SE-BERT: A SPEECH-ENABLED BEGINNING READING TUTOR

by

AMANDA ELIZABETH JONES

A THESIS

Presented to the Department of Computer and
Information Science
and the Honors College of the University of Oregon
in partial fulfillment of the requirements
for the degree of
Bachelor of Science

June 2000

An Abstract of the Thesis of

Amanda Elizabeth Jones for the degree of Bachelor of Science

In the Department of Computer and Information Science to be taken June 2000

Title: SE-BERT: A SPEECH-ENABLED BEGINNING READING TUTOR

Approved: [redacted]
Dr. Zary Segall

This thesis will discuss the design and initial development of SE-BERT, a computer program that uses speech recognition technology to teach beginning reading skills to young schoolchildren. Developed in conjunction with the Oregon Center for Applied Science, the project has been awarded a federal grant of $100,000, and will undergo continued development over the next year.

This document will also discuss the educational needs addressed by this system and the reasons motivating its development, explain how speech recognition technology works, examine the literature related to children's speech recognition, and describe the program's design and technical implementation.

TABLE OF CONTENTS

INTRODUCTION

This project is a response to a solicitation by the National Institute for Child Health and Development for programs that address reading deficiency in children. The program has been under development since 1998, in conjunction with my employer, Oregon Center for Applied Science (ORCAS), a producer of educational and health-related multimedia software.

At the inception of the project, there was a great deal of research information and tools available to assist in the development of speech recognition applications tailored to adult speakers. However, there was considerably less material available in support of products tailored to children's speech. In particular, the commercially-available speech engines that were investigated in the early stages of this project had a significant limitation: the engine of every system was trained on adult speech and was therefore inappropriate for use in this project.

This presented a problem. In developing speech-based systems, it is important, if not critical, to train the engine with speech that closely resembles the speech of the target users. This applies not only to the age of the target users – such factors as accent, dialect and even gender can negatively affect the recognition performance if the engine has not been exposed to similar speech samples in the training process.

The original plan to overcome this obstacle involved purchasing a $12,000 speech processing system from Entropic; this would have necessitated recording and transcribing hours of children's speech data, and then retraining the engine with the collected samples. However, thanks to a toolkit developed at Oregon Graduate Institute's Center for Speech

and Language Understanding (CSLU), engine retraining is no longer necessary. The CSLU toolkit includes two recognition engines, one of which has already been trained on children's speech collected from Portland schoolchildren. Furthermore, the user-friendly software development environment included in the toolkit has simplified the prototype implementation.

REASONS FOR DEVELOPING SE-BERT

Educational Needs Addressed by the Project

According to educational researchers, unresolved deficiency in reading skills in the early school years often leads to future academic failure (Noell 1993). Children who lack reading proficiency experience increasing difficulty keeping pace with their peers, and will likely fall behind in later academic pursuits unless the problem is addressed early on. Persistent academic failure, an obvious consequence of reading impairment, has been linked to such problems as increased incidence of negative behavior, delinquency, substance abuse, and eventual dropout (Patterson 1992, 1989; Hawkins 1992; Becker 1986; qtd. in Noell 1993).

Unfortunately, reading impairment exists at an unacceptable level among American schoolchildren. As recently as 1993, between 25% and 40% of the fourth-, eighth- and twelfth-graders tested as part of the National Assessment of Educational Progress performed below the lowest level. This was defined as "[demonstrating] partial mastery of the knowledge and skills fundamental for proficient work at each grade" (Mullis, Campbell & Farstrup 1993, qtd. in Noell 1993). As school funding has continued to decrease nationwide (Larson 1992, qtd. in Noell 1993), many teachers have been left with fewer resources and limited personnel, while the demands on their time and energy have simultaneously increased (Kauffman, Gerber, & Semmel 1998, qtd. in Noell 1993).

The Case for Speech Recognition-based Reading Instruction

At the same time, computers have become increasingly prevalent in American homes and classrooms over the past five years. Recent advances in multimedia technology have permitted the development of an abundance of educational software titles. Often integrating audio, video, text, and animation into self-paced interactive learning environments, these packages present material on everything from basic language skills to mathematics and science fundamentals (Ehsani & Knodt 1998). While programs of this nature certainly have the potential to supplement classroom learning and present opportunities for extra practice, reading programs that lack the ability to analyze the student's speech and respond appropriately are necessarily limited in their educational utility.

