# Serving Customers With Automatic Speech Recognition—Human-Factors Issues

**Blake L. Wattenbarger**
**Roger B. Garberg**
**Edward S. Halpern**
**Barry L. Lively**

This paper explores the human-factors issues in designing human/machine interfaces employing automatic speech recognition (ASR) technology. It describes how a designer of a product or service who uses this technology must ensure successful interactions between computers and end-users, leading to customer satisfaction and success of the product or service. The paper discusses how ASR technology can support human/machine interactions in telecommunications applications, if the human-factors issues that arise in ASR user-interface (UI) design can be appropriately addressed.

## Introduction

HAL, the computer in the movie *2001—A Space Odyssey*, could speak clearly in a voice that sounded human. It could understand human speech—not just the individual words, but also their meanings. And it could easily recognize the voice of the person with whom it was speaking.

When HAL spoke and listened, the human with whom it was conversing did not need to compensate for the fact that HAL was a computer. HAL's abilities as a speech processor are only distant dreams for us today, but pale outlines of these dreams have already reached a state of practical application for some situations.

## Reasons for Using ASR

Why use ASR? An obvious answer is that it holds the promise of UIs that are more natural and easy to use. Today's interactive telecommunications services, such as voice messaging, require users to translate their wishes into arbitrary strings of numbers that are entered on a touch-tone telephone keypad. However, many people would prefer to tell an operator or attendant what they want, so that there is no need to learn which buttons to press or how to navigate through a complex hierarchical series of menus. ASR seems to offer a degree of naturalness that is more like talking to an actual operator.

Other factors merit consideration of ASR technology. Interactive network services are unavailable to customers who use pulse-dialing telephones. Estimates vary and accurate numbers are difficult to obtain, but current telephone industry research shows that between 25 and 40 percent of all U. S. households are now using pulse-dialing telephones, either rotary-dial or push-button. If service providers are to meet these customers' needs, the only alternative to ASR is the employment of full-time attendants.

Replacing attendants or operators with speech-processing equipment offers the possibility of substantial cost savings. In some existing applications, such as operator services, the potential savings are enormous. In new applications, a service might be economically viable only if it can be automated to avoid the high cost of attendants.

In some applications, safety is the main concern, as in dialing a cellular telephone while driving. Whenever hands and eyes are busy with other tasks, speech may be the only practical way for a user to provide information to the system.

## Current ASR Technology

To understand UI design issues, it is first necessary to grasp some of the characteristics and limitations of today's ASR systems. These systems must be engineered for specific applications. Many applications do not require a system with the sophisticated recognition capabilities of a human being. A system that can make consistent discriminations between spoken words in a set can provide users with an effective tool for

**Panel 1. Abbreviations, Acronyms, and Terms**

ASCII—American standard code for information interchange

ASR—automatic speech recognition

ASR models—words entered into a system vocabulary with which an end-user's utterances are compared and matched

barge-in—permits speaking during a prompt, eliminating the need for a user to listen to an entire message before responding

concept spotting—allows users to respond to prompts with alternate words, phrases, or sentences expressing their intent

early decision—ability of an ASR system to respond immediately to a user's utterance

motherese—a form of speech used by adults when speaking to children that is characterized by simplified vocabulary and changes in pitch, loudness, and sentence length

natural-language processing—a technology that allows ASR machines to "understand" complete thoughts

non-keyword rejection—ASR capability that prevents out-of-vocabulary speech or noise from being recognized as a legitimate word

phoneme—one of the set of the smallest units of speech that distinguish one utterance or word from another

PIN—personal identification number

platform—location of an ASR system, either in a network or in a user's terminal equipment

prosody—elements of meaning that are conveyed by features, such as stress and pitch, that span boundaries of sound segments. For example, in the word "window", the sound segments comprise /w/I/n/d/o. The prosodic feature of stress is present across the first three segments.

recognizer—ASR subsystem that matches a user's utterances with stored models

SDL—specifications and description graphical computer language

speaker-dependent—an ASR system having word models formed by an individual user

speaker-independent—an ASR system having word models formed in advance by system designers

SV—speaker verification

"training"—ASR system storage of utterances used as word models

TTS—text-to-speech, usually used to precede the word "synthesis"

UI—user interface

VIP—voice-interactive phone service

*Wizard of Oz* technique—simulation of the action of an automated device using a human agent, in such a way that subjects believe they are interacting with a machine, not a human

word spotting—a technique that improves recognition accuracy by ignoring extraneous words or sounds in an utterance that would otherwise impair or prevent recognition

completing specific tasks. ASR provides that capability for an increasing number of applications.

**Comparisons of ASR Technology.** An ASR device operates by comparing a user's utterance with models of words or phrases held in memory. There is virtually never an exact match between an utterance and any of the models. A match decision is based on how far a user's utterance is from various active models. Recognizers that simply pick the closest alternative model generally do not perform as well as those that require one model to be clearly superior in the comparisons. The former method leads to correct and incorrect recognitions, while the latter—called *non-keyword rejection*—allows the recognizer to report that it does not have enough information for a clear decision. ("Please repeat that.") Non-keyword rejection substantially reduces the number of incorrect recognitions, which is often more important to users than the simple probability of a correct recognition.

The ASR models with which a user's utterances are compared come into existence in one of two ways. They can be formed by the individual user, thus creating a *speaker-dependent system*. Or, they can be formed in advance by usability engineers, creating a *speaker-independent system*.

During "training" of a speaker-dependent recognizer, a typical system prompts the user to speak a

command word at least twice. The recognizer averages its coding of these utterances and stores the resulting model.

Two potential problems arise at this point. One problem concerns the consistency of utterances used in "training." Suppose one utterance was "aahhhhh, Harry," and the other was "Harry." Averaging the coding of these two utterances could well produce a model that would never be matched. An effective method for solving this problem is to match the two utterances at the time of "training". If they differ too widely, prompting for utterances is continued until a sufficiently similar pair is obtained.

The second problem shows up when one model is so similar to another already in memory that it is unlikely that the system can discriminate between them ("Murray" and "Mary"). The system can spot this similarity and prompt a user to change one of the names. It is not desirable to prompt for changing only the last model developed, because the system does not know which one has more "synonyms." These problems do not arise with a speaker-independent system, because it is "trained" as part of product or service development before a user is exposed to it.

