An Online System for Entering and Annotating non-Native Mandarin Chinese Speech for Language Teaching

by

Andrea Johanna Hawksley

Submitted to the
Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the degree of

Master of Engineering in Computer Science and Engineering

at the Massachusetts Institute of Technology

August 2008

Copyright 2008 Andrea J. Hawksley. All rights reserved.

The author hereby grants to M.I.T. permission to reproduce and to distribute publicly paper and electronic copies of this thesis document in whole and in part in any medium now known or hereafter created.

Author . . . .
Department of Electrical Engineering and Computer Science
August 22nd, 2008

Certified by . . . .
Stephanie Seneff
Principal Research Scientist
Thesis Supervisor

Accepted by . . . .
Arthur C. Smith
Chairman, Department Committee on Graduate Theses

An Online System for Entering and Annotating non-Native Mandarin Chinese Speech for Language Teaching

by

Andrea Johanna Hawksley

Submitted to the Department of Electrical Engineering and Computer Science
on August 22nd, 2008, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Computer Science and Engineering

Abstract

This thesis describes the design and implementation of an intuitive online system for the annotation of non-native Mandarin Chinese speech by native Chinese speakers. This system will allow speech recognition researchers to easily generate a corpus of labeled non-native speech. We have five native Chinese speakers test the annotation system on a sample bank of 250 Chinese utterances and observe fair to moderate inter-rater agreement scores. In addition to giving us a benchmark for inter-rater agreement, this also demonstrates the feasibility of having remote graders annotate sets of utterances. Finally, we extend our work to Chinese language instruction by creating a web-based interface for Chinese reading assignments. Our design is a simple, integrated solution for completing and correcting of spoken reading assignments, that also streamlines the compilation of a corpus of labeled non-native speech for use in future research.

Thesis Supervisor: Stephanie Seneff
Title: Principal Research Scientist

Acknowledgments

The author would like to thank the Spoken Language Systems (SLS) group, the Computer Science and Artificial Intelligence Laboratories, and the Massachussetts Institute of Technology for providing her with the opportunity and funding to complete this masters thesis. She particularly wants to thank her advisor Stephanie Seneff for her indispensable help, guidance, and patience throughout this project. She would also like to thank Ian McGraw and Mitchell Peabody for their help and guidance. The Industrial Technology Research Institute (ITRI), Ruby Chou, and Lii-miin Hawksley helped provide suitable native speaking annotators for her user study. Scott Kominers provided support throughout the projected and aided particularly in the writing and editing of this thesis. The author would also like to thank John Hawksley and Lii-miin Hawksley for their help in editing this thesis. Finally, she would like to thank Ibrahim Badr, Christopher Hawksley, and Eric Tung for their support throughout.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Speech recognition has great potential to improve Computer Aided Language Learning (CALL) systems by enabling students to practice speaking without a human language partner. A major problem in foreign language classes is that students often lack opportunities to practice speaking - even though the aim of most language students is to learn to speak the language fluently. Ideally, a language class might have close to a one-to-one ratio between students and native speakers. In practice there are typically many students per teacher and students may only get to say a sentence or two per class.

Besides the lack of in-class interaction, it is difficult for teachers to assign spoken homework assignments. The one type of spoken assignment in frequent use is a spoken reading assignment. In this type of assignment, a teacher assigns a paragraph to be read. The students then read the assignment into a recording device and turn in the resultant recording. To grade the assignment, the teacher listens to the recording and corrects the students' pronunciation by marking up the original paragraph. Some time after the initial recording, the student has their recording returned to them with the corrections indicated. Spoken free responses to a simple prompt are a modification of this assignment that require more time on the part of the teacher but allow for the practice of spontaneous speech.

There are several flaws with this system. Many schools still use cassette tapes for recording even though cassettes are an obsolete medium. Students probably will not remember what they originally said for the assignment and how they pronounced it by the time their assignments are returned. While re-listening to the cassette tape in the context of the teacher corrections may be useful, a student may have difficulty understanding what they are saying wrong and correcting it without more feedback. Finally, the purpose of a spoken reading assignment is to develop both speaking and reading fluency. Without any timely feedback, a spoken reading assignment may not be the most pedagogically efficient way of improving reading fluency. Mostow showed that both human and automated reading tutors that supplied immediate feedback were superior to classroom instruction where there was less individual attention and feedback [31].

A computer system for this type of assignment offers many advantages over the standard described above. Students can easily log in and record themselves speaking from any computer, and a computer with a sufficiently good speech model could potentially offer immediate feedback.

In a standard classroom situation, the teacher would then be able to log in to the system, listen to the student recordings, transcribe the speech in the case of a spontaneous (rather than read) assignment, and indicate the students' mispronunciations. However, it is easy to imagine a fully automated system where students without access to language teachers could listen to their own speech and transcribe and correct it with the help of the computer system.

We believe that this type of fully automated system is fully realizable within specified conversation domains. For example, a student could record themselves saying, "wo3 shi2 san1 hao4 fan3 hui2", a phrase from the "airline travel" domain. He might then attempt to transcribe his own speech as, "wo1 shi2 san1 hao4 fan3 hui2", incorrectly marking the tone of "wo3" as "wo1". The computer would then recognize this mistranscription and suggest the correct transcription as an alternative. The student could then mark any pronunciation errors that he notices in his speech and the computer could also indicate any mispronunciations that it notices based on its internal speech model. Finally, the computer could even use correct speech from the student in the past to splice together a waveform file of what the student would sound like saying the phrase correctly.

The transcriptions and corrections obtained from computerized version of the reading assignment would also prove invaluable to speech researchers seeking to develop non-native speech models and automatic error detection systems. In particular, existing speech recognition systems based on and intended for native speakers are not as effective at understanding the heavily accented speech of language learners. Moreover, they have no ability to indicate where a language learner has mispronounced a word - an ability that would be extremely useful for language learners.

For this thesis, I developed the first version of the system described above for Mandarin Chinese learners. As a first step, I developed a transcription and correction system that simplifies and expedites the usually tedious process of obtaining corpora of non-native Mandarin Chinese speech [11] by providing a mechanism for easily transcribing and correcting spoken utterances collected as waveform files. I then combined this annotation system with a system that allows teachers to create and assign spoken reading assignments and students to record the assignments to create a basic assignment and correction system similar to the one described above.

1.1 Motivation for the Annotation System

Most current speech recognition systems depend on speech models derived from native speech. However, the hesitant and accented speech of language learners is not well reflected by those models; thus current systems generally perform significantly worse with non-native speech. Additionally, current Chinese speech models may ignore tones, even though Chinese is a tonal language where the tone of a word can change its meaning entirely. For English-speaking Chinese language learners, learning to pronounce tones correctly is frequently one of the most difficult and most important things they must learn in order to attain Chinese fluency. Since native Chinese speakers often don't enunciate or pronounce tones in quite the same way as non-native speakers, the development of a non-native tone model would be particularly helpful for CALL.

As non-native speakers are prone to making pronunciation errors, any corpus of non-native speech needs these errors to be marked. These annotations can then be used to create models of "correct" and "incorrect" non-native speech which can be used by speech recognition systems intended for non-native speakers to identify mispronunciations and respond accordingly.

One difficulty with obtaining corrected transcripts for non-native speech is that these transcripts need to be corrected by native speakers of the language, but it is difficult to find reasonable numbers of non-native and native speakers of a language living in close vicinity. We attempt to get around this problem by introducing a web-based annotation system for Mandarin Chinese where the speech files and transcription information are stored centrally, but the Chinese speech fragments can be recorded, transcribed, and corrected from any computer with a web browser and an internet connection. Thus, non-native speech can be recorded into the database by Chinese language learners anywhere in the world, and then graded by native speakers in a country where Chinese is the native language.

1.2 Background

Chinese is a character-based language which does not allow for "sounding out" words. Thus, we expect students to commit gross mispronunciation errors rather than minor errors from incorrectly sounding out words. Additionally, as Chinese is a tonal language and the non-native speakers we are focusing on speak English, a non-tonal language, we expect to see tone mispronunciation errors as well. Thus, we expect a different set of oral reading miscues than described in [32].

Both systems build substantially on the existing web based speech recognition systems that have been created by the Spoken Language Systems (SLS) group. They utilize the Galaxy system for accessing on-line information using speech [40], as well as Chinese language processing capabilities originally developed in Yinhe, its Mandarin Chinese correlate [46]. It will also build upon previous speech-enabled web-based games such as those incorporating incremental understanding [17] and various other Chinese language learning games already implemented in the Spoken Language Systems lab [39].

The ultimate objective for the assignment system is for it to be able to "listen" to a speaker as he reads a passage, and identify and correct errors. This program differs from the reading tutors described earlier [19, 31–33], as it is directed at adult second language learners and is intended to be used for both graded assignments and tutoring. We expect these differences to lead to a different pattern of reading errors.

In the case where a student stumbles, it is more likely that they will start the assignment over to get a clean recording rather than simply regressing to the first word of the sentence [43]. This is in direct contrast to a reading tutor where the goal is the understanding of the passage, not necessarily fluent speech, and the expectation would be for the student to continue stumbling over the sentence until they get it correct.

This Chinese language learning system is superior to existing language learning games because it allows teachers to tailor the assignments to different lessons and student ability levels, making it more appropriate for a classroom setting. Although the system described in [26] is fully immersive, it is limited to the domain of family relations. Other "domain" based games from the SLS laboratory are similarly restricted [7,39,47]. The underlying architecture for these multilingual computer systems is the same as the one previously described by the SLS laboratory for English-based systems [41, 48]. Johnson's Tactical Language Training System contains only proscribed lessons relevant to military personnel and is not appropriate for a standard classroom setting [22].

The assignment system also has the potential to improve existing Chinese language learning systems in the Spoken Language Systems lab. The current acoustic models behind the recognizers are based on native speakers of Mandarin. Within a few years of collecting non-native speech data with this system, it should be possible to build non-native acoustic models to improve recognition accuracy.

Chapter 2

Related Work

A substantial body of related work has provided motivation for both the annotation and assignment systems.