For this reason, "listening and responding to children [as they] read aloud has been the holy grail of technology-based literacy instruction," according to Peter Fairweather, senior manager of the Education Solutions group at IBM's Watson Research Center (Benedek 1997). While the small number of speech-enabled reading programs on the market are capable, to an extent, of "listening and responding" to the user's speech, most are not based on proven educational models. Furthermore, a more serious shortcoming among these programs is that they fail to teach basic reading skills, assuming instead that the users are already proficient readers that could benefit from additional practice. This completely overlooks the needs of children who have not yet mastered basic phonemic, graphemic, and word recognition tasks, all of which are essential in order for reading practice to be of any benefit to the user (Noell 1993). As

will be discussed later, the reading instruction workbook used in SE-BERT directly addresses this need; the curriculum is based on Direct Instruction, an empirically-verified educational technique (Adams 1996).

AN INTRODUCTION TO SPEECH RECOGNITION TECHNOLOGY

An understanding of how speech recognition technology works will help the reader appreciate the challenges this project presents. The following section is based on information presented in Ehsani & Knodt (1998) and Brewer (1996).

An Overview of Speech Recognition

According to Bernstein & Franco (1996, qtd. in Ehsani & Knodt 1998), humans and machines process speech in fundamentally different ways. Complex cognitive processes account for humans' ability to associate incoming acoustic signals with meanings and intentions; computers, however, represent acoustic signals as a series of binary values (0 or 1), no different from any other type of digital information. Furthermore, computers use algorithms to perform analysis and lookup functions to "recognize" phonemes and words in a stream of incoming speech data. Digital speech recognition systems are limited in several other ways: for one, most systems are limited only to audio processing, which prevents taking advantage of the rich array of non-verbal cues transmitted in face-to-face human communication. Another significant limitation is the static, finite nature of the vocabulary tables in speech systems – in most cases, the engine is able to recognize only those words that are programmed into the system at the time of deployment.

Despite these differences, both humans and computers face the same basic task when recognizing speech: to find the word string that best matches what was spoken, in

order to decode the speaker's intended message. Speech recognition technology attempts to simulate and optimize this process computationally.

Hidden Markov Models in Speech Recognition Technology

A number of technical approaches to speech recognition have been proposed and implemented since the early 1970s, but the most successful technique in use today is Hidden Markov Modeling. Hidden Markov Modeling involves constructing a statistical model of a process that includes transitions from one state to another. The goal of this modeling process is to predict future states given a set of previous state information.

An informative example of HMMs involves a simple three-state weather system, depicted in Figure 1:

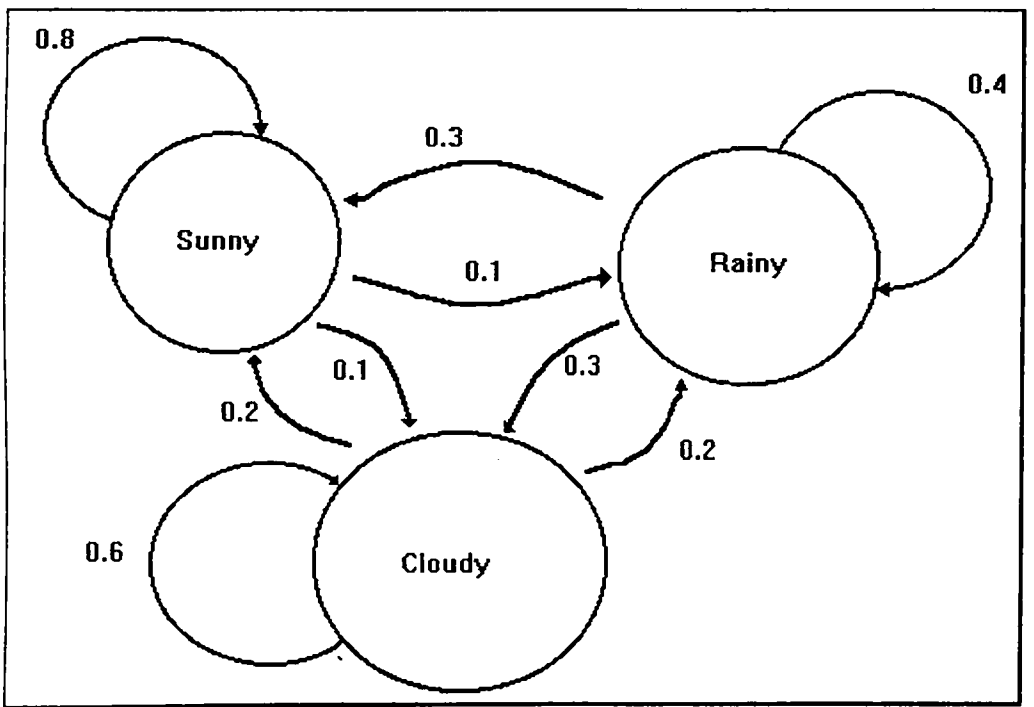

Fig. 1: A simple Markov model for weather patterns. (Juang, Perdue, & Thomson 1995, qtd. in Brewer 1996).

