Language Learning & Technology
http://llt.msu.edu/vol2num1/article3/

July 1998, Volume 2, Number 1
pp. 54-73
(page numbers in PDF differ and should not be used for reference)

SPEECH TECHNOLOGY IN COMPUTER-AIDED LANGUAGE LEARNING: STRENGTHS AND LIMITATIONS OF A NEW CALL PARADIGM

Farzad Ehsani
Sehda, Inc.

Eva Knodt
Sehda, Inc.

ABSTRACT

We investigate the suitability of deploying speech technology in computer-based systems that can be used to teach foreign language skills. In reviewing the current state of speech recognition and speech processing technology and by examining a number of voice-interactive CALL applications, we suggest how to create robust interactive learning environments that exploit the strengths of speech technology while working around its limitations. In the conclusion, we draw on our review of these applications to identify directions of future research that might improve both the design and the overall performance of voice-interactive CALL systems.

INTRODUCTION

During the past two decades, the exercise of spoken language skills has received increasing attention among educators. Foreign language curricula focus on productive skills with special emphasis on communicative competence. Students' ability to engage in meaningful conversational interaction in the target language is considered an important, if not the most important, goal of second language education. This shift of emphasis has generated a growing need for instructional materials that provide an opportunity for controlled interactive speaking practice outside the classroom.

With recent advances in multimedia technology, computer-aided language learning (CALL) has emerged as a tempting alternative to traditional modes of supplementing or replacing direct student-teacher interaction, such as the language laboratory or audio-tape-based self-study. The integration of sound, voice interaction, text, video, and animation has made it possible to create self-paced interactive learning environments that promise to enhance the classroom model of language learning significantly. A growing number of textbook publishers now offer educational software of some sort, and educators can choose among a large variety of different products. Yet, the practical impact of CALL in the field of foreign language education has been rather modest. Many educators are reluctant to embrace a technology that still seeks acceptance by the language teaching community as a whole (Kenning & Kenning, 1990).

A number of reasons have been cited for the limited practical impact of computer-based language instruction. Among them are the lack of a unified theoretical framework for designing and evaluating CALL systems (Chapelle, 1997; Hubbard, 1988; Ng & Olivier, 1987); the absence of conclusive empirical evidence for the pedagogical benefits of computers in language learning (Chapelle, 1997; Dunkel, 1991; Salaberry, 1996); and finally, the current limitations of the technology itself (Holland, 1995; Warschauer, 1996). The rapid technological advances of the 1980s have raised both the expectations and the demands placed on the computer as a potential learning tool. Educators and second language acquisition (SLA) researchers alike are now demanding intelligent, user-adaptive CALL systems that offer not only sophisticated diagnostic tools, but also effective feedback mechanisms capable of focusing the learner on areas that need remedial practice. As Warschauer puts it, a computerized language teacher should be able to

> understand a user's spoken input and evaluate it not just for correctness but also for appropriateness. It should be able to diagnose a student's problems with pronunciation, syntax, or usage, and then intelligently decide among a range of options (e.g., repeating, paraphrasing, slowing down, correcting, or directing the student to background explanations). (Warschauer, 1996, p. 6)

Salaberry (1996) demands nothing short of a system capable of simulating the complex socio-communicative competence of a live tutor--in other words, the linguistic intelligence of a human--only to conclude that the attempt to create an "intelligent language tutoring system is a fallacy" (p. 11). Because speech technology isn't perfect, it is of no use at all. If it "cannot account for the full complexity of human language," why even bother modeling more constrained aspects of language use (Higgins, 1988, p. vii)? This sort of all-or-nothing reasoning seems symptomatic of much of the latest pedagogical literature on CALL. The quest for a theoretical grounding of CALL system design and evaluation (Chapelle, 1997) tends to lead to exaggerated expectations as to what the technology ought to accomplish. When combined with little or no knowledge of the underlying technology, the inevitable result is disappointment.

In this paper, we make a case for using automatic speech recognition (ASR) and speech processing technology in CALL. We propose not only that speech technology is an essential component of CALL, but that it is, in fact, ready to be deployed successfully in second language education, provided that the current limitations of the technology are understood and systems are designed in ways that work around these limitations.

In order to appreciate the potential benefit of using speech technology in CALL, a basic understanding of both the core technology and its limitations--what it can and cannot do--is therefore essential. In the following section, we will present an overview of speech recognition. We will then cover design considerations as they relate to the performance of specific speech applications. An overview of current research trends will help identify the kinds of technological advances that lend themselves to being deployed in computer-based language instruction. Next, to illustrate the potential use of speech technology, we will examine a number of innovative language learning applications that offer voice-interactive capabilities. We will evaluate these applications in view of how they integrate speech technology within an overall technical and pedagogical design, and how effectively they deal with current technological limitations. In the final section of the paper, we will draw on our review of these applications, as well as on our own experience in building a voice-interactive system for learning Japanese, to identify directions of future research that might improve both the design and the overall performance of voice-interactive CALL systems.

PRINCIPLES OF ASR TECHNOLOGY

Consider the following four scenarios:

1. A court reporter listens to the opening arguments of the defense and types the words into a steno-machine attached to a word-processor.
2. A medical doctor activates a dictation device and speaks his or her patient's name, date of birth, symptoms, and diagnosis into the computer. He or she then pushes "end input" and "print" to produce a written record of the patient's diagnosis.
3. A mother tells her three-year old, "Hey Jimmy, get me my slippers, will you?" The toddler smiles, goes to the bedroom, and returns with papa's hiking boots.
4. A first-grader reads aloud a sentence displayed by an automated Reading Tutor. When he or she stumbles over a difficult word, the system highlights the word, and a voice reads the word aloud. The student repeats the sentence--this time correctly--and the system responds by displaying the next sentence.

At some level, all four scenarios involve speech recognition. An incoming speech signal elicits a response from a "listener." In the first two instances, the response consists of a written transcript of the spoken input, whereas in the latter two cases, an action is performed in response to a spoken command. In all four cases, the "success" of the voice interaction is relative to a given task as embodied in a set of expectations that accompany the input. The interaction succeeds when the response--by a machine or human "listener"--matches these expectations.

Recognizing and understanding human speech requires a considerable amount of linguistic knowledge: a command of the phonological, lexical, semantic, grammatical, and pragmatic conventions that constitute a language. The listener's command of the language must be "up" to the recognition task or else the interaction fails. Jimmy returns with the wrong items, because he cannot yet verbally discriminate between different kinds of shoes. Likewise, the reading tutor would miserably fail in performing the court-reporter's job or transcribing medical patient information, just as the medical dictation device would be a poor choice for diagnosing a student's reading errors. On the other hand, the human court reporter--assuming he or she is an adult native speaker--would have no problem performing any of the tasks mentioned under (1) through (4). The linguistic competence of an adult native speaker covers a broad range of recognition tasks and communicative activities. Computers, on the other hand, perform best when designed to operate in clearly circumscribed linguistic sub-domains.

Humans and machines process speech in fundamentally different ways (Bernstein & Franco, 1996). Complex cognitive processes account for the human ability to associate acoustic signals with meanings and intentions. For a computer, on the other hand, speech is essentially a series of digital values. However, despite these differences, the core problem of speech recognition is the same for both humans and machines: namely, of finding the best match between a given speech sound and its corresponding word string. Automatic speech recognition technology attempts to simulate and optimize this process computationally.