Hincks observed the importance of pronunciation training in language learning [21]. This key skill is sometimes neglected, resulting in "fossilization of mispronunciations" [21]. These fossilized mispronunciations, in turn, dilute speech data and increase the difficulty of automatic speech recognition (ASR).

ASR may be used to train pronunciation in second language learning, but specially trained systems are needed [35]. However, the development of such technology is difficult, as there are many issues with data collection, annotation, and evaluation [37].

2.1 Annotation

In 1998, Ehsani and Knodt laid out a program for the application of speech technology to the teaching of foreign language skills [11]. They argued that "large corpora of non-native transcribed speech data" are "one of the most needed resources for developing [...] CALL applications" [11]. They also highlighted the importance of having both read and conversational speech data. Some progress has been made towards such a data set in the last decade, but additional work is needed.

2.1.1 Read Speech

Menzel et al. have collected a corpus of non-native spoken English read speech from second-language learning exercises [1, 3, 20, 28]. Their data, obtained from Italian and German learners, is manually annotated at both the word and phone levels for pronunciation quality by trained linguist judges. Although their data are highly detailed, Menzel et al. consistently obtained low levels of inter- and intra-judge annotation agreement with best inter-annotator hit rates being around 55%. Also, the interface of the system used is complicated, requiring expert annotaters.

Slightly better agreement levels were found by Bratt et al., who assembled a large database of read Latin-American Spanish speech [4, 13]. These data were collected with a portable offline tool from both native and non-native speakers and include utterance- and phone-level annotations.

2.1.2 Spontaneous Speech

Read speech samples often contain fewer errors and more confident pronunciation than do spontaneous speech samples. Thus, read speech is less effective for training language learning systems than is spontaneous speech. Consequently, researchers have sought databases of spontaneous speech. Such samples are more difficult to obtain, as they require not only annotation but also transcription of each utterance.

In an early study, Byrne et al. compiled a large sample of English speech from Hispanic speakers [5]. Their data include both read and conversational speech with detailed, time-aligned transcriptions. The spontaneous speech data were collected from telephone conversations; it is unclear exactly how much data was obtained.

More recently, Maekawa et al. collected morphologically annotated transcriptions of Japanese speech [14, 25]. A subset of these data also include segmental and intonation labeling. Although this speech was presented spontaneously, it was prepared in advance. Thus, the speech was not "spontaneous" in the most rigorous sense, but Maekwava et al. found that even the least-spontaneous samples were "more spontaneous than typical read speech."

2.1.3 Technical Concerns

All of the studies heretofore discussed produced large data sets, but also suffer from two key difficulties. First, they established complex annotation schemes and often used complicated annotation tools. Thus, highly skilled annotators were required. Second, these studies were all conducted offline, hence transcription and annotation had to be conducted at preassigned sites.

In 2002, Milde and Gut alleviated the second of these difficulties in their compilation of German and English second language speech [30]. In this study, read, prepared, and free speech samples were obtained and partially annotated. A multi-user Web annotation toolkit called TASX was developed; this system stores XML-annotated corpus data in a relational database. Although promising, the development of TASX appears to have stagnated [38]; hence, only a small corpus was actually collected [30]. A refactored version called Eclipse Annotator has been released [2] but it is not clear that any new corpus generation projects have been undertaken using this tool.

Like Milde and Gut, we design and implement an online annotation system, allowing transcription and annotation to be conducted by native language speakers in their home countries. Our system is intuitive and easy to use, so that expert transcribers are not required. Thus, it will be possible to collect a large corpus of non-native speech samples annotated by native language speakers. Additionally—and unlike any prior studies—we focus our attention on Chinese, a tonal language.

2.2 Assignment

Significant attention has been given to the development of systems both for automated reading tutoring and for CALL. These systems are excellent sources of non-native speech utterances; however they have rarely been utilized for such purposes because of the difficulty of obtaining transcriptions.

2.2.1 Automated Reading Tutoring

Carnegie Melon University's Project LISTEN has developed an automated reading coach for children learning to read their native languages [33]. This system prompts children to read single sentences out loud and uses speech recognition to detect oral reading miscues [32,43]. A subsequent year-long study of this reading tutor showed that students trained by the speech recognition-based system improved in most reading skills [31]. These improvements were nearly as great as those observed in children working with a human tutor and substantially greater than those seen in untutored students. While the enforced break between sentences makes speech recognition easier and creates an easy location to stop and prompt students, it is not necessary. Indeed, Hagen et al. integrated children's speech recognition into a real-time interactive literacy tutoring system that did not require unnatural pauses between sentences [19].

2.2.2 Language Learning Games

Some research has attempted to capture salient mispronunciations and language skills of foreign language speakers. Meng et al. identify major phonological disparities between the native language (Cantonese) and the foreign language being studied (English) and use them to effectively derive salient mispronunciations [27]. Similarly, Chandel et al. have developed Sensei, a web-based application that assesses the English skills of call center agents. The correlation between Sensei's analysis and that of human assessors compares favorably to the correlation between human raters' analyses [6].

Existing speech-enabled language learning systems are of particular relevance to this project. The Tactical Language Training System—designed to give military personnel the language skills required to carry out civil affairs missions—teaches a foreign language via interaction with AI characters in an elaborate computer world via speech and mouse gestures [22]. This system also provides an AI tutor that gives language assistance and feedback.

The Spoken Language Systems (SLS) group at MIT's Computer Science and Artificial Intelligence Laboratory (CSAIL) has also been involved in the development of language learning games. Their early work involved Language Tutor, a basic language learning system in which students learn correct pronunciation by listening to and repeating correctly spoken words. This system provides automatic feedback on students' pronunciation [48]. More recently, CSAIL SLS has developed a series of language learning games focused on specific domains [26, 41]. These games might be thought of as a family of language learning lessons incorporating speech recognition and dialogue. Current games in this family include a dialogue based system in the "weather" domain [41] and an immersive dialogue system in the "family" domain appropriate for pre-beginners [26].

Translation games comprise another class of speech-enabled language learning games developed by the SLS group [7, 47]. These games present students with a list of sentences in their native language and prompt students to provide spontaneous translations of the sentences into the language being learned. Pimsleur advocates this learning approach [36]. Translation type game developed by SLS include an interactive interpretation game in the "activity scheduling" domain [7] and a similar game in the "airline travel" domain [47]. We use the spontaneous utterances recorded from the user study of the "airline travel" game to test the annotation system in this thesis.

2.2.3 Opportunities for Annotation

Ehsani and Knodt describe current uses of speech technology in CALL and suggest further directions for speech-enabled CALL systems [11]. They describe the importance of reading coaches and other potential language learning games. They then explain that a large database of non-native transcribed speech is one of the most essential and needed resources for developing CALL systems. The dearth of such databases is a problem that remains true to this day and therefore continues to confound work on CALL systems. Our system offers a potential solution to this problem: eventually, it may provide an automatic corpus of transcribed non-native speech without the labor intensity required by current methods.

Finally, Luis von Ahn has developed web-based games which motivate people to engage in the otherwise tedious task of labeling online images [44, 45]. If enough people play such games, then a substantial corpus of images may be labeled in a short period of time. Our system is partially motivated by this idea—individuals will perform a labeling task because it provides them the personal benefit of stored and organized information.

Chapter 3

Annotation System

This chapter describes an online system for correcting and transcribing non-native Chinese speech segments. This system enables a set of waveform files to be easily corrected and transcribed by native speakers with internet access anywhere on the world. The corresponding corrections and waveform files are stored in a centralized location for ease of use and access by speech researchers seeking to develop non-native speech models.

This system thus allows speech recognition researchers to easily obtain a corpus of corrected non-native speech by streamlining the correction process and avoiding the need to find local non-native and native speakers of a language.

3.1 Back End

The web-based transcription and annotation system builds on several other systems already in existence. We use the Eclipse Integrated Development Environment (IDE) [10] with CVS version control [9] to organize and program our code base, which is mostly written in Java [29].

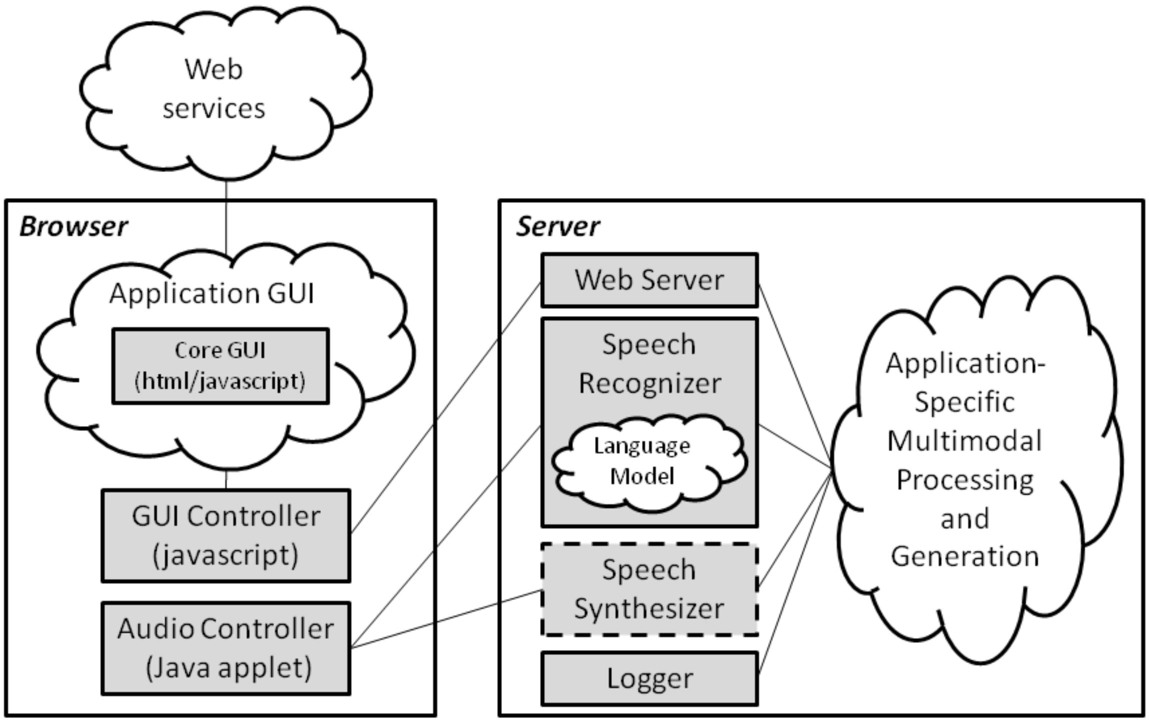

Figure 3-1: The WAMI architecture [18]

3.1.1 Asynchronous Javascript and XML (AJAX)

The annotation system is programmed using Asychronous JavaScript and XML, or AJAX, in order to achieve an interactive web environment. Our system uses the Web-Accessible Multimodal Interfaces (WAMI) toolkit (formerly ajaxgui) developed by the Spoken Language Systems group [17, 18, 39]. The WAMI architecture is depicted in figure 3-1.