In Figure 1, the labeled circles represent discrete weather states ("Sunny," "Cloudy," and "Rainy"), and the arrows leaving each circle represent the probability of the next day's weather changing to the indicated state (or remaining unchanged). For example, if today's weather is cloudy, this model would predict a 60% chance that tomorrow will also be cloudy, a 20% chance of rain, and a 20% chance that the sun will shine.

Given the current state of the weather and a model with the state-change probabilities, it is a fairly straightforward process to predict the most likely weather state for the following day. (The difficult part, of course, comes with calculating the probabilities themselves.) This weather system is an example of a Markov Model, in which the current state information is known in advance and used in generating the prediction.

By contrast, a Hidden Markov Model attempts to predict future states, but the information on the current state is partial or inconclusive. For example, consider the task of predicting the following day's weather using Figure 1, given only the temperature and humidity information for the current day (that is, without a conclusive label for the current state). In this situation, the predictions take on an added dimension of uncertainty. Not only is it impossible to make a completely accurate prediction for the following day, but the current state is also reduced to a "best guess" based on external information about the typical temperature and humidity characteristics of the individual states (some of which may even overlap).

This process is similar to the tasks a speech engine must perform to accurately identify the patterns that represent particular phonemes and words in the input signal.

The recognition engine may not have conclusive data on the current phoneme or word being pronounced, but the input signal provides acoustic data that describes it. In both weather prediction and speech recognition, a Markov Model, coupled with additional knowledge about the states' characteristics, can lead to a fairly reliable estimate of the actual input sequence of state changes. For this reason, speech recognition systems that take advantage of this mathematical modeling technique are among the most accurate on the market (Juang, Perdue, & Thompson 1995, qtd. in Brewer 1996).

Speech Recognition System Design

A typical HMM-based speech recognition engine consists of five basic components (Ehsani & Knodt 1998):

1. An acoustic signal analyzer that generates the spectral representation of an input signal. The acoustic signal analyzer breaks down a complex sound and measures the sound level at each frequency.

2. A set of HMMs trained on large amounts of representative speech data. This enables the system to make "educated" guesses about the most likely phoneme to occur next in the input stream.

3. A hard-coded lexicon, which is a set of rules for accurately converting sub-word phoneme sequences into words.

4. A statistical language model or grammar network, which specifies valid word sequences at the sentence level.

5. A decoded, which is a search algorithm for computing the "best match" between a spoken utterance and its textual representation.

Fig. 2 shows three representations of an input signal. The first is the English text for the phrase, "He was shot in the back." This is followed by a "waveform," which is

simply a graphical representation of the amplitude (or volume) over time. The last frame is a "spectrogram," a graph of the sound intensity at each frequency over time (Fig. 2):

