THE ROLE OF SPEECH PROCESSING IN HUMAN-COMPUTER INTELLIGENT COMMUNICATION

Candace Kamm, Marilyn Walker and Lawrence Rabiner
Speech and Image Processing Services Research Laboratory
AT&T Labs-Research, Florham Park, NJ 07932

Abstract:

We are currently in the midst of a revolution in communications that promises to provide ubiquitous access to multimedia communication services. In order to succeed, this revolution demands seamless, easy-to-use, high quality interfaces to support broadband communication between people and machines. In this paper we argue that spoken language interfaces (SLIs) are essential to making this vision a reality. We discuss potential applications of SLIs, the technologies underlying them, the principles we have developed for designing them, and key areas for future research in both spoken language processing and human computer interfaces.

1. Introduction

Around the turn of the twentieth century, it became clear to key people in the Bell System that the concept of Universal Service was rapidly becoming technologically feasible, i.e., the dream of automatically connecting any telephone user to any other telephone user, without the need for operator assistance, became the vision for the future of telecommunications. Of course a number of very hard technical problems had to be solved before the vision could become reality, but by the end of 1915 the first automatic transcontinental telephone call was successfully completed, and within a very few years the dream of Universal Service became a reality in the United States.

We are now in the midst of another revolution in communications, one which holds the promise of providing ubiquitous service in multimedia communications. The vision for this revolution is to provide seamless, easy-to-use, high quality, affordable communications between people and machines, anywhere, and anytime. There are three key aspects of the vision which characterize the changes that will occur in communications, namely:

- the basic currency of communications switches from narrowband voice telephony to seamlessly integrated, high quality, broadband, transmission of voice, audio, image, video, handwriting, and data.

- the basic access method switches from wireline connections to combinations of wired and wireless, including copper cable, fiber, cell sites, satellite, and even electrical power lines.
- the basic mode of communications expands from people-to-people to include people-to-machines.

It is the third aspect of this vision that most impacts research in human computer intelligent communication. Although there exist a large number of modalities by which a human can have intelligent interactions with a machine, e.g., speech, text, graphical, touch screen, mouse, etc., it can be argued that speech is the most intuitive and most natural communicative modality for most of the user population. The argument for speech interfaces is further strengthened by the ubiquity of both the telephone and microphones attached to personal computers, which affords universal remote as well as direct access to intelligent services.

In order to maximize the benefits of using a speech interface to provide natural and intelligent interactions with a machine, the strengths and limitations of several technologies need to be fully understood. These technologies include:

- coding technologies that allow people to efficiently capture, store, transmit, and present high quality speech and audio;
- speech synthesis, speech recognition, and spoken language understanding technologies that provide machines with a mouth to converse (via text-to-speech synthesis), with ears to listen (via speech recognition), and with the ability to understand what is being said (via spoken language understanding);
- user interface technologies which enable system designers to create habitable human-machine interfaces and dialogues which maintain natural and sustainable interactions with the machine.

In the remainder of this paper we first motivate the growing need for SLIs by describing how telecommunication networks have and are continuing to evolve. Then we discuss the wide range of current and potential applications of SLIs in the future communications network. We next describe the current status of speech and human computer interaction technologies that are key to providing high quality SLIs, including speech and audio coding, speech synthesis, speech recognition, and spoken language processing. We next turn to the question of how to put these technologies together to create a natural, easy-to-use, spoken language interface. We then discuss a number of design principles for SLIs that we have developed and show how they have been instantiated in a range of SLIs. We conclude with a brief discussion of areas for further research.

2. **The Evolution of Telecommunication Networks**

Figure 1 shows a simplified picture of how the fundamental telecommunications networks have evolved over their first 100 years of existence (Bryan Carne, 1995). Basically there are two distinct networks, namely POTS (Plain Old Telephone Service) and PACKET (often called the Internet).

The POTS network is a connection-oriented, narrowband, voice centric network whose main functionality is digital switching and transmission of 3 kHz voice signals (from a standard telephone handset) digitized to 64 Kbps digital channels. Data (in the form of modem signals from a PC or a FAX machine) can be handled rather clumsily by the POTS network through the use of a voiceband modem which limits the speed of transmission to rates below 64 Kbps. Services on the POTS network (e.g., basic long-distance, 800 number calling, call forwarding, directory services, conferencing of calls, etc.) are handled via a distributed architecture of circuit switches, databases and switch adjuncts. Signaling (for call setup and call breakdown, look ahead, routing, database lookup and information passing) is handled by a side (out of band) digital channel (the so-called SS7 (Signaling System 7) system) which is essentially a parallel 64 Kbps digital network.

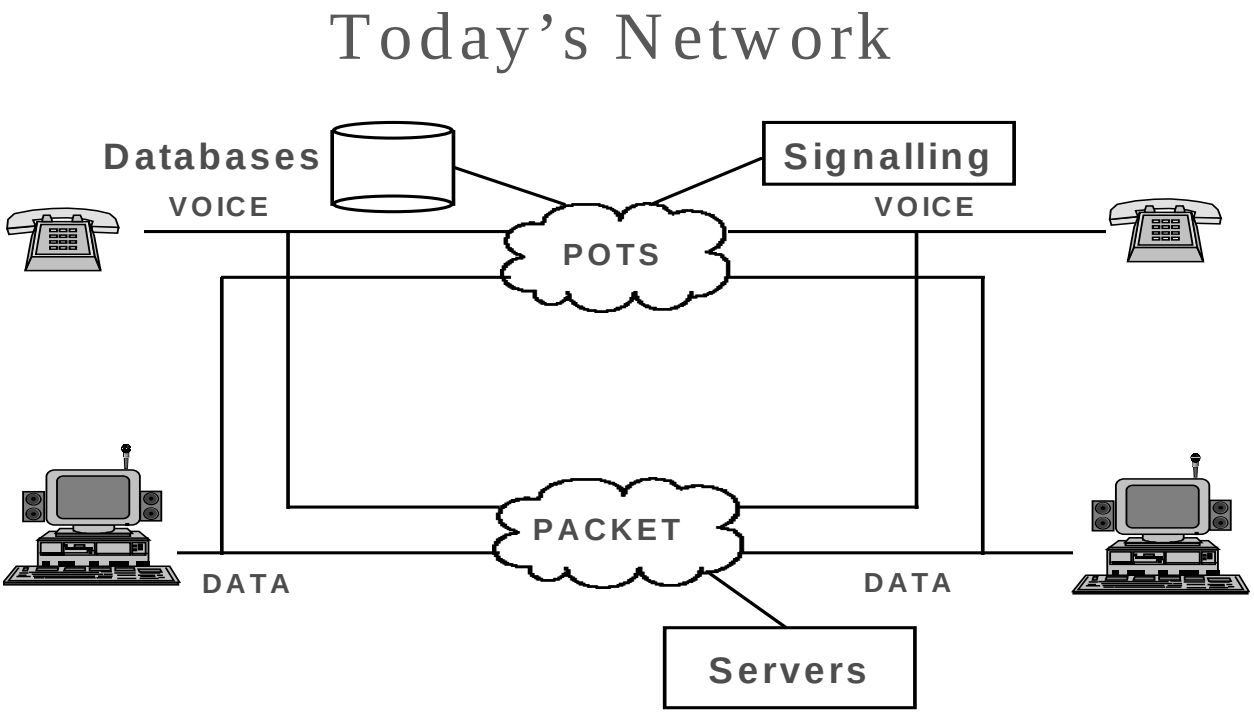

Fig. 1. The telecommunications network of today.