We use Apache Tomcat version 5.5 [12] as the webserver, and incorporate the GALAXY system for audio processing [40]. For core user interface development, we use the Google Web Toolkit (GWT) [15] with the MyGWT java library [34] to automatically produce AJAX code from Java.

3.1.2 Database

The original speech segments are stored as ".wav" files on a centralized computer server along with a basic transcription of each speech file if it exists. Further annotations and transcriptions made by the transcribers are stored in a PostgreSQL database along with the user id of the transcriber who made each annotation [16]. The PostgreSQL database is accessed and changed using standard SQL commands generated in Java. Additionally, some administration of the database occurs using the web-based administration tool, phpPgAdmin [42].

3.1.3 Translation Protocols

The transcriptions for our annotation system are stored phonetically as tone-marked pinyin. However, most native speakers of Chinese prefer to read Chinese characters, making a pinyin to characters translation protocol highly desirable. Unfortunately, pinyin to character translation is extremely difficult as there are nearly 5,000 characters in regular use in the Chinese language. Moreover, most of these characters are homonyms. For example, the pinyin word “dao4” could reference any one of seven different characters and the pinyin “dao3” which differs only in tone from “dao4” can also reference seven different characters - two of which are the same as those represented by “dao4”.

We accomplish pinyin to character translation in our system by restricting our domain of possible words to those in the “airline travel” domain from which most of the waveform files that we are currently interested in annotating derive. Additionally, we try to match larger phrases first which makes it more likely that we translate the pinyin phonemes into the correct character rather than one of its homonyms.

3.1.4 Galaxy

The annotation system builds substantially on existing web based speech recognition systems that have been created by the Spoken Language Systems group. Most notably, it utilizes the Galaxy system for accessing on-line information using speech [40], as well as Chinese language processing capabilities originally developed in Yinhe, its Mandarin Chinese correlate [46].

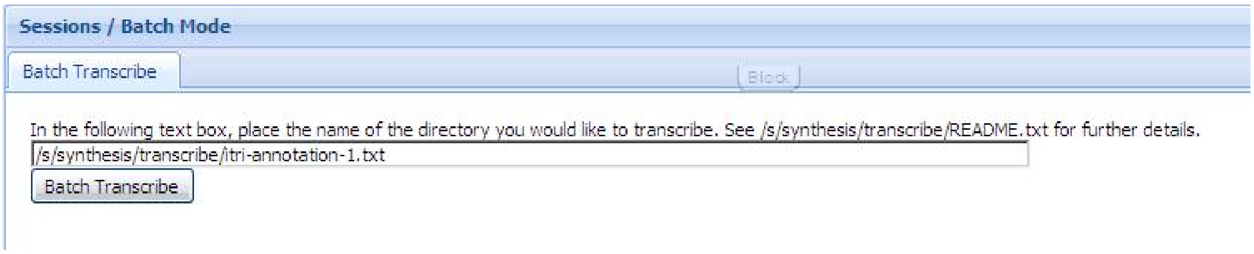

Figure 3-2: The transcriber inputs the file referencing the wav files that they would like to annotate

3.2 User Interface

Transcribers access our web-based annotation system by navigating to our login site and logging in using an assigned username and password linked to a randomly assigned id number. Each transcriber then navigates to the batch transcription page where they select the file referencing the waveform files that they would like to annotate or transcribe (figure 3-2).

The waveform files are loaded up into a table along with any matching transcription files that are found. If a matching transcription file has been found, the transcriber can select the utterance in the table, and cause the transcription to appear in both pinyin and characters underneath the table, as in figure 3-3. We refer to this view as the annotation view. The annotator can then play the associated speech file by selecting the "Play Audio" button underneath the transcription. Clicking a word in the transcription will cause a red 'X' to appear, indicative of a mispronunciation error.

Otherwise, a simple transcription interface appears, enabling the transcriber to listen to the audio file and add a transcription to the database, as in figures 3-4 and 3-5. Selecting the "Transcription Mode" button will then load the newly created transcription into the annotation view described in the previous paragraph (figure 3-6).

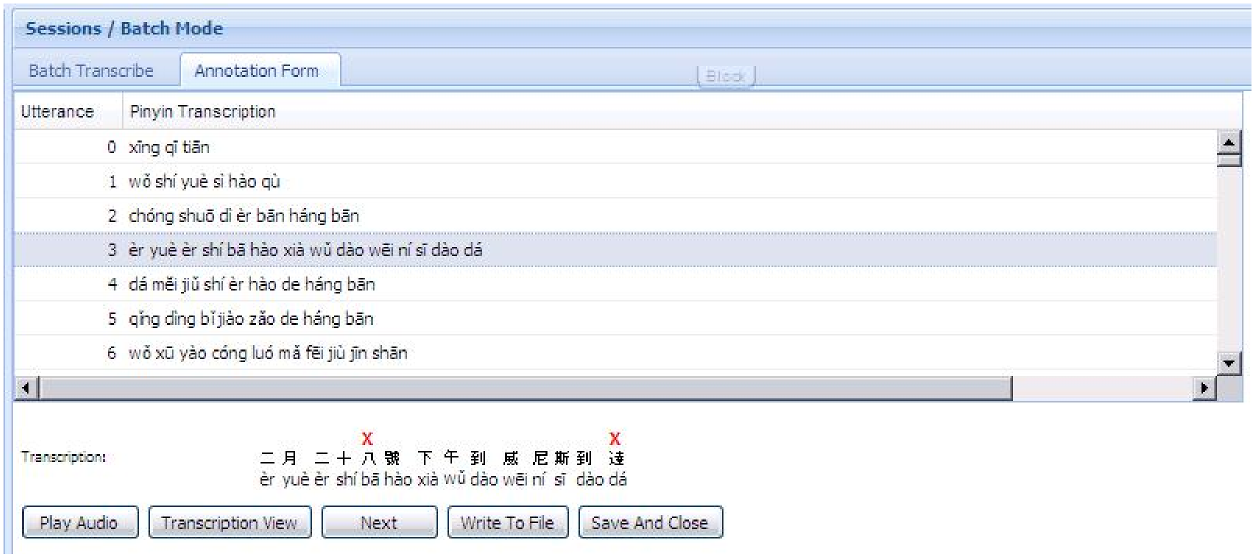

Figure 3-3: The speech files are loaded and the transcriber annotates.

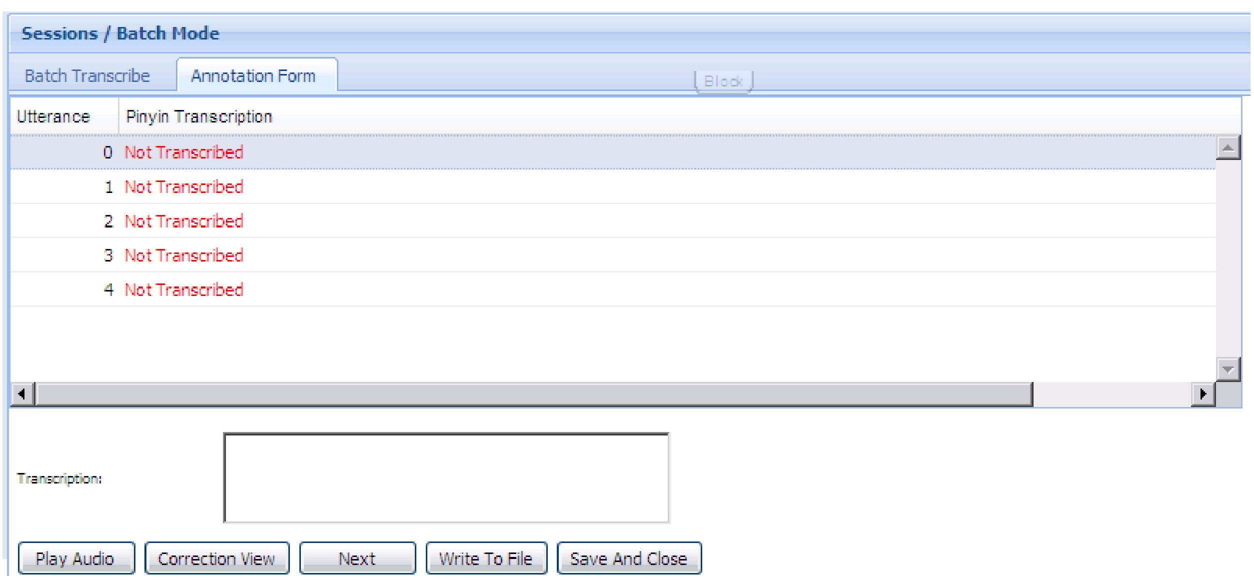

Figure 3-4: A transcription interface appears if there is no pre-existing transcription.