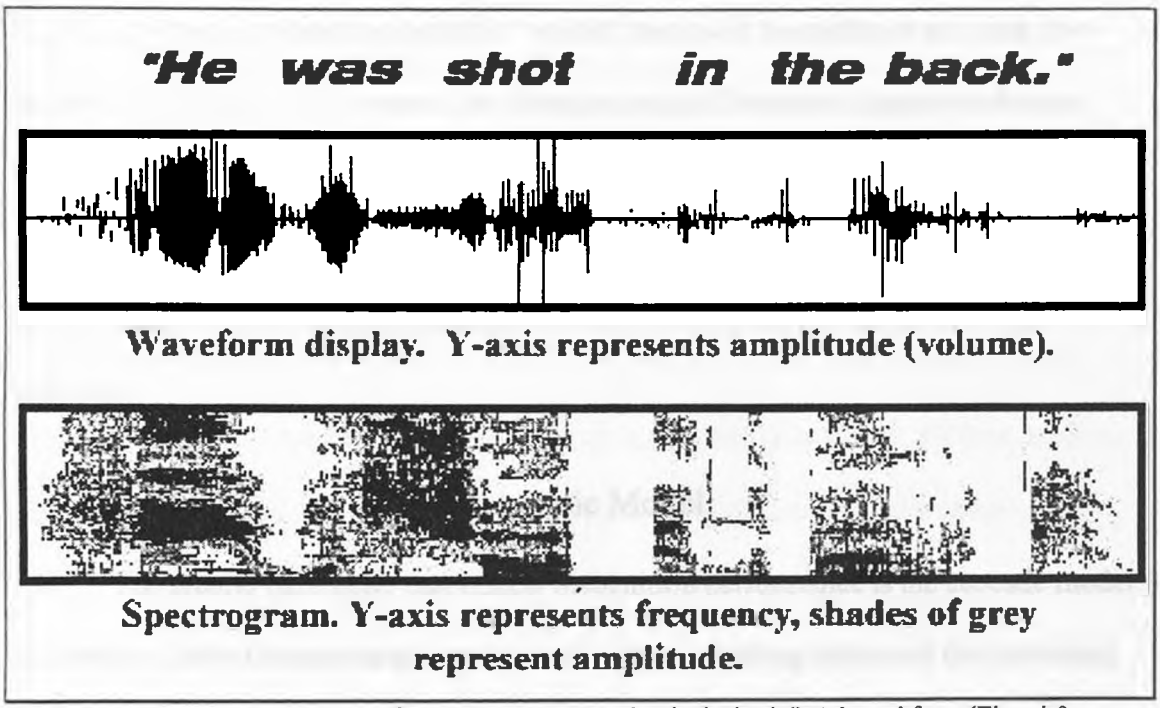

Fig. 2: Waveform and spectrogram of the phrase, "He was shot in the back." Adapted from (Ehsani & Knodt 1998).

Performance Parameters in Speech Recognition Systems

There are three main parameters that define the type of speech a recognition engine is programmed to analyze. These parameters – the task definition, acoustic models, and input modality – all have a significant impact on the accuracy of a given system (Brewer 1996, Ehsani & Knodt 1998).

Task Definition

"Task definition" refers to both the vocabulary size (the number of words the engine has been taught to recognize) and the perplexity of the grammar network. To illustrate these concepts, compare an automatic telephone dialing system to a 40,000-

word dictation engine. The dialing system is an example of a small-vocabulary low-perplexity system; the input is constrained to a fairly specific format (either seven or ten digits), and there are only ten possible "words" that could be spoken at any time (the names of the digits). By contrast, the dictation engine illustrates a large-vocabulary, high-perplexity system. The system is trained on thousands of words, many of which are valid candidates for the next word at any given point in the input stream. Most programs of this nature require complex grammar networks to help the recognizer determine what was said.

Acoustic Model

The second parameter that affects recognition performance is the acoustic model. This refers to how the system adapts to users' unique speaking styles and the individual qualities of their voices. There are three common models: speaker-dependent, speaker-adaptive, and speaker-independent; each of the three has its strengths and weaknesses. A speaker-dependent engine can only be used by a single user, who must train the recognizer on every word in the vocabulary. This technique results in much higher accuracy compared to the other acoustic models, but it renders the engine almost unusable for other users. In addition, the intensive training process can be tedious for the user.

Speaker-independent engines can "recognize the speech patterns of a large class of people, such as speakers of American English" (Brewer 1996). This can be difficult to implement, because pronunciation style can vary dramatically from person to person. Additionally, the error rate is usually higher on speaker-independent engines. A speaker-adaptive engine is a compromise between the other two models. The training process

begins with a set of speaker-independent models and adjusts them to more closely match the speaking patterns of the individual user.

Input Modality

The input modality of a speech engine refers to the flow of input speech. Older engines that were limited to "discrete recognition" required the user to pause briefly after each word. Modern dictation engines are theoretically capable of "continuous recognition," which does not require pauses to distinguish the ends of words. Users, however, must still speak slowly and enunciate words clearly to obtain the best accuracy. Discrete recognition tends to result in much higher accuracy, since it eliminates a great deal of ambiguity in the input signal; for this reason, this input modality is a good choice when accuracy is critical and when the system design allows for it.

LITERATURE REVIEW

The literature review conducted for this paper included journal articles, newsgroup discussions, web documents, and reviews of existing software programs that claim to teach children to read using speech recognition. The purpose of the review was to find information to help determine whether the proposed system was technically feasible.