The PACKET network is a connectionless, wideband, data centric network whose main functionality is routing and switching of data packets (so-called IP (Internet Protocol) packets) from one location to another in the network using the standard transmission protocol, TCP/IP (Transmission Control Protocol/Internet Protocol), associated with the Internet. The packets consist of a header and a payload, where the header contains information about the source and destination addresses of the packet, and the payload is the actual data being transmitted. The PACKET network was designed to efficiently handle the high burstiness of data transmission. Voice signals can be handled by the PACKET network, albeit rather clumsily, because of several inherent problems including the long delays associated with routing and switching in the PACKET network, the irregularity of transmission due to variable congestion in the network, and the need for sophisticated signal processing to compress and digitize speech into appropriate size packets. Services in the PACKET network are provided by servers attached to the PACKET network and running in a client-server mode. Typical services include browsing, searching, access to newspapers and magazines, access to directories, access to bookstores, stock offerings, etc. Since packets are self routing, there is no outside signaling system associated with PACKET networks.

A major problem in today's network is that, because of their separate evolution, the POTS and PACKET networks are only weakly coupled, at best. Hence services that are available on the POTS network cannot be accessed from a PC connected to the PACKET network; similarly services that are available on the PACKET network cannot be accessed from a telephone connected to the POTS network. This is both wasteful and inefficient, since essentially identical services are often offered over both networks (e.g., directory services, call centers, etc.), and such services need to be duplicated in their entirety.

Recently, however, telecommunications networks have begun to evolve to the network architecture shown in Figure 2. In this architecture, which already exists in some early instantiations, there is tight coupling between the POTS and PACKET networks via a Gateway which serves to move POTS traffic to the PACKET network, and PACKET traffic to the POTS network. Intelligence in the network is both local (i.e., at the desktop in the form of a PC, a screen phone, a Personal Information Manager, etc.), and distributed throughout the network in the form of active databases (e.g., an Active User Registry (AUR) of names and reach numbers), and services implemented at and attached to the Gateway between the POTS and PACKET networks. In this manner any given service is, in theory, accessible over both the POTS and PACKET networks, and from any device connected to the network.

The evolving telecommunications network of Figure 2 reflects a major trend in communications, namely that an ever increasing percentage of the traffic in the network is between people and machines. This traffic is of the form of voice and electronic mail messages, Interactive Voice Response (IVR) systems, and direct machine queries to access information.

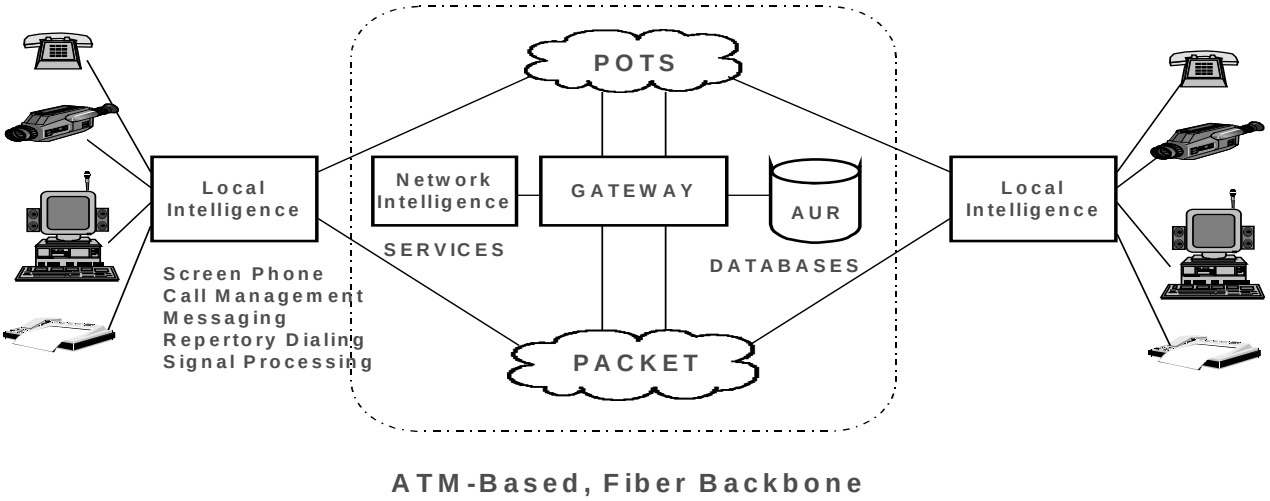

Fig. 2. The telecommunications network of tomorrow.

A major implication of the evolving telecommunications network is that in order to access any service made available within the network, a compelling, customizable, seamless, easy-to-use, and high quality spoken language interface (SLI) is required. Although other user interface modes, such as text, graphical, mouse, touch screen, and touch-tone, might also be satisfactory in some circumstances, the SLI is the only one that is ubiquitously available (via both telephone handsets and stand-alone microphones), permits hands-free and eyes-free communication, is efficient in communication, and is a natural mode for many common tasks. Section 3 discusses a range of current and future applications of SLIs. The remainder of the paper discusses what is required to build such interfaces.

3. Applications of Spoken Language Interfaces

The precursors to spoken language interfaces in telecommunications services are called Interactive Voice Response (IVR) systems, where the system output is speech (either pre-recorded and coded, or generated synthetically), and user inputs are generally limited to touch-tone key-presses. Currently, IVR is used in many different application domains, such as electronic banking, accessing train schedule information, and retrieval of voice messages. Early implementations of SLIs often simply replaced the key-press commands of IVR systems with single word voice commands (e.g., push or say 'one' for service 1). More advanced systems allowed the users to speak the service name associated with the key-push (e.g., "For help, push 'one' or say 'help'"). As the complexity of the task associated with the IVR applications increased, the systems tended to become confusing and cumbersome, and the need for a more

flexible and more habitable user interface became obvious to most system developers. It should also be clear that the task domains appropriate for SLIs are a superset of the applications that are handled by existing IVR systems today.

Application domains for Spoken Language Interfaces (SLIs) can be grouped into two broad areas, namely:

- Human-Computer Communication Applications
- Computer-Mediated Human-Human Communication Applications.

3.1 Human-computer communications applications

Human-computer communication applications access information sources or services on a communication network. Many such information sources are already available in electronic form. These include personal calendars and files, stock market quotes, business inventory information, product catalogues, weather reports, restaurant information, movie schedules and reviews, cultural events, classified advertisements, and train and airline schedules. There is also a range of applications that provide individualized customer service, including personal banking, customer care, and support services.

Several research labs have developed prototype SLIs for accessing personal calendars (Marx & Schmandt, 1996; Yankelovich et al., 1995). Some of these SLIs allow querying of a personal calendar as well as the calendars of other users, e.g., the user can say "What meetings does Don have today?". There are also many prototype SLIs that focus on information accessible from the web such as stock market quotes (Yankelovich et al., 1995), weather reports (Sadek et al., 1996), movie schedules and reviews (Wisatowaty, 1995), and classified advertisements (Meng et al., 1996). Some of these SLIs are available now in early market offerings. Given the rapid increase in the number and variety of such information sources, we would expect many new SLI applications to appear over the next few years.