Since the early 1970s, a number of different approaches to ASR have been proposed and implemented, including Dynamic Time Warping, template matching, knowledge-based expert systems, neural nets, and Hidden Markov Modeling (HMM) (Levinson & Liberman, 1981; Weinstein, McCandless, Mondshein, & Zue, 1975; for a review, see Bernstein & Franco, 1996). HMM-based modeling applies sophisticated statistical and probabilistic computations to the problem of pattern matching at the sub-word level. The generalized HMM-based approach to speech recognition has proven an effective, if not the most effective, method for creating high-performance speaker-independent recognition engines that can cope with large vocabularies; the vast majority of today's commercial systems deploy this technique. Therefore, we focus our technical discussion on an explanation of this technique.

An HMM-based speech recognizer consists of five basic components: (a) an acoustic signal analyzer which computes a spectral representation of the incoming speech; (b) a set of phone models (HMMs) trained on large amounts of actual speech data; (c) a lexicon for converting sub-word phone sequences into words; (d) a statistical language model or grammar network that defines the recognition task in terms of legitimate word combinations at the sentence level; (e) a decoder, which is a search algorithm for computing the best match between a spoken utterance and its corresponding word string. Figure 1 shows a schematic representation of the components of a speech recognizer and their functional interaction.

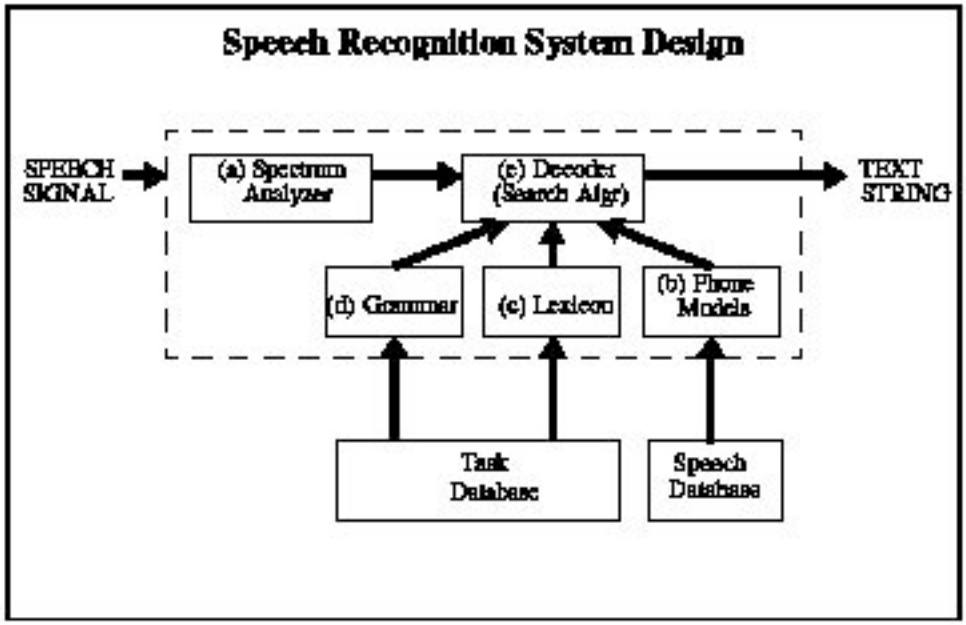

Figure 1. Components of a speech recognition device

A. Signal Analysis

The first step in automatic speech recognition consists of analyzing the incoming speech signal. When a person speaks into an ASR device--usually through a high quality noise-canceling microphone--the computer samples the analog input into a series of 16- or 8-bit values at a particular sampling frequency (ranging from 8 to 22KHz). These values are grouped together in predetermined overlapping temporal intervals called "frames." These numbers provide a precise description of the speech signal's amplitude. In a second step, a number of acoustically relevant parameters such as energy, spectral features, and pitch information, are extracted from the speech signal (for a visual representation of some of these parameters, see Figure 2 on page 53). During training, this information is used to model that particular portion of the speech signal. During recognition, this information is matched against the pre-existing model of the signal.

B. Phone Models

Training a machine to recognize spoken language amounts to modeling the basic sounds of speech (phones). Automatic speech recognition strings together these models to form words. Recognizing an incoming speech signal involves matching the observed acoustic sequence with a set of HMM models. An HMM can model either phones or other sub-word units or it can model words or even whole sentences. Phones are either modeled as individual sounds--so-called monophones--or as phone combinations that model several phones and the transitions between them (biphones or triphones). After comparing the incoming acoustic signal with the HMMs representing the sounds of language, the system computes a hypothesis based on the sequence of models that most closely resembles the incoming signal. The HMM model for each linguistic unit (phone or word) contains a probabilistic representation of all the possible pronunciations for that unit--just as the model of the handwritten cursive *b* would have many different representations.

Building HMMs--a process called training--requires a large amount of speech data of the type the system is expected to recognize. Large-vocabulary speaker-independent continuous dictation systems are typically trained on tens of thousands of read utterances by a cross-section of the population, including

members of different dialect regions and age-groups. As a general rule, an automatic speech recognizer cannot correctly process speech that differs in kind from the speech it has been trained on. This is why most commercial dictation systems, when trained on standard American English, perform poorly when encountering accented speech, whether by non-native speakers or by speakers of different dialects. We will return to this point in our discussion of voice-interactive CALL applications.

C. Lexicon

The lexicon, or dictionary, contains the phonetic spelling for all the words that are expected to be observed by the recognizer. It serves as a reference for converting the phone sequence determined by the search algorithm into a word. It must be carefully designed to cover the entire lexical domain in which the system is expected to perform. If the recognizer encounters a word it does not "know" (i.e., a word not defined in the lexicon), it will either choose the closest match or return an out-of-vocabulary recognition error. Whether a recognition error is registered as a misrecognition or an out-of-vocabulary error depends in part on the vocabulary size. If, for example, the vocabulary is too small for an unrestricted dictation task--let's say less than 3K--the out-of-vocabulary errors are likely to be very high. If the vocabulary is too large, the chance of misrecognition errors increases because with more similar-sounding words, the confusability increases. The vocabulary size in most commercial dictation systems tends to vary between 5K and 60K.

D. The Language Model

The language model predicts the most likely continuation of an utterance on the basis of statistical information about the frequency in which word sequences occur on average in the language to be recognized. For example, the word sequence *A bare attacked him* will have a very low probability in any language model based on standard English usage, whereas the sequence *A bear attacked him* will have a higher probability of occurring. Thus the language model helps constrain the recognition hypothesis produced on the basis of the acoustic decoding just as the context helps decipher an unintelligible word in a handwritten note. Like the HMMs, an efficient language model must be trained on large amounts of data, in this case texts collected from the target domain.

In ASR applications with constrained lexical domain and/or simple task definition, the language model consists of a grammatical network that defines the possible word sequences to be accepted by the system without providing any statistical information. This type of design is suitable for CALL applications in which the possible word combinations and phrases are known in advance and can be easily anticipated (e.g., based on user data collected with a system pre-prototype). Because of the *a priori* constraining function of a grammar network, applications with clearly defined task grammars tend to perform at much higher accuracy rates than the quality of the acoustic recognition would suggest.