Engine Training Issues

Commercial speech recognition products are generally trained on adult speech only, which results in extremely poor performance when using the engine for children's speech (Aist et al. 1998; Cole et al. 1998; Das, Nix & Picheny 1998; Lacayo 1997; Miller 1996; Wilpon & Jacobsen 1996). The length of the human vocal tract increases throughout childhood, with development peaking around puberty for males and continuing more gradually for females (Wilpon 1996). As this increase in length occurs, important speech parameters such as formant frequencies tend to change dramatically (Das, Nix & Picheny 1998). Because speech recognition engines rely on spectral analysis to distill important acoustic features from the input signal, it is important that the base frequencies match the samples on which the recognizer has been trained. The solution for this application is to train the engine with samples of children's speech.

Training a speech engine, however, is a non-trivial task; it requires recording and phonetically transcribing a large amount of speech data. The cost of obtaining and transcribing this data can be prohibitively high even for adult speech, and compiling a

similar database of children's speech can cost several times as much (Aist et al. 1998), since children's pronunciations are often atypical, which complicates the transcription process. Additionally, the collection process itself can often be automated when dealing with adults, but children require constant supervision and prompting during the recording process. My own efforts at gathering and transcribing a relatively small amount of speech data from kindergartners and first- and second-graders at a local elementary school (for the purpose of engine evaluation, not training) confirmed these assertions.

Additional Techniques for Improving Performance

In addition to engine retraining, there are a number of other techniques that have been used to try to improve accuracy for children's speech. These include frequency warping and collecting language model data from children. Frequency warping involves applying a remapping function to transpose and/or stretch the frequency data in the original input signal, with the goal of reducing the mismatch between the engine's model and the input data. Collecting language model data consists of conducting an analysis of children's natural speech patterns as well as the speech data itself; this can help the engine handle the higher rate of disfluencies often present in children's speech. Both of these techniques are shown to substantially reduce error rates for children's speech recognition in (Das, Nix & Picheny 1998). Two other techniques include separate training sets based on gender and age (Wilpon & Jacobsen 1996, Aist et al. 1998), and training with automatically-collected data (Aist et al. 1998), both of which have had positive results.

Although most studies involving speech recognition accuracy include error percentages, such statistics can often be misleading. Recognition accuracy is highly dependent on a number of factors, such as the underlying engine, number of speakers, training procedure, number of words, actual words, and even sound quality of the input data. Although the results are internally valid since each study compared the error rate after changing only one of those variables, nearly all of these factors differed from study to study; this invalidates any comparison of accuracy rates across studies.

Existing Reading Instruction Programs Using Speech Recognition

While researching this project, three similar reading programs have been discovered: Carnegie Mellon's Project LISTEN, IBM's Watch Me Read, and Edmark Corporation's Let's Go Read! I and II. These programs all "listen to children read" and offer limited feedback to the user, although they vary in task complexity and functionality. Project LISTEN is the most ambitious of the three; it has been under development since 1993, and several prototypes have been pilot-tested at inner city elementary schools in Pittsburgh. The other two programs, Watch Me Read and Let's Go Read! are both commercial ventures based on IBM's ViaVoice recognition engine (Benedek 1997, Edmark Corp. 1998). All three of these programs share an important limitation: they offer an opportunity for extra practice to children with existing reading abilities, but do not address the needs of children most at risk of growing up illiterate – those who lack basic reading skills.

Watch Me Read

Developed at IBM's Watson Research Center, Watch Me Read is an interactive program that claims to "help children learn to read by listening to them as they read aloud." According to Benedek (1997), the interface was designed by an experienced video game developer. An animated panda reads a story to the child, and when the panda reaches a target sentence, it first reads the sentence in its entirety to the child, then repeats it with one word missing. The child is then prompted to read the missing word; if the child mispronounces it, the panda prompts the child to read it again.

The program is designed to adjust to the child's perceived reading level: "If the child knows most of the words, the panda hangs back and lets the child read. If the child is hesitant, the panda reads most of the text, leaving only a few words for the child" (Benedek 1997). The program also records the story as the panda and the child read it. At the end of the story, the recording is played back in the form of a performance, complete with the raising of a theatrical curtain and applause.

Although this program claims to "help kids learn to read," it is beneficial only to those children who are reasonably proficient readers. The system actually rewards poor readers by making them read less, when they probably need the practice more than their already-capable peers do.

Project LISTEN

Project LISTEN is described as "a reading coach that listens" and responds to children as they practice reading (Mostow et al. 1993). In a standard session with the Reading Tutor, a sentence from a pre-selected text is displayed on the screen, and the system "listens" for missed words. It also listens for requests for help, and provides

assistance when needed. In addition to the ability to "listen" to children's speech, it also incorporates synthesized speech to generate responses to the reader.