A number of government funded, research projects have focused on creating SLIs for accessing train and airline schedule information in both the United States and Europe in the early 1990's. This led to the development of a number of Air Travel Information Systems (ATIS) at various research labs in the United States (Hirschman et al. 1993; Levin and Pieraccini,1995), and Train Timetable systems at various research labs in Europe (Danieli et al, 1992; Danieli and Gerbino, 1995, Bennacef et al, 1996; Simpson and Fraser, 1993). Some of these systems are close to deployment; field studies have recently been carried out to compare a version of a Train Timetable SLI that restricts the user to single word utterances to a version that allows users more freedom in what they say to the system (Danieli and Gerbino, 1995; Billi et al., 1996)

3.2 **Computer-Mediated Human-Human Communication Applications**

In contrast to human-computer communication applications, computer-mediated human-human communication applications are based on functionality for accessing and using the network to communicate with other humans. These applications include SLIs for voice calling, for retrieving and sending email and voice mail, for paging and faxing, and for translating what a user says in one language into a language that the other user understands. In addition to supporting remote access that is both hands and eyes-free, SLIs for these applications can also provide functionality that is difficult or impossible to provide with touch-tone inputs or other modalities (Walker & Whittaker, 1989; Walker, 1989).

By way of example, voice calling allows users to simply pick up the phone and say "Call Julia" or "Call the electric company" rather than needing to find or remember the number and then enter it by touch tone (Perdue and Scherer, 1996, Kamm, 1995). SLIs for voice calling rely on several types of information sources, such as personalized directories, community directories, and 800 number directories. Personalized directories support voice calling by storing a user's personal address book. Similarly, community directories store information that is geographically relevant to the user such as the electric company or the community theater.

There are several prototype SLIs for accessing email or voice mail by phone (Yankelovich et al., 1995; Marx and Schmandt, 1996; Walker et al., 1997a). MailCall (Marx and Schmandt, 1996) makes novel use of information from the user's personal calendar to prioritize mail. ELVIS (Walker et al., 1997a) provides the ability to search and prioritize mail by content or by sender, by supporting queries such as "List my messages from Julia about the dialogue workshop". This type of SLI makes it possible to support random access to messages by particular time-relevant or subject-relevant search criteria rather than simply by the linear order in which messages were received and stored.

SLIs for speech-to-speech translation are under development in research labs in the United States, Europe and Asia (Alshawi, 1996; Bub & Schwinn, 1996, Sumita & Iida, 1995). Currently these SLIs focus on domains related to business travel, such as scheduling meetings or making reservations with car rental agencies or hotels. These projects have focused on a small set of common languages including English, Chinese, Spanish, Japanese and German.

The boundary between computer-mediated human-human communication and human-computer communication blurs somewhat in voice enabled personal agent applications. In these applications, an anthropomorphic personal agent metaphor can be used as an interface to a suite of call control, messaging, and information retrieval services (Kamm et al., 1997).

The next two sections describe the current state of the art in spoken language and human computer interface technologies that make it possible to build SLIs to successfully implement this wide range of applications.

4. Speech Technologies

There are three key speech technologies that provide the underlying components for spoken language interfaces, namely: (1) speech and audio coding; (2) text-to-speech synthesis; and, (3) speech recognition and spoken language understanding. These technologies have an obvious role in SLIs, namely they provide the computer with a "mouth", an "ear" and a "brain", respectively. The role of speech and audio coding in an SLI is more subtle, but equally important, namely providing the high quality audio signals that are critical to intelligibility and user acceptance of a system with speech output. The remainder of this section describes each of these technologies and illustrates how different types of applications have widely differing technology requirements.

4.1 Speech and Audio Coding Technology

Speech coding technology is used for both efficient transmission and storage of speech (Kleijn & Paliwal, 1995). For transmission applications, the goal is to conserve bandwidth or bit rate, while maintaining adequate voice quality. For storage applications the goal is to maintain a desired level of voice quality at the lowest possible bit rate.

Speech coding plays a major role in three broad areas: namely, the wired telephone network, the wireless network (including cordless and cellular), and for voice security for both privacy (low level of security) and encryption (high level of security). Within the wired network the requirements on speech coding are rather tight with strong restrictions on quality, delay, and complexity. Within the wireless network, because of the noisy environments that are often encountered, the requirements on quality and delay are often relaxed; however, because of limited channel capacity the requirements on bit rate are generally tighter (i.e., lower bit rate is required) than for the wired network. Finally, for security applications, the requirements on quality, delay, and complexity are generally quite lax. This is because secure speech coding is often a requirement on low-bandwidth channels (e.g., military communications) where the available bit rate is relatively low. Hence, lower quality, long delay, low bit rate algorithms have generally been used for these applications.

All (digital) speech coders can be characterized in terms of four attributes; namely, bit rate, quality, signal delay, and complexity. The *bit rate* is a measure of how much the "speech model" has been exploited in the coder; the lower the bit rate, the greater the reliance on the

speech production model. *Quality* is a measure of degradation of the coded speech signal and can be measured in terms of speech intelligibility and perceived speech naturalness. *Signal delay* is a measure of the duration of the speech signal used to estimate coder parameters reliably for both the encoder and the decoder, plus any delay inherent in the transmission channel. (Overall coder delay is the sum of the encoder delay, the decoder delay, and the delay in the transmission channel.) Generally the longer the allowed delay in the coder, the better the coder can estimate the synthesis parameters. However, long delays (on the order of 100 ms) are often perceived as quality impairments and sometimes even as echo in a two-way communications system with feedback. Finally, *complexity* is a measure of computation required to implement the coder in digital signal processing (DSP) hardware.

A key factor in determining the number of bits per second required to code a speech (or audio) signal is the signal bandwidth. Figure 3 shows a plot of speech and audio signal bandwidth for four conventional transmission and/or broadcast modes; namely, conventional telephony, AM-radio, FM-radio, and compact disc (CD). Conventional telephone channels occupy a bandwidth from 200 to 3400 Hz; AM-radio extends the bandwidth on both ends of the spectrum to cover the band from 50 to 7000 Hz (this is also the bandwidth that most audio/video teleconferencing systems use for transmission of wideband speech); FM-radio extends the spectrum further (primarily for music) to the range 20 to 15,000 Hz; and the range for CD audio is from 10 to 20,000 Hz.

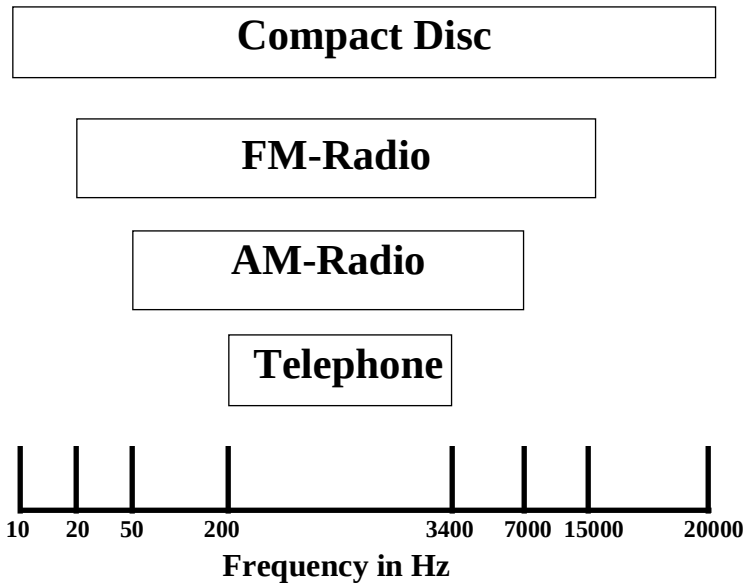

Fig. 3. Plot of speech and audio frequency bands for telephone, AM-radio, FM-radio, and compact disc audio signals.

