Towards a Reading Coach that Listens: Automated Detection of Oral Reading Errors

Jack Mostow, Alexander G. Hauptmann, Lin Lawrance Chase, and Steven Roth

Project LISTEN, CMT-UCC 215, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213-3891
mostow@cs.cmu.edu

Abstract[1]

What skill is more important to teach than reading? Unfortunately, millions of Americans cannot read. Although a large body of educational software exists to help teach reading, its inability to hear the student limits what it can do.

This paper reports a significant step toward using automatic speech recognition to help children learn to read: an implemented system that displays a text, follows as a student reads it aloud, and automatically identifies which words he or she missed. We describe how the system works, and evaluate its performance on a corpus of second graders' oral reading that we have recorded and transcribed.

1. Introduction

Deficiency in reading comprehension has become a critical national problem; workplace illiteracy costs over $225 billion dollars a year (Herrick, 1990) in corporate retraining, industrial accidents, and reduced competitiveness. Although intelligent tutoring systems might help, their inability to see or hear students limits their effectiveness in diagnosing and remediating deficits in comprehension.

In an attempt to address this fundamental limitation, we are building on recent advances in automated speech processing, reading research, and high-speed computing. We have dubbed this effort Project LISTEN (for "Language Instruction that Speech Technology ENables"). This paper reports our initial results.

To place these results in context, imagine how automated speech recognition may eventually be used in an interactive system for assisting oral reading. The system displays text on the screen and listens while the student (a child, illiterate adult, or foreign speaker) reads it aloud. When the student gets stuck or makes a serious mistake, the system intervenes with the sort of assistance a parent or teacher might provide, such as saying the word, giving a hint, or explaining unfamiliar vocabulary. Afterwards, it reviews the passages where the student had difficulty, giving an opportunity to reread them, and providing appropriate feedback.

This paper reports on a scaled-down version of such a system. Along the way it points out some current limitations and directions for future improvement.

2. What Evelyn does (and doesn't)

Our implemented prototype, named Evelyn, displays a page of text on a screen, and listens while someone reads it. While the user is reading, Evelyn dynamically displays what it thinks is the reader's current position in the text, by highlighting the next word to read. This position does not necessarily progress linearly through the text, since the reader may repeat, misread, sound out, insert, or skip words. Due to the nature of the speech recognition process, the display lags behind the reader; it is intended to show us what the system is doing, rather than to be of pedagogical benefit.

When the reader finishes, Evelyn identifies substitutions, deletions, and insertions relative to the original text. Evelyn treats these phenomena as follows:

- **Substitutions:** Evelyn provides contrastive feedback. To focus the reader's attention, it visually highlights the misread passage of the text on the screen. It plays back what the reader said, and then speaks the same passage correctly using synthesized or pre-digitized speech.
- **Deletions:** Evelyn provides corrective feedback. It highlights and speaks the text it thinks the reader skipped. For this case there is nothing to play back.
- **Insertions:** Evelyn deliberately ignores them. Insertions are usually hesitations, sounding out, self-corrections, repetitions, or interjections, rather than genuine misreadings of the text.

Notice that Evelyn uses "missed words" -- words in the text that were not read correctly at least once -- as its criterion for what to give feedback on. This criterion is based on the pedagogical assumption that whether the reader eventually succeeded in reading the word matters more than whether he or she got it right on the first try. Although word misses are a reasonable first-order approximation, finer-grained criteria will be needed to

[1]This research was supported in part by the National Science Foundation under Grant Number MDR-9154059; in part by the Defense Advanced Research Projects Agency, DoD, through DARPA Order 5167, monitored by the Air Force Avionics Laboratory under contract N00039-85-C-0163; in part by a grant from the Microelectronics and Computer Technology Corporation (MCC); in part by the Rutgers Center for Computer Aids to Industrial Productivity, an Advanced Technology Center of the New Jersey Commission on Science and Technology, at Rutgers University; and in part by a Howard Hughes Doctoral Fellowship to the third author from the Hughes Research Laboratory, where she is a Member of Technical Staff. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the sponsors or of the United States Government.

trigger an expanded range of pedagogically useful interventions.