The training process is one that holds no intrinsic value for a user. It is analogous to teaching each new acquaintance how to understand one's speech. It is a challenge for UI engineers to design a training session to be as quick and innocuous as possible.

Designers provide a speaker-independent system with speech samples for a particular application. To build a composite model of a single command word, a speaker-independent system will require numerous samples of each command word from all dialects throughout the country. A significant sample size is required because of the large variation in the acoustic properties of speech produced by different speakers, even when they say the same word. The process of obtaining these samples—and ensuring that an adequate cross-section of users by age, gender, and region is represented—is costly and time consuming. However, once a model is completed, any new services may use it. Thus, many common words are currently available for use in speaker-independent services, words such as the numbers "zero" through "nine", "oh", "yes", and "no."

When designing a UI for a new speaker-independent service, identification of the necessary vocabulary set is a critical decision, one that has substantial cost implications if new word models must be constructed. If new words are needed, they should be carefully selected so that they are meaningful to a user's task and acoustically distinguishable by the recognizer. Balancing the recognizer's need for easily distinguished words with a user's requirement for easily remembered commands is a critically important UI design goal.

A speaker-independent ASR system allows access to the service by anyone with a telephone, including those with pulse-dialing phones. In addition, speaker-independent ASR can be used even when customers are not subscribers, such as in automated operator applications. Because speech models are developed with samples across all environmental conditions (for example, different microphones, line noise, background noise, etc.), these models may be more robust in services where calls originate across all environmental conditions. Finally, while speech-model development is laborious and costly, it is transparent to users who need not "train" the system for their own voice.

Speaker-dependent ASR has an advantage in applications that require very large vocabularies (speech dictation systems), that are idiosyncratic to users (name-dialing telephones), or that have a highly specialized vocabulary (speech control of an industrial process). Because speaker-independent systems can require a lengthy period of data collection and model building, some applications employ speaker-dependent technology to meet a market window. One example comes from international applications that, for marketing reasons, needs to be deployed rapidly. There is no reason in principle why a speaker-independent system could not be designed to handle multiple languages, but there may not be enough time to build command-word models for many countries. Valuable marketing opportunities may be lost when product development takes too long.

**Platforms.** The ASR system can be located either in the network or in a user's terminal equipment. These two locations have very different advantages and limitations. In the network, many users can share ASR hardware and software. Sharing spreads costs and permits use of more expensive technology than would otherwise be acceptable in terminal equipment. On the other hand, the network device must accommodate the network's limited bandwidth that filters out much of the information contained in users' utterances. Network devices must also accommodate carbon-button transmitters,

**Panel 2. Other Supporting Speech Technologies**

ASR is only part of the story of HAL's linguistic behavior. Other speech technologies, which HAL had mastered, are being developed today. While a meaningful ability to identify speakers by the sound of their voices is still largely beyond reach, the capability to verify a person's claimed identity with a speech sample (speaker verification) is currently being developed. And programs that translate stored text into articulate speech are now available.

**Speaker Verification (SV).** This technique requires the system to verify that the speaker is the person claimed, using a speech sample in a manner analogous to a fingerprint. Telecommunications fraud is a billion-dollar-per-year problem that might be solved—or at least greatly reduced—by using SV technology.

From a user's point of view, substitution of a simple spoken password in place of an easily forgotten personal identification number (PIN) is very appealing. SV requires a user to "train" a system to "understand" human voice characteristics by providing a number of speech samples.

Speaker consistency is very important. If a user speaks with the same tone and pace for each training sample, as well as later during actual use, SV accuracy will be high. One major challenge for human-factors engineers is to find ways to assure such consistency. Inevitably, the verifier will reject certain words and phrases. Some of these rejections will be legitimate; others will be incorrect, rejecting valid users for some unknown reason. The human-factors challenge is to achieve an optimal tradeoff between rejection of impostors and acceptance of valid users.

**Speech Synthesis.** In the movie, HAL appeared able to think, formulate new thoughts, convert thoughts into natural language, and produce speech as output. Today, thoughts are stored in computers as text and coded as ASCII (American standard code for information interchange) characters. Text-to-speech (TTS) synthesis is arguably the most advanced of the three technologies discussed in this paper. Conceptually, TTS synthesis is simply a process that takes text as input and produces speech as output. However, TTS synthesis is not as simple as it appears.

There are many pairs of words that are spelled the same way but have different pronunciations and meanings (for example, "contract", "record", "lead", and "object" used as verbs or as nouns, and "read" used as a present- or past-tense verb). Also, orthographic strings often have pronunciations that depend on context (for example, note the diverse pronunciations associated with "gh" in "ghost", "tough", and "through"). In addition, meaning is often determined by the *prosody* of speech. Nevertheless, the use of linguistic analysis makes it possible to solve these problems, and usable—if imperfect—TTS-synthesis systems are now available.

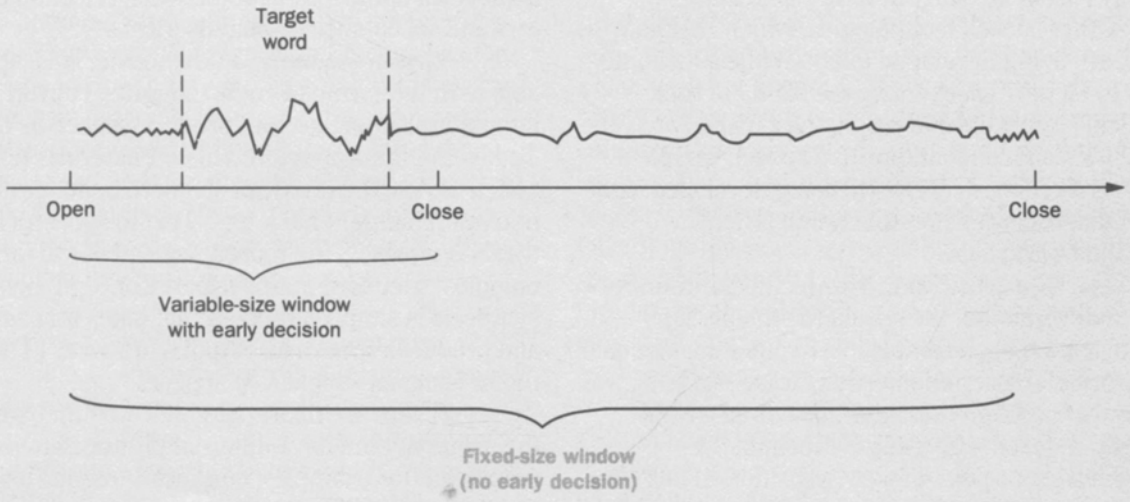
Some TTS systems have undergone field trials[19,20], and some are now available in commercial products such as the AT&T AYC12 text-to-speech synthesis board used in the AINet service circuit node and Conversant® voice information systems.

