Note, for example, that /ah/ and /l/ are very similar, and it is not impossible that the minor parametric variability that caused the segment boundary at the lower level is just free variation. These phoneme hypotheses give rise to a multiplicity of word hypotheses such as 'ah but', 'all out', 'all about', 'all but', 'are about', and so on.

If, instead of selecting several alternate segmentations and following their consequences, we were to select a single segmentation and associate a single label with each segment, the resulting errors might make it impossible to verify and validate the correct word. Thus some form of network representation of alternate hypotheses at all levels is necessary in systems requiring high accuracy.

Even the lowest level decision about segmentation sometimes requires the active mediation of higher level knowledge such as the plausibility of a given word occurring in that position. Fig. 12 can be used to illustrate the point. The segment boundary at the word juncture of 'all' and 'about' is usually very difficult to find since the spectral characteristics of /l/ and the reduced vowel /ax/ tend to be very similar. In the event that a higher level process is fairly confident about this word sequence but there is no segment boundary, it could call upon the segmenter for a closer look, possibly using a different parametric representation. In general, SUS's require flexible means of cooperation and communication among different

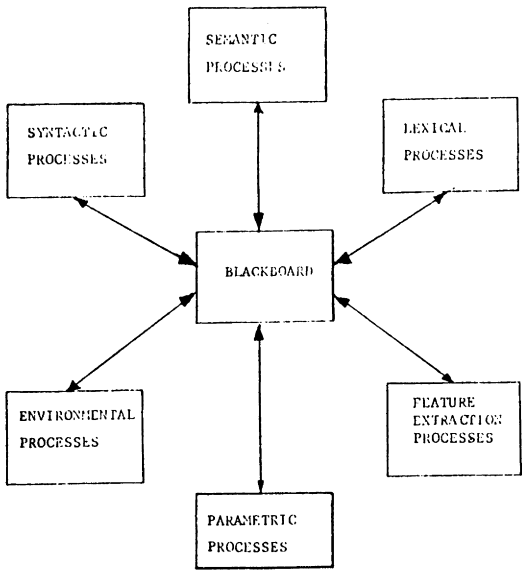

Fig. 13. Blackboard model used in Hearsay-II. (From Lesser *et al.* [86].)

knowledge sources (KS's). Since an SU system tends to have many more KS's than a CSR system, the system should be designed so that knowledge processes can easily be added and deleted.

Finally, the requirements of representation of *understanding, response generation, conversational context,* and *task* all add to the difficulty and overall complexity of an SUS.

Approaches to Speech Understanding: Given the difficulties that arise in SUS, it is clear that one needs significantly more sophisticated system design than those used in current CSR systems. At present, there is no clear agreement among researchers as to what an ideal system organization for an SUS might be.

In the VDMS system [127], the parser coordinates the operation of the system. In many respects the control flow resembles the one for CSR systems (Fig. 10) and is based on the best-first strategy. However, the simplistic notion of an ordered list of word sequences is replaced by a *parse net* mechanism which permits sharing of results and does not require strict left-to-right analysis of the utterance [115]. A language definition facility permits efficient internal representation of various KS's [128].

In the SPEECHLIS system [166], control strategies and system organization are derived through incremental simulation [168]. People located in different rooms simulate the various components and attempt to analyze an utterance by communicating via teletypewriter. Then one by one, people are replaced by computer algorithms having specified interface characteristics. A control strategy for SUS derived in this manner is described by Rovner *et al.* [131]. The final control structure is not available yet but is expected to be within the near future.

Perhaps the most ambitious of all the system organizations is the one used by the Hearsay-II system [86], [37]. Though it was designed with speech understanding systems in mind, it is viewed as one of the potential solutions to the problem of knowledge-based systems (KBS) architecture that is of general interest in artificial intelligence research. Other proposed solutions to the KBS architecture problem include Planner [62], production systems [106], and QA-4 [132].

Hearsay-II is based on a *blackboard* model (Fig. 13). The blackboard model conceives of each KS as an information gathering and dispensing process. When a KS generates a hypothesis about the utterance that might be useful for others, it broadcasts the hypothesis by writing it on the "blackboard"—a structurally uniform global data base. The hypothesis-and-test paradigm (see Section I-A) serves as the basic medium of communication among KS's. The way KS's communicate and cooperate with each other is to validate or reject each other's hypotheses. The KS's are treated uniformly by the system and are independent (i.e., anonymous to each other) and therefore relatively easy to modify and replace. The activation of a KS is data-driven, based on the occurrence of patterns on the blackboard which match the templates specified by the KS.

Most of the control difficulties associated with SUS appear to have a solution within the Hearsay framework. It is easy to delete, add, or replace KS's. The system can continue to function even in the absence of one or more of these KS's as long as there are some hypothesis generators and some verifiers in the aggregate. The blackboard consists of a uniform multilevel network (similar to the one in Fig. 12, but containing all the levels) and permits generation and linkage of alternate hypotheses at all the levels. A higher level KS can generate hypotheses at a lower level and vice versa. It is not necessary for the acoustic processing to be bottom-up and the language model to be top-down.