We make no claims for the pedagogical efficacy of Evelyn's feedback. Rather, its purpose is to show us what the speech analysis is doing, and to test the feasibility of features that might later support various interventions. To identify more effective forms of feedback to implement, we are currently performing "Wizard of Oz" studies in which a human experimenter controls the display by hand to simulate possible system behavior -- including interrupting the reader to provide assistance in the context where it is needed.

3. Relation to previous work

There have been some uses of automated speech recognition in language learning. For example, the Indiana Speech Training Aid (Watson et al, 1989) helps hearing-impaired people improve their speech, by comparing their pronunciation of problem words to that of fluent speakers. (Newton, 1990) describes a commercial product that uses automated speech recognition for a similar purpose in foreign language training. However, these systems are based on isolated word recognition technology, which requires as input a single word or phrase chosen from a fixed vocabulary. The techniques used in isolated word recognition do not handle the continuous speech that occurs in reading connected text.

Some more recent work has used continuous speech recognition. (Bernstein et al, 1990) automatically estimated the intelligibility of foreign speakers based on how well their readings of a few sentences matched models trained on native speakers. A system developed at MIT uses a connected speech recognizer to follow the reading of a known text, providing verbal feedback via DECtalk (McCandless, 1992, Phillips et al, 1992). However, systematic evaluation of its accuracy has been limited by the lack of a corpus of disfluent reading. Instead, fluent utterances were used to simulate disfluent reading by pretending that one word in each utterance should have been some other word selected randomly from the lexicon -- a methodology that admittedly fails to capture important characteristics of disfluent reading (McCandless, 1992, p. 12).

There has been more use of speech in the system-to-student direction, thanks to the availability of synthesized or digitized speech. In particular, previous research has documented the benefits of making speech feedback on demand available to children with reading difficulties (Wise et al, 1989, Roth & Beck, 1987, McConkie & Zola, 1987, Reitsma, 1988), and this capability is now available in some commercial educational software (e.g., (Discis, 1991)). Pronouncing a word on demand supports students' reading comprehension -- both directly, by relaxing the bottleneck caused by their deficits in word recognition, and indirectly, by freeing them to devote more of their attentional resources to comprehension processes. Although such assistance can therefore be very useful, its utility is limited by the students' ability and willingness to ask for help when they need it; struggling readers often misread words without realizing it (McConkie, 1990).

Alternatives to the approach reported here include using an eyetracker or a user-controlled pointing device to track the reader's position in the text. These alternatives might indeed facilitate text tracking, but at best they could only indicate what the reader was trying to read -- not whether the outcome was successful.

Our project differs from previous efforts by drawing on the best available technology, in the form of Bellcore's ORATOR™ speech synthesizer[2] (Spiegel, 1992) and CMU's Sphinx-II speech recognizer (Huang et al, 1993). ORATOR produces high-quality speech, and is especially good at pronouncing names. Sphinx-II represents the current state of the art in speaker-independent connected speech recognizers, insofar as it was ranked at the top in DARPA's November 1992 evaluations of such systems. However, analysis of oral reading differs from speech recognition in an important way. In speech recognition, the problem is to reconstruct from the speech signal what sequence of words the speaker said. In contrast, the problem addressed in this paper is to figure out, given the text, where the speaker departed from it.

4. How Evelyn works

In this section we explain how the Evelyn system works. Since Evelyn is built on top of the Sphinx-II speech recognizer, we start with a minimal description of Sphinx-II to distinguish what it already provides from what Evelyn contributes. (For a more detailed description of Sphinx-II, please see (Huang et al, 1993).) Then we explain how we use the Sphinx-II recognizer within the Evelyn system -- that is, how we generate Sphinx-II's knowledge sources from a given text, and how we process Sphinx-II's output.

4.1. Speech recognition with the Sphinx-II system

Sphinx-II's input consists of digitized speech in the form of 16,000 16-bit samples per second from a microphone via an analog-to-digital converter. Sphinx-II's output consists of a segmentation of the input signal into a string of words, noises, and silences.