used in some older telephones, which significantly distort the signal. These problems can be avoided if the device is located in the terminal equipment.

**Applications.** ASR telephone network applications can serve a variety of different markets. These markets may be identified by network providers interested in cutting the cost of their current services, or other vendors wishing to offer new products.

Over the last several years, a number of speaker-independent services has been made available, either in field trials or as actual service offerings. Several corporations—including AT&T, BBN Systems & Technologies, Bell Communications Research, Northern Telecom, and the regional Bell operating companies—have been involved. Versions of an automated operator have been offered by Ameritech Services, AT&T, and New Jersey Bell.[1] Northern Telecom offered a stock-exchange quotation service that could be accessed by saying the name of the company about which information was sought. BBN Systems & Technologies has offered a voice-dialing system, allowing callers to say the name of the person they want to call. And MetroCel Cellular has offered a voice-dialing system over the cellular network.

AT&T partnered with US West to trial voice-interactive phone service (VIP), allowing customers to say the name of the custom-calling feature they wished to activate instead of entering a two- or three-digit access




**Panel 3. New Capabilities**

Several new developments have made great improvements to ASR service capabilities and to the UI. ASR technology has only recently begun to fulfill the promise of natural, conversational interactions.

**Early Decision.** When ASR systems were first being developed, speech recognition occurred during fixed-duration "windows." For example, if a six-second window was used and a person spoke only during the first second, there would be a five-second pause before the system would respond. Early decision is the ability of a system to close the recognition window following the end of a user's utterance and to respond immediately (see figure above).

Although a time delay of only a few seconds seems insignificant, in reality the unnatural pause often prompts inexperienced users to repeat their utterance, thinking they have been ignored or not heard, just as they would when speaking to another person. Experienced ASR users know they have to wait for the system to catch up, but are often frustrated by the delayed responses. Early-decision technology permits a much smoother, more natural human/machine interaction.

**Non-keyword Rejection.** The first speech recognition algorithms simply compared an input utterance with stored models or templates of vocabulary words and produced the best match as output. This resulted in out-of-vocabulary speech, or even noise, being "recognized" as a word it did not even resemble. More recent algorithms can determine when a match is so poor that the input should be rejected. This capability is known as *non-keyword rejection*. It permits dialogue between a user and machine to be more natural and human-like. The system can ask for a repeat, or for new input if the original utterance is unintelligible.

code. In collaboration with BellSouth Enterprises, AT&T collected speech models over the cellular network for the southern region of the country and completed a trial using voice-digit dialing. This allowed customers to say the seven-digit number they wanted to call.

Speaker-dependent recognizers also find a wide range of applications, and some of these were discussed earlier. Another application of this technology is in voice-controlled programming, which simplifies the setup process for video cassette recorders.[2] Another example is voice dialing of cellular telephones.[3] Without speech-recognition capability, a cellular telephone user usually steers with the left hand while dialing manually with the right thumb, holding the handset (which has

the dial pad located on the back) up to eye level. Callers are often pushed to the limits of their ability to maintain vehicle control in this situation. With a speaker-dependent system, a user lifts the handset, speaks a name, and the connection is automatically completed. Attention to safe driving has not been diverted.

**Customer Reactions to ASR.** Customers respond positively to new service offerings that use ASR. In a residential trial of VIP, participants were selected because they subscribed to two or more custom calling features. About 66% of trial participants said they would either probably or definitely use the service if it were put into place permanently.

Focus-group results revealed the expected ASR accuracy rate (correct procedure being executed on the first prompt) to be about 90 to 95 percent for a viable service offering. Of those participants who said they would use the service, 96 percent felt comfortable speaking to a computer, and 85 percent of those who claimed they wouldn't use the service still felt comfortable with it. In addition, 75 percent of respondents preferred speech input to touch-tone input, while only 16 percent preferred touch-tone input.

The success of cellular applications depends on task demands. Voice name-dialing over a handset—with the recognizer located in the car—was well received. There was little to interfere with speech recognition other than road rumble and noise contributed by the car. These noises are essentially continuous, and although often in excess of 70 dB, the recognizer can adapt to them. The recognizer tends to mask much of the impulse noise that might otherwise interfere with performance. Speech recognition in a car was good, and having the speech-recognition capability available kept to a minimum any additional cognitive load imposed by dialing while driving. That is, a user did not have to think much about what needed to be done to complete a call.

For example, in a road test with experienced users, voice name-dialing was judged to be as easy as reading one of the large green-and-white information signs commonly found along interstate highways.[3] In contrast, manual dialing was judged even more difficult than tuning in a car radio without using the preprogrammed selection buttons.

Another cellular trial of voice-digit dialing technology was conducted using a car-installed speakerphone and a network-located recognizer. While users liked the concept of hands-free dialing, the noise that is presently an inescapable part of the cellular network interfered with recognition and substantially degraded performance.

In market trials of an AT&T version of automated operators, the majority of users rated the service excellent or good. Users included customers making "zero-plus" (0+) telephone calls in a specific region. The reason given most often for liking the service was that users could place and complete calls quickly, eliminating the formality of talking to an operator. The reasons given most often for disliking the service were that many users preferred talking to another person, rather than a machine, and that the service was too slow.

To summarize the results: users are willing to talk to a machine if doing so provides a way to meet a need. As with any telephone service, users find products or services most appealing when they help solve a particular problem.