The “ideal” speech coder has a low bit rate, high perceived quality, low signal delay, and low complexity. No ideal coder as yet exists with all these attributes. Real coders make tradeoffs among these attributes, e.g., trading off higher quality for increased bit rate, increased delay, or increased complexity.

Figure 4 shows a plot of speech coding quality (measured as a subjective Mean Opinion Score (MOS)) versus bit rate for a wide range of standards-based and experimental laboratory coders. Also shown are the curve of MOS versus bit rate that was achieved in 1980 and in 1990.

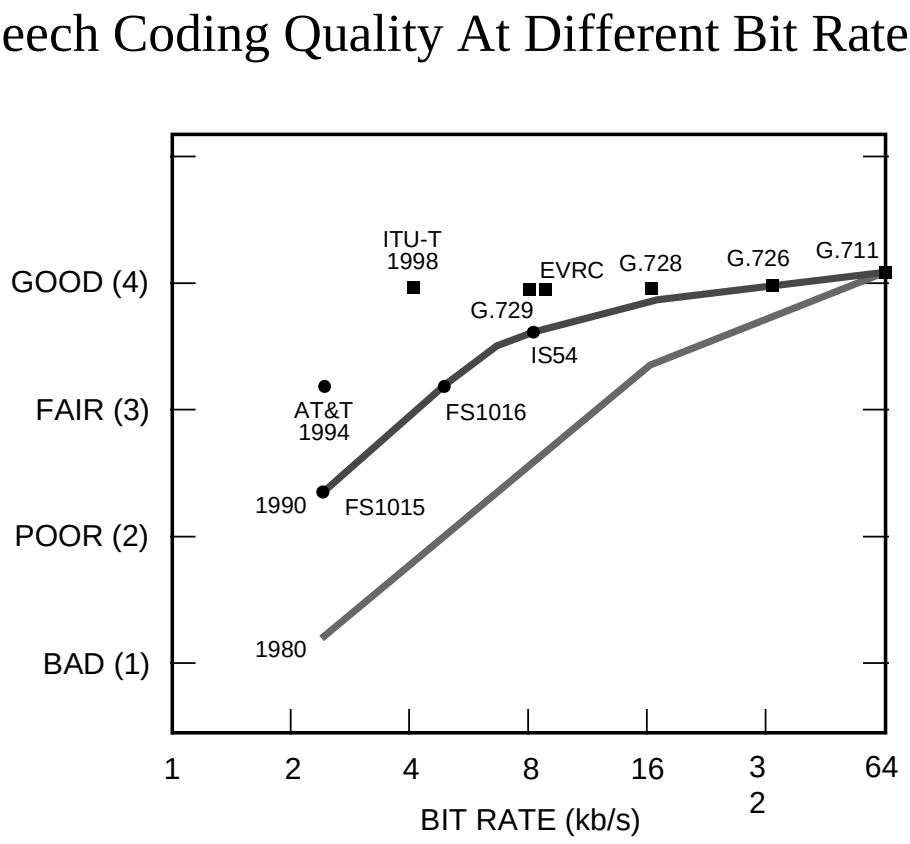

Fig. 4. Speech quality mean opinion scores of several conventional telephone coders as a function of bit rate**.**

It can be seen that it is anticipated that by the turn of the century, there will be high subjective quality coders at bit rates as low as 4 Kbps. Further, there are indications that high quality speech coders may ultimately be achieved at bit rates as low as 1 Kbps. This will require a great deal of innovative research, but theoretically such low bit rate coders are achievable.

Wideband speech coding provides more natural sounding speech than telephone bandwidth speech, and leads to the perception of being in the same room with the speaker. This attribute could have a strong, positive, and pervasive effect on a user's perceptions of the quality of SLIs, especially SLIs involving voice messaging and conferencing. Based on the growing usage and demand for wideband speech in telecommunications, standard coding methods have been applied and have been shown capable of providing high-quality speech (MOS scores of 4.0 or higher) in the 32-64 Kbps range. Current research is focused on lowering the bit rate to the 8-16 Kbps range, while maintaining high quality so as to provide audio/video teleconferencing at 128 Kbps with 112-120 Kbps provided for video coding, and 8-16 Kbps for high quality audio coding.

CD quality audio coding is essential for any SLI application involving digital audio, e.g., SLIs to preview new CD's, as well as to access and listen to music samples provided by many web sites. With the advent of mass marketed devices for digital coding and storage of high-fidelity audio, including the Compact Disc (CD), the digital audio tape (DAT), and most recently the minidisk (MD), and the digital compact cassette (DCC), the area of efficient digital coding of high-fidelity audio has become a topic of great interest and a great deal of activity. Also driving this activity is the need for a digital audio standard for the sound for high-definition TV (HDTV) and for digital audio broadcasting (DAB) of FM-channels.

To appreciate the importance of coding digital audio efficiently and with quality which is essentially indistinguishable from that of an original CD, consider the bit rate that current CD's use to code audio. The sampling rate of a CD is approximately 44.1 kHz and each sample (for both channels of a stereo broadcast) is coded with 16-b accuracy. Hence, a total of 44.1 x 2 x 16 or 1.41 Mbps is used to code digital audio on a CD. Current state-of-the-art coding algorithms, such as the Perceptual Audio Coder or PAC developed at AT&T, are capable of coding 2 channels of digital audio at a total bit rate in the range of 64-128 Kbps with essentially no loss in quality from that of the original CD coding (Johnston & Brandenburg, 1991).

4.2 Speech Synthesis

Spoken language interfaces rely on speech synthesis, or text-to-speech (TTS) systems to provide a broad range of capability for having a machine speak information to a user (Sproat & Olive, 1995; Van Santen et al., 1996). While, for some applications, it is possible to concatenate and play pre-recorded segments, this is not possible in general. Many applications are based on

dynamic underlying information sources for which it is difficult to predict what the system will need to say. These include applications that involve generating natural language sentences from structured information, such as a database record, as well as those involving unstructured information, such as email messages or the contents of a WEB page.

TTS systems are evaluated along two dimensions, namely the intelligibility of the resulting speech, and the naturalness of the speech. Although system performance varies greatly for different tasks and different synthesizers, the best TTS systems achieve word intelligibility scores of close to 97% (natural speech achieves 99% scores); hence the intelligibility of the best TTS systems approaches that of natural speech. Naturalness scores for the best TTS systems, as measured by conventional MOS scores are in the 3.0-3.5 range, indicating that the current quality of TTS is judged in the fair-to-good range, but most TTS systems still do not match the quality and prosody of natural speech.

4.3 Speech Recognition and Spoken Language Understanding

The ultimate goal of speech recognition is to enable a machine to literally be able to transcribe spoken inputs into individual words, while the goal of spoken language understanding research is to extract meaning from whatever was recognized (Rabiner & Juang, 1993; Rabiner et al., 1996). The various SLI applications discussed earlier have widely differing requirements for speech recognition and spoken language understanding; hence there is a range of different performance measures on the various systems that reflect both the task constraints and the application requirements.

