Inclusion of Deaf Students in Computer Science Classes using Real-time Speech Transcription

Richard Kheir and Thomas Way
Applied Computing Technology Laboratory
Department of Computing Sciences
Villanova University
Villanova, PA 19085

richard.kheir@villanova.edu
thomas.way@villanova.edu

ABSTRACT

Computers increasingly are prevalent in the classroom, with student laptops becoming the norm, yet some beneficial uses of this widespread technology are being overlooked. Speech recognition software is maturing, and possesses the potential to provide real-time note taking assistance in the classroom, particularly for deaf and hard of hearing students. This paper reports on a practical, portable and readily deployed application that provides a cost-effective, automatic transcription system with the goal of making computer science lectures inclusive of deaf and hard of hearing students. The design of the system is described, some specific technology choices and implementation approaches are discussed, and results of two phases of an in-class evaluation of the system are analyzed. Ideas for student research projects that could extend and enhance the system also are proposed.

Categories and Subject Descriptors

K.4.2 [**Computers and Society**]: Social Issues – *Assistive technologies for persons with disabilities*. H.5.2 [**Information Interfaces and Presentation**]: User Interfaces – *Voice I/O.*

General Terms

Design, Experimentation, Human Factors.

Keywords

Speech recognition, computer science education, inclusion, accessibility, deaf students, hard of hearing students, assistive technology.

1. INTRODUCTION

Advances in affordable portable computing technology have led to wider availability, making it possible to deploy automatic speech recognition (ASR) in the classroom, although challenges remain [3]. The ability of ASR systems to transcribe continuous speech faster than a note taker can write, with reasonable accuracy and minimal training, make them a viable option to assist deaf and hard of hearing students with note taking [5]. Computer science continues to be a popular choice of college major for high school students with hearing disabilities [1], although these students can find traditional accommodations such as sign language interpreters or lip-reading insufficient [10]. Technology such as speech recognition can provide a viable solution, but awareness of accessibility issues continues to be the most significant hurdle to inclusion [4].

Obstacles to relying on ASR for note taking include recognizing multiple or random speakers [5], synchronizing and incorporating visual cues [9], balancing real-time automated speech text against the potential for distraction [6], insufficient accuracy in recognizing domain-specific jargon [5], configuring, training and deploying the ASR system for classroom use [2], and achieving acceptable accuracy through microphone selection, improved software and additional training of the ASR system [11].

Active research in ASR for college classrooms is being done by the Liberated Learning Project (LLP), among others [5,6,2,11]. The LLP has the goal of enabling students with various disabilities, including hearing impairment, to maximize the benefits of the college lecture experience [8]. Significantly, the LLP has collaborated with IBM to develop the ViaScribe software that is specifically designed for real-time captioning, including ASR, of natural, extemporaneous speech. ViaScribe improves readability by detecting pauses in speech and inserting sentence and paragraph breaks, provides phonetic spellings when the recognizer is uncertain, and even has a less-accurate speaker-independent mode to accommodate multiple speakers [3].

Accuracy of reasonably well-trained ASR systems typically is better than 75-85% in classroom lecture settings, with rates over 90% for particularly consistent and clear lecturers [5,11], a rate that a significant majority of students find acceptable and useful [6]. A centralized ASR system producing real-time captioning on a projection screen with post-lecture access to a transcription has been used successfully in the classroom [11], although a more individualized approach often may be preferable [3,6,11].

This paper presents the design and evaluation of the Villanova University Speech Transcriber (VUST) system that increases accessibility of computer science lectures for deaf and hard of hearing students using real-time speech recognition software. This study was conducted at the Applied Computing Technology Laboratory at Villanova University (actlab.csc.villanova.edu), and

ITiCSE'07, June 23–27, 2007, Dundee, Scotland, United Kingdom.

evaluates the impact of the VUST system paired with our Dictionary Building Software utility (DiBS) [7] on the effectiveness of a portable, centralized, affordable, laptop-based ASR system designed to augment note taking by deaf and hard of hearing students in the college classroom. Although the original motivation for development of the system was to improve accessibility of computer science lectures specifically, the system holds potential for much wider applicability.