Sphinx-II uses three primary knowledge sources: a database of phonetic Hidden Markov Models, a dictionary of pronunciations, and a language model of word pair transition probabilities. The Hidden Markov Models use weighted transitions between a series of states to specify the acoustic probability of a given phone or noise. The pronunciation dictionary represents the pronunciation of each word as a sequence of phonemes. The language model specifies the linguistic probability that the second word in each pair will follow the first.

Sphinx-II operates as follows. The digitized speech is compressed to produce four 16-bit numbers every 10 msecs. This stream of values is matched against the Hidden Markov Models to compute the acoustic probability of each phone at each point in the speech signal. Hypothesized phones are concatenated into words

[2]ORATOR is a registered trademark of Bellcore.

using the pronunciation dictionary, and hypothesized words are concatenated into word strings using the language model. Beam search is used to pursue the most likely hypotheses, while unlikely paths are pruned. At the end of the utterance, Sphinx-II outputs the highest-rated word string as its best guess.

4.2. How Evelyn applies Sphinx-II to oral reading

In order to use Sphinx-II, we must supply the phonetic, lexical, and linguistic knowledge it requires to recognize oral reading of a given text.

Evelyn's phonetic knowledge currently consists of Sphinx-II's standard 7000 Hidden Markov Models trained from 7200 utterances produced by 84 adult speakers (42 male and 42 female). Sphinx-II uses separate models for male and female speakers. The results reported here used the female models, which we assume work better for children's higher-pitched speech.

Evelyn's lexical knowledge is created by a combination of lookup and computation. First the given ASCII text is segmented into individual words, and the words are sequentially numbered in order to distinguish multiple occurrences of the same word. The phonetic pronunciation of each word is then looked up in a pronunciation lexicon of about 32,000 words. However, some words may not be found in this lexicon, such as idiosyncratic words or proper names. Pronunciations for these missing words are generated by the ORATOR speech synthesizer. Finally, the pronunciation lexicon is augmented with individual phonemes to allow recognition of non-word phonetic events that occur when readers sound out difficult words.

Evelyn's linguistic knowledge consists of a probabilistic word pair transition grammar. This grammar is generated automatically from the numbered word sequence. It consists of pairs of (numbered) words and the likelihood of a transition from one word to the next. There are currently three kinds of transitions in our language model. The highest-probability transition is from one word to the next one in the text, which models correct reading. Second, a transition to an arbitrary other word in the text models a repetition or skip. Third, a transition to a phoneme models a non-word acoustic event.

The constraint provided by this grammar is critical to the accuracy of the speech recognition. If the grammar weights transitions to correct words too strongly, then word misses will not be detected when they occur. However, if it weights them too weakly, recognition accuracy for correct words will be low. To represent a particular tradeoff, Evelyn uses a linear combination of the three kinds of transitions.

As Figure 4-1 shows, the recognized string is Sphinx-II's best guess as to what the reader actually said into the microphone, given the phonetic, lexical, and linguistic knowledge provided. Evelyn compares this string against the original text, and segments the input utterance into correctly spoken words from the text, substitutions, deletions, and insertions. Based on this analysis, Evelyn provides the feedback described earlier in Section 2.