E. Decoder

Simply put, the decoder is an algorithm that tries to find the utterance that maximizes the probability that a given sequence of speech sounds corresponds to that utterance. This is a search problem, and especially in large vocabulary systems careful consideration must be given to questions of efficiency and optimization, for example to whether the decoder should pursue only the most likely hypothesis or a number of them in parallel (Young, 1996). An exhaustive search of all possible completions of an utterance might ultimately be more accurate but of questionable value if one has to wait two days to get a result. Trade-offs are therefore necessary to maximize the search results while at the same time minimizing the amount of CPU and recognition time.

PERFORMANCE AND DESIGN ISSUES IN SPEECH APPLICATIONS

For educators and developers interested in deploying ASR in CALL applications, perhaps the most important consideration is recognition performance: How good is the technology? Is it ready to be

deployed in language learning? These questions cannot be answered except with reference to particular applications of the technology, and therefore touch on a key issue in ASR development: the issue of human-machine interface design.

As we recall, speech recognition performance is always domain specific--a machine can only do what it is programmed to do, and a recognizer with models trained to recognize business news dictation under laboratory conditions will be unable to handle spontaneous conversational speech transmitted over noisy telephone channels. The question that needs to be answered is therefore not simply "How good is ASR technology?" but rather, "What do we want to use it for?" and "How do we get it to perform the task?"

In the following section, we will address the issue of system performance as it relates to a number of successful commercial speech applications. By emphasizing the distinction between recognizer performance on the one hand--understood in terms of "raw" recognition accuracy--and system performance on the other; we suggest how the latter can be optimized within an overall design that takes into account not only the factors that affect recognizer performance as such, but also, and perhaps even more importantly, considerations of human-machine interface design.

Historically, basic speech recognition research has focused almost exclusively on optimizing large vocabulary speaker-independent recognition of continuous dictation. A major impetus for this research has come from US government sponsored competitions held annually by the Defense Advanced Research Projects Agency (DARPA). The main emphasis of these competitions has been on improving the "raw" recognition accuracy--calculated in terms of average omissions, insertions, and substitutions--of large-vocabulary continuous speech recognizers (LVCSRs) in the task of recognizing read sentence material from a number of standard sources (e.g., *The Wall Street Journal* or *The New York Times*). The best laboratory systems that participated in the WSJ large-vocabulary continuous dictation task have achieved word error rates as low as 5%, that is, on average, one recognition error in every twenty words (Pallet, 1994).

Although the annual DARPA benchmark tests have yielded significant technological advances, they are a poor indicator of ASR performance as it relates to the technology's potential commercial use. Very few of the laboratory systems participating in these contests are commercially viable, due in large measure to the narrow focus on recognition accuracy. Experimental LVCSR systems generally run on very large computers, and recognition speed is not an issue. By comparison, the base-line recognizer performance of commercial dictation systems with roughly similar task definition and vocabularies of 20K to 60K is much lower. Dragon's Naturally Speaking or IBM's ViaVoice, for example, start out with a baseline recognition accuracy of only 60% to 80% (again depending on accent, background noise, type of utterance, etc.). But these systems run on affordable PC platforms with modest memory requirements, operate near real-time, and support speaker adaptation features that allow the user to train the system. Training a recognizer is a reciprocal process: the system adapts to the acoustic characteristics of the user's voice by analyzing and learning from speech samples collected during the setup phase; the user, over time, adjusts his or her speaking style to "dictation mode," a clearly articulated speech input that conforms to the grammatical conventions of written discourse. Haskin (1997) reports post-training error rates as low as 5% while more conservative estimates range between 11-13% on average (Jecker, 1998). As the apparent commercial success of these systems shows, such a performance range may be acceptable, provided that the system offers convenient editing features. Continuous dictation, however, remains limited in scope and is still far from recognizing spontaneous conversational speech.

An important lesson learned in the development of LVCSR systems is that the technology itself is highly adaptable, yielding increasing robustness when tailored to a specific recognition task. This insight has led to the successful commercialization of speech technology in telephony applications with constrained task domains such as voice-dialing, directory assistance, and information retrieval. The key to designing such applications lies in choosing the right task and in optimizing the variables that affect recognition

performance. In what follows, we will discuss some of these variables and show how system performance can be maximized when speech technology is integrated within a carefuly designed user interface.

Task Definition

This is the most important step in designing a speech recognizer. Delimiting the performance domain imposes constraints on both the vocabulary size and what is referred to as "perplexity," which is usually defined as the average branching factor within any given grammar network. A small vocabulary recognizer with limited perplexity (e.g., of the type used in automatic voice dialing), tends to be much more robust than a high-perplexity large-vocabulary dictation system.

In general, recognizers perform faster and more accurately when the incoming speech is enunciated clearly and in a noise-free environment, when the task perplexity is low, and when the dictionary is small. In this case, the system needs less CPU time and memory to process alternative recognition hypotheses, and word error rates tend to be lower. A simple "yes/no" recognition task is trivial compared to a ticket reservation system that uses a natural dialog user interface. The relationship between perplexity and performance can work to our advantage when we are developing voice-interactive instructional materials since words and phrases used by language learners are usually limited to a relatively small set of clearly circumscribed tasks. However, for systems with limited task domain to perform as expected, all potential user responses must be known in advance and anticipated in the system's grammar and vocabulary. Therefore, it is important to collect authentic user data in the early stages of developing such systems.

Acoustic Models

Recognizers tend to perform best when trained on (or adapted to) the voice characteristics or speaking style of the speaker. Speaker independent recognizers contain acoustic models obtained by averaging over large variations in the speech patterns of large populations of various ages and dialect groups. By contrast, speaker-dependent systems are trained specifically on the voice of the speaker(s) for whom they are designed. A third option is speaker adaptation, a technique in which acoustic parameters obtained from a subset of speakers, or one speaker, are used to augment or modify the generalized models of a speaker-independent system. Speaker adaptation can reduce recognition error rates by 30% to 70% depending on the acoustic environment and the original acoustic models used (Neumeyer, Sankar, & Digalakis, 1995; Woodland, Pye, & Gales, 1996; Zavaliagkos, Schwartz, McDonough, & Makhoul, 1995). The importance for CALL is that native acoustic models can be adapted to recognize the speech of language learners. Doing so involves collecting the appropriate speech data and training non-native models. The resulting acoustic models can be made exclusively from non-native data (Neumeyer et al., 1996), or by adapting native models to the non-native data (Ehsani, 1996).

Input Modality

Another variable that affects recognition performance is the manner in which the system processes the incoming speech signal. In systems with a discrete speech input modality, the recognizer processes each word separately. Therefore each word must be spoken separately with distinct pauses between them. In systems with continuous input mode, no such pauses are necessary. Continuous Speech Recognition (CSR) systems use more extensive search algorithms in the decoding stage to optimize not only the phone strings, but also the word strings. The trade-off in recognition accuracy can be formidable, and under conditions where system resources are limited and high degrees of accuracy are necessary, discrete input may be the design of choice.

Input Quality

For optimal recognition performance, the incoming speech signal must be of high acoustic quality. A number of standard techniques can remove noise from the signal or adapt acoustical models to noisy data (for a review, see Young, 1996). However, not only noise interference can affect the quality of the speech

input. A number of other factors, often overlooked in the literature, play a role as well, such as the type of sound card and microphone used, or whether the speech is run through a pre-amplifier. Sound cards have their own internal amplifiers, but they tend to amplify the noise along with the speech. The amplitude of the speech signal needs to be carefully adjusted for best recognition performance. More specifically, the amplitude needs to be kept within a certain limited range. If the amplitude exceeds an upper limit, the signal is clipped and the signal analyzer cannot extract all of the relevant features. If the amplitude is too low, background noise becomes more prominent and can overpower the signal.