2. SYSTEM DESIGN

The VUST system consists of three major components: the speech recognition software, a dictionary enhancement tool, and a transcription distribution application. Figure 1 illustrates the VUST architecture, showing these major components and other elements of the system.

The dotted line in Figure 1 indicates the physical computer on which the speech recognition engine, VUST server application, wireless microphone receiver and other elements are located. One or more client applications can connect to the server, and a wireless headset microphone transmits speech to the server for processing.

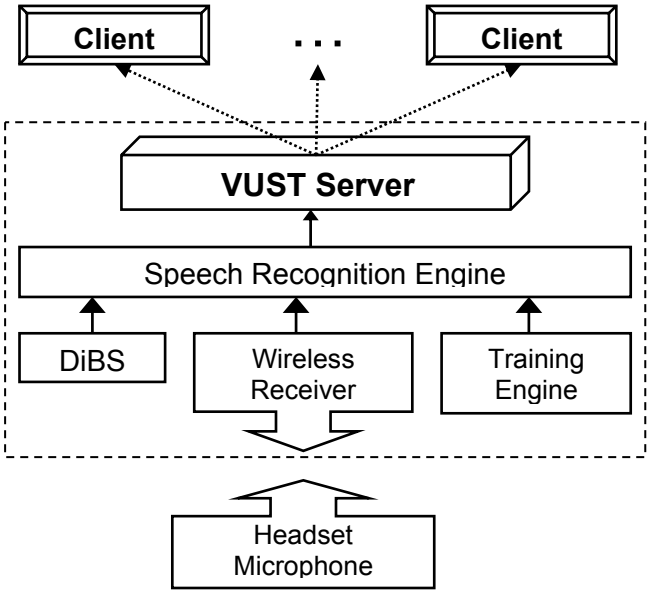

Figure 1. VUST System Design.

2.1 Speech Recognition System

The speech recognition system uses an ASR system designed to be affordable, accurate and easy to set up and use. The Microsoft Speech Recognition Engine (MSRE) was selected due to the wide availability in academic institutions of the Microsoft XP platform, which includes the MSRE, effectively providing the ASR engine for our system at no additional cost.

The Nady Systems UHF-3 wireless unidirectional headset microphone was selected as a cost-effective solution ($120-$140), with unrestricted movement, high directionality and good tolerance of interference being key considerations when selecting a microphone for ASR [7].

The MSRE is trained by an instructor via a control panel included with the engine. The instructor reads from a selection of available text scripts into a microphone, enabling the recognition engine to learn to recognize the specific words as spoken by the specific instructor. The maximum level of training that was tested in our evaluation required less than one hour, with 30 minutes of script-based training, 5 minutes to run the dictionary tool, and 10 minutes of additional training to record pronunciations of domain-specific words.

Setting up and running the system involves ensuring the instructor's computer is appropriately networked, connecting the wireless microphone receiver and putting on the wireless headset, activating the MSRE via the Windows Speech control panel, and starting the server application. Once the system is running, students can connect via a simple web page containing the client application. The instructor controls the location and content of this web page.

2.2 Dictionary Tool

The Dictionary Building Software tool (Figure 2) analyzes textual input, scanning for domain-specific terminology to add to the speech recognition system custom dictionary (i.e., "custom.dic"). DiBS parses an input file into words, filtering words below a minimum length threshold, that appear in a standard system dictionary, and that already appear in the custom dictionary. The minimum length threshold of six characters limits the words considered to those with a higher likelihood of being domain-specific, which tend to be longer in length.