## Challenges to Current Technology

ASR is far from equaling HAL's performance. However, intensive development efforts have led to greater flexibility in ASR technology (see Panel 3). In addition, some of the remaining obstacles to HAL-like competence can be overcome, at least in part, by special efforts in designing a UI.

**Accommodating Inaccuracies.** Human speech is normally directed to other persons, who bring a certain level of recognition accuracy—plus an array of strategies—for correcting misunderstandings and clarifying doubts. Human speech is highly redundant[4,5,6], and listeners can understand many words from the context of an utterance, even if some words are omitted or garbled.

Humans can employ many error-correcting strategies, such as echoing or asking about parts of a speaker's utterance that are unclear, if the inference from context is uncertain. Recognizers, however, presently rely mainly on simple algorithms that make little or no use of redundancy or of human error-correcting strategies in conversation.

ASR algorithms do not "understand" speech as human listeners do. They are not able to make intelligent guesses about garbled or missing information—beyond offering second or third guesses based on a match between system input and active models or templates. Therefore, a major challenge is to design ASR systems to

**Panel 4. Anticipated Technical Advances in ASR Technology**

In addition to making improvements to ASR technology, anticipated technical advances focus on making ASR services seem more natural or conversational. Ideally, this would require a system to recognize streams of continuous speech containing many words. The goal of ASR research is to allow speakers to use conversational speech without restrictions on speed or vocabulary.[21] Achieving this goal demands, among other advances, enlarged ASR vocabularies.

**Concept Spotting.** Each transaction or human/machine interaction involves recognition of a single word or short phrase from a menu. There is a one-to-one relation between menu items and target words. With concept-spotting capability, users can respond to prompts with alternate words, one or more phrases, or sentences expressing their intent.

Using subword technology, along with enlarged vocabularies and word spotting, recognition of multiple words representing the same concept is possible. Ultimately, it will be necessary to use the techniques of *natural-language processing*, an emerging technology, for machines to "understand" complete thoughts.

**Enlarged Vocabularies.** Vocabulary size is presently restricted by the limited accuracy of recognition algorithms—which leads to substitution errors as vocabulary size grows—and by delays resulting from necessary processing. Better algorithms and faster computers will inevitably increase the practical vocabulary size that recognition systems can handle.

As vocabulary size grows, users will be the first to benefit. Applications will be able to accept common synonyms for keywords. With even larger vocabularies, it will be possible to design much more user-friendly interfaces that avoid complex hierarchical menu trees.

**Subword-based ASR.** To avoid the laborious process of collecting speech data to build models for specific words, engineers are now developing models based on smaller speech units such as syllables and phonemes. These subword unit models will make it possible to create new vocabularies quickly, and will allow the trialing of new ASR services months earlier than currently feasible. Subword-based ASR will also facilitate development of custom vocabularies for individual network-services subscribers without the need for speaker-dependent ASR system "training."

**Connected Words/Digits/Natural Numbers.** Humans can distinguish words as separate entities even though, on the basis of objective physical analysis, they may appear to be inextricable.[22] Continued improvements to connected-digit recognition will open up the ASR applications market. Recognition of digit strings allows development of new services requiring users to key in telephone numbers, calling-card or credit-card numbers, and catalog orders.

accommodate less-than-perfect accuracy in ways users will find smooth, natural, and convenient.

**Application Environments.** Another challenge to ASR deployment is variability in human behavior. Users can obtain excellent performance in a controlled laboratory setting by knowing just what they may and may not do. But performance in the real world may be far less favorable.[7] In actual applications, people often don't know exactly what to say and may be distracted by other tasks. Their expectation is of a system that will adapt to their needs, not vice versa.

Another issue that arises in many applications is the problem of noise. A study was completed in which telephone conversations with service representatives were catalogued.[1] The study reported that over 40 percent of calls contained noticeable background noise that was unrelated to telephone line quality. Examples include talking in the background, television sounds or music, and traffic noise. Transmission noises include static, pops, and clicks.

In a recent AT&T study, noise from radios, TVs, and conversations unrelated to the automated interaction was present in almost nine percent of calls, while line noise was present in over six percent of calls. Another seven and one-half percent of calls contained both types of noise. Because most users do not understand how ASR technology works, it may fail for reasons unknown to them in such noisy conditions.

Noise is particularly problematical in the cellular environment. In addition to the sources already mentioned, there are additional noise sources, such as cell fading, wind, air conditioning fans, and honking horns.

Cell fading is peculiar to systems that have network-situated recognizers. The other sources of noise are problems for any recognizer, whether in a car or a network. But, as discussed earlier, the effects of impulse noise arising inside or outside a car are somewhat less than they might otherwise be due to masking by wind and road noise.

### Optimizing a UI to ASR Applications

As previously noted, current ASR technology is favorably received by most users. They are willing to use an ASR-based service if it provides the means to complete tasks accurately, efficiently, and in a satisfying way. In other words, most users are willing to talk to machines, but they expect the machines to respond intelligently. There is a risk that users will reject a new ASR application if it fails to understand spoken instructions.

This risk is reduced by anticipating and leading speech to decrease the chances of users saying unexpected words, and by identifying the appropriate responses to expected words. Whether the use of ASR technology involves single-word or concept-spotting approaches to dialog design (see Panel 4), much of the risk associated with deploying individual ASR applications involves uncertainty about the range of speech evoked by the new application.

**ASR Influence on Speech Patterns.** Research results suggest that people are sensitive to the linguistic capabilities of their listeners, and that they are willing and able to adapt their speech to listeners (including machines) with limited language competence.

When adults speak to children, they use a form of speech that they believe to be appropriate. This speech, called *motherese*, involves diverse alterations in speaking, ranging from simplifications in vocabulary to changes in prosodic characteristics, such as pitch and loudness.[8] Adults speaking to children shorten the average length of sentences and less often substitute pronouns for nouns.