Project LISTEN has been prototyped and field-tested over the last eight years, and many extensions have been added during this time (Mostow & Aist 1998). Originally, text selections were culled from the Weekly Reader, a popular children's magazine, the system now allows children to author stories for their peers to read (Aist et al. 1998). Another recent addition is the ability to provide backchanneling feedback (Mostow & Aist 1998). This feature was added in an effort to model more closely the corrective activities of human reading tutors.

Despite this attempt to emulate effective tutoring techniques, this program shares the central limitation of Watch Me Read. Although it claims to target problem readers who are at a high risk for illiteracy, the program fails to offer opportunities for users to improve basic reading skills. It also targets older readers as opposed to those in kindergarten through second grade, the target group for SE-BERT.

Let's Go Read!

Let's Go Read! uses the same underlying speech recognition architecture as Watch Me Read, but it incorporates more basic learning activities, such as differentiating vowel sounds (Edmark Corp. 1998). The program literature claims that the curriculum combines "the best of the Whole Language and Phonics methodologies," two teaching techniques that differ dramatically in their approaches to reading instruction (in fact, some educators consider these methodologies to be mutually exclusive). Furthermore, the program lacks other beginning-reading activities that are included in SE-BERT, such as sounding out words.

THE IMPLEMENTATION OF SE-BERT

The ability of a speech engine to recognize sounded-out words and phonemes is an extremely specific task, but one that is central to the implementation of the reading program in question. Because dictation engines are trained to recognize specific words spoken normally, a more versatile speech package is called for. The speech toolkit developed at Oregon Graduate Institute's Center for Spoken Language Understanding (CSLU) has several unique features that led to the decision to use it to develop SE-BERT.

The CSLU Toolkit

According to OGI's website, "the CSLU Toolkit is a huge repository of software tools and ingredients that can be used to create spoken language interfaces." It is built in C and Tcl/Tk, and includes a user-friendly graphical Rapid Application Development environment (RAD). The modules in RAD support functionality such as branching, looping, and presenting visual aids in sync with the spoken prompts, as well as a host of other specific features required by SE-BERT. Additionally, the toolkit also includes a realistic three-dimensional animated talking head ("Baldi") that "speaks" the synthesized and recorded voice prompts used in the program. Most importantly, the latest release includes a speech recognition engine trained on children's speech, eliminating the need for collecting a large volume of children's speech data and for retraining the recognition engine.

Features of the CSLU Toolkit

There are several features included in the CSLU toolkit that have significantly impacted the design and implementation of SE-BERT. These fall into three groups: features bundled in the RAD environment that support the desired program design, aspects of the speech engine that make it particularly suitable to the recognition tasks at hand, and other additional properties that support specific requirements of the program or enhance it in some way.

RAD is a graphical object-oriented environment that allows developers to build speech systems by arranging and graphically connecting nodes that represent speech modules (Fig. 3):

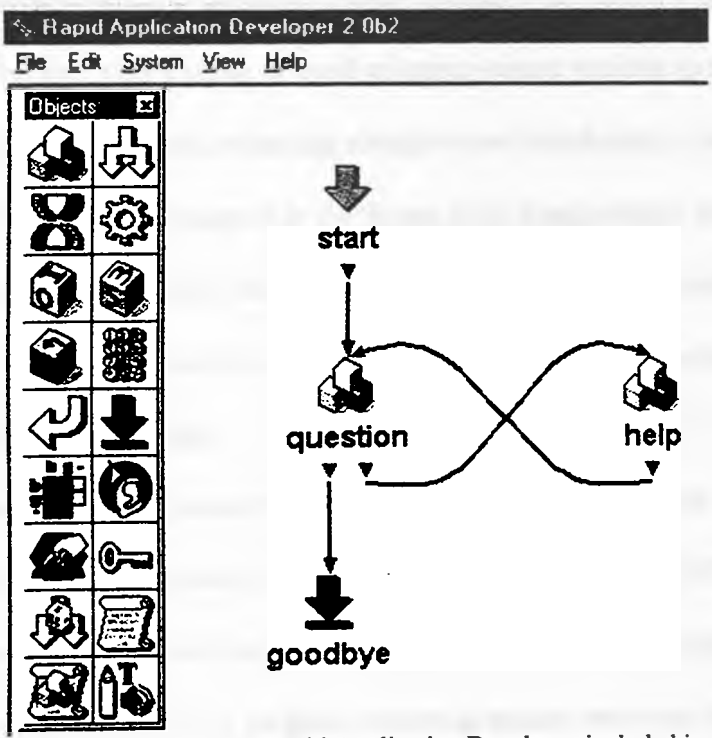

Fig. 3: Screenshot of the Rapid Application Developer included in the CSLU Toolkit. (Courtesy of Oregon Graduate Institute's web page)