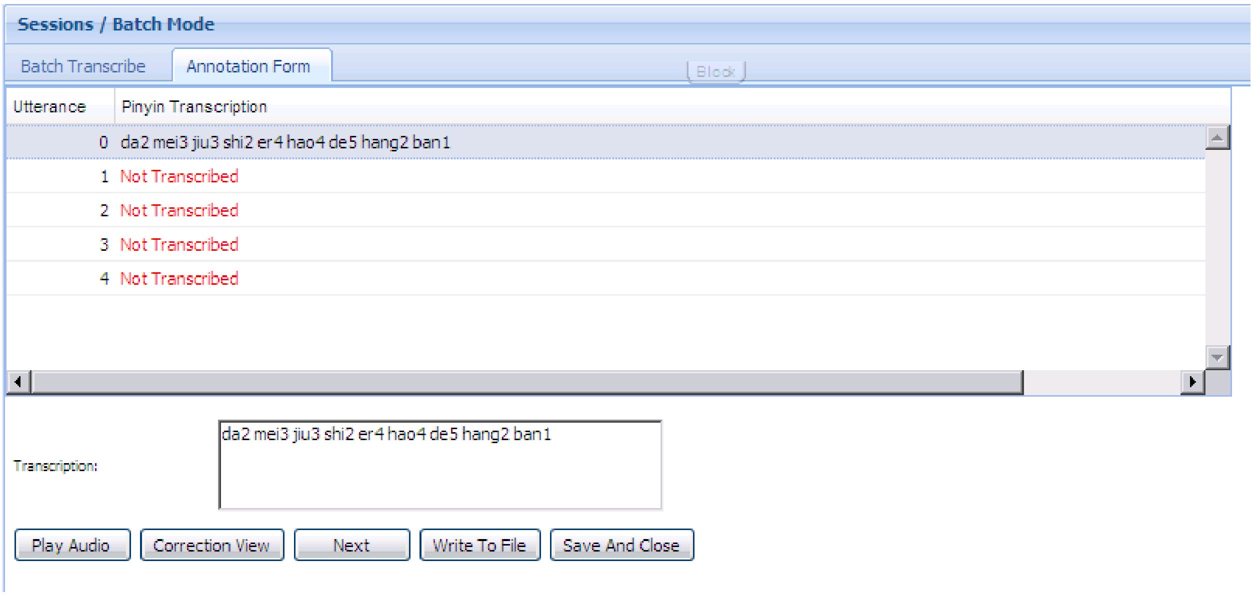

Figure 3-5: The transcriber adds a transcription to a previously untranscribed speech file.

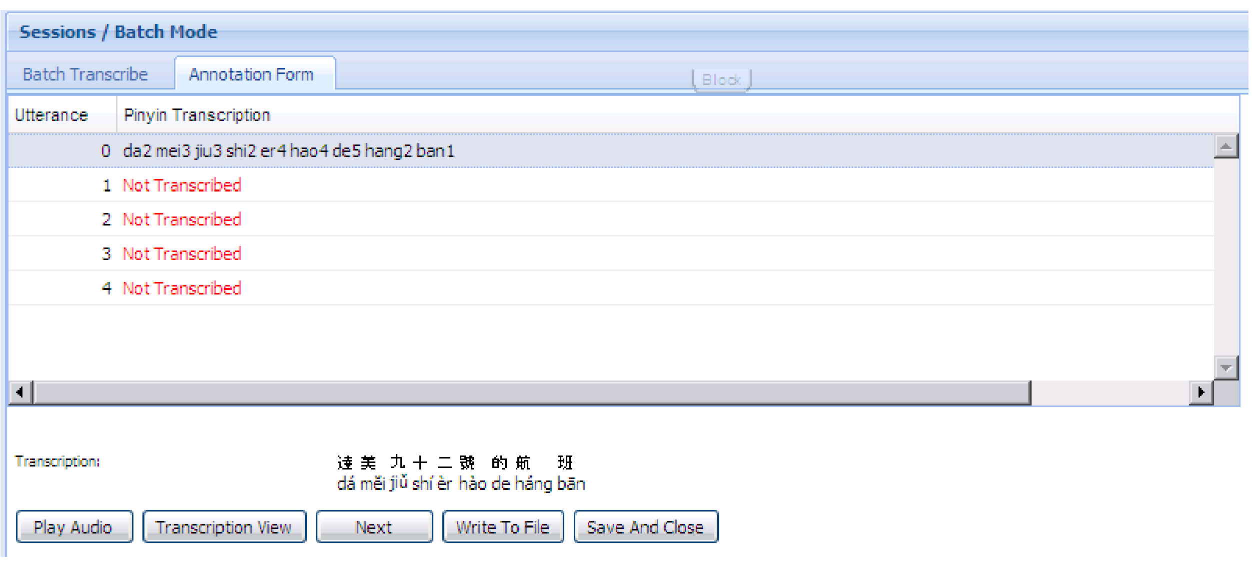

Figure 3-6: The added transcription is loaded for correction.

Chapter 4

Chinese Spoken Reading Assignments

This chapter describes an online system for spoken reading assignments intended to replace the current, paper-based system. This system is easier to access and use for both students and teachers.

The system will also allow for more effective language learning by supplying pronunciation information and timely feedback. Our eventual goal is to develop a tool capable of automatically identifying pronunciation errors. This system will also be beneficial to the Chinese learner in a non-classroom setting, as they would be able to upload and record their own paragraphs and get useful pronunciation feedback.

Finally, this system allows speech recognition researchers to obtain a corpus of labeled non-native speech - complete with an indication of the most egregious pronunciation errors. Our system puts the work of labeling the speech on Chinese teachers who would be doing the task anyway.

To put this in another perspective, we are hoping to entice Chinese teachers to use this system extensively in the classroom, because it serves their needs much better than the tools they are currently using. As a crucial side benefit, they will be providing annotated digital data that will be invaluable to researchers hoping to further automate some of their tasks, such as identifying mispronounced phonemes or tones. Ultimately, if computers can be effective in these decisions, students working individually outside of the classroom will be able to obtain valid feedback even in the absense of a human teacher's support.

4.1 Back End

The assignment system shares much of its back end and framework with the transcription system as described in section 3.1. Like the transcription system, it is a web-based system developed using the WAMI framework [17, 18, 39]. In addition to the framework used in the annotation system, the assignment system also makes use of the Galaxy system as a basis for both speech recording and playback - the annotation system uses the Galaxy system only for playback [40, 46]. We also build on previous speech-enabled web-based games such as those incorporating incremental understanding [17] and various other Chinese language learning games already implemented in the Spoken Language Systems lab [39].

4.1.1 Translation Protocols

The assignment system expects Chinese character input that must be translated to phonetic pinyin with tones. This differs from the annotation system which incorporates pinyin to character translation within a small domain as described in section 3.1.3. Character to pinyin translation is difficult because of the exceedingly large number of characters in the Chinese language (around 50,000, although only 5,000 or so are in regular use). Moreover, characters may be pronounced differently depending on their location in a phrase. We accomplish the automatic translation of Chinese characters to pinyin phonemes via the Adsotrans system developed by David Lancashire [23].

4.2 User Interface

The login screen for both the teacher and the student interfaces is the same as that for the transcription system; however the subsequent screens differ depending on whether the user is a teacher or a student.

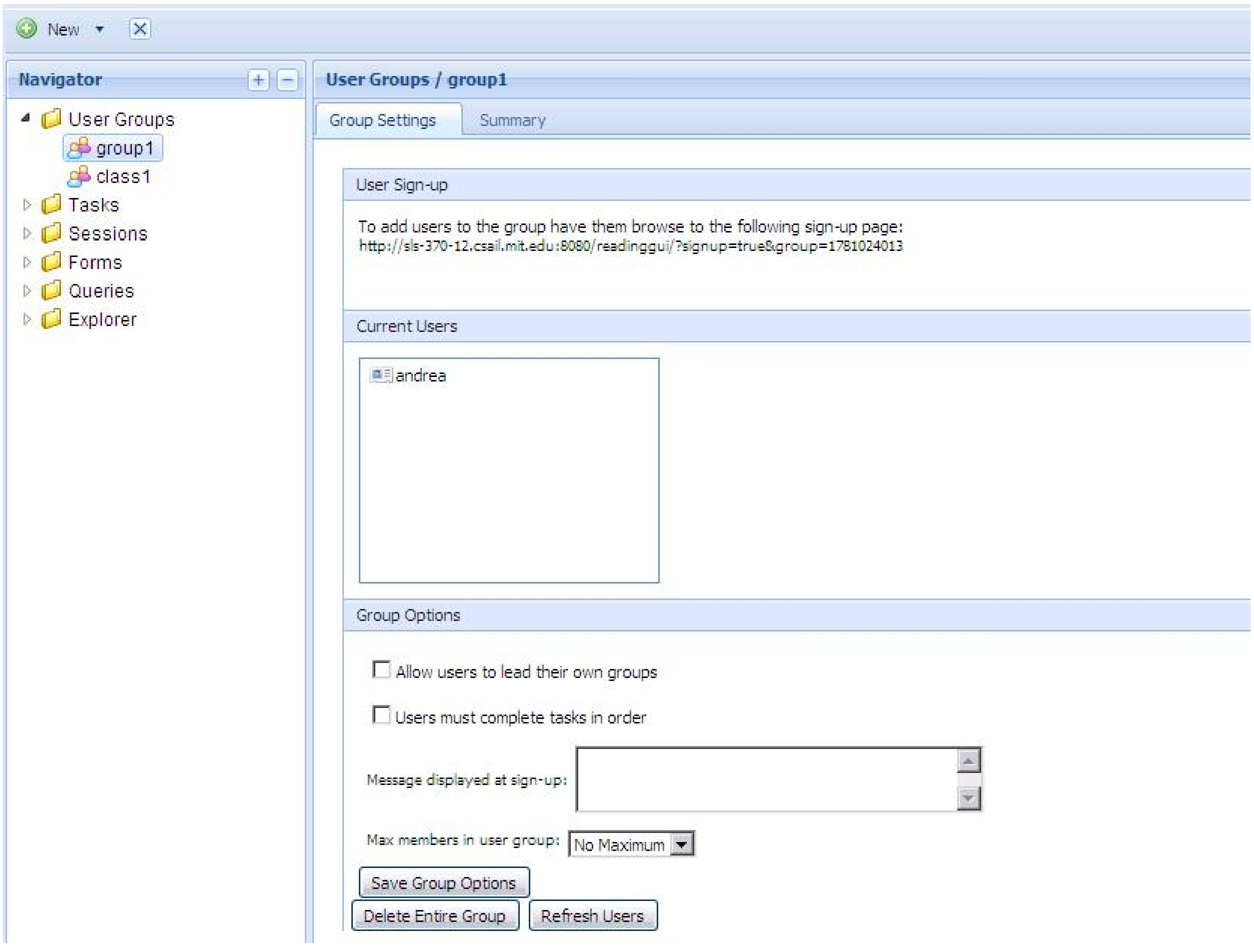

Figure 4-1: The group creation interface

A teacher has the ability to create "Groups" or classes which students can join, as seen in figure 4-1. She can then create reading assignments to be read by her class by typing or copy-pasting the paragraph to be read into a textbox (figure 4-2). Our system automatically converts her paragraph to a tone-marked pinyin format using Adsotrans [23]. Teachers then have the opportunity to review and fix the pinyin before saving the assignment onto our system (figure 4-3). This pinyin also serves as a phonetic transcription for the purpose of developing a non-native speech corpus.