Some SLI applications require a speech recognizer to do word-for-word transcription. For example, sending a textual response to an email message requires capabilities for voice dictation, and entering stock information or ordering from a catalogue may require entering number sequences or lists of data. For these types of systems, *word error rate* is an excellent measure of how well the speech recognizer does at producing a word-for-word transcription of the user's utterance. The current capabilities in speech recognition and natural language understanding, in terms of word error rates, are summarized in Table 1. It can be seen that performance is pretty good for constrained tasks (e.g., digit strings, travel reservations), but that the word error rate increases rapidly for unconstrained conversational speech. Although methods of adaptation can improve performance by as much as a factor of two, this is still inadequate performance for use in many interesting tasks.

For some applications, complete word-for-word speech recognition is not required; instead, tasks can be accomplished successfully even if the machine only detects certain keywords or key phrases within the speech. For such systems, the job of the machine is to categorize the user's utterance into one of a relatively small set of categories; the category identified is then mapped to an appropriate action or response (Gorin et al., 1997). An example of this type of system is AT&T's 'How May I Help You' (HMIHY) task in which the goal is to

classify the user's natural language spoken input (the reason for calling the agent), into one of fifteen possible categories, such as billing credit, collect call, etc. Once this initial classification is done, the system transfers the caller to a category specific subsystem, either another artificial agent or a human operator. (Gorin et al., 1997; Boyce and Gorin, 1996). *Concept accuracy* is a more appropriate measure of performance for this class of tasks than word accuracy. In the HMIHY task, word accuracy is only about 50 percent, but concept accuracy approaches 87 percent (Gorin et al., 1997).

CORPUS	TYPE	VOCABULARY SIZE	WORD ERROR RATE
Connected Digit Strings	Spontaneous	10	0.3%
Airline Travel Information	Spontaneous	2500	2.0%
Wall Street Journal	Read Text	64,000	8.0%
Radio (Marketplace)	Mixed	64,000	27%
Switchboard	Conversational Telephone	10,000	38%
Call Home	Conversational Telephone	10,000	50%

Table 1
Word Error Rates for Speech Recognition and Natural Language Understanding Tasks
(Table courtesy of John Makhoul, BBN)

Another set of applications of speech recognition technology are so-called Spoken Language Understanding (SLU) systems, where the user is unconstrained in terms of what can be spoken and in what manner, but is highly constrained in terms of the context in which the machine is queried. Examples of this type of application include AT&T's CHRONUS system for air travel information (Levin & Pieraccini, 1995), and a number of prototype railway information systems described in Section 3. As in the HMIHY example, results reported by Bennacef et al. (1996) show speech understanding error rates of 6-10 percent, despite recognition error rates of 20-23 percent. These results demonstrate how a powerful language model can achieve high understanding performance despite imperfect ASR technology.

5. Human Computer Interface Technologies

Section 3 discussed the wide range of information services which we anticipate will be available on tomorrow's communication network that will be accessible with spoken language interfaces (SLI). Section 4 described the current state of the art in the underlying spoken

language technologies, which has progressed to the point that many prototype applications now appear in research laboratories and in early market offerings. However, in order to create successful speech-enabled applications with these rapidly maturing but still imperfect technologies, user interfaces to spoken dialogue systems must mitigate the limitations of both current speech technologies and human cognitive processing. Previous research has focused primarily on advancing the performance of the technologies; the new research challenge is to understand how to integrate these technologies into viable, easy-to-use spoken language systems. Previous work in both interface design and dialogue systems is applicable to the design of SLIs. Figure 5 shows a widely-accepted, general paradigm for iterative design and usability testing of user interfaces. The steps in the process are:

1. Design the UI to match the task and the operating conditions.
2. Build a trial system with the UI designed in step 1.
3. Experiment with real users.
4. Evaluate the success measures associated with overall system performance.
5. Loop back to step 1 and make improvements in the UI design based on results of the evaluation.

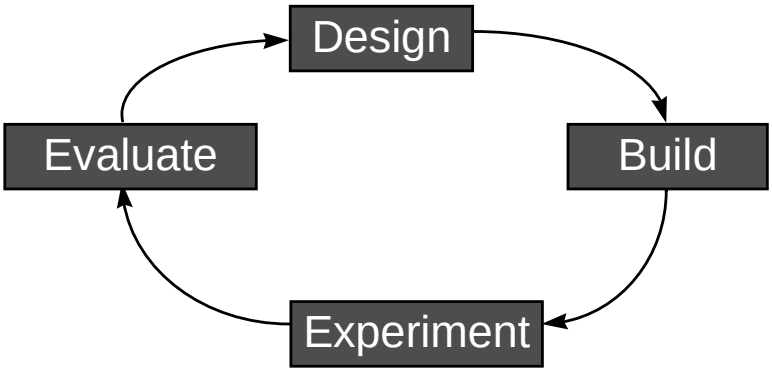

Fig. 5. The spoken dialogue development cycle.

Some of the basic design principles that are well-understood in graphical user interfaces (GUI) are equally applicable to SLIs. Three key design principles for GUI's are (Shneiderman, 1986):

- *continuous representation* of the objects and actions of interest. Here the goal is to keep the graphical objects 'in the face' of the user so that it becomes both obvious and intuitive what can be done next.

- *rapid, incremental, reversible* operations whose *impact* on the object of interest is *immediately* visible. Here the goal is to make sure that every action has an immediate and unmistakable response that can easily be undone if the resulting state is incorrect or not the one that the user desires.
- physical actions or labeled button presses instead of complex syntax with natural language text commands.

To some extent, all these principles reflect an understanding of the limitations of human cognitive processing (Whittaker & Walker, 1991). Humans are limited both in their ability to recall known information, as well as to retain large amounts of information in working memory (Miller, 1956; Baddeley, 1986). Continuous representation addresses the user's difficulty remembering what options are available, while having physical actions associated with labeled button presses means that users do not have to remember details of a command syntax. Similarly, rapid operations with immediate, visible impact maintain the user's attention on the task at hand, facilitating efficient task completion.

Well-designed, spoken language interfaces also address human cognitive limitations. However, the non-persistent, temporal nature of audio signals, coupled with the limited capacity of auditory memory, impose additional requirements for SLI design. Obviously, without a visual display, different methods are necessary to instantiate design principles like "continuous representation" and "immediate impact". Furthermore, whereas a persistent visual display permits presentation of large amounts of information in tabular format, which can be easily scanned and browsed, audio-only interfaces must summarize and aggregate information into manageable pieces that a user can process and cope with effectively.

For example, consider a complex task like information retrieval, where the amount of information to be provided to the user is large. In this case, the limitations on human auditory processing capacity impose constraints on the system that drive SLIs toward dialog-based interactions. In these dialogs, the system presents information incrementally, and multiple exchanges between the agent and the user are often necessary to extract the specific information the user requires.

Thus, in addition to SLI design principles covering a single exchange between the user and the agent, SLI design extends to the broader problem of dialogue management (Abella et al., 1996). For example, dialogue management requires that the system keep track of the current context, including what the user has already said, what the system has already provided, and previous misrecognitions and misunderstandings by the system.