There are several features included in the RAD environment that address specific design issues central to the implementation of SE-BERT. The written program is designed in such a way that the user is presented with a spoken sample to model the correct pronunciation of a single phoneme, a simple word sounded out slowly, or a similar speech prompt. The user is then prompted to repeat the spoken example; if an error is made, the program gives specific feedback according to the nature of the error. For example, it is possible to distinguish whether the child substituted an incorrect letter when speaking a word, or paused in between the letters during a sounding-out task, or even if they read the word slowly instead of saying it fast. The branching functionality included in each RAD speech object allows specific corrections to be loaded at each step of the lesson. Furthermore, it is possible to loop through a correction sequence repeatedly until the user gets it right, or until an upper-bound variable is reached. This prevents users from getting stuck repeating a single word indefinitely. One additional feature that will be taken advantage of in the future is the Login object, which allows for the creation of individual profiles, data sets and even automatically-recorded speech files from each user. This will be used to evaluate the system's performance when it is deployed for testing in September.

In addition to features present in the RAD environment, there are several aspects of the recognition engine that make it particularly well-suited to SE-BERT. The inclusion of an engine pre-trained on children's speech is perhaps the most important of all. Also important is the ability to program custom grammar networks; this is central to the implementation of SE-BERT because it enables careful customization of the recognition at every step in the program. For example, each step in the lessons involves

listening for a specific sound or word, and there are common mistakes that most children tend to make when learning to read. The ability to program the grammar network makes it possible to program the recognizer to listen for those specific mispronunciations, in addition to rejecting any other erroneous answers from the user. Finally, the system's Dynamic Rejection Adjustment parameter makes the system responsive to the error rate. Depending on the configuration, it is possible to dynamically adjust the system to be less or more strict by a fixed amount after a given number of consecutive misrecognitions (instances where the system cannot confidently determine what was said) or correct recognitions (which indicates the system may not be catching mispronunciations).

Other useful functionality includes the ability to measure and branch according to the input duration, the ability to override and extend the set of accepted pronunciations, and BaldiSync, the utility that enables synchronizing the animated face to recorded natural speech prompts. The input duration can indicate whether the user spoke the target speech at the correct speed and whether they paused inappropriately; this is important for proper user error correction. The ability to set alternate pronunciation models will allow the program to be customized for different dialects across the country. Finally, the use of BaldiSync will allow the inclusion of both recorded and synthesized voice prompts, to be selected by the user.

Curriculum Presented by SE-BERT

The program is divided into 100 "lessons," each of which consists of 6 or more "tasks" for the student. Each task focuses on a single activity, such as learning a group of sounds, slowly sounding out words that use the new sounds, and speaking the words quickly, without pausing between the individual phonemes.

The written lessons are geared toward parents who are home-schooling their children (Engelmann 1983), but a computerized adaptation of the program would be effective in either a home or classroom environment. It is based on Direct Instruction, an empirically-based instructional theory developed by Engelmann in 1964 (Adams 1996) that places great emphasis on repetition and standardized delivery of the material. Human teachers who use DI programs in classrooms are required to undergo training and follow pre-defined scripts. Using a computer to deliver the lessons would prevent any deviations; this will assist in evaluating the educational effectiveness of the program when it is eventually tested in schools.

Future Plans

SE-BERT is currently under development, and classroom tests are planned for September 2000. For Phase I of the NICHD grant, the first ten lessons of the written program will be converted to use speech recognition. In December, another grant application will be submitted for the Phase II portion of the program. If the second phase funding is received (approximately $975,000), the other ninety lessons will be developed as well.

GLOSSARY

Acoustic – Relating to the sense or organs of hearing, to sound, or to the science of sounds.

Acoustic Model – The method used by the recognition system to adapt to differences in individual users' voices.

Algorithm – A step-by-step procedure for solving a problem or for accomplishing some end, especially by a computer.

Amplitude – The intensity of a signal. In the case of speech signals, this would be perceived as the volume.

Backchanneling – Sub-word vocalizations that listeners tend to produce while another person is speaking ("mmm," "mm-hmmm," and "uh-huh" are typical examples).

Continuous Recognition – A speech engine which does not require pauses to distinguish the ends of words.

Decoder – A search algorithm for computing the best match between a spoken utterance and its textual representation

Discrete Recognition – A speech engine which requires users to pause between words.