Furthermore, the microphone can make a tremendous difference in recognition performance. Most recognizers perform best when used with a noise-canceling head-mounted microphone. Not only do these microphones filter out extraneous noise, but the head-mounted position ensures that the distance between the speaker's mouth and the microphone is kept constant and the amplitude remains stable throughout the utterances. Finally, some kind of mechanism for automatically adjusting (or telling the user to adjust) the amplifier or the pre-amp setting is of value. Most commercial voice-interactive CALL systems offer this feature.

Careful consideration of the factors described above must enter into the design of commercial speech applications, if they are to perform well in practical applications.

CURRENT TRENDS IN VOICE-INTERACTIVE CALL

In recent years, an increasing number of speech laboratories have begun deploying speech technology in CALL applications. Results include voice-interactive prototype systems for teaching pronunciation, reading, and limited conversational skills in semi-constrained contexts. Our review of these applications is far from exhaustive. It covers a select number of mostly experimental systems that explore paths we found promising and worth pursuing. We will discuss the range of voice-interactions these systems offer for practicing certain language skills, explain their technical implementation, and comment on the pedagogical value of these implementations. Apart from giving a brief system overview, we report experimental results if available and provide an assessment of how far away the technology is from being deployed in the commercial and educational environments.

Pronunciation Training

A useful and remarkably successful application of speech recognition and processing technology has been demonstrated by a number of research and commercial laboratories in the area of pronunciation training. Voice-interactive pronunciation tutors prompt students to repeat spoken words and phrases or to read aloud sentences in the target language for the purpose of practicing both the sounds and the intonation of the language. The key to teaching pronunciation successfully is corrective feedback, more specifically, a type of feedback that does not rely on the student's own perception. A number of experimental systems have implemented automatic pronunciation scoring as a means to evaluate spoken learner productions in terms of fluency, segmental quality (phonemes) and supra-segmental features (intonation). The automatically generated proficiency score can then be used as a basis for providing other modes of corrective feedback. We discuss segmental and supra-segmental feedback in more detail below.

Segmental Feedback. Technically, designing a voice-interactive pronunciation tutor goes beyond the state of the art required by commercial dictation systems. While the grammar and vocabulary of a pronunciation tutor is comparatively simple, the underlying speech processing technology tends to be complex since it must be customized to recognize and evaluate the disfluent speech of language learners. A conventional speech recognizer is designed to generate the most charitable reading of a speaker's utterance. Acoustic models are generalized so as to accept and recognize correctly a wide range of different accents and pronunciations. A pronunciation tutor, by contrast, must be trained to both recognize and correct subtle deviations from standard native pronunciations.

A number of techniques have been suggested for automatic recognition and scoring of non-native speech (Bernstein, 1997; Franco, Neumeyer, Kim, & Ronen, 1997; Kim, Franco, & Neumeyer, 1997; Witt & Young, 1997). In general terms, the procedure consists of building native pronunciation models and then measuring the non-native responses against the native models. This requires models trained on both native and non-native speech data in the target language, and supplemented by a set of algorithms for measuring acoustic variables that have proven useful in distinguishing native from non-native speech. These variables include response latency, segment duration, inter-word pauses (in phrases), spectral likelihood, and fundamental frequency (F0). Machine scores are calculated from statistics derived from comparing non-native values for these variables to the native models.

In a final step, machine generated pronunciation scores are validated by correlating these scores with the judgment of human expert listeners. As one would expect, the accuracy of scores increases with the duration of the utterance to be evaluated. Stanford Research Institute (SRI) has demonstrated a 0.44 correlation between machine scores and human scores at the phone level. At the sentence level, the machine-human correlation was 0.58, and at the speaker level it was 0.72 for a total of 50 utterances per speaker (Franco et al., 1997; Kim et al., 1997). These results compare with 0.55, 0.65, and 0.80 for phone, utterance, and speaker level correlation between human graders. A study conducted at Entropic shows that based on about 20 to 30 utterances per speaker and on a linear combination of the above techniques, it is possible to obtain machine-human grader correlation levels as high as 0.85 (Bernstein, 1997).

Others have used expert knowledge about systematic pronunciation errors made by L2 adult learners in order to diagnose and correct such errors. One such system is the European Community project SPELL for automated assessment and improvement of foreign language pronunciation (Hiller, Rooney, Vaughan, Eckert, Laver, & Jack, 1994). This system uses advanced speech processing and recognition technologies to assess pronunciation errors by L2 learners of English (French or Italian speakers) and provide immediate corrective feedback. One technique for detecting consonant errors induced by inter-language transfer was to include students' L1 pronunciations into the grammar network. In addition to the English /th/ sound, for example, the grammar network also includes /t/ or /s/, that is, errors typical of non-native Italian speakers of English. This system, although quite simple in the use of ASR technology, can be very effective in diagnosing and correcting known problems of L1 interference. However, it is less effective in detecting rare and more idiosyncratic pronunciation errors. Furthermore, it assumes that the phonetic system of the target language (e.g., English) can be accurately mapped to the learners' native language (e.g., Italian). While this assumption may work well for an Italian learner of English, it certainly does not for a Chinese learner; that is, there are sounds in Chinese that do not resemble any sounds in English.

A system for teaching the pronunciation of Japanese long vowels, the mora nasal, and mora obstruents was recently built at the University of Tokyo. This system enables students to practice phonemic differences in Japanese that are known to present special challenges to L2 learners. It prompts students to pronounce minimal pairs (e.g., long and short vowels) and returns immediate feedback on segment duration. Based on the limited data, the system seems quite effective at this particular task. Learners quickly mastered the relevant duration cues, and the time spent on learning these pronunciation skills was well within the constraints of Japanese L2 curricula (Kawai & Hirose, 1997). However, the study provides no data on long-term effects of using the system.

Supra-segmental Feedback. Correct usage of supra-segmental features such as intonation and stress has been shown to improve the syntactic and semantic intelligibility of spoken language (Crystal, 1981). In spoken conversation, intonation and stress information not only helps listeners to locate phrase boundaries and word emphasis, but also to identify the pragmatic thrust of the utterance (e.g., interrogative vs. declarative). One of the main acoustical correlates of stress and intonation is fundamental frequency (F0); other acoustical characteristics include loudness, duration, and tempo. Most commercial signal processing software have tools for tracking and visually displaying F0 contours (see Figure 2). Such displays can and have been used to provide valuable pronunciation feedback to students.

Experiments have shown that a visual F0 display of supra-segmental features combined with audio feedback is more effective than audio feedback alone (de Bot, 1983; James, 1976), especially if the student's F0 contour is displayed along with a native model. The feasibility of this type of visual feedback has been demonstrated by a number of simple prototypes (Abberton & Fourcin, 1975; Anderson-Hsieh, 1994; Hiller et al., 1994; Spaai & Hermes, 1993; Stibbard, 1996). We believe that this technology has a good potential for being incorporated into commercial CALL systems.