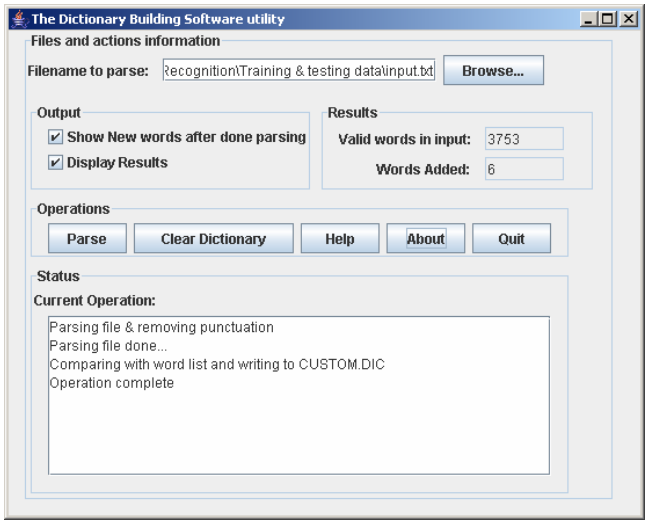

Figure 2. Dictionary Building Software (DiBS) utility.

The key innovation of the DiBS tool is the ability for the user easily to add domain-specific terminology to the MSRE custom dictionary in one, simple step. Prior to DiBS, the method for customizing the dictionary and improving recognizer accuracy was well hidden in obscure documentation, and involved a number of non-intuitive steps. The DiBS tool streamlines the process so that minimal time and no technical expertise is required in order to customize the dictionary, thereby improving

the accuracy of the recognition engine, and therefore likelihood that the speech recognition system will be used.

The speech recognition engine relies on a static system dictionary for its basic recognition, with syntax rules built into the recognizer that phonetically match utterances with corresponding words. Secondly, the recognizer uses words in the custom dictionary in a similar way. DiBS improves recognition accuracy by adding terminology to this custom dictionary.

If a user notes that some terminology is still not being recognized, which can happen if the word uses exceptions to typical rules of pronunciation or is particularly complicated, word-specific training can be performed by the user. This training is part of the underlying Windows XP speech recognition system, and is done using a training interface linked to the custom dictionary.

2.3 Transcription Distributor

The VUST consists of a text distribution server application and corresponding client application, both implemented in Java. The server and client are based on common chat server architecture, modified to accept input from the speech recognition engine and with client chat-back disabled. The design of VUST was kept minimal and straightforward to support a design goal of ease of use. Capture and acquisition of a lecture transcription had to be easy so that any instructor could deploy and use the system, and any student would find it easy to read and save the result. Java was selected as the implementation language to ensure portability across platforms, including Macs, PCs and Linux machines.

The VUST server receives the textual output of the recognition engine, and immediately forwards it to any client applications that are connected. The client application is a Java applet (Figure 3), embedded on a simple web page provided by the instructor, and automatically connects to the VUST server when the page is accessed. If the client fails to connect to the server, a message appears on the client indicating this failure.

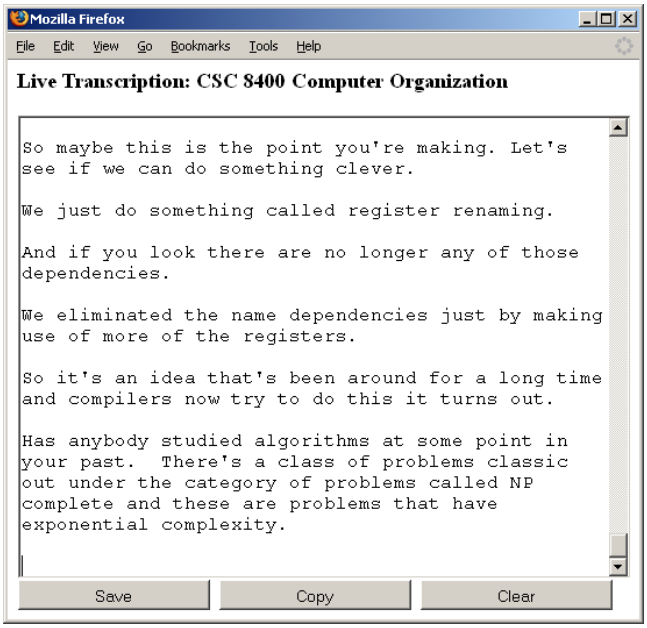

Figure 3. VUST Transcription Client applet.