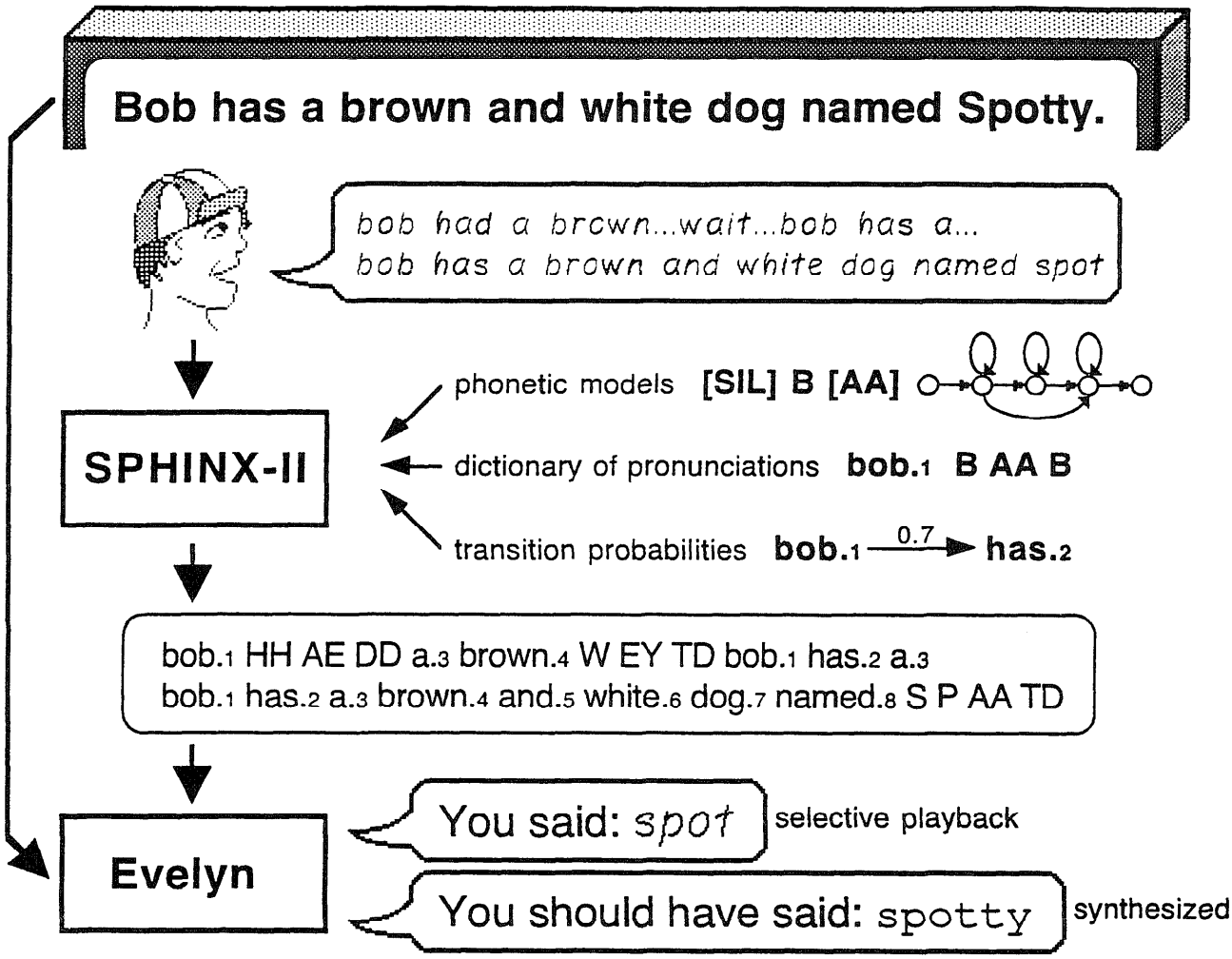
4.3. Text following

Although Evelyn provides corrective feedback only after the reader finishes the page, in the future we would like to interrupt with help when appropriate. Evelyn does provide one prerequisite capability for making such interruptions, namely text following. At present this capability is used merely to provide a visible dynamic indication of where Evelyn thinks the reader is. However, experience with this capability has exposed some challenging technical issues.

In order to track the reader's position in the text, Evelyn obtains partial recognition results from Sphinx-II four times a second in the form of the currently highest rated word string. However, these partial results are subject to change. For example, suppose the word "alter" was spoken. The first partial hypothesis may be the determiner "a". After more speech has been processed, the word "all" might be a candidate. It is not until the whole spoken word has been processed that Sphinx-II will return the correct hypothesis. Moreover, if the subsequent word is "candle", the hypothesis may be revised to reflect the high probability of the phrase "altar candle". In contrast, the phrase "alter ego" would require no modification of the earlier hypothesis of "alter".

The point of this discussion is that one cannot merely look at the last word in the current partial recognition result for a reliable estimate. Our initial text following algorithm, which did just that, caused the cursor that indicates the current location in the text to skip around wildly as it tried to follow the reader. The problem is that in order to select reliably among competing hypotheses, the recognizer needs to know the context that follows. This **"hindsight dependency"** suggests that for some time to come, a real-time speech recognizer will have to lag behind the speaker by a word or two to reduce inaccuracy -- *no matter how fast the machine it runs on.* Thus pedagogical interruptions triggered by recognition will face a tradeoff between interrupting promptly and waiting for more reliable recognition results.