Other types of visual pronunciation feedback include the graphical display of a native speaker's face, the vocal tract, spectrum information, and speech waveforms (see Figure 2). Experiments have shown that a visual display of the talker improves not only word identification accuracy (Bernstein & Christian, 1996), but also speech rhythm and timing (Markham & Nagano-Madesen, 1997). A large number of commercial pronunciation tutors on the market today offer this kind of feedback. Yet others have experimented with using a real-time spectrogram or waveform display of speech to provide pronunciation feedback. Molholt (1990) and Manuel (1990) report anecdotal success in using such displays along with guidance on how to interpret the displays to improve the pronunciation of suprasegmental features in L2 learners of English. However, the authors do not provide experimental evidence for the effectiveness of this type of visual feedback. Our own experience with real-time spectrum and waveform displays suggests their potential use as pronunciation feedback provided they are presented along with other types of feedback, as well as with instructions on how to interpret the displays.

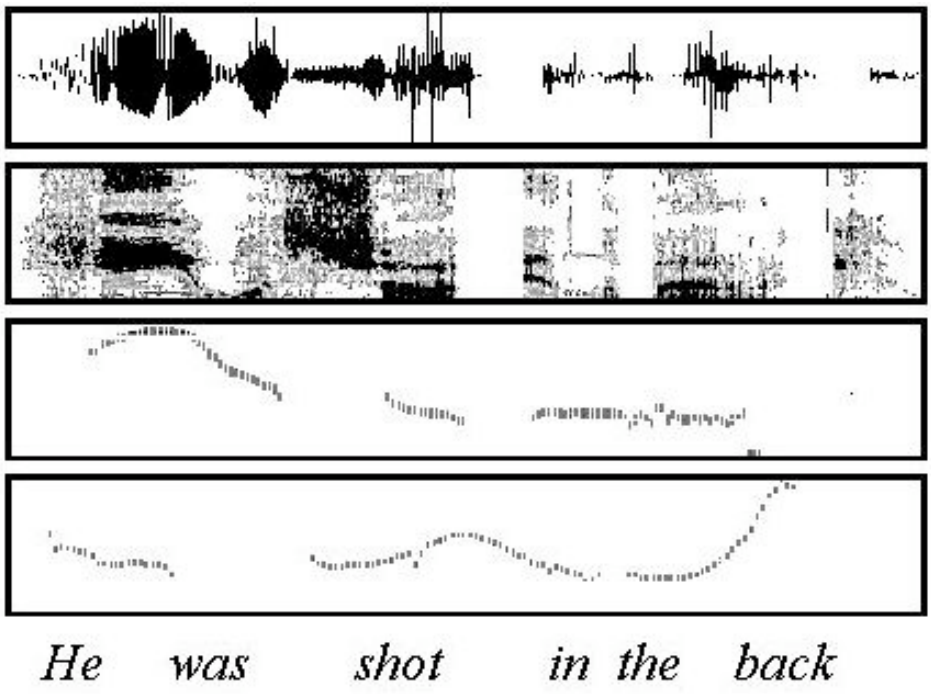

Figure 2. Alternative speech display modes of the phrase *He was shot in the back* generated with Entropic's signal processing software. (1)

Reading Aloud

Reading aloud exercises literacy skills in both second language and literacy education. Intensive practice in reading aloud helps students establish the conventional association between sounds and their written form, a skill that requires years of practice in young children and students of languages with non-phonetic writing, such as Japanese or Chinese. Teaching children and students how to read their own native or a foreign language is thus an area where speech recognition technology can make a significant difference.

Imagine a reading tutor that not only listens to children and students reading aloud a story presented on the screen, but intervenes to provide help when needed and corrects mistakes.

Designing a basic recognition network for a voice-interactive reading tutor is relatively straightforward. There is only one correct spoken response to any given written prompt, and the system "knows" in advance what the student will be trying to say. However, the technical challenge is to recognize and respond adequately to the disfluencies of inexperienced readers. Such disfluencies include hesitations, mispronunciations, false starts, and self-corrections.

In the early 1990s, Cowan and Jones (1991), McCandless (1992), and Phillips, Zue, and McCandless (1993) among others demonstrated the technical feasibility of a voice-interactive reading tutor, without, however, providing empirical user data. One of the first fielded prototype systems for teaching reading to young children was developed by the Center for Teaching and Learning (CTL) in 1991 (Kantrov, 1991). The simple but robust multimedia application used an isolated, speaker-dependent recognizer and limited reading vocabulary (18+ words). The system was designed to expand children's reading vocabulary by embedding new words within the context of a goal-oriented game: children are called upon to help a bear overcome obstacles on his way home; reading the word correctly removes the obstacle. Results of three field trials in two Boston-area public schools indicated that the problems with the application were related to the human interface and input mode (microphones), rather than the speech recognition component per se. Ironically, recognition errors, especially misrecognition of correctly read words, contributed positively to the pedagogical effect of the application: the children got additional reading practice, because they had to repeat the words several times until the machine responded appropriately.

One of the most ambitious automated reading coaches currently being developed is the ongoing Project LISTEN at Carnegie Mellon University (CMU). Designed to combat illiteracy, the fully automated prototype uses continuous speech recognition to listen to children read continuous text and automatically trigger pedagogically appropriate interventions (Mostow, Roth, Hauptmann, & Kane, 1994). The system features a personalized agent, "Emily," who provides feedback and assistance when necessary. The system incorporates expert knowledge on individual reading assistance that is both pedagogically relevant and technically feasible. Emily intervenes when the child misreads one or more words in the current sentence, gets stuck, or clicks on a word to get help. On the other hand, to reduce frustration in children with reading difficulties, the system deliberately refrains from treating false starts, self-corrections, or hesitations as "mistakes." Instead, errors of this type are modeled and included into the recognition grammar as acceptable.

An experimental trial of the system was conducted among 12 second graders at an urban school in Pittsburgh. Results showed that the children could read at a reading level 0.6 years more advanced when using the automated reading coach, and the average number of reading mistakes fell from 12.3% (without assistance) to 2.6% (with assistance) in texts with similar difficulty.

An improved version of CMU's reading coach running real-time on an affordable PC platform was fielded in 1996 among 8 of the poorest third grade readers at Fort Pitt, PA to measure improvements in reading performance over an 8 month period of using the system (Mostow, 1997; Mostow & Aist, 1997). While the earlier study measured reading performance only in terms of student word error rates, the improved system implements algorithms for measuring reading fluency in young children. Relevant performance variables include reading rate, inter-word latency (silence), disfluency (false starts, self-corrections, omissions) and time spent with the assistant. Comparing subjects' reading fluency levels at the beginning of using the system with those at the end, the experiments suggest an overall improvement in reading accuracy of 16% and a 35% decrease in inter-word latency. After using the system for eight months, students' reading levels improved by an average of two years. These results are encouraging in that they show how careful system design and evaluation based on user data can lead to useful and practical applications.

Teaching Linguistic Structures and Limited Conversation

Apart from supporting systems for teaching basic pronunciation and literacy skills, ASR technology is being deployed in automated language tutors that offer practice in a variety of higher-level linguistic skills ranging from highly constrained grammar and vocabulary drills to limited conversational skills in simulated real-life situations. Prior to implementing any such system, a choice needs to be made between two fundamentally different system design types: *closed response* vs. *open response* design. In both designs, students are prompted for speech input by a combination of written, spoken, or graphical stimuli. However, the designs differ significantly with reference to the type of verbal computer-student interaction they support. In closed response systems, students must choose one response from a limited number of possible responses presented on the screen. Students know exactly what they are allowed to say in response to any given prompt. By contrast, in systems with open response design, the network remains hidden and the student is challenged to generate the appropriate response without any cues from the system.