Disfluency – Any spontaneous self-repair that interrupts the smooth flow of an otherwise coherent sentence or individual word (Oviatt 1996).

Frequency – The number of oscillations of energy per second in a sound wave.

Garbage Threshold – The cutoff point for rejecting words that do not appear to match any entries in the recognition vocabulary.

Grammar Network – A programmed recognition network that specifies valid word sequences at the sentence level.

Grapheme – All of the letters and letter combinations that represent a single phoneme, as *f*, *ph*, and *gh* for the phoneme /f/.

Input Modality – The flow of input speech to the speech recognition system. Either discrete or continuous.

Lexicon – The system dictionary, or set of rules for accurately converting sub-word phoneme sequences into words.

Phoneme – Any of the abstract units of the phonetic system of a language that correspond to a set of similar speech sounds that are perceived to be a single distinctive sound in the language.

Speaker-Adaptive – A speaker-independent engine that adapts to the individual speech characteristics of one user.

Speaker-Dependent – A speech engine that is customized to recognize the speech of a single user.

Speaker-Independent – A speech engine trained on the speech of a generic set of users, such as adult native speakers.

Spectral Analysis – The process of analyzing a spectrogram.

Spectrogram – The graph obtained by analyzing a sound waveform into its frequency components and plotting the result. The X-axis often represents time and the Y-axis represents the frequency; the shading density or color indicates the intensity.

Statistical language model – See "Grammar Network".

Waveform – A graphic representation of the shape of a sound wave that indicates the wave's amplitude (Y-axis) over time(X-axis).

Works Cited

Adams, G., and S. Engelmann. "Research on Direct Instruction: 25 Years Beyond DISTAR." Educational Achievement Systems, Seattle, WA. 1996.

Aist, G., Chan, P., Huang, X., Jiang, L., Kennedy, R., Latimer, D., Mostow, J., and C. Yeung. "Automatic Data Collection and Acoustic Model Training for Children's Speech." 1998. http://www.cs.cmu.edu/~listen

------. "How Effective is Unsupervised Data Collection for Children's Speech Recognition?" Proceedings for the upcoming International Conference on Speech and Language Processing (*ICSLP98*). December 1998. http://www.cs.cmu.edu/~aist/icslp1998-acoustic-training.doc

Brewer, D. "A Brief Survey of Speech Recognition." Lost website; formerly at http://www.linfield.edu/~dbrewer. Accessed Sept. 1998; last updated May 1996.

Benedek, E. "Watch Me Read." IBM Research Magazine, IBM Corporation, 1997. http://www.research.ibm.com/resources/magazine/1997/issue_4/solutions497.html

Das S., Nix D., and M. Picheny. "Improvements in Children's Speech Recognition Performance." Proceedings of the 1998 International Conference on Acoustics, Speech and Signal Processing, 1998.

Edmark. "Let's Go Read! An Island Adventure." Edmark Corporation. http://www.edmark.com/prod/lgr/island

Ehsani, F., and E. Knodt. "Speech Technology in Computer-Aided Language Learning: Strengths and Limitations of a New CALL Paradigm." Language Learning & Technology, Vol. 2, No. 1, July 1998. http://polyglot.cal.msu.edu/llt/vol2num1/article3/index.html

Lacayo, M. "Voice Interfaces for Children in the Elementary School Years." Proceedings of the Workshop on Perceptual User Interfaces (PUI), 1997. http://research.microsoft.com/vision/puiworkshop97/%5Fpapers/p054.txt

Mostow, J., Hauptmann, A. G., Chase, L. L., and S. Roth. "Towards a Reading Coach that Listens: Automated Detection of Oral Reading Errors," Proceedings of the Eleventh National Conference on Artificial Intelligence (*AAAI93*), Washington, DC, American Association for Artificial Intelligence, 1993.

Mostow, J. and G. Aist. "Project LISTEN: A Reading Tutor that Listens." CHIKids 1999 Technology Workout submission. http://www.cs.cmu.edu/~aist/CHI99-CHIkids.doc

Noell, J. "Research Plan." (Unpublished grant application). Oregon Center for Applied Science, 1993.

Oviatt, S. L. "User-Centered Modeling for Spoken Language and Multimodal Interfaces." IEEE Multimedia 4(3), 1996. http://www.cse.ogi.edu/~lchen/SharonPaper/Ieee/Ieee.html

Wilpon J., and C. N. Jacobsen. "A Study of Speech Recognition for Children and the Elderly." Proceedings of the 1996 International Conference on Acoustics, Speech and Signal Processing, 1996.