To attack this problem, we developed a more sophisticated heuristic text-following algorithm. It exploits the expectation that the reader will step through the text in a sequential fashion, and yet allows for reading errors without considering short-lived spurious candidates. The revised algorithm has a certain amount of inertia. As it digests partial results, it refrains from moving the cursor until at least n (currently $n=2$) consecutive words in the text have been recognized. This heuristic gives us reasonable confidence that a portion of the text has been read correctly and that recognition is stable. If Sphinx-II recognizes a word other than the expected next one, this method prevents the cursor from immediately jumping to a new location in the text, since it is likely to represent either a misreading by the reader or a misrecognition by the recognizer. However, when the reader really does skip to another place in the text, the method allows the cursor to catch up, after a short delay during which consecutive words from the new location in the text are recognized.



In this example, the reader self-corrected "had" to "has", but misread "Spotty" as "spot". Sphinx-II's actual recognition was much less accurate than the ideal result shown.

Figure 4-1: Example of how Evelyn should detect a word miss

5. How well Evelyn works

Since Evelyn is only a precursor of an educational system, a true pedagogical evaluation of it would be premature. However, we did measure its performance in a way that would help guide our work. In particular, we needed a suitable evaluation scheme to help us develop, assess, and improve the language models described in Section 4.2 by allowing us to test alternative language models against the same data. We now describe the data we collected for this purpose, the methodology we used to measure performance, and the results we obtained.

5.1. Corpus of oral reading

To evaluate Evelyn's performance, we collected and transcribed a corpus of oral reading, which we are continuing to expand. This paper is based on readings by second graders at two schools in Pennsylvania. Of the 27 speakers, 17 are from Turner School, a public school in Wilkinsburg with a predominantly minority student body. The other 10 are from the Winchester Thurston School, a private school in Pittsburgh.

We made the recordings at the schools, using special-purpose software running on a NeXT workstation to display the texts and digitally record the speech. We used a Sennheiser close-talking headset microphone to reduce the amount of non-task information in our acoustic signal, but did not by any means eliminate it. Our corpus contains many sounds, both speech and non-speech, that are not examples of a reader decoding a text. (Teachers and children are talking in the hallway, doors are slamming, and readers are shuffling and bumping the microphone with their chins.)

We selected the reading materials from the Spache graded reading tests (Spache, 1981), since their levels of difficulty are well calibrated and their accompanying comprehension probes have been carefully validated. To

accommodate a large display font, we split each text into two or three one-page passages, advancing to each "page" when the child finished the previous one, and administering a comprehension probe when the child completed the text. To obtain examples of reading errors, we chose texts for each subject somewhat above his or her independent reading level, which we estimated by administering Spache's word list test. We wanted text challenging enough to cause errors, but easy enough to produce complete readings.

The evaluation corpus used for this paper consists of 99 spoken passages, totalling 4624 text words. The passages average about 47 words in length and 45 seconds in duration. The pace -- about one text word per second -- reflects the slow reading speed typical of early readers. The number of actual spoken words is higher, due to repetitions and insertions.

We carefully transcribed each spoken passage to capture such phenomena as hesitations, "sounding out" behaviors, restarts, mispronunciations, substitutions, deletions, insertions, and background noises. The transcripts contain correctly spoken words, phonetic transcriptions of non-words, and noise symbols such as `[breath]`.

5.2. Accuracy of recognition and missed-word detection

Using the original texts, transcripts, and recognizer outputs over the corpus, we measured both "raw" recognition accuracy and missed-word detection accuracy.

To compute recognition accuracy, we compared the string of symbols output by the recognizer against the string of symbols obtained from the transcript. We used a standard dynamic programming algorithm to align symbols from these two strings and count substitutions, deletions, and insertions. These "raw" scores appear mediocre: 4.2% substitutions, 23.9% deletions, and 0.5% insertions. Thus 28.1% of the transcribed symbols are misrecognized, and the total error rate, including insertions, is 28.6%.