Closed Response Designs. One of the first implementations of a closed response design was the Voice Interactive Language Instruction System (VILIS) developed at SRI (Bernstein & Rtischev, 1991). This system elicits spoken student responses by presenting queries about graphical displays of maps and charts. Students infer the right answers to a set of multiple-choice questions and produce spoken responses.

A more recent prototype currently under development in SRI is the Voice Interactive Language Training System (VILTS), a system designed to foster speaking and listening skills for beginning through advanced L2 learners of French (Egan, 1996; Neumeyer et al., 1996; Rypa, 1996). The system incorporates authentic, unscripted conversational materials collected from French speakers into an engaging, flexible, and user-centered lesson architecture. The system deploys speech recognition to guide students through the lessons and automatic pronunciation scoring to provide feedback on the fluency of student responses. As far as we know, only the pronunciation scoring aspect of the system has been validated in experimental trials (Neumeyer et al., 1996).

In pedagogically more sophisticated systems, the query-response mode is highly contextualized and presented as part of a simulated conversation with a virtual interlocutor. To stimulate student interest, closed response queries are often presented in the form of games or goal-driven tasks. One commercial system that exploits the full potential of this design is TraciTalk (Courseware Publishing International, Inc., Cupertino, CA), a voice-driven multimedia CALL system aimed at more advanced ESL learners. In a series of loosely connected scenarios, the system engages students in solving a mystery. Prior to each scenario, students are given a task (e.g., eliciting a certain type of information), and they accomplish this task by verbally interacting with characters on the screen. Each voice interaction offers several possible responses, and each spoken response moves the conversation in a slightly different direction. There are many paths through each scenario, and not every path yields the desired information. This motivates students to return to the beginning of the scene and try out a different interrogation strategy. Moreover, TraciTalk features an agent that students can ask for assistance and accepts spoken commands for navigating the system. Apart from being more fun and interesting, games and task-oriented programs implicitly provide positive feedback by giving students the feeling of having solved a problem solely by communicating in the target language.

The speech recognition technology underlying closed response query implementations is very simple, even in the more sophisticated systems. For any given interaction, the task perplexity is low and the vocabulary size is comparatively small. As a result, these systems tend to be very robust. Recognition accuracy rates in the low to upper 90% range can be expected depending on task definition, vocabulary size, and the degree of non-native disfluency.

Open Response Design. The basic principle of an open response design is that students have to come up with a response entirely on their own, without any help from the system. Such systems present a greater challenge to the student and consequently lend themselves to pedagogically more ambitious implementations. Internally, however, systems of this type process students' responses *as if* they were selected from a multiple-choice list (Waters, 1994). As a minimum, all possible correct responses must be included in the grammar network. If, in addition, the system is supposed to provide detailed feedback to incorrect or questionable input, any potential mistakes must be modeled and anticipated in the grammar network. An open response design can be either very simple or dauntingly complex. While it is easy to implement an open response design for simple question-answer drills (e.g., "What's the color of grass?"), designing a system capable of holding up a prolonged conversation on "How do I get to the train station?" requires a multi-level network grammar based on data collected from students, natural language processing capabilities, and strategies for recovering from misunderstandings. In the following, we provide a sense of the range of possibilities associated with this type of CALL design.

Stimulus-response queries

A recent implementation of an open response design for teaching beginning Spanish is *The Auto Interactive Tutor* (TAIT) by Mitsubishi Research Laboratories (Waters, 1995). The system presents study material in the form of stimulus-response pairs and is organized around a set of primitive items to be learned such as "What is the Spanish word for 'left'?" It uses speech recognition to process student input and to move forward. What distinguishes this system from the others discussed in this review is the fact that it is user-adaptive. It constructs an evolving model of the user's knowledge by keeping track of the average error rate, and it presents subsequent material accordingly. Even though the system was never fielded with users, and despite the fact that it uses a rather primitive, small-vocabulary, discrete-input, speaker-dependent speech recognizer, informal evidence suggests that TAIT makes clever use of design in order to get the most out of a simple implementation of speech technology.

Simulated real-life conversation

In the past few years, a number of speech laboratories have tried to build systems that can understand and judge continuous spoken language and maintain a conversation through several turns. The goal is to emulate essential features of human-human communication for the purpose of teaching and practicing conversational skills in the target language. Interactions should work without requiring collateral cues from a mouse or keyboard, operate at an appropriate conversational pace, and incorporate verbal strategies for resolving misunderstandings.

A prototype system for simulating human-human interactions was recently developed at Entropic (Ehsani, Bernstein, Najmi, & Todic, 1997; Ehsani, Bernstein, & Najmi, in press). The system, called *Subarashii* (Japanese for "wonderful"), offers beginning students of Japanese the opportunity to solve simple problems through (virtual) spoken interactions with monolingual Japanese natives. *Subarashii* is designed to understand what a student is saying in Japanese (within a constrained context) and to respond in a meaningful way in spoken Japanese.

In a series of loosely connected everyday situations, the system poses problems in written English (e.g., inviting a friend to go to a movie) and offers occasional support to the student in the form of written reminders, but problems can only be solved by speaking and understanding Japanese. Despite the restricted communicative competence of beginning L2 learners, there is a variety of potentially valid utterances that the student can produce in any situation, even if some of these may be grammatically or pragmatically incorrect. *Subarashii* will not only properly process correct responses, but it will also recognize and reject (with an appropriate message) many incorrect inputs. However, in order to give appropriate feedback on student errors, the system must be able to anticipate such errors along with the expected responses. In other words, they must be included in the recognition network. To create such a network, each encounter was prototyped in a traditional Hypercard environment on a Macintosh with text

input. Hypercard provides an effective means of modeling each encounter on the basis of actual input from a test group of students.

The acoustic models for this system were originally built with a low rejection in order to be more forgiving of a student's accent. However, this approach resulted in a relatively large proportion of misrecognized utterances, and false acceptance. Two trials conducted among 32 students from Silver Creek High School, San Jose, and 13 Stanford students yielded alarmingly low recognition accuracy rates (41.6% and 36.6% respectively) due to insufficient training data. These results imply that only one-third of the students' responses were correctly recognized. However, the functional accuracy, that is, the percentage of times the system responded appropriately, was significantly higher (66.9% and 71.4% for Silver Creek and Stanford respectively). For example, if the student said, "Hi, how are you doing?" and the question was misrecognized as, "Hello, how are you?," this is technically a recognition error. Functionally, however, the system will respond appropriately which suggests that near perfect recognition accuracy may not be a necessary requirement for an effective speech dialog system.

FUTURE TRENDS IN VOICE-INTERACTIVE CALL

In the previous sections, we reviewed the current state of speech technology, discussed some of the factors affecting recognition performance, and introduced a number of research prototypes that illustrate the range of speech-enabled CALL applications that are currently technically and pedagogically feasible. With the exception of a few exploratory open response dialog systems, most of these systems are designed to teach and evaluate linguistic form (pronunciation, fluency, vocabulary study, or grammatical structure). This is no coincidence. Formal features can be clearly identified and integrated into a focused task design. This means that robust performance can be expected. Furthermore, mastering linguistic form remains an important component of L2 instruction, despite the emphasis on communication (Holland, 1995). Prolonged, focused practice of a large number of items is still considered an effective means of expanding and reinforcing linguistic competence (Waters, 1994). However, such practice is time consuming. CALL can automate these aspects of language training, thereby freeing up valuable class time that would otherwise be spent on drills.