The remainder of this section presents several examples of the application of the principles of continuous representation, immediate impact, reversibility, incrementality, and summarization/aggregation in audio-only speech-enabled interfaces.

Simulating a “continuous representation” in an audio-only interface requires a trade-off between providing a full representation of the available options in each audio prompt and not providing sufficient information. Novice users might desire (and require) a full representation of all the options, but this is likely to be unacceptably long for an experienced user. One compromise strategy is to provide prompts in a “question - pause - options” format, where the “options” serve as a reminder of what the user can say at this point in the dialogue. Figure 6 shows an example of this strategy, in the context of the personal communications agent application described in Section 3.

Agent: Maxwell here, what can I do for you?
User: (says nothing within allocated two second speech interval)
Agent: You can say “call”, followed by a name or number, “Get my messages”, or “Get me the employee directory”. For more options, say “Help me out”.

Figure 6. Design Principle: Continuous Representation

In this example, the user is reminded of the available options when the agent detects that the dialogue is not progressing as expected. That is, because the system has not detected speech within two seconds after the agent’s initial prompt (“What can I do for you?”), the assumption is that the user may not know what to do. At that point, the agent explains how to perform the most frequently used functions, as well as how to find out about more options.

In addition to reminders when a time-out has occurred, another strategy for establishing a “continuous representation” in an audio-only interface is to provide consistency across features in a multi-featured application (e.g., the personal agent for communications services). This consistency can be achieved by using a vocabulary that is “almost always available” and serves the same function regardless of where the user is in the feature space. For example, at any point in the interaction, the command “help me out” obtains context dependent instructions and the command “cancel” undoes the preceding request, as shown in the example in Figure 7. In this example, the user gives the command “help me out” first at the top level of the quasi-hierarchical feature space, and then later inside a module that provides access to an employee directory. In each case, the user receives a help message from the agent appropriate to the options available at that point in the interaction.

A third strategy for providing continuous representation in an audio-only interface is to give users audible cues to orient them with the task space. These cues can be brief, distinctive sound effects (“earcons”) or spoken prompts that the agent says to indicate the user’s “location”. In the example shown in Figure 7, the prompt “Maxwell here” cues the user that the top level vocabulary is active, and the prompt “Employee Directory” indicates that the interaction has entered the Employee Directory module.

Agent: Maxwell here. What can I do for you?

User: Help me out.
Agent: You can say “call”, followed by a name or number, “Get my messages”, or “Get me the employee directory”. You can always say “I’m done…..
User: (Barging in over Agent’s speech) Get me the employee directory.
Agent: First message…
User: (Barging in over Agent’s speech) Cancel.
Agent: Maxwell here.
User: Get me the employee directory.
Agent: Employee directory.
User: Help me out.
Agent: Say the first and last name of the person you want to call.

Figure 7. Design Principle: Continuous Representations & Immediate Impact

The agent’s response of “Employee Directory” also satisfies the design principle of providing immediate, observable impact, by providing implicit feedback that the user’s previous request was understood and acted on correctly. In contrast, the user’s first request of “Get me the Employee Directory” in Figure 7 was misrecognized by the system as “Get my messages”. In this case, the agent’s implicit confirmation statement “First message” was an immediate cue to the user that an error had occurred. The user then spoke the command “cancel”, which caused the system to undo the previous action (the erroneous foray into the messaging module), and return to the top level of the interaction. In order to achieve the design principle of immediacy, the system must have real-time, low latency responses, and must also allow the user to “barge-in” while the system is speaking (Krauss & Bricker, 1967). Effective audio interfaces must be able to listen even while they are presenting prompts and must stop speaking when they are interrupted by the user (following human conversational etiquette). This principle provides a sense of continuity and natural flow to the dialogue.

The principle of incrementality is illustrated in the interaction in Figure 8, taken from the Air Travel Information System application described in Sections 3 and 4. In this case, the user has provided the originating airport but has not given sufficient information for the agent to disambiguate the destination airport, so the agent asks a clarifying question to determine the destination, and then proceeds to ask about the next piece of information that the system needs in order to complete the query to the air travel database.

Incrementality is also a useful design strategy for handling instances when the speech recognizer only recognizes a portion of the utterance and needs to reprompt for the missing information, as in the call completion example in Figure 9.

User: I want to go from Boston to San Francisco.
Agent: San Francisco is served by two airports, San Francisco International and Oakland International. Which airport do you want?
User: San Francisco International.
Agent: When do you want to travel from Boston to San Francisco International Airport?

Figure 8 . Design Principle: Incrementality I

User: Call John Doe.
Agent: Call who?
User: John Doe.
Agent: Calling John Doe.

Figure 9. Design Principle: Incrementality II

The principle of summarization and aggregation of information is particularly important to audio-only interfaces, where the options for scanning and browsing information are quite limited. An example of the use of summarization and aggregation from a system that provides voice access to electronic mail is shown in Figure 10. In this example, rather than reading out a list of twenty electronic mail message subjects and senders, the system tells the user how many messages there are, and provides some additional information that has been previously specified as of importance to this user (i.e., "important messages from President Clinton and from Vice President Gore").

Agent: You have 20 new e-mail messages, with important messages from President Clinton and from Vice President Gore.
User: Do I have any messages about meetings for today?
Agent: You have 3 messages about meetings for today: one from Gore about "NII meeting at 3 pm", one from Shalala about "Budget Meeting" and one from your secretary about "Staff Meeting at 8 am".
User: Please read the messages about meetings.

Figure 10. Design Principle: Summarization and Aggregation

In addition to these general design principles, successful spoken dialogue interactions may be facilitated by customizing and adapting the system to the needs and preferences of individual users. To effectively customize the interaction, an accurate user model is required. The user model includes assumptions about what the user knows about the system and how the user prefers to interact with the system.

For example, for the personal communication agent application, the user model for an "expert" user might include the following: a) the expert knows and remembers what the system feature set is and what commands invoke those features, b) the expert prefers terse, implicit confirmation strategies that move the dialogue along as quickly as possible, and c) the expert typically speaks to the system in terse telegraphic commands.

In contrast, the user model for a "novice" user might be that: a) the novice remembers a few commands, but often will need reminders of what is available, b) the novice may prefer more thorough confirmation to assure him/her that the dialogue is progressing correctly, and c) the novice is apt to provide only partial information, requiring more frequent use of incremental strategies.

Based on these differences in underlying assumptions about expert and novice users, the customized dialogs shown in Figures 11 and 12 might be proposed for the two user populations. As a user gains experience with the system, the system could track the user's behaviors, and after the number of successful uses of a feature exceeds some critical threshold, the agent could offer the user the option of shifting from the novice interaction to the expert version of the system.

Agent: Annie here.
Expert: Get my labels.
Agent: Labels.
Expert: Delete the label for Candy Kamm.
Agent: Deleting Candy Kamm. Labels, what next?

Figure 11. Expert Dialog

Agent: Annie here. What would you like me to do?
Novice: Get my labels.
Agent: I've got your labels. What would you like me to do?
Novice: Delete a label.
Agent: Say the name or number you want to delete.
Novice: Candy Kamm.
Agent: Are you sure you want to delete Candy Kamm?
Novice: Yes.
Agent: OK, I'll delete Candy Kamm and I'll get right back to you.