However, since Evelyn's purpose is to detect missed words, a more useful criterion in this domain is the accuracy of this detection. (Recall that a word that is misread or "sounded out", but eventually produced correctly, is not considered a missed word.) To measure the *accuracy of missed-word detection*, we counted the words in the text that Evelyn misclassified, either as correct, or as missed.

Measuring the accuracy of missed-word detection requires a three-way comparison between the original text (what the reader was supposed to say), a transcript of the utterance (what the reader actually said), and the recognizer's output (what the recognizer thinks the reader said). First, the actual word misses in the corpus are identified by comparing the transcripts against the original text, using the alignment routine described earlier. Then the hypothesized misses are identified by comparing the recognizer output against the original text, using the same alignment routine. Finally we check the hypothesized misses against the actual ones.

Table 5-1: Accuracy of Missed-Word Detection

Reader Disfluency	Evelyn Coverage	Evelyn Precision
2.5%	63.6%	60.9%

Disfluency = (missed words) / (words in text)

Coverage = (misses detected) / (words missed)

Precision = (misses detected) / (missed-word reports)

Table 5-1 summarizes Evelyn's ability to detect word misses in our current corpus: how frequently word misses occurred, what fraction of them Evelyn reported, and what fraction of such reports were true. We computed these three numbers separately for each reading, and then averaged them across the readings, so as to avoid counting the longer, more difficult texts more heavily than the shorter, easier ones. (This methodology also served to discount some "outlier" runs in which our language model caused the recognizer to get lost without recovering.) The first number measures reading disfluency as the percentage of words actually missed by the reader, which varied from zero to 20%, but averaged only 2.5%. That is, the average reader missed only about one word in 40. The second number shows that Evelyn detected these misses almost two thirds of the time -- probably enough to be pedagogically useful. The third number reflects a moderate rate of false alarms. For each properly reported miss, Evelyn often classified a correctly read word as a miss, but a majority of such reports were true.

It is instructive to compare these numbers against a "strawman" algorithm that classifies 2.5% of the words in the text as missed, but chooses them randomly. That is, how well can we do if all we know is the average reader's disfluency? Since the strawman chooses these words independently of whether they are actual misses, its expected coverage and precision will also each be 2.5%. How well does Evelyn do by comparison? Its coverage and precision are each about twenty-five times better. Thus the additional information contributed by speech recognition, although imperfect, is nevertheless significant.

These results represent the best of the few language models we have tested so far on the corpus. Further improvements in accuracy may require devising better language models of oral reading and training new phonetic models on a large number of young readers.

Besides accuracy, we are concerned with speed, since timely intervention will require keeping up with the reader. For our language model and corpus, the recognizer already runs consistently in between real time and two times real time on a 100+ MIPS DEC Alpha workstation. A modest increase in processing power due to faster hardware should therefore produce the speed required for real time response.

6. Conclusion

The principal contribution of this work is an **implemented system for a new task** -- automatically following the reading of a known text so as to detect an important class of oral reading errors (namely, missed words). Its **model of oral reading** constitutes an initial solution to the problem of constraining a speech recognizer's search when the text is known but the reading is disfluent. We identified **hindsight dependency** as causing an intrinsic tradeoff between accuracy and immediacy for recognition-driven interrupts, and developed a **heuristic text-following algorithm** based on this tradeoff.

To establish an initial baseline for performance at this new task, we **evaluated the performance** of Evelyn and its underlying model. We defined **performance evaluation criteria** that are more appropriate for the task of detecting word misses than is the traditional definition of accuracy in speech recognition. To evaluate our algorithms, we recorded and transcribed a **corpus of oral reading** by second graders of varying fluency. This corpus is a contribution in itself, since the speech recognition community has not previously had access to a corpus of disfluent oral reading. It is essential to our continued efforts to improve on the baseline defined by Evelyn.

The social significance of our work, if it succeeds, will be its impact on illiteracy: even a one percent reduction in illiteracy would save the nation over two billion dollars each year. But in the long run, the broader scientific significance of this work may be its role in helping to open a powerful new channel between student and computer based on two-way speech communication.