When a student logs in to the assignment system, they see a list of the assignments that their teachers have assigned to them (figure 4-4). If they select an assignment, the paragraph for that assignment, in Chinese characters, is displayed along with a green "Hold

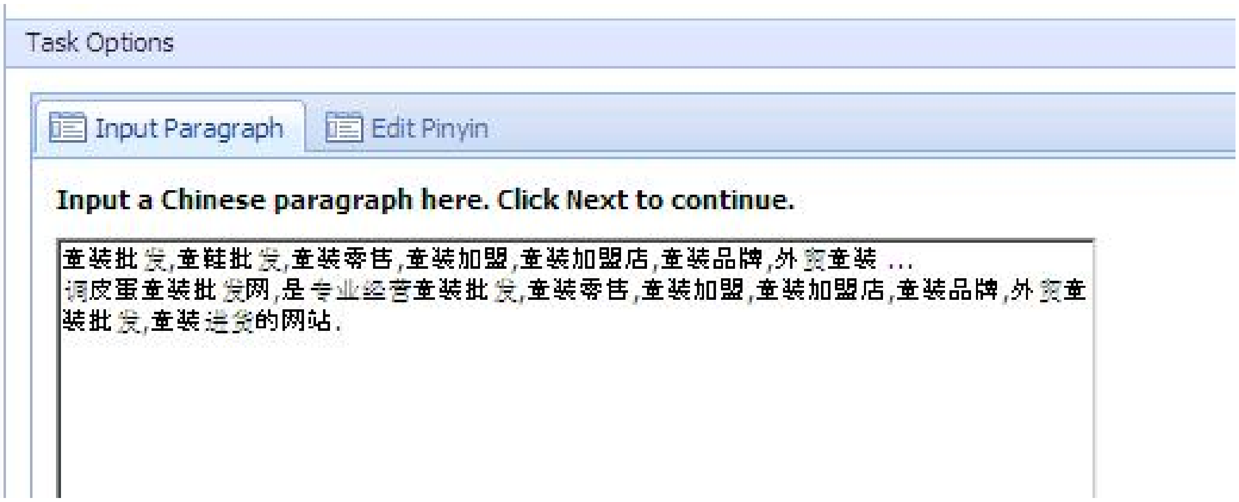

Figure 4-2: The teacher inputs a Chinese paragraph for the reading assignment

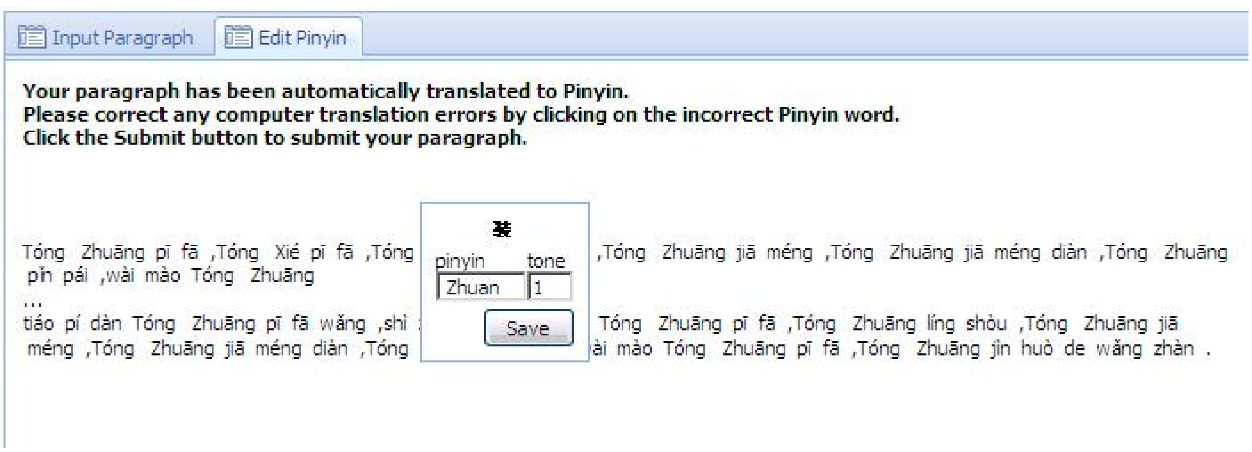

Figure 4-3: The teacher edits the automatically translated pinyin

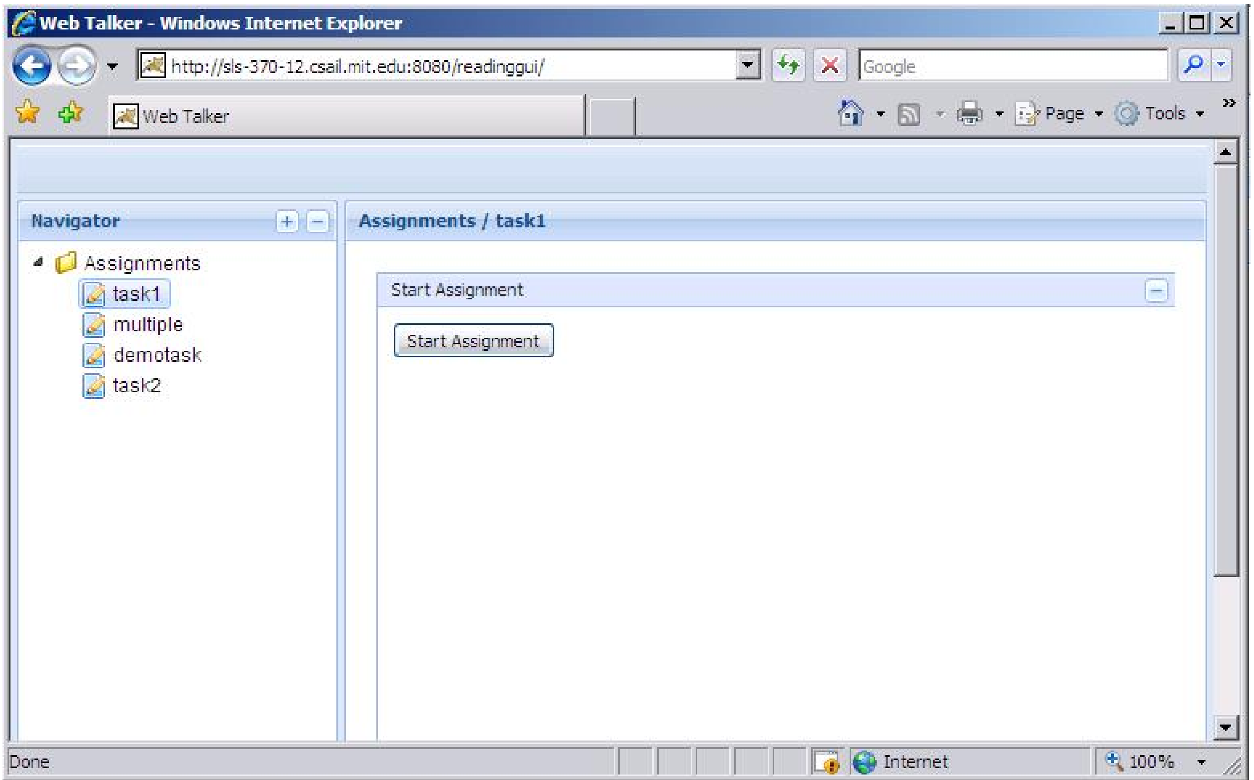

Figure 4-4: When the student logs in, he sees a list of the assignments he needs to complete

to Talk" button (figure 4-5). Students press the button at the start of their utterance and may re-record as many times as they wish.

Once the students have recorded assignments, the teachers can log in to see a list of which students have recorded audio files and to grade the assignments. Teachers can select and listen to the student recordings and click on incorrect characters to indicate mispronunciations. This correction interface is similar to that for the transcription system, but provides more information about the specific error committed by each student. Rather than just an "X" appearing above incorrect characters, various specific types of errors can be selected. A single click indicates a tone error and causes the correct tone to appear above the character, two clicks indicate a non-tone pronunciation error causing the correct pronunciation to appear, three clicks indicate that both tone and pronunciation were incorrect, and four clicks returns the character to the original correct state (figure 4-6). As in the transcription system, teachers can use this interface to quickly listen to and correct a spoken assignment.