Figure 12: Novice Dialog

User-specific usage patterns and preferences for system features offer an additional information source for customizing and adapting the interface to these systems. For example, if the personal agent detects that the user has called a particular telephone number repeatedly over a short time period, the agent may offer the user the option of adding that number to the user's personal voice dialing list. The impact of customization and system adaptation on task performance and user satisfaction with complex human-computer interactions has not been explored systematically, and, along with the research areas described in the following section, is one of many aspects of SLIs that need further study.

6. Future Research Directions

In order to progress from the current capability of limited domain prototype systems, to SLIs with improved performance and wider functionality, research is needed in several key areas. These research areas encompass not only the underlying speech technologies, but also include the integration of the component technologies, the relevant information sources, and the user interface.

For the underlying speech technologies one key research goal is improving the naturalness of TTS to approach that of prerecorded speech. Experience with the current generation of TTS systems has shown that the use of TTS systems will increase significantly as the voice quality becomes more acceptable (natural sounding). There are several systems that have recently achieved significantly higher naturalness scores, based on improved synthesis methods and on improved waveform selection techniques (Campbell and Black, 1997; Sagisaka et al., 1992, Bigorgne et al., 1993). Another goal for future research is to provide the capability for TTS systems to have an arbitrary voice, accent, and language, so as to customize the system for different applications (e.g., having a "celebrity" voice for an agent) (Sorin, 1994). Such a capability demands rapid and automatic training of synthesis units, and a facility for matching the intonation, pitch, and duration of an individual talker rapidly and accurately. Another area of research, which may prove beneficial in both improving the naturalness and efficacy of human-computer dialog, involves using higher level discourse information in determining the appropriate prosodic contours for the dialog generated by the system (Hirschberg 1993; Hirschberg and Nakatani, 1996).

Research goals in automatic speech recognition include trying to improve the robustness of ASR to variations in the acoustic environment, user populations (e.g., children, adults), and transmission characteristics (telephone, speakerphone, open microphone on desktop) of the system. We also believe that new and interesting systems will evolve, with better user interface designs, as we obtain a better understanding of how to rapidly bootstrap and build language models for new task domains. The effective use of systems with more flexible grammars is a research goal that requires effort in both speech recognition and user interface technologies.

A critical research topic, related to the integration of user interface technologies into dialogue systems, is the basic issue of how to evaluate spoken dialogue systems (Danieli et al., 1992; Danieli & Gerbino, 1995; Hirschman et al., 1993; Simpson & Fraser, 1993; Sparck-Jones & Galliers, 1996; Whittaker & Stenton, 1989). In order to test the effects of different dialogue strategies in SLIs, an objective performance measure is needed that combines task-based success measures (e.g., information elements that are correctly obtained) and a variety of dialogue-based cost measures (e.g., number of error correction turns, time to task completion, etc.) (Hirschman & Pao, 1993). A general framework for evaluation of dialogue systems across tasks is essential in order to determine optimal strategies for successful dialogue systems (Walker et al., 1997b; Cole et al., 1996). Using such a framework, different instantiations of dialogue strategies and dialogue systems can be compared and methods for automatic selection of optimal dialogue strategies can be developed.

In this paper we have focused on voice only interfaces, but the user interface technologies can also be applied to mixed mode interfaces. One type of mixed mode interface that is currently generating considerable interest is that of a lifelike computer character in which spoken language understanding and TTS are integrated with models of a talking head, thereby providing an animated agent that provides consistent and synergistic auditory and visual cues as to the sounds being spoken. Such systems are starting to appear as the face and voice of email and voice mail readers on PCs, but these capabilities can be used in any SLI application. Another research area that is only beginning to be explored involves understanding how to use speech optimally in dialog systems when multiple input modes (e.g., speech, keyboard, mouse) and multiple output modes (e.g., audio, display) are available (Allen et al., 1996; Smith, Hipp & Bierman 1992; Cohen & Oviatt, 1994).

7. Summary

In this paper, we have argued that spoken language interfaces will play an ever increasing role in the telecommunications systems of the twenty-first century as they provide the required natural voice interfaces to the enhanced and novel services that will become the framework of the network of the future. The opportunities and challenges for building such interfaces are essentially unlimited. The resultant interfaces will provide ubiquitous access to and support for richer human-computer and human-human communicative interactions.

REFERENCES.

A. Abella, M. K. Brown and B. Buntschuh (1996), "Development Principles for Dialog-Based Interfaces", *ECAI-96 Spoken Dialog Processing Workshop*, Budapest, Hungary.

J. F. Allen, B. W. Miller, E. K. Rinnger and T. Sikorski (1996), "A Robust System for Natural Spoken Dialogue", *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, pp. 62-70.

H. Alshawi (1996). "Head Automata and Bilingual Tiling: Translation with Minimal Representations", *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics*, Santa Cruz, Ca., pp. 167-176.

A. Baddeley (1986), **Working Memory** (Oxford University Press, Oxford).

S. Bennacef, L. Devillers, S. Rosset and L. Lamel (1996), "Dialog in the RAILTEL Telephone-Based System", *Proceedings ISSD '96*, pp. 173-176.

D. Bigorgne, O. Boeffard, B. Cherbonnel, F. Emerard, D. Larreur, J. L. Le Saint-Milon, I. Metayer, C. Sorin, and S. White (1993). Multilingual PSOLA text-to-speech system. *Proc. ICASSP'93*, pp. 187-190.

R. Billi, G. Castagneri and M. Danieli (1996), "Field Trial Evaluations of Two Different Information Inquiry Systems", *Proceedings IVTTA 96*, pp. 129-135.

S. Boyce and A. L. Gorin (1996), "User Interface Issues for Natural Spoken Dialogue Systems", *Proceedings of International Symposium on Spoken Dialogue, ISSD*, pp. 65-68.

E. Bryan Carne (1995), **Telecommunications Primer: Signals, Building Blocks and Networks** (Prentice-Hall, New York).

T. Bub and J. Schwinn (1996) "VERBMOBIL: The Evolution of a Complex Large Speech-to-Speech Translation System", *Proceedings of the International Conference on Spoken Language Processing, Philadelphia, PA.*, pp. 2371-2374.

N. Campbell and A. W. Black (1997), "Prosody and the Selection of Source Units for Concatenative Synthesis". In J. Van Santen, R. W. Sproat, J. P. Olive, and J. Hirschberg (eds**.), Progress in Speech Synthesis** (Springer-Verlag, New York), pp. 279-292.

P. R. Cohen and S. L. Oviatt (1994), "The role of voice in human-machine communication". In David B. Roe and J. Wilpon (eds.), **Voice Communication between Humans and Machines** (National Academy of Sciences Press, Washington, DC), pp. 34-75.

R. A. Cole, J. Mariani, H. Uszkoreit, A. Zaenen, and V. Zue (eds.) (1996), **Survey of the State of the Art in Human Language Technology.** (http://www.cse.ogi.edu/CSLU/HLTsurvey/HLTsurvey.html).

M. Danieli, W. Eckert, N. Fraser, N. Gilbert, M. Guyomard , P. Heisterkamp, M. Kharoune, J. Magadur, S. McGlashan, D. Sadek, J. Siroux and N. Youd (1992), "Dialogue Manager Design Evaluation", Technical Report Project Esprit 2218 SUNDIAL, WP6000-D3.

M. Danieli and E. Gerbino (1995), "Metrics for Evaluating Dialogue Strategies in a Spoken Language System", *Proceedings of the 1995 AAAI Spring Symposium on Empirical Methods in Discourse Interpretation and Generation*, pp. 34-39.