There has been considerable debate about whether motherese is necessary, harmful, or just irrelevant to language development among children. But it is relatively uncontested that adults overwhelmingly alter their speech when talking to children. Motherese has been observed in every language that has been examined by linguists interested in language acquisition.[9]

Do adults make adjustments in their speech when talking to machines? Evidence suggests they do. In one study, users speaking to an information service over the telephone were observed.[10] The study determined that many aspects of verbal behavior were affected by whether users thought they were talking to another person or to a machine. Another study reported that adults who interacted with an automated operator service shortened the length of their spoken requests, as compared with the length of requests made to operators.[11] Researchers observed this effect regardless of whether or not the initial prompt indicated that a caller had reached an automated service. Interestingly, studies also showed that children's requests were of equal length whether speaking to an automated service or to an operator. It is possible that most children have not learned the linguistic skill of adapting their speech to minimally-competent listeners.

The results of this research must be reflected in UI design for ASR systems. A designer cannot expect to encounter the same speech patterns in talk directed to an automated system and an operator or attendant.

**Improving Naturalness.** Much UI design work is aimed at letting users know what words and phrases are recognized at any given point in a dialogue. Consequently, the typical approach up to now has been for the ASR system to take charge of interaction with a user, leading to an unnatural interface that can give the impression of a cold and unfriendly system. For example, contrast what an attendant might say:

*"Welcome to XYZ service. How may I help you?"*

with what an ASR system might say:

*"Welcome to XYZ service. Please say A, B, C, or D now."*

Until recently, a user had to avoid uttering extraneous words or sounds. Even a simple "uh" before the vocabulary word, or a polite "please" afterward, would impair or prevent recognition. For this same reason, a human/machine interface was designed to give explicit instructions about what to say.

In the last two years, a new technique called *word spotting* has largely overcome this problem. Users are allowed to say "Collect, please" or even "I'd like a collect call, please" with little or no decrease in recognition accuracy as compared with an isolated "collect." This permits a much friendlier interface style, and encourages

users to respond more naturally, as they normally would.

A new capability called *barge-in* permits a customer to speak during the prompting message. Without barge-in, the recognition system must be turned off during the prompting announcement to prevent the system from "hearing" the prompt itself as user input. With barge-in, users can interrupt the prompt, just as they might interrupt an operator. This capability has been available for many years for touch-tone input, and can greatly accelerate the interaction process. Barge-in allows the interaction between user and system to be more natural and conversational.

To take advantage of these capabilities, studies have been performed to test a more conversational style of human/machine voice communication. For example, instead of

*"Welcome to AT&T. Please say collect, calling card, third number, person-to-person, or operator, now."*

the prompt said

*"Welcome to AT&T. What type of call would you like to make? (pause) Please say collect, calling card, third number, person-to-person, or operator, now."*

The researchers anticipated that the question and pause inserted in the original prompt would accomplish three objectives:

- As permitted by barge-in, experienced users would be encouraged by the conversational pause to complete the transaction without waiting to hear the entire list of options repeated.
- Due to unfamiliarity with the system, novice users would wait long enough to hear the entire list of options.
- All users would find the question helpful in clearly understanding what to do next, reducing the number of "freezes"—customers who hear the prompt but say nothing.

Results of the laboratory research reinforced original predictions. On average, customers can complete transactions more quickly than with the shorter prompt because they are more likely to barge in. More calls were automated, fewer customers called an operator for assistance, and the system was more highly rated with the question-pause format. In a following field study using live traffic, time spent on the transaction was found to be shorter, and fewer customers called for an operator when prompted with the question-pause format than with a prompt that did not begin with a question.

In addition to the menu transactions just discussed, the same series of laboratory studies examined the simpler yes/no transaction, in which a yes or no response is requested by the system. In this transaction, the same approach was tried. Instead of

*"You have a collect call from (caller's name). Please say yes if you accept the charges, no if you refuse the charges, or operator if you need assistance, now."*

the system said

*"You have a collect call from (caller's name). Will you accept the charges?" (pause) "Please say yes, no, or operator, now."*

The results, paralleling those discussed earlier, reveal an advantage to a simple question, followed by a pause, and then the available options. AT&T Bell Laboratories is planning additional field studies at selected sites.

Researchers are currently studying other types of transactions. The most significant of these are transactions that involve collecting strings of connected digits from users. While recognition accuracy is quite high for a single digit, a string of 10 connected digits, as in a telephone number, is less accurately recognized.

A system with a one-percent chance of error on individual digits will make some error on 10 percent of 10-digit telephone numbers. In addition, users of deployed ASR systems often make false starts, inject pauses or extraneous speech, or simply misspeak (for example, reverse adjacent digits). Researchers are solving this problem through work on both ASR technology and UI design. In examining the UI design problem, the human-factors focus must be on improving accuracy while also maintaining good user acceptance and task efficiency.

**Feedback and Error Impact.** A case of using ASR for connected digits illustrates particularly well the importance of designing for error.[12] This approach involves taking steps to minimize error frequency, to maximize likelihood of error detection, and to ensure that recovery from error is natural and straightforward.[13]

Observing users keying in characters and numbers, researchers found that 70 percent of all errors were self-detected. Keyboard users knew, in most cases, that they made an error without any special system feedback, using instead their own self-monitoring techniques to confirm successful completion of an act. In contrast,

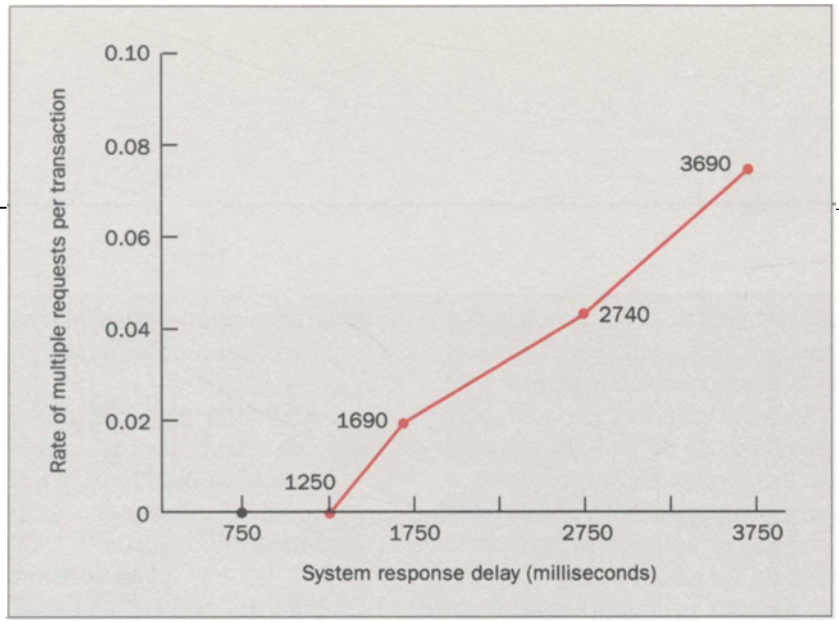



**Figure 1. Rate of multiple requests as a function of system response delay. Data are from trials in which no barge-in occurred.**

researchers noted that users of speech input were heavily reliant on externally provided system feedback.[13]