Acknowledgements

We thank Marcel Just, Leslie Thyberg, and Margaret McKeown for their expertise on reading; Matthew Kane, Cindy Neelan, Bob Weide, Adam Swift, Nanci Miller, and Lee Ann Galasso for their various essential contributions; the entire CMU Speech Group for their advice on speech in general and Sphinx in particular; Murray Spiegel and Bellcore for use of their ORATOR speech synthesis system; CTB Macmillan/McGraw-Hill for permission to use copyrighted reading materials from George Spache's *Diagnostic Reading Scales*; the pupils we recorded at Irving School in Highland Park, NJ, Winchester Thurston School in Pittsburgh, PA, and Turner School in Wilkinsburg, PA, and the educators who facilitated it; and the many friends who provided advice, encouragement, and assistance to get Project LISTEN started.

References

J. Bernstein, M. Cohen, H. Murveit, D. Rtischev, and M. Weintraub. (1990). Automatic evaluation and training in English pronunciation. *International Conference on Speech and Language Processing (ICSLP-90)*. Kobe, Japan.

Discis Knowledge Research Inc. *DISCIS Books*. 45 Sheppard Ave. E, Suite 802, Toronto, Canada M2N 5W9. Commercial implementation of Computer Aided Reading for the MacIntosh computer.

E. Herrick. (1990). *Literacy Questions and Answers*. Pamphlet. P. O. 81826, Lincoln, NE 68501: Contact Center, Inc.

X. D. Huang, F. Alleva, H. W. Hon, M. Y. Hwang, K. F. Lee, and R. Rosenfeld. (1993). The SPHINX-II speech recognition system: An overview. *Computer Speech and Language*, (in press).

M. McCandless. (May 1992). *Word Rejection for a Literacy Tutor*. S.B. Thesis. Cambridge, MA: MIT Department of Electrical and Computer Engineering.

G. W. McConkie. (November 1990). Electronic Vocabulary Assistance Facilitates Reading Comprehension: Computer Aided Reading. Unpublished manuscript.

G. W. McConkie and D. Zola. (1987). Two examples of computer-based research on reading: Eye movement tracking and computer aided reading. In D. Reinking (Eds.), *Computers and Reading: Issues for Theory and Practice*. New York: Teachers College Press.

F. Newton. (1990). Foreign language training. *Speakeasy*, Vol. *1*(2). Internal publication distributed to customers by Scott Instruments, 1111 Willow Springs Drive, Denton, TX 76205.

M. Phillips, M. McCandless, and V. Zue. (September 1992). *Literacy Tutor: An Interactive Reading Aid* (Tech. Rep.). Spoken Language Systems Group, 545 Technology Square, NE43-601, Cambridge, MA 02139: MIT Laboratory for Computer Science.

P. Reitsma. (1988). Reading practice for beginners: Effects of guided reading, reading-while-listening, and independent reading with computer-based speech feedback. *Reading Research Quarterly*, *23*(2), 219-235.

S. F. Roth and I. L. Beck. (Spring 1987). Theoretical and Instructional Implications of the Assessment of Two Microcomputer Programs. *Reading Research Quarterly*, *22*(2), 197-218.

G. D. Spache. (1981). *Diagnostic Reading Scales*. Del Monte Research Park, Monterey, CA 93940: CTB Macmillan/McGraw-Hill.

M. F. Spiegel. (January 1992). *The Orator System User's Manual - Release 10*. Morristown, NJ: Bell Communications Research Labs,

C. S. Watson, D. Reed, D. Kewley-Port, and D. Maki. (1989). The Indiana Speech Training Aid (ISTRA) I: Comparisons between human and computer-based evaluation of speech quality. *Journal of Speech and Hearing Research*, *32*, 245-251.

B. Wise, R. Olson, M. Anstett, L. Andrews, M. Terjak, V. Schneider, J. Kostuch, and L. Kriho. (1989). Implementing a long-term computerized remedial reading program with synthetic speech feedback: Hardware, software, and real-world issues. *Behavior Research Methods, Instruments, & Computers*, *21*, 173-180.