While such systems are an important step in the right direction, other more complex and ambitious applications are conceivable and no doubt desirable. Imagine a student being able to access the Internet, find the language of his or her choice, and tap into a comprehensive voice-interactive multimedia language program that would provide the equivalent of an entire first year of college instruction. The computer would evaluate the student's proficiency level and design a course of study tailored to his or her needs. Or think of using the same Internet resources and a set of high-level authoring tools to put together a series of virtual encounters surrounding the task of finding an apartment in Berlin. As a minimum, one would hope that natural speech input capacity becomes a routine feature of any CALL application.

To many educators, these may still seem like distant goals, and yet we believe that they are not beyond reach. In what follows, we identify four of the most persistent issues in building speech-enabled language learning applications and suggest how they might be resolved to enable a more widespread commercial implementation of speech technology in CALL.

1. More research is necessary on modeling and predicting multi-turn dialogs.

An intelligent open response language tutor must not only correctly *recognize* a given speech input, but in addition *understand* what has been said and *evaluate* the meaning of the utterance for pragmatic appropriateness. Automatic speech understanding requires Natural Language Processing (NLP) capabilities, a technology for extracting grammatical, semantic, and pragmatic information from written or spoken discourse. NLP has been successfully deployed in expert systems and information retrieval. One of the first voice-interactive dialog systems using NLP was the DARPA-sponsored Air Travel Information System (Pallett, 1995), which enables the user to obtain flight information and make ticket

reservations over the telephone. Similar commercial systems have been implemented for automatic retrieval of weather and restaurant information, virtual environments, and telephone auto-attendants. Many of the lessons learned in developing such systems can be valuable for designing CALL applications for practicing conversational skills.

2. More and better training data are needed to support basic research on modeling non-native conversational speech.

One of the most needed resources for developing open response conversational CALL applications is large corpora of non-native transcribed speech data, of both read and conversational speech. Since accents vary depending on the student's first language, separate databases must either be collected for each L1 subgroup, or a representative sample of speakers of different languages must be included in the database. Creating such databases is extremely labor and cost intensive--a phone level transcription of spontaneous conversational data can cost up to one dollar per phone. A number of multilingual conversational databases of telephone speech are publicly available through the Linguistic Data Consortium (LDC), including Switchboard (US English) and CALLHOME (English, Japanese, Spanish, Chinese, Arabic, German). Our own effort in collaboration with John Hopkins University (Byrne, Knodt, Khudanpur, & Bernstein, 1998; Knodt, Bernstein, & Todic,1998) has been to collect and model spontaneous English conversations between Hispanic natives. All of these efforts will improve our understanding of the disfluent speech of language learners and help model this speech type for the purpose of human-machine communication.

3. Authoring tools and API's must become more widely available and easier to use.

Speech recognition functionality is more likely to become a standard feature of CALL if it can be easily incorporated into language learning applications. The best way of accomplishing this is by using a standardized speech API. A speech API consists of a set of program modules that allow application developers to access the functionality of a speech decoder without the need for a full understanding of the underlying technology. Most vendors have their own private or publicized speech API's, and currently there are several competing "standards." Entropic has recently built a speech API specifically tailored to the needs of language educators. This API integrates a state-of-the-art speech recognizer and a set of high level programming routines with existing authoring environments for incorporating speech recognition into PC-based language training applications. Entropic's speech API provides a versatile development environment for educational software compatible with other multimedia authoring tools such as Java or Authorware. The API is easy to use and supports powerful functionality for CALL applications, including access to F0 information, timing, confidence scores, and automatic mapping to known non-native pronunciations.

4. As voice-interactive CALL systems become more widely accepted, the quality of commercial systems is likely to improve.

The lion's share of funding for CALL comes from government sources. Since these funds are modest and production costs high, most of the funded systems remain at an experimental stage; few have been tested with end users and fewer still have entered the commercial market. A common argument assumes that CALL will have a greater practical impact on learning, when more well-designed CALL applications are readily available. No doubt, we do need better and more thoroughly tested systems. Such systems must become more affordable, easy to install, and platform-independent. However, the increasing commercial success of telephone-based voice applications suggests that the technology will rapidly improve once it enters the commercial market on a larger scale. As a result, large amounts of user data will become available to augment and improve the technology even further.

ABOUT THE AUTHORS

Farzad Ehsani is the Chief Technology Officer at Sehda, Inc., a startup focusing on large vocabulary applications for dialogue and captioning. Previously, he was the Head of Language Systems at Entropic Research Laboratory. He holds a Bachelors in Computer Science and a Masters and an Engineers degree in Electrical Engineering all from MIT. Before joining Entropic, Mr. Ehsani worked at NEC, Motorola, MIT, and DEC as a researcher focusing on language modeling and speech recognition.

E-mail: farzad@sehda.com

Eva Knodt is the VP of Marketing at Sehda, Inc., a startup focusing on large vocabulary applications for dialogue and captioning. She holds a PhD in German literature from the University of Virginia, and has taught at Indiana and Stanford Universities. Previously, she was a researcher in Entropic Research Laboratory working in the areas of experimental design and protocol implementation, as well as building language and acoustic models for English and Japanese.

E-mail: knodt@sehda.com

REFERENCES

Abberton, E., & Fourcin, A. (1975). Visual feedback and the acquisition of intonation. In E. H. Lenneberg, & E. Lenneberg (Eds.), *Foundations of language development* (2nd ed., pp. 157-165). New York: Academic Press.

Anderson-Hsieh, J. (1994). Interpreting visual feedback on suprasegmentals in computer assisted pronunciation instruction. *The CALICO Journal*, *11*(4), 5-21.

Bernstein, J. (1997). *Automatic spoken language assessment by telephone* (Tech. Rep. No. 5-97), Menlo Park, CA: Entropic, Inc.

Bernstein, J., & Horacio, F. (1996). Speech recognition by computer. In N. Lass (Ed.), *Principles of experimental phonetics* (pp. 408-434). St. Louis: Mosby.

Bernstein, J., & Rtischev, D. (1991). A voice interactive language instruction system, *Proceedings of Eurospeech*, Sept., 981-983.

Bernstein, L., & Christian, B. (1996). For speech perceptions by humans or machines, three senses are better than one. *Proceedings of the International Conference on Spoken Language Processing*, Oct., 1477-1480.

Byrne, W., Knodt, E., Khudanpur, S., & Bernstein, J. (1998). Is automatic speech recognition ready for non-native speech? A data collection effort and initial experiments in modeling conversational Hispanic English, *Proceedings of Speech Technology in Language Learning,* May.

Chapelle, C. (1997). CALL in the year 2000: Still in search of research paradigms? *Language Learning & Technology*, *1*(1), 19-43. Retrieved February 1, 1998 from the World Wide Web: http://polyglot.cal.msu.edu/llt/vol1num1/chapelle/default.html.

Cowan, H., & Jones, B. (1991). Reaching students with reading problems. *Electronic Learning 11*(1), 36-38.

Crystal, D. (1981). *Clinical linguistics*. New York: Harper Press.

de Bot, K. (1983). Visual feedback of intonation: Effectiveness and induced practice behavior. *Language and Speech, 26*(4), 331-350.

Dunkel, P. (Ed.). (1991). *Computer-assisted language learning and testing: Research issues and practice.* Philadelphia: Penn State University Press.

Egan, K. (1996). Speech recognition application to language learning: Echos. *Proceedings of CALICO*, July.

Ehsani, F. (1996). *Air traffic control task for Japanese* (Tech. Rep. No. 7-96), Menlo Park, CA: Entropic, Inc.