When feedback alerts users to a recognition error, the system must then permit easy error recovery. Even in normal conversation, people sometimes must ask for repetition of a word or phrase they did not understand. It is very important to ask for repetition quickly, clearly, and without offending the ASR user. Researchers have already begun studying the best ways to accomplish this.

In one of these studies, automated-operator re-promptings were classified and counted. Researchers found that, in most instances, the re-promptings followed an utterance containing the correct keyword, but that for some reason such as poor timing or background noise, the recognizer missed it. Researchers subsequently tried a shortened prompt. Instead of

*"Your response was not understood. Please say collect, calling card, third number, person to person, or operator, now."*

the system said

*"Sorry, please repeat." (pause) "Please say collect, calling card, third number, person to person, or operator, now."*

In laboratory tests, the command "Sorry, please repeat," followed by a pause, resulted in much faster system performance and better customer acceptance than did the original sentence, "Your response was not understood." As before, the pause induced many people to respond immediately, rather than wait for the system to repeat all the alternate responses.

These strategies seem to work because they effectively conform to the conversational style and pace to which people are accustomed. It is clear from watching participants in these studies that frustration and impatience build rapidly when someone knows what to do but must wait until the system is ready for them to proceed.

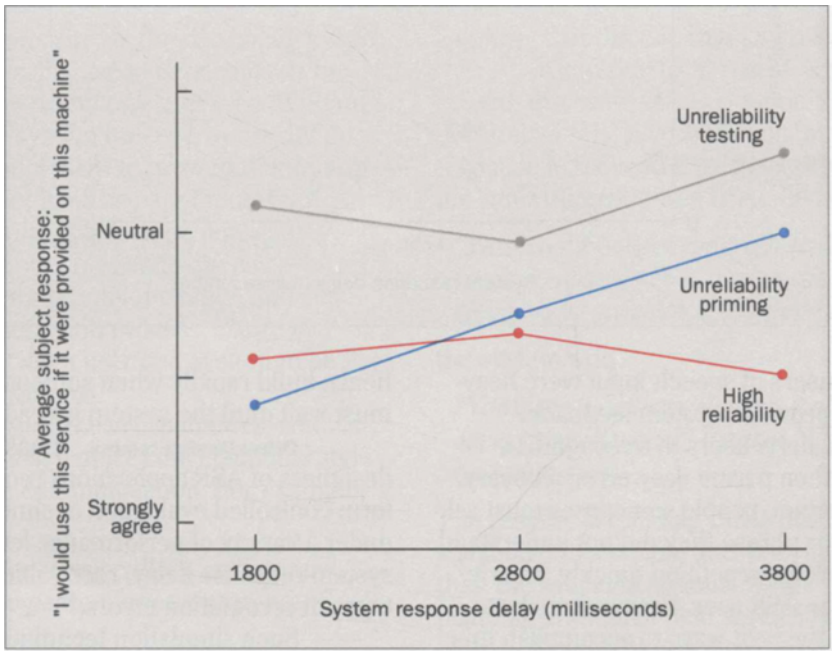
**Other Design Issues.** It has been argued that designers of ASR applications require the ability to perform controlled evaluation of simulated ASR dialogs under a variety of performance levels, including length of system-response delay, recognition accuracy level, and types of recognition errors.[14]

Such simulation techniques are suggested to be the primary means of addressing important issues. These issues include speed and accuracy requirements for various applications, criticality of errors by type, appropriate forms of error correction, the need for connected or continuous speech and speaker independence, the effects of large vocabulary size, and the human ability to constrain speech in terms of vocabulary, syntax, and speaking patterns.

Simulation experiments conducted by AT&T Bell Laboratories provided results concerning speed and accuracy requirements for a variety of network services using ASR. One topic addressed in these experiments is system-response delay—the time interval between the end of a user's utterance and the start of system feedback. In one experiment, subjects calling from their homes interacted with each of four different ASR services.

An ASR system provided separate response delays for each task, ranging from 0.75 seconds to 3.7 seconds. Results indicated that even small response-delay differences had noticeable effects on service responsiveness ratings. Response delays negatively affected overall service ratings when they were very short (<0.8 seconds) as well as when they were longer (>3 seconds). However, longer delays caused difficulty only when users waited until after the prompt to begin speaking. For users who barged in on prompts, service-response speed ratings remained unaffected.

**Figure 2. System response delay and reliability effects on users' ratings of their willingness to use an ASR service.**




AT&T Bell Laboratories researchers counted the number of times a system request was repeated: users repeat a request when system delay causes them to think they have not been heard. Figure 1 shows the results of this count as a function of response delay on trials with no user barge-in. Multiple requests only start to occur when delays exceed 1.25 seconds, and they increase steadily thereafter. In contrast, on barge-in trials, multiple requests are more frequent and unrelated to response delay. These results are consistent with users' expectations: a system should respond at about the same time another person would in normal conversation.

In a second experiment, users interacted with one of two ASR service simulations. Users were also randomly assigned to one of three ASR accuracy conditions. The distribution and frequency of preplanned system recognition failures differed for each condition. In one condition, users experienced recognition errors (in "priming" trials) before the experimental trials. Figure 2 summarizes the results of this experiment. Users who interacted with a highly reliable ASR system in experimental trials said they were willing to use the service regardless of response delay. Users who interacted with an unreliable ASR system in experimental trials were neutral in their willingness-to-use ratings, regardless of response delay. But for users who experienced errors only in priming trials, willingness to use the service was strongly dependent on a short response delay.