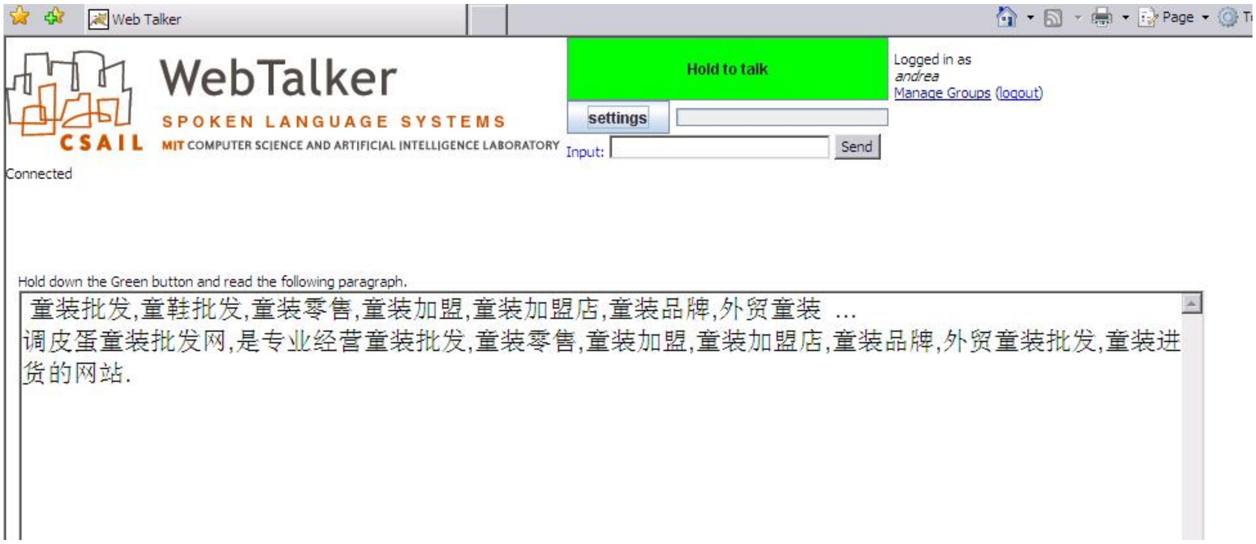

Figure 4-5: The student clicks the button and reads the paragraph to perform the assignment

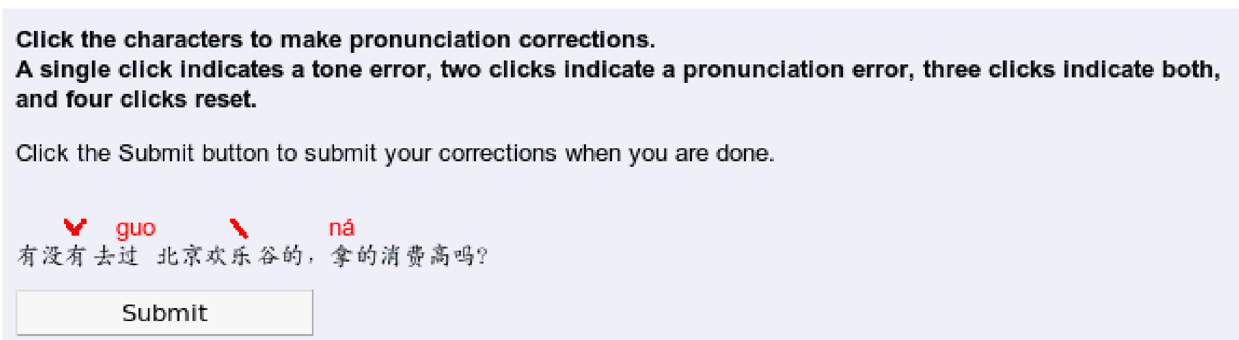

Figure 4-6: Example of a teacher correcting student pronunciation errors.

These corrections can then be made available to researchers working on automatic language assessment – for training and testing the technology they develop.

Chapter 5

User Study

We tested the annotation system using previously transcribed non-native speech recordings gathered from the user study of a spoken translation game developed by Chao Wang and Stephanie Seneff [47]. In this study, 12 Chinese language learners translated a series of sentences in the domain of airline travel from English to Chinese. Students were not generally given a Chinese translation of the sentence to read unless they needed and prompted the system for help; thus, most of the utterances that we study here are spontaneous rather than read.

These utterances had been previously manually annotated by a native Chinese speaker without the use of this annotation system. For this user study, we sought to test the efficacy of our annotation system and determine inter-rater agreement for the annotations. Several native Chinese speakers used our annotation system to annotate the utterance set, and we calculated Fleiss' and Cohen's kappa agreement scores between these raters and the original manual annotation.

5.1 Methods

As a preliminary test of our system, we recruited two native Chinese speakers from within the lab to annotate a small sample of 76 and 94 utterances respectively from the corpus using the annotation system. This test was used to uncover any previously undetected bugs in the system and make sure that the system worked as expected when being used remotely by multiple users.

The revised system was then used by one remote annotator in Michigan and two remote annotators in Taiwan to annotate the same set of 250 utterances. Annotators were provided with basic directions for using the system, as seen in Appendix A, and instructed to ask questions and give comments as necessary. These annotators were all native Chinese speakers. The annotator from Michigan had spent the first 22 years of her life living in Taiwan and currently teaches Chinese to local children, while the annotators from Taiwan currently live in a Chinese speaking country.

The annotations obtained were converted to special annotation files ("tanno" files) from the SQL database using a perl script (Appendix B). Additional perl scripts were then used to calculate Fleiss' and Cohen's Kappa scores based off of both the preliminary annotations, the later remote annotations, and the set of annotations to make judgements about the agreement between raters (Appendix C). Kappa scores were calculated on the syllable or phoneme level with the annotation of each pinyin word considered to be a single trial. Thus, the utterance "da2 mei3 jiu3 shi2 er4 hao4 de5 hang2 ban1" consists of nine trials even though "hang2 ban1" might be considered a single Chinese word by some metrics.

All annotators, including the preliminary annotators, made corrections on the same set of 250 utterances; however the preliminary annotators transcribed only the first 94 and 76 utterances of the set, respectively. The total number of trials and utterances transcribed by each annotator can be seen in table 5.1. We refer to the annotator who had previously annotated the utterances without my tool as 'previous', the two preliminary annotators as 'prelim1' and 'prelim2', and the three final annotators as 'final1', 'final2', and 'final3'.

5.2 Results

Fleiss' and Cohen's Kappa scores are both standard metrics for inter-rater agreement. Fleiss' Kappa is used to calculate agreement between any number of raters, while Cohen's Kappa

Annotator	**Trials**	**Utterances**
previous	>2173	>250
prelim1	790	94
prelim2	625	76
final1	2173	250
final2	2173	250
final3	2173	250

Table 5.1: Total number of trials and utterances annotated by each annotator.

κ **Value**	*Interpretation*
< 0	Poor agreement
$0.0 - 0.20$	Slight agreement
$0.21 - 0.40$	Fair agreement
$0.41 - 0.60$	Moderate agreement
$0.61 - 0.80$	Substantial agreement
$0.81 - 1.00$	Almost perfect agreement

Table 5.2: Landis and Koch's Table for Interpreting Kappa Scores

can only be used to calculate agreement between exactly two raters. By using both metrics we are able to see the overall observed agreement level between all raters while also observing whether or not some pairs of raters tend to agree more than others.

For both the Fleiss' Kappa and the Cohen's Kappa metrics, a score of 1 indicates perfect agreement, a score of 0 indicates that the level of agreement seen is no more than would be expected by chance, and any score less than 0 indicates negative agreement. Landis and Koch proposed the divisions shown in table 5.2, for interpreting kappa scores [24].

We used Fleiss' kappa to get the agreement level between all four of the raters that annotated all 250 utterances: This kappa score is 0.36. When the two preliminary raters are added, causing the number utterances being considered to decrease, the score increases slightly, to 0.38.

We then used Cohen's kappa to get the agreement measures between each set of two raters, including the preliminary raters. These kappa scores can be seen in table 5.3. The scores range between 0.22, indicating fair agreement, and 0.53, indicating moderate agreement. We note that the smaller number of utterances available for the preliminary raters does not appear to have substantially changed the kappa scores.

	previous	**prelim1**	**prelim2**	**final1**	**final2**	**final3**
previous		0.50	0.47	0.48	0.39	0.33
prelim1	0.50		0.53	0.38	0.34	0.27
prelim2	0.47	0.53		0.47	0.35	0.26
final1	0.48	0.38	0.47		0.45	0.26
final2	0.39	0.34	0.35	0.45		0.22
final3	0.33	0.27	0.26	0.26	0.22	

Table 5.3: Cohen's kappa scores.

From the Cohen's kappa analyses, we observed that the five worst inter-rater agreements all involved the annotator 'final3'. Based on this observation, we tried removing the rater 'final3' from the Fleiss' kappa analysis. The Fleiss' kappa score over all 250 utterances for the three remaining raters when 'final3' is not considered is 0.44, an improvement of 0.08 over the kappa score when 'final3' is included. The Fleiss' kappa score omitting 'final3' but including the preliminary raters improves to 0.41, just over the 'moderate agreement' level.

5.3 Discussion

This study accomplished two major objectives: It demonstrated the feasibility of remote annotators and provided us with inter-rater agreement measures for the annotation of non-native Chinese pronunciation.

5.3.1 Remote Annotation

All our annotators accessed the system remotely by browsing to a specific website in their browser of choice, generally either Internet Explorer 7.0 or Firefox 3.0. However, some annotators tested the system from within Cambridge, Massachusetts, where the audio files are being stored, while other annotators used the system from Michigan or Taiwan. We successfully demonstrated that distance from the central server did not substantially affect the ease of annotation. In fact, one of the raters in Taiwan finished annotating all 250 utterances in about two hours.

This has important implications for the annotation of non-native speech in general, as it demonstrates that such speech can be annotated by native speakers even if there are very few native speakers available locally. Additionally, the annotators can make their ratings without having to download and learn to use specialized software programs.

Unfortunately, we also observed that the annotators who used Internet Explorer tended to have more problems than those using Firefox, particularly with connecting the server and playing audio files.

5.3.2 Inter-rater Agreement

Although we know of no previous studies that have compared inter-rater agreement for assessing the pronunciation of a tonal language, studies that have tried to assess inter-rater agreement for the pronunciation of non-tonal languages have generally had relatively poor agreement. For example, Menzel et al. found inter-judge hit rates of between 30% and 55% for phone-level annotations of read, non-native English speech [28]. Thus, we hypothesized that it is probably diffficult to calibrate two people on the decision of what constitutes "good" and "bad" tones, particularly when annotating spontaneous speech, which is more likely to have all types of disfluencies.

This hypothesis was born out by our study, where we observed kappa scores for inter-rater agreement in the 'fair' and 'moderate' agreement ranges proposed by Landis and Koch [24]. In particular, we found Cohen's kappa scores ranging between 0.22 and 0.53 and a Fleiss' kappa score of 0.36. After dropping the ratings made by the rater, 'final3', whose ratings have comparatively low correlation to all of the other raters, our Cohen's kappa scores improve to between 0.34 and 0.53 and the Fleiss' kappa score improves to 0.43.