A. Gorin, G. Riccardi and J. Wright (1997), "How may I help you?", to appear in *Speech Communication*.

J. Hirschberg and C. Nakatani (1996), "A prosodic analysis of discourse segments in direction-giving monologues", *34th Annual Meeting of the Association for Computational Linguistics*, pp. 286-293.

J. B. Hirschberg (1993), "Pitch Accent in Context: predicting intonational prominence from text", *Artificial Intelligence Journal*, Vol. 63, pp. 305-340.

L. Hirschman, M. Bates, D. Dahl, W. Fisher, J. Garofolo, D. Pallett, K. Hunicke-Smith, P. Price, A. Rudnicky and E. Tzoukermann (1993), "Multi-Site Data Collection and Evaluation in Spoken Language Understanding", *Proceedings of the Human Language Technology Workshop*, pp. 19-24.

L. Hirschman and C. Pao (1993), "The Cost of Errors in a Spoken Language System", *Proceedings of the Third European Conference on Speech Communication and Technology*, pp. 1419-1422.

J. D. Johnston and K. Brandenburg (1991), “Wideband Coding-Perceptual Considerations for Speech and Music” in **Advances in Speech Signal Processing** (Marcel Dekker, New York), S. Furui and M. M. Sondhi (eds.), pp.109-140.

C. A. Kamm (1994), "User Interfaces for Voice Applications", **in Voice Communication between Humans and Machines** (National Academy of Sciences Press, Washington, D.C.), D. Roe and J. Wilpon eds., pp. 422-442.

C. A. Kamm, S. Narayanan, D. Dutton and R. Ritenour (1997), "Evaluating Spoken Dialog Systems for Telecommunications Services", *Proceedings of EUROSPEECH 1997*, pp. 2203-2206.

W. B. Kleijn and K. K. Paliwal (1995), “An Introduction to Speech Coding”, in **Speech Coding and Synthesis** (Elsevier, New York), W. B. Kleijn and K. K. Paliwal (eds.), pp. 1-47.

R. M. Krauss and P.D. Bricker (1967), “Effects of transmission delay and access delay on the efficiency of verbal communication", *J. Acoustical Society of America*, Vol. 41.2.

E. Levin and R. Pieraccini (1995), "CHRONUS, The Next Generation", *Proceedings of 1995 ARPA Spoken Language Systems Technology Workshop, Austin Texas*.

M. Marx and C. Schmandt (1996), "MailCall: Message Presentation and Navigation in a Nonvisual Environment", *Proceedings of the Conference on Human Computer Interaction, CHI96*.

G. A. Miller (1956), “The magical number seven, plus or minus two: Some limits on our capacity for processing information", *Psychological Review*, Vol. 3, pp. 81-97.

H. Meng and S. Busayapongchai and J. Glass and D. Goddeau and L. Hetherington and E. Hurley and C. Pao and J. Polifroni and S. Seneff and V. Zue (1996), "Wheels: A conversational system in the automobile classifieds domain", *Proceedings of the 1996 International Symposium on Spoken Dialogue*, pp. 165-168.

R. J. Perdue and J. B. Scherer (1996), “The Way We Were: Speech Technology, Platforms and Applications in the `Old' ATT”. *Proceedings IVTTA 96*, pp. 7-11.

L. R. Rabiner, B. H. Juang, and C. H. Lee (1996), “An Overview of Automatic Speech Recognition”, in **Automatic Speech and Speaker Recognition, Advanced Topics** (Kluwer Academic Publishers, New York), C. H. Lee, F. K. Soong, and K. K. Paliwal (eds.), pp. 1-30.

L. R. Rabiner and B. H. Juang (1993), **Fundamentals of Speech Recognition** (Prentice-Hall, New York).

M. D. Sadek, A. Ferrieux, A. Cosannet, P. Bretier, F. Panaget and J. Simonin (1996), "Effective Human-Computer Cooperative Spoken Dialogue: The AGS Demonstrator", *Proceedings of the 1996 International Symposium on Spoken Dialogue*, pp. 169-173.

Y. Sagisaka, N. Kaiki, N. Iwahashi, and K. Mimura (1992) “ATR-v-TALK Speech Synthesis System”, *Proceedings of ICSLP 92*, vol. 1, pp. 483-486.

B. Shneiderman (1986), **Designing the User Interface: Strategies for Effective Human-Computer Interaction** (Addison Wesley, Menlo Park, California).

A. Simpson and N. A. Fraser (1993), "Black Box and Glass Box Evaluation of the SUNDIAL System", *Proceedings of the Third European Conference on Speech Communication and Technology*, pp.1423-1426.

R. Smith, D. R. Hipp, and A. W. Biermann (1992), "A Dialogue Control Algorithm and its performance", *Proceedings of the Third Conference on Applied Natural Language Processing*.

C. Sorin (1994), "Towards High-Quality Multilingual Text-to-Speech", in **Progress and Prospects of Speech Research and Technology** (Infix Publishing Company, Sankt Augustin), H. Nieman (ed.).

K. Sparck-Jones and J. R. Galliers (1996), **Evaluating Natural Language Processing Systems**, (Springer-Verlag, New York).

R. Sproat and J. Olive (1995), "An Approach to Text-to-Speech Synthesis", in **Speech Coding and Synthesis** (Elsevier, New York), W. B. Kleijn and K. K. Paliwal, (eds.), pp. 611-633.

E. Sumita and H. Iida (1995), "Heterogeneous Computing for Example-based Translation of Spoken Language*". Proceedings of TMI-95: the International Conference on Theoretical and Methodological Issues in Machine Translation*, pp. 273-286.

J. P. Van Santen, R. W. Sproat, J. P. Olive, and J. Hirschberg (eds.) (1996), **Progress in Speech Synthesis** (Springer-Verlag, New York).

M. A. Walker (1989), "Natural Language in a Desk-top Environment", *Proceedings of HCI89, 3rd International Conference on Human-Computer Interaction*, Boston, Mass, pp. 502-509.

M. A. Walker, D. Hindle, J. Fromer, G. Di Fabbrizio, and C. Mestel (1997a), "Evaluating Competing Agent Strategies for a Voice Email Agent", Proceedings of EUROSPEECH 1997, pp. 2219-2222.

M. A. Walker, D. Litman, C. Kamm, and A. Abella (1997b), "PARADISE: A Framework for Evaluating Spoken Dialog Agents", *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and 8th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 271-280.

M. A. Walker and S. J. Whittaker (1989), “When Natural Language is Better than Menus: A Field Study”, Hewlett Packard Laboratories Technical Report HPL-BRC-TR-89-020.

S. Whittaker and P. Stenton (1989), "User Studies and the Design of Natural Language Systems", *Proceedings of the European Association of Computational Linguistics Conference EACL89*, pp. 116-123.

S. Whittaker and M. Walker (1991), "Towards a theory of multimodal interaction", *Proceedings of the AAAI Workshop on Multimodal Interaction*.

J. Wisowaty (1995), "Continuous Speech Interface for a Movie Locator Service", *Proceedings of the Human Factors and Ergonomics Society*.

N. Yankelovich, G. Levow, and M. Marx (1995), “Designing Speech Acts: Issues in Speech User Interfaces”, *Conference on Human Factors in Computing Systems CHI95*.