Ehsani, F., Bernstein, J., Najmi, A., & Todic, O. (1997). Subarashii: Japanese interactive spoken language education. *Proceedings of Eurospeech,* Sept., 681-684.

Ehsani, F., Bernstein, J., & Najmi, A. (in press). An interactive dialog system for learning Japanese. *Speech Communication*.

Franco, H., Neumeyer, L., Kim, Y., & Ronen, O. (1997). Automatic pronunciation scoring for language instruction. *Proceedings of ICASSP*, April, 1471-1474.

Haskin, D. (1997, September 23). Voice recognition reaches new height with dragon naturally speaking. *PC Magazine*, *16*.

Higgins, J. (1988). *Language learners and computers: Human intelligence and artificial unintelligence*. Singapore: Longman Group.

Hiller, S., Rooney, E., Vaughan, R., Eckert, M., Laver, J., & Jack, M. (1994). An automated system for computer-aided pronunciation learning. *Computer Assisted Language Learning, 7*(1), 51-63.

Holland, M. (1995). The case for intelligent CALL. In M. Holland, J. D. Kaplan, & M. R. Sams (Eds.), *Intelligent language tutors: Theory shaping technology*. Mahwah, NJ: Lawrence Erlbaum Associates.

Hubbard, P. (1988). An integrated framework for CALL courseware evaluation. *CALICO Journal*, Dec., 51-72.

James, E. (1976). The acquisition of prosodic features of speech using a speech visualizer. *International Review of Applied Linguistics 14*, 227-243.

Jecker, D. (1998, March 10). Speech recognition - Performance tests. *PC Magazine*, *17*.

Kantrov, I. (1991). *Talking to computer: A prototype speech recognition system for early reading instruction* (Tech. Rep. No. 91-3). Newton, MA: Education Development Center.

Kawai, G., & Hirose, K. (1997). A CALL system using speech recognition to train the pronunciation of Japanese long vowels, the mora nasal and mora obstruents. *Proceedings of Eurospeech*, Sept., 657-660.

Kenning, M. M., & Kenning, M. J. (1990). *Computers and language learning: Current theory and practice*. London: Ellis Horwood.

Kim, Y., Franco, H., & Neumeyer, L. (1997). Automatic pronunciation scoring of specific phone segments for language instruction. *Proceedings of Eurospeech*, Sept., 645-648.

Knodt, E., Bernstein, J., & Todic, O. (1998). A protocol for collecting a corpus of spontaneous, conversational, Hispanic English. *Proceedings of the First International Conference on Language Resources and Evaluation,* May, 341-345.

Levinson, S., & Liberman, M. (1981). Speech recognition by computer. *Scientific American*, April, 64-76.

McCandless, M. (1992). *Word rejection for a literacy tutor*. Unpublished master's thesis, Massachusetts Institute of Technology, Cambridge.

Manuel, S. (1990). *Speech communication report* (Progress Report No. 133, pp. 289-299). Cambridge, MA: Massachusetts Institute of Technology, The Research Laboratory of Electronics.

Markham, D., & Nagano-Madesen, Y. (1996). *Proceeding of the International Conference on Spoken Language Processing*, Oct., 1473-1476.

Molholt, G. (1990). Spectrographic analysis and patterns in pronunciation. *Computers and the Humanities, 24*, 81-92.

Mostow, J. (1997). Collaborative research on learning technologies: An automated reading assistant that listens. *Proceedings of the National Science Foundation Interactive Systems Grantees Workshop*, Aug.

Mostow, J., Roth, S., Hauptmann, A. G., & Kane, M. (1994). A prototype reading coach that listens. *Proceedings of the 12th National Conference on Artificial Intelligence*, Aug., 785-792.

Mostow, J., & Aist, G. (1997). The sounds of silence: Towards automated evaluation of student learning in a reading tutor that listens. *Proceedings of the 14th National Conference on Artificial Intelligence,* July, 355-361.

Neumeyer, L., Sankar, A., & Digalakis, V. (1995). A comparative study of speaker adaptation techniques. *Proceedings of Eurospeech*, Sept., 1127-1130.

Neumeyer, L., Franco, H., Weintraub, M., & Price, P. (1996). Automatic text-independent pronunciation scoring of foreign language student speech. *International Conference on Spoken Language Processing*, Sept., 1457-1460.

Ng, K. L. E., & Olivier, W. (1987). Computer-assisted language learning: An investigation on some design and implementation issues. *System*, *15*(1), 1-17.

Pallet, D., Fiscus, J., Fisher, W., Garofolo, J., Lund, B., Martin, A., & Przybocki, M. (1994). 1993 benchmark tests for the arpa spoken language program. *Proceedings of the Human Language Technology Workshop* (pp. 49-74). San Francisco: Morgan Kaufmann Publishers, Inc.

Pallet, D., Fiscus, J., Fisher, W., Garofolo, J., Lund, B., Martin, A., & Przybocki, M. (1995). 1994 benchmark tests for the arpa spoken language program. *Proceedings of the Human Language Technology Workshop* (pp. 5-36). San Francisco: Morgan Kaufmann Publishers, Inc.

Phillips, M., Zue, V., & McCandless, M. (1993). Language tutor: An interactive aid for teaching English and Japanese. In V. Zue (Ed.), *Annual research summary* (pp. 29-30). Cambridge MA: Spoken Language Systems Group, MIT LCS.

Rypa, M. (1996). VILTS: The voice interactive language training system. *Proceedings of CALICO*, July.

Salaberry, M. R. (1996). A theoretical foundation for the development of pedagogical tasks in computer mediated communication. *CALICO Journal, 14*(1), 5-34.

Spaai, G. & Hermes, D. (1993). A visual display for teaching intonation. *CALICO Journal 10*(3), 19-30.

Stibbard, R. (1996). Teaching English intonation with a visual display of fundamental frequency. *The Internet TESOL Journal, 2*(8). Retrieved February 1, 1998 from the World Wide Web: http://www.aitech.ac.jp/~iteslj/Articles/Stibbard-Intonation.

Warschauer, M. (1996). Computer-assisted language learning: An introduction. In S. Fotos (Ed.), *Multimedia language teaching* (pp. 3-20). Logos International: Tokyo.

Waters, R. (1994). *The audio interactive tutor* (Tech. Rep. No. 94-04). Cambridge Research Center, MA: Mitsubishi Electronic Research Laboratories.

Waters, R. (1995). The audio interactive tutor. *Computer Assisted Language Learning, 8*(4), 325-354.

Weinstein, C., McCandless, S., Mondshein, L., & Zue, V. (1975). A system for acoustic-phonetic analysis of continuous speech. *IEEE trans. Acoust. Speech and Signal Processing, 23*(1), 54-67.

Witt, S., & Young, S. (1997). Language learning based on non-native speech recognition. *Proceedings of Eurospeech*, Sept., 633-636.

Woodland, P., Pye, D., & Gales, M. (1996). Iterative unsupervised adaptation using maximum likelihood linear regression. *Proceedings of the International Conference on Spoken Language Processing*, Oct., 1133-1136.

Young, S. (1996). A review of large-vocabulary continuous speech recognition. *IEEE Signal Processing Magazine*, Sept., 45-57.

Zavaliagkos, G., Schwartz, R., McDonough, J., & Makhoul, J. (1995). Adaptation algorithms for large scale hmm recognizers. *Proceedings of Eurospeech*, Sept., 1131-1134.