Bratt et al. studied inter-rater agreement in rating non-native Spanish and French speech, but looked only at sentence-level annotations, where they found higher agreement rates, with agreement scores around 0.75 [4]. This is unsurprising, as it is presumably easier to identify that an error exists in an utterance than to determine which specific phoneme was the source of the error. Although we did not compare inter-rater agreement on the utterance-level, we would also expect higher agreement rates on that level.

Chapter 6

Future Directions

As with all research systems, our annotation and assignment systems are works in progress, and we envision a number of potential avenues for improving and expanding upon these systems.

6.1 Future Directions for the Annotation System

Although the transcribers in our user study each graded a significant portion of the utterances from Chao Wang's spoken translation game [47], there are still a large number of utterances from both that study and other studies done by our lab that have yet to be annotated. Completing the transcription and annotation of these utterances will vastly increase the size of our non-native Chinese speech corpus. This, in turn, will aid in the development of speech recognition and error detection systems for non-native Chinese speech.

We also anticipate that our annotation system may be implemented in any number of language learning systems similar to the assignment system that make use of and allow us to obtain corpora of transcribed non-native speech. For example, the Advanced Placement exams for foreign languages have speaking sections where students are given a short (between 20 and 90 second) time frame to respond verbally to an oral or visual prompt [8]. Most classes preparing for these exams will have similar assignments. Implementing a web-based version of this assignment would provide an interesting source of spontaneous non-native speech. Requiring students to transcribe their speech after the fact, using our annotation system, makes sense from a pedagogical point of view as it forces students to listen carefully to themselves in order to identify their own errors.

6.2 Future Directions for the Assignment System

There are a number of user interface changes that we believe would improve our reading assignment system. We would like to modify the basic interface for recording an assignment such that the current sentence being read by the student is in a larger font and the other sentences are grayed out. Students would then be expected to select each sentence as it is being read. Additionally, we plan to further improve the user friendliness of the system by adding a prompting feature. Selecting a specific character in the paragraph should result in a prompt of the correct pinyin pronunciation of the character, possibly with the option to hear the pinyin pronounced by a native speaker.

The graying out of text not in the current sentence encourages the students to remember to click the sentence they are reading while still letting them see the context of the sentence they are reading. By separating each sentence in this way, we will make it easier to correctly align the spoken words with the sentence, detect, and correct student errors in the future. After finishing the recording, students should have the option to playback all or part of the recording and decide whether they are ready to submit the assignment.

A user study is also necessary to determine the efficacy and ease of use of our system. We would like to create a small study where some Chinese students are given assignments using our system. The user study will allow us to get feedback on the accessibility and user friendliness of our system compared to the tools that the students were previously using for their spoken reading assignments, and to get suggestions for improvements to our system. It will also present an opportunity for us to solicit input from students and teachers on a more complicated reading system that provides automatic feedback on likely speech errors.

Following the small user study, we would like to implement the assignment system on a larger scale. This would enable us to collect and analyze large quantities of speech and corrections data from the system. This speech data could then be automatically aligned to the correct words in the assigned paragraphs. By comparing the alignments to the corrections data from the teachers, we should be able to create a tool that can identify correct and incorrect non-native speech.

By integrating the automatic error identification tool into our spoken reading assignment system, we will be able to provide instantaneous feedback to students in the case an error is committed, perhaps by highlighting the incorrectly pronounced characters in red. Students would then be able to replay the offending sentence and view the correct pinyin of the mispronounced character as an aid to improving their pronunciation and overall performance on the assignment.

The spoken reading assignment described in this proposal is a type of assignment in frequent use in the language learning community. However, with the advent of computerized speech recognition, many more exciting speech-enabled language learning systems are possible [26]. In the future, we would like to incorporate this assignment into a broader system of speech recognition enabled language learning games that could also be used as spoken assignments.

Chapter 7

Conclusions

In this thesis we described and implemented two related systems for Chinese language learning research. The annotation system provides an easy to use interface for transcribing and annotating Mandarin Chinese from remote computers. The assignment system incorporates our annotation system into an interface for basic Chinese reading assignments. Both systems provide a mechanism for language researchers to collect corpora of non-native speech.

7.1 Annotation System

Our annotation system provides an easy-to-use user interface for transcribing and annotating collections of non-native Chinese speech. Additionally, the system enables the audio and transcription files to be stored in an entirely different location from the annotators. Thus, a set of utterances can be recorded by a Chinese language learner in the United States and graded by a native speaker living in Taiwan or China.

Gauging the pronunciation of speech is not a binary, cut-and-dry decision, as demonstrated by the relatively low annotator agreement rates in previous studies. We performed a user study using our annotation system to gauge inter-annotator agreement rates for non-native speech annotation. This study demonstrated that raters tend to show fair to moderate levels of inter-rater agreement, with a kappa scores around 0.4 for both Fleiss' and Cohen's kappas. This rate is similar to that seen in studies of other languages.

The agreement level among humans obtained here provides a benchmark for judging the success of an automated annotation system – if it can realize a similar agreement level when compared with the human annotations, then it can be inferred that it should perform comparably to humans in the task.

7.2 Assignment System

Our online system for spoken Chinese language assignments simplifies the process of updating and grading the assignments for teachers. It also makes it easier for students to complete these assignments as they can access the system easily from any computer.

Future versions of the system have the potential to offer immediate, automated feedback in response to the students' readings, which we expect to be more effective for language learning than the delayed feedback of the standard spoken reading assignment.

Finally, our assignment system is integrated with the previously described annotation system. This creates a mutually beneficial system where teachers can give and grade standard reading assignments while language researchers collect a corpus of labeled non-native speech, which can be used to develop better language learning software.

Appendix A

Directions for User Study

Directions for Using the Chinese Annotation Tool:

Go to http://sls-370-12.csail.mit.edu:8080/readinggui/

Log in.

Select the arrow next to the "Sessions" folder. Select the "Batch Mode" option

In the text box enter the name of the file you are interested in transcribing. It should default to `/s/synthesis/transcribe/itri-annotation-1.txt` Within the `/s/synthesis/transcribe/` folder there are 7 files we are interested in transcribing, itri-annotation-1.txt – itri-annotation-7.txt

Click the "Batch Transcribe" button. This will load up all of the files referenced in the file that you selected. This will probably take a while, as there are a couple hundred wav files associated with each text file.

Select the utterance that you are interested in. The utterance should appear below the table in both pinyin and traditional chinese characters. The characters are automatically translated and may not always be correct. If they don't seem to line up correctly with the pinyin, you should trust the pinyin and ignore the characters for that utterance.

Click "Play Audio" to play the wav file associated with the utterance you have selected. In the event of a pronunciation error, clicking on the incorrect pinyin word once will put an **X** above that word marking it as wrong. Clicking again will remove the **X**.

To get to the next utterance, click the “Next” button or select it in the table.

Feel free to contact me at hawksley@csail.mit.edu if you have any additional questions.

Appendix B

“tanno” Files

Special text files referred to as “tanno” files were used to store annotation information particularly for the purposes of calculating agreement measures. Each tanno file is specific to one annotator and one utterance. The name of each tanno file is of the format, utterance.annotator.tanno, indicating which utterance and annotator is associated with it. The contents of each tanno file are a transcription of the utterance associated with it with asterisks indicating which words in the utterance had been marked as incorrect by that annotator.

As an example, consider the hypothetical tanno file, helloworld.hawksley.tanno. The utterance associated with this tanno file would be called helloworld.wav, and the associated annotator would be ‘hawksley’. The contents of helloworld.hawksley.tanno might be “da4 jia1* hao3”. These contents indicate that the helloworld utterance consists of three pinyin words, and that the annotator hawksley believes that the second word, jia1, has been mispronounced.

This file can be easily compared to other tanno files also associated with the utterance helloworld, but annotated by different annotators, such as helloworld.andrea.tanno.

Appendix C

Scripts for calculating Kappa Scores

We measured agreement using both Fleiss' Kappa and Cohen's Kappa for the agreement measures. Fleiss' Kappa is used to calculate agreement between any number of raters, while Cohen's Kappa is used to calculate agreement between only two raters. The Kappa scores were calculated using the following perl scripts, written with the assumption of a binary rating system.

C.1 Fleiss' Kappa

```
#!/usr/local/bin/perl

use strict;

my $dirtoget="/s/summit/wavs/mandarin/scill/ftgame/anna/";
opendir(IMD, $dirtoget) || die("Cannot open directory");
my @thefiles= readdir(IMD);
closedir(IMD);

my %raters;      #list of raters
my %files;       #key = files, value = num of raters who have rated the file
my $numfiles;

my %ignorelist; #These raters should be ignored
```

```
$ignorelist{"hawksley"} = 1;

foreach my $f (@thefiles) {
    unless ( ($f eq ".") || ($f eq "..") ) {
      if ($f =~ m/.tanno$/) {
        my @filename = split('\.', $f);

        if (!exists($ignorelist{@filename[1]})) {
          if (exists($files{@filename[0]})) {   # increments the number of raters
                                                # who have rated a given file
            $files{@filename[0]} = $files{@filename[0]} + 1;
          } else {
            $files{@filename[0]} = 1;
          }
          $raters{@filename[1]} = 1;    # Adds a rater to the hash, nothing
                                        # happens if the rater is already there
        }
      }
    }
}

my $numraters += scalar keys %raters;
my @kappatable;
my $tablelength = 0;

while ((my $filename, my $count) = each %files) {
    if ($count == $numraters) {
      $numfiles++;
     my $length = 0;
     while ((my $rater, my $exists) = each %raters) {

      open(DAT, $filename . "." . $rater . ".tanno") || die("Could not open file!");
      my @lines=<DAT>;
      close(DAT);

        $tablelength = $tablelength - $length;

        foreach my $line (@lines) {
          my @words = split(' ', $line);
          $length = @words;

          foreach my $word (@words) {
```

```
            if (!exists($kappatable[$tablelength])) {
              $kappatable[$tablelength] = 0;
            }

            if ($word =~ m/\*$/) {
              $kappatable[$tablelength] = $kappatable[$tablelength] + 1;
            }

            $tablelength = $tablelength + 1;
          }
        }
      }
    }
}

my $totalnumratings = $numraters*($tablelength);
my $p1=0;   # column 1 probability = sum of first row/total num ratings
my $p2=0;   # column 2 probability = sum of second row/total num ratings
my @p;      # row probabilities = 1/(numraters*(numraters-1))*
            #  (first column^2+second column^2-numraters)
my $sump=0; # sum of row probabilities
my $po;     # observed probability
my $pe;     # expected probability = p1^2+p2^2;
my $k;      # our significance measure, kappa = (po-pe)/(1-pe)

my $i = 0;
foreach my $rating (@kappatable) {
    $p1 = $p1 + $numraters - $rating;
    $p2 = $p2 + $rating;
    $p[$i] = 1/($numraters*($numraters-1))*(($numraters-$rating)**2+
                                               ($rating)**2-$numraters);
    $sump = $sump + $p[$i];
    $i = $i + 1;
}
$p1 = $p1/$totalnumratings;
$p2 = $p2/$totalnumratings;

$pe = $p1**2 + $p2**2;
$po = 1/(($tablelength)*($numraters*($numraters-1)))*$sump*$numraters*($numraters-1);

$k = ($po-$pe)/(1-$pe);
```

```
print "The kappa score is $k out of " . $totalnumratings/3 .
                              " ratings and $numfiles utterances\n";
```