These results suggest that ASR algorithms may allow for as much as a doubling of response delays if the number of correct recognitions increases. Results also suggest that positive service evaluations depend on the interaction between feedback promptness and expected accuracy. When expected accuracy is low, users require rapid feedback. In addition, users who indicate dissatisfaction with response time may actually be expressing a lack of confidence in the application's reliability.

Other factors may subtly contribute to ASR service acceptance. For example, a user's age has a bearing on reception of new technology.[15] Researchers using an ASR-based data-retrieval task found that older users were slower to recover from system recognition errors.[16] Such an effect illustrates the importance of carefully considering a user group's demography.

### Designing for Users

Observations of a number of attempted ASR applications suggest that a range of factors affects service usability and acceptability.[14] Implied in the observations are several suggestions for developing successful

applications for telephony. The suggestions are:

- Planning for ASR incorporation should begin at project inception.
- A staged process of ASR development should be employed that includes regular user checks and tests.
- Speaker-dependent, ASR system training should be performed, when needed, in the context of an operational task—that is, under noise conditions comparable to ordinary environments.
- No disruption of a task sequence, either by long response delays or by recognition errors, should be permitted.
- Users should be assured of receiving appropriate recognition feedback.

UI design activity can be used at all stages of development. Before incurring the cost of developing an ASR service, designers can learn much about market acceptance and usability. Later, by setting up prototypes and running limited field trials, designers can obtain vital data about the service concept itself, as well as about ease of use.

Throughout the project, technical advances in ASR algorithms must be integrated into the UI design. Through iterative design and testing, design decisions focus on circumventing technical limitations and increasing a user's likelihood of success. In some cases, forward-looking efforts can help identify future directions the technology must take to support users, and how designers can anticipate future application opportunities.

**Case Study One.** In the case of the AT&T voice-response operator service, UI contributions have been made in each stage of the product development cycle:

- Early service evaluation
- Iterative design
- Live-traffic testing
- Improvements and refinement.

During the mid- to late-1980's, AT&T completed a number of voice-response operator service field trials. Thereafter, researchers concluded that the service concept was viable, but that system performance needed improvement. A series of laboratory studies, which examined various prompting strategies to optimize the UI, was then completed.[17] As the capabilities of word spotting, early decision, and barge-in became available, and as ASR accuracy improved, a new round of live-traffic field trials began in 1991. UI design experiments again focused on announcement wording, but the studies grew to include selection of an announcement voice and a call-flow structure that accommodated a fully deployed service. With the technical capability to discriminate between valid input, invalid input, and no input, it was possible to provide a user with context-specific feedback at any time, such as by a confirmation announcement, a re-prompt, or operator help.

Next, work on refining the UI included shortening prompts wherever possible, routing confused callers to an operator, and encouraging users to barge in—all to shorten call time. In addition, a demonstration system was developed for ongoing call-in studies. It assessed user behavior, problems, and preferences. Even though the service is now deployed, studies continue on how prompts affect user behavior and how response delays affect service-quality perception. As new features are discovered, each one is trialed in live-traffic situations and is evaluated by examining user successes and ratings. The new prompting strategy previously discussed (*"What type of call would you like to make? [pause] Please say..."*) is one example of this ongoing work.

**Case Study Two.** In the case of the AT&T and US West trial of voice-interactive phone service (VIP), considerable user-focused testing went into its development. VIP offers residential customers a speech interface into custom-calling features. VIP's introductory prompt characterizes the service:

*"Please say the name of the service you want or say help for a list of the services you subscribe to, now."*

The first step in developing VIP was market research. Feedback from participants indicated that the service could become a marketing success. A custom vocabulary was to be used, and the words, phrases, and synonyms were quickly determined. During data collection and model building, additional user tests and service design work were completed.

Use of a *Wizard of Oz* technique allowed the design team to collect data about an early version of VIP. This technique involves simulating the action of an automated device using a human agent, in such a way that subjects believe they are interacting with a machine, rather than with a human. In the VIP study, the *Wizard of Oz* technique involved presenting users with a simulated VIP service—one in which a human performed the speech recog-

nition function—and directing the system's responses to user input. Results of the study were used to complete design work and to finalize vocabulary content.

Next, a call-in demonstration was developed, allowing the design team to conduct focus groups using an actual version of VIP. Based on focus-group input, additional changes were made for a preliminary trial with volunteer users. The trial demonstrated that callers found VIP easy to use, and they supported marketability of VIP. A comprehensive, two-month field trial was then conducted.[18]

The call flow for VIP was written in the graphical specifications and description language (SDL), which compiles into C language. SDL facilitated the rapid design and testing of one version of VIP. Several human-factors studies were conducted to test alternative announcements. The studies also investigated ways of incorporating help messages into the service. Currently, US West is conducting a phase-two trial to assess whether the new announcements are improvements, and to test the usefulness of help messages.

In both the voice-response operator service and VIP cases, UI designers contributed their expertise to improve and optimize usability throughout development and field testing.

## Conclusion

HAL's speech abilities, which made it appear completely human, did not happen by accident. HAL seemed lifelike not just because its ASR algorithms worked well, but because all its speech behavior met the expectations of those who talked to it.

Unlike human speech skills, which seem to develop naturally with maturation, HAL had to be "taught"—programmed, actually—to use every aspect of its language. Many more years of work on the issues discussed in this paper will be required before machines will be capable of duplicating HAL's seemingly effortless and natural speech competence.

Nevertheless, today's ASR technology offers many capabilities that can be used to meet current needs. The frustration that some customers feel with a touch-tone based interface may best be overcome by using speech-based technology. But to achieve success, designers must plan from the beginning to use speech in the interface, and to engage the expertise of a UI designer at a project's inception.

## Acknowledgment

The authors wish to thank M. J. Cosky for his substantial support and work on this paper.