How does the recognition proceed in an asynchronously activated data-driven system such as Hearsay-II? Since there are not many systems of this type around, it is difficult for most people to visualize what happens. It is difficult to explain using flow charts which are primarily useful for explaining sequential flow of control. What we have here is an activity equivalent to a set of cooperating asynchronous parallel processors even when it runs on a uniprocessor. Generating and verifying hypotheses using several KS's is analogous to several persons attempting to solve a jigsaw puzzle with each person working on a different part of the puzzle but with each modifying his strategies based on the progress being made by the others.

What is important to realize is that within the Hearsay framework one can create the effects of a strictly bottom-up system, top-down system, or system which works one way at one time and the other way the next time, depending on cost and utility considerations. The ratings policy process, a global KS, combines and propagates ratings across levels facilitating focus of attention, goal-directed scheduling, and eventual recognition of the utterance. The focus-of-attention KS is used to determine an optimal set of alternative paths which should be explored further based on notions such as effort spent, desirability of further effort, important areas yet to be explored, and goal lists.

Knowledge Sources: Fig. 14 shows the design choices made by the three systems. Many of the low-level issues are common with CSR and do not require much discussion (see also Sections III, IV, and V). Here we will discuss the nature of the higher level knowledge sources used in each system.

The task for Hearsay-II is news retrieval, i.e., retrieval of daily wire-service news stories upon voice request by the user. The vocabulary size for the task is approximately 1200 words. The syntax for the task permits simple English-like sentences and uses the ACORN network representation developed by Hayes-Roth and Mostow [58]. The semantic and pragmatic model uses subselection mechanisms based on news items of

	Hearsay-II	Speechlis	VDMS
Microphone	Close speaking microphone	Close speaking microphone	Sony ECM-377 condenser microphone
Noise Level	Terminal room	Quiet office	Low (sound booth)
Parametric Rep.	LPC using Itakura metric	Formants and features	Formants and features
Segmentation	Parameter based	Feature based	Classification based
Labeling	Prototype matching	Heuristic	Heuristic
Word Hypothesis	Syllable based	Segment based	Syllable based
Word Verification	Markov Process	Analysis-by-Synthesis	Heuristic match with A-Matrix
Syntax	Restricted English	English-like	English-like
Semantics	Acorn net	Semantic net	Semantic net
Discourse Model	Topic based subselection	User model	Ellipsis and anaphora
Task	News retrieval	Travel Budget Manager	Submarine Data Base Management
Systems Control	Blackboard Model	Centralized Controller	Parser-Based

Fig. 14. Design choices of the three speech understanding systems. (Compiled from Lesser *et al.* [86], Woods [166], Ritea [127], and other related publications.)

1	2	3	4	5				6			
				Acoustic + Syntax				Acoustics + Syntax + Semantics			
Data set: Task/Speaker	Words in lexicon	No. of sentences	No. of words	%Sentences %Near miss	%Words	Average Time per sentence (sec)	Average Time per second of speech	%Sentences %Near miss	%Words	Average Time per sentence (sec)	Average Time per second of speech
1 Chess/Rn	31	14	82	43 100	87	11	6	100 100	100	9	5
2 Chess/Jb	31	19	86	74 95	93	12	9	100 100	100	8	6
3 Chess/Jb	31	21	105	15 50	69	15	8	48 90	88	13	7
4 Chess/(Tel) Bl	31	25	99	52 84	78	7	6	80 88	88	7	6
Totals		79	352	46 80	81	11	7	79 93	93	9	6

Fig. 15. Performance of Hearsay-I speech understanding system. (From Erman [35].) Column 1 gives data set number, task, and speaker identification. Column 2 gives number of words in task lexicon. Column 3 shows number of sentences in data set. Column 4 gives total number of word tokens in data set. Column 5 gives results for HS-I system recognition with Acoustics module and Syntax module both operating. First subcolumn indicates percent of sentences recognized completely correctly. "Near miss" (indicated below that number in first subcolumn) indicates percent of times that recognized utterance differed from actual utterance by at most one word of approximate similar phonetic structure. Second subcolumn gives percent of words recognized correctly. Mean computation times on PDP-10 computer (in seconds per sentence and in seconds per second of speech) are shown in subcolumns three and four. Column 6 shows results for recognition using all three sources of knowledge (for Chess task only): Acoustics, Syntax, and Semantics modules. Subcolumns are similar to those of Column 5.

the data, analysis of the conversation, and the presence of certain content words in the blackboard.

The task for SPEECHLIS is to act as an assistant to a travel budget manager. It permits interactive query and manipulation of a travel budget of a company. It is meant to help a manager keep track of the trips taken or proposed, and to produce summary information such as the total money spent or allocated. The vocabulary is about 1000 words. The syntax permits a wide variety of English sentences and is based on the augmented transition network (ATN) formalism developed by Woods [164]. The parser is driven by a modified ATN grammar [15], [16] which permits parsing to start anywhere,