In the sample of captured text in Figure 3, when brief pauses are detected, a period is inserted in the text, while longer pauses lead to the insertion of a paragraph break. In the last block of recognized text, even though the last sentence obviously contains some errors, it still maintains the intended meaning of the spoken sentence. This is typical of an acceptable form of recognition error.

In addition to presenting the live transcription of the lecture, the client also allows the student to export the transcription to a text file, copy and past it to another program, or clear the current transcription from the screen. A pop-up dialog prevents the student from accidentally clearing a transcription in progress without first confirming the desire to do so.

3. Evaluation

The VUST system was evaluated as a standalone, centralized speech transcription system for recognition accuracy, perceived accessibility and deployability. The system was tested in a controlled environment in an empty classroom using prepared lecture notes, and in a real classroom setting. An initial study was performed to measure the effectiveness of the DiBS tool on improving recognition accuracy. A follow-up study making use of the full VUST system was conducted to determine how the system would perform in an authentic lecture setting.

3.1 Improving Accuracy

The initial study measured the effectiveness of the DiBS utility to improve the recognition accuracy of the Microsoft Speech Recognition Engine (MSRE). The engine was prepared and tested using five training scenarios: untrained, minimally trained, moderately trained, moderately trained with a customized dictionary, and moderately trained with a customized dictionary and selected customized pronunciations.

The DiBS utility analyzed a number of text files containing the content of technical papers and lecture notes related to the subject matter of selected computer science lectures. Custom pronunciations were recorded using the MSRE training interface for approximately 10 domain-specific words that the MSRE had difficulty recognizing.

Tests were performed using spoken lectures containing terminology-rich material from undergraduate and graduate courses in computer architecture, totaling approximately 3,700 words or 30 minutes of continuous speech. The lectures were conducted in a classroom by a computer science professor wearing a wireless headset microphone, using a very clear and consistent speaking style, and were digitally captured to WAV files. To enable valid comparison, these digitized lectures were then replayed to the MSRE running on a university-issued laptop, under five training scenarios, with the transcription output captured into a Microsoft Word file. Objective measures of accuracy were made using a free text file comparison tool called DiffDoc (softinterface.com) by comparing the output of the speech recognizer with a human transcription of the original lecture. Results of the file comparison tool were analyzed manually for verification.

Table 1 shows the results of evaluation of the recognition engine for accuracy and accessibility under the five training scenarios. Accuracy improved with additional training, with marked

improvements when going from an untrained to a minimally trained system (from 75% to 88% accurate) and with the addition of a customized dictionary and pronunciations to a moderately trained system (from 91% to 94%). The recognition accuracy varied greatly (plus or minus 5-10%) depending on the prevalence of terminology that was not found in the default ASR dictionary. Adding terminology from the domain of the lecture helped, and additional recording of pronunciations of specific terminology that the recognizer still misrecognized helped more.

Table 1. Comparison of recognition accuracy, range of accuracy, and accessibility.

Description	Accuracy	Range	Accessibility
Untrained	75%	64-83%	poor to fair
Minimal training (default script, 10 minutes total)	88%	78-93%	sufficient
Moderate training (3 additional scripts, 30 minutes total)	90%	81-96%	good
Moderate training, customized dictionary	91%	83-96%	good
Moderate training, customized dictionary, customized pronunciations	94%	86-98%	very good

Accessibility of the resulting transcription was measured by reading the transcript and in effect grading it as if it were a student report summarizing the content of the lecture. This more subjective accessibility of each transcript was judged broadly to be: poor, fair, sufficient, good, very good, excellent. Even with minimal training, the results were passable (sufficient), although they required careful reading and some editing to make them usable as notes. With moderate training, transcripts were usable (good) as class notes with only minor editing, such as inserting paragraph breaks.