## References

1. D. Karis and K. M. Dobroth, "Automating Services with Speech Recognition Over the Public Switched Telephone Network: Human-Factors Considerations," *IEEE Journal on Selected Areas in Communications*, Sect. 9(4), May 1991, pp. 574–585.
2. H. Somerfield, "Technically Speaking; A Guide to the Brave New World of Electronic Gadgets," *San Francisco Chronicle*, Final Edition, Home Section, January 13, 1993, pp. 1/Z1.
3. E. C. Schwab et al., "Human-Factors Contributions to the Development of a Speech-Recognition Cellular Telephone," *Behavioral Aspects of Speech Technology*, Elsevier Publishing Company, Amsterdam, The Netherlands, 1993.
4. C. E. Shannon, "A Mathematical Theory of Communication," *Bell System Technical Journal*, Vol. 27, July 1948, pp. 379–423; October 1948, pp. 623–656.
5. C. E. Shannon, "Prediction and Entropy of Printed English," *Bell System Technical Journal*, Vol. 30, January 1951, pp. 50–64.
6. J. R. Pierce and J. E. Karlin, "Reading Rates and the Information Rate of a Human Channel," *Bell System Technical Journal*, Vol. 36, March 1957, pp. 497–516.
7. J. G. Wilpon et al., "Speech Recognition: From the Laboratory to the Real World," *AT&T Technical Journal*, Vol. 69, No. 5, 1990, pp. 14–24.
8. C. Snow and C. Ferguson, *Talking to Children*, Cambridge University Press, Cambridge, Massachusetts, 1977.
9. N. Smith, *The Twitter Machine*, Basil Blackwell Ltd., Oxford, United Kingdom, 1989, Chapter 13, pp. 142–155.
10. M. A. Richards and K. Underwood, "Talking to Machines: How Are People Naturally Inclined to Speak," *Contemporary Ergonomics*, 1984, pp. 62–67.
11. J. Rubin-Spitz and D. Yashchin, "Effects of Dialogue Design on Customer Responses in Automated Operator Services," *Proceedings of Speech Technology, 1989*, New York City, pp. 126–129.
12. C. Lewis and D. A. Norman, "Designing for Error," *User-Centered System Design: New Directions in Human-Computer Interaction*, Lawrence Earlbaum Assoc., Hillsdale, New Jersey, 1986, pp. 411–432
13. C. Frankish and J. Noyes, "Sources of Human Error in Data-Entry Tasks Using Speech Input," *Human Factors*, 1990, Vol. 32, No. 6, pp. 697–716.
14. C. A. Simpson et al., "System Design for Speech Recognition and Generation," *Human Factors*, 1985, Vol. 27, No. 2, pp. 115–141.
15. M. C. Clark, "The Use of Technology in the Home by Older Adults," *Proceedings of the Human-Factors Society, 30th Annual Meeting*, Santa Monica, California, 1986, pp. 1164–1166.
16. S. P. Casali, B. H. Williges, and R. D. Dryden, "Effects of Recognition Accuracy and Vocabulary Size of a Speech-Recognition System on Task Performance and User Acceptance," *Human Factors*, 1990, Vol. 32, No. 2, pp. 183–196.
17. L. H. Gellman and W. B. Whitten II, "Simulating an Automatic Operator Service to Optimize Customer Success," *Proceedings of the 12th International Symposium on Human Factors in Telecommunication*, The Hague, The Netherlands, May 1988, pp. 1–10.
18. J. Boulware, K. Kinder, and G. Batcha, "The First Network-Based

Application of Speech Recognition for Easy Access to Telephone Service: A Field Trial Report," *Proceedings of AVIOS* (American Voice I/O Society), Minneapolis, Minnesota, September 1992, pp. 295–300.
19. Y. C. Tsao, "A Lexical Study of Sentences Typed by Hearing-Impaired TDD Users," *Proceedings of the 13th International Symposium on Human Factors in Telecommunications*, Torino, Italy, September 10–14, 1990.
20. Y. C. Tsao, "Text-to-Speech Technology for Dual-Party Relay Services," *Proceedings of the Human-Factors Society, 35th Annual Meeting*, San Francisco, California, September 1991, pp. 213–216.
21. R. W. Bennett et al., "Speaking To, From, and Through Computers: Speech Technologies and User-Interface Design," *AT&T Technical Journal*, Vol. 68, No. 5, 1989, pp. 17–30.
22. D. Jones, K. Hapeshi, and C. Frankish, "Design Guidelines for Speech Recognition Interfaces," *Applied Ergonomics*, Vol. 20, No. 1, 1989, pp. 47–52.

*(Manuscript approved July 1993)*

***Blake L. Wattenbarger*** *is a technical manager in the Consumer Services Performance and Operations Department at AT&T Bell Laboratories in Holmdel, New Jersey. He is responsible for human-factors support of all AT&T consumer services. Mr. Wattenbarger joined AT&T in 1970. He has a B.A. in physics and mathematics from Oklahoma City University, Oklahoma, M.A. degrees in both psychology and mathematics from the University of Michigan, Ann Arbor, and a Ph.D. in experimental psychology, also from the University of Michigan.*

***Roger B. Garberg*** *is a member of the technical staff in the Services and Speech Technology Department at AT&T Bell Laboratories in Naperville, Illinois. He is responsible for exploring the properties of speech-based user interfaces that optimize system performance and end-user satisfaction. Mr. Garberg joined the company in 1986. He has an A.B. in social science from Shimer College in Mt. Carroll, Illinois, and both an M.A. and Ph.D. in applied cognitive psychology from Southern Illinois University in Carbondale.*

***Edward S. Halpern*** *is a member of the technical staff in the Services and Speech Technology Department at AT&T Bell Laboratories in Naperville, Illinois. He is responsible for human-factors design and evaluation of products and services using automatic speech recognition. Mr. Halpern joined the company in 1983. He has both a B.S. in economics and a Ph.D. in instructional systems technology from Indiana University in Bloomington.*

***Barry L. Lively*** *is a technical manager in the Core Human-Factors Group with AT&T Consumer Products in Indianapolis, Indiana. He is responsible for user-interface development work and management for all of AT&T Consumer Products. Mr. Lively joined the company in 1976. He has a B.S. in psychology from Pennsylvania State University in University Park, an M.S. in experimental psychology from Kent State University in Kent, Ohio, and a Ph.D. in experimental psychology from the University of Michigan, Ann Arbor.*