Although very good accessibility was achieved with the addition of some customized pronunciations, excellent accessibility was not achieved in any of the scenarios, reinforcing the need for continued research in speech recognition technology [1]. It is important to note that, although recognition at times reached well above 90% accuracy, a very good result, these results may be artificially optimistic due to the constrained nature of the quiet test environment, consistent speech and chosen material. The second phase of evaluation was designed to measure recognition in a more realistic classroom setting.

3.2 Measuring Deployability

To determine whether speech recognition could be a beneficial classroom technology for increasing accessibility of computer science lectures for deaf and hard of hearing students, the VUST system was deployed in a real lecture setting. For this experiment, the full system was used by the instructor in a regular computer architecture class meeting which included a hard of hearing student.

An entire 90 minute lecture consisting of nearly 10,000 words was transcribed using the VUST system, and the transcription output was saved to a text file and also transcribed manually for comparison. The instructor then analyzed the transcript and identified all misrecognitions, within reasonable constraints (e.g., singular vs. plural and homonym misses were allowed when the meaning was intact, while obviously incorrect recognition or anything that hurt the meaning was marked as incorrect). The automatic and manual transcriptions were then compared for accuracy. Sections of the transcript were classified based on their speech content, as: roll-call (list of names or otherwise discontinuous speech), planning (assignments, dates, general classroom business), discussion (interaction including student discussion), and lecture (continuous instructor speech).

Not surprisingly, the best recognition accuracy was achieved with prepared lecture, resulting from the MSRE preference for continuous speech. Note that the DiBS utility was not used in this phase of experiments to enable clear distinction among classifications of speech and effectiveness of the client-server approach. Overall accuracy was 85%. Planning, lecture and discussion were all consistent with this average, with roll-call scoring the lowest (61%). Table 2 summarizes the results obtained using the VUST.

Table 2. Comparison of VUST recognition accuracy with four classifications of speech content.

Classification	Words Correct	Total Words	Percent Recognized
Planning	628	758	83%
Lecture	5930	6925	86%
Roll-call	155	254	61%
Discussion	1556	1846	84%
TOTAL	**8269**	**9783**	**85%**

The low recognition accuracy (61%) of roll-call speech was not unexpected. A student name can be a form of domain-specific terminology all to itself, and are not likely to be found in the static system dictionary. Planning speech scored next lowest (83%), due to its disjoint, bullet-item nature, also lacking the continuous flow that the MSRE prefers. Discussion and lecture speech were both recognized at relatively acceptable rates, deemed very usable by the instructor and student who participated.

Student reaction to the VUST system was striking. The experience of real-time transcription was described as a “totally new experience” and of enormous benefit. The hearing-impaired student found himself raising his hand to contribute to a classroom discussion for the first time, having followed along with the help of the VUST transcript. Other (hearing) students who had access to the transcript following the class found it to be

a useful supplement to their notes, and they remarked at how closely the transcript matched what occurred in class.

4. CONCLUSIONS AND FUTURE WORK

The VUST system shows significant promise as an affordable and beneficial assistive system to make the computer science classroom more inclusive for deaf and hard of hearing students. Although the benefits of a sign language interpreter or prepared lecture note handouts is recognized, both require additional and regular cost or preparation. By enabling the use of a automated, real-time transcription, cost and preparation overhead is reduced and accessibility is increased.

Providing easy to use software that can improve recognition accuracy and make distribution of a real-time lecture transcription contribute to making VUST very usable by instructors and students. Customizing the dictionary of speech recognition system with domain-specific terminology is effective at improving accuracy. The DiBS tool provides an efficient means to automatically cull such uncommon jargon from large amounts of text and customize the recognition engine, in this case the MSRE. Although DiBS only considers new terms that are six characters or greater in length as an optimization, shorter domain-specific terms can be added manually by an instructor.

An alternative use of VUST could be in stand-alone mode, running on a student laptop. In this configuration, a student would provide a wireless microphone to the instructor and capture the lecture transcription directly on the student computer. However, effective use of ASR in this way requires the student laptop to contain a speech profile trained by the instructor. Using the Speech Recognition Profile Manager Tool (microsoft.com), a speech profile can be imported or exported, making possible distribution of the profile, along with custom dictionaries for specific topics, via a central repository such as a university or department web site. In this way, a student can install such a speech profile of a particular instructor and immediately improve recognition accuracy.

It is important to note that although VUST generates a transcription that can improve accessibility, it is not a replacement for attendance and the very real benefits of being physically present and interactive in a lecture setting. Recognition technology has advanced considerably in recent years, yet accuracy is still far from producing lecture notes on par with what an instructor would prepare by hand. The VUST transcript is best used to assist and augment note taking, much as a student uses a spoken lecture to add detail and clarification to material gleaned from slides or board work.

Because VUST and DiBS are implemented using Java, and the system consists of distinct software components, there are many opportunities for student research and development projects. One project could involve improving the DiBS tool to harvest more domain-specific terminology from a variety of sources. DiBS currently only accepts text input, but available Java add-ons could make it possible to parse PDF and MS Word documents, further improving the usability of the system.

Another potential project is the development of a corpus of domain-specific terminology, ready-made for computer science that could be used as customization input to the DiBS tool. This collection could be extended to other terminology rich subjects, such as biology, engineering, philosophy, and others, further increasing accessibility to real-time lecture transcription.

Future work includes plans to produce a commercial-quality version of the VUST and DiBS software, design of a centralized repository system for domain specific terminologies and speech profiles, and evaluation of other cost-effective speech engines.

5. REFERENCES

[1] J. Blackorby, R. Cameto, A. Lewis and K. Hebbeler. Study of persons with disabilities in science, mathematics, engineering and technology. SRI International, Menlo Park, CA, 1997.

[2] K. Bain, S. Basson and M. Wald. Speech recognition in university classrooms. Procs. of the Fifth International ACM SIGCAPH Conference on Assistive Technologies, ACM Press, pp. 192-196, 2002.

[3] K. Bain, S. Basson, A. Faisman and D. Kanevsky. Accessibility, transcription, and access everywhere. IBM Systems Journal, 2005, Vol. 44, No. 3, pp. 589-603, 2005.

[4] R. Cohen, A. Fairley, D. Gerry and G. Lima. Accessibility in introductory computer science. Procs. of the 36th SIGCSE Technical Symposium on Computer Science Education, pp. 17-21, 2005.

[5] C. Davis. Automatic speech recognition and access: 20 years, 20 months, or tomorrow? Hearing Loss, 2001, 22(4), pp. 11-14, 2001.

[6] A. Hede. Student reaction to speech recognition technology in lectures. In S. McNamara and E. Stacey (Eds.), Untangling the Web: Establishing Learning Links. Procs. of the Australian Society for Educational Technology (ASET) Conference, Melbourne, July 2002.

[7] R. Kheir and T. Way. Improving speech recognition to assist real-time classroom note taking. Rehabilitation Engineering Society of North America (RESNA). Conference, Atlanta, USA, June 2006.

[8] Liberated Learning Project. Coordinated by Saint Mary's University, Halifax, Canada, online at http://liberatedlearning.com, accessed Dec. 18, 2006.

[9] R. Stuckless. Recognition means more than just getting the words right: Beyond accuracy to readability. Speech Technology, Oct./Nov. 1999, pp. 30-35, 1999.

[10] M. Wald. Hearing disability and technology. Access All Areas: disability, technology and learning, JISC TechDis and ALT, pp. 19-23, 2002.

[11] M. Wald and K. Bain. Using automatic speech recognition to assist communication and learning. Procs. of the 11th International Conference on Human-Computer Interaction, Las Vegas, 2005.