C.2 Cohen's Kappa

```
#!/usr/local/bin/perl

use strict;

my $dirtoget="/s/summit/wavs/mandarin/scill/ftgame/anna/";
opendir(IMD, $dirtoget) || die("Cannot open directory");
my @thefiles= readdir(IMD);
closedir(IMD);

my $rater1 = 'andrea';
my $rater2 = 'hawksley';
my %files; #key = files, value = 2 if both have rated, 0 or 1 otherwise

foreach my $f (@thefiles) {
    unless ( ($f eq ".") || ($f eq "..") ) {
        if ($f =~ m/.tanno$/) {
            my @filename = split('\.', $f);

            if (@filename[1] eq $rater1 || @filename[1] eq $rater2) {
                #increments the number of raters who have rated a given file
                if (exists($files{@filename[0]})) {
                    $files{@filename[0]} = $files{@filename[0]} + 1;
                } else {
                    $files{@filename[0]} = 1;
                }
            }
        }
    }
}

my $numraters = 2;
my $numfiles = 0;
my $bothcorrect = 0;
my $r1correct_r2incorrect = 0;
my $r1incorrect_r2correct = 0;
```

```
my $bothincorrect = 0;

while ((my $filename, my $count) = each %files) {
  if ($count == $numraters) {
      $numfiles++;

      #Rater 1
      open(DAT, $filename . "." . $rater1 . ".tanno") || die("Could not open file!");
      my @lines1=<DAT>;
      close(DAT);

      #Rater 2
      open(DAT, $filename . "." . $rater2 . ".tanno") || die ("Could not open file!");
      my @lines2=<DAT>;
      close(DAT);

      my $numberlines = @lines1;
      for (my $i = 0; $i < $numberlines; $i++) {
          my @words1 = split(' ', @lines1[$i]);
          my @words2 = split(' ', @lines2[$i]);
           for (my $j = 0; $j < @words1; $j++) {
              if ((@words1[$j] =~ m/\*$/) && (@words2[$j] =~ m/\*$/)) {
                  #if the word has been marked incorrect
                  $bothincorrect++;
              } elsif (@words1[$j] =~ m/\*$/) {
                  $r1incorrect_r2correct++;
              } elsif (@words2[$j] =~ m/\*$/) {
                  $r1correct_r2incorrect++;
              } else {
                  $bothcorrect++;
              }
          }
      }
  }
}

#row and column totals
my $r1correct = $bothcorrect + $r1correct_r2incorrect;
my $r1incorrect = $bothincorrect + $r1incorrect_r2correct;
my $r2correct = $bothcorrect + $r1incorrect_r2correct;
my $r2incorrect = $bothincorrect + $r1correct_r2incorrect;
```

```
#overall total
my $total = $r1correct + $r1incorrect;

#expected results given no agreement
my $e_bothcorrect = $r1correct*$r2correct/$total;
my $e_1correct_2incorrect = $r1correct*$r2incorrect/$total;
my $e_1incorrect_2correct = $r1incorrect*$r2correct/$total;
my $e_bothincorrect = $r1incorrect*$r2incorrect/$total;

#actual and expected agreement
my $agree = $bothcorrect+$bothincorrect;
my $e_agree = $e_bothcorrect+$e_bothincorrect;

#the kappa score
my $k = ($agree-$e_agree)/($total-$e_agree);

print "The kappa score is $k out of $total trials and $numfiles utterances\n";
```

Bibliography

[1] Eric Atwell, Peter Howarth, and Clive Souter. The isle corpus: Italian and german spoken learners' english. *ICAME Journal*, 27:5–18, April 2003.

[2] Fabian Behrens and Jan-Torsten Milde. The eclipse annotator: an extensible system for multimodal corpus creation. In *Proceedings of the 5th International Conference on Language Resources and Evaluation(LREC 2006)*, Genoa, Italy, 2006.

[3] P. Bonaventura, P. Howarth, and W. Menzel. Phonetic annotation of a non-native speech corpus. In *Proceedings International Workshop on Integrating Speech Technology in the (Language) Learning and Assistive Interface, InStil 2000*, pages 10–17, Dundee, 2000.

[4] H. Bratt, L. Neumeyer, and H. Franco. Collection and detailed transcription of a speech database for development of language learning technologies. In *Proc. of ICSLP 98*, pages 1539–1542, Sydney, Australia, 1997.

[5] William Byrne, Eva Knodt, Sanjeev Khudanpur, and Jared Bernstein. Is automatic speech recognition ready for non-native speech? a data collection effort and initial experiments in modeling conversational hispanic english. In *ESCA Conference on Speech Technology in Language Learning*, Marhomen, Sweden, 1998.

[6] A. Chandel, M. Madathingal A. Parate, H. Pant, N. Rajput, S. Ikbal, O. Deshmukh, and A. Verma. Sensei: Spoken language assessment for call center agents. *ASRU 2007*, 2007.

[7] C. Chao, S. Seneff, and C. Wang. An interactive interpretation game for learning chinese. In *Proceedings of the Speech and Language Technology in Education (SLaTE) Workshop*, Farmington, Pennsylvania, October 2007.

[8] CollegeBoard. `http://www.collegeboard.com`, 2007.

[9] Concurrent versions system. `http://www.ximbiot.com`.

[10] Eclipse ide for java ee developers. `http://www.eclipse.org`, 2008.

[11] F. Ehsani and E. Knodt. Speech technology in computer-aided language learning: Strengths and limitations of a new call paradigm. *Language Learning and Technology*, 2(1):54–73, July 1998.

[12] The Apache Software Foundation. Apache tomcat. `http://tomcat.apache.org/`, 1999–2007.

[13] H. Franco, L. Neumeyer, M. Ramos, and H. Bratt. Automatic detection of phone-level mispronunciation for language learning. In *Proceedings of Eurospeech 99*, volume 2, pages 851–854, Budapest, Hungary, 1999.

[14] Sadaoki Furui. Recent advances in spontaneous speech recognition and understanding. In *Proceedings ISCA-IEEE Workshop on Spontaneous Speech Processing and Recognition*, Tokyo, Japan, 2003.

[15] Google. Google web toolkit. `http://code.google.com/webtoolkit/`, 2008.

[16] PostgreSQL Global Development Group. Postgresql. `http://www.postgresql.org/`, 1996–2008.

[17] A. Gruenstein. Shape game: A multimodal game featuring *incremental* understanding. Term project, Massachusetts Institute of Technology, May 2007.

[18] Alex Gruenstein. Building applications with the wami toolkit. `https://projects.csail.mit.edu/cgi-bin/wiki/view/SLS/WamiTutorial`, May 2008.

[19] A. Hagen, B. Pellom, S. Van Vuuren, and R. Cole. Advances in children's speech recognition within an interactive literacy tutor. *HLT-NAACL*, 2004.

[20] Dan Herron, Wolfgang Menzel, Eric Atwell, Roberto Bisiani, Fabio Daneluzzi, Rachel Morton, and Juergen Schmidt. Automatic localization and diagnosis of pronunciation errors for second language learners of english. In *Proceedings of EUROSPEECH99: 6th European Conference on Speech Communication and Technology*, volume 2, pages 855–858, Budapest, Hungary, 1999.

[21] Rebecca Hincks. Speech technologies for pronunciation feedback and evaluation. *Re-CALL*, 15(1):3–20, 2003.

[22] W.L. Johnson, C. Beal, A. Foweles-Winkler, S. Narayanan, D. Papachristou, S. Marsella, and H. Vilhjalmsson. Tactical language training system: An interim report. *ITS 2004*, 2004.

[23] D. Lancashire. Adsotrans. `http://www.adsotrans.com`, 2005.

[24] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33:159–174, 1977.

[25] Kikuo Maekawa, Hanae Koiso, Sadaoki Furui, and Hitoshi Isahara. Spontaneous speech corpus of japanese. In *Proceedings LREC2000*, pages 947–952, Athens, Greece, 2000.

[26] I. McGraw and S. Seneff. Immersive second language acquisition in narrow domains: A prototype island dialogue system. In *Proceedings of the Speech and Language Technology in Education (SLaTE) Workshop*, Farmington, Pennsylvania, October 2007.

[27] H. Meng, Y. Lo, L. Wang, and W. Lau. Deriving salient learners' mispronunciations from cross-language phonological comparisons. *ASRU 2007*, 2007.

[28] Wolfgang Menzel, Eric Atwell, Patricia Bonaventura, Dan Herron, Peter Howarth, Rachel Morton, and Clive Souter. The isle corpus of non-native spoken english. In M. Gavrilidou, G. Carayannis, S. Markantionatou, S. Piperidis, and G. Stainhaouer, editors, *Proceedings of LREC2000: Second International Conference on Language Resources and Evaluation*, volume 2, pages 957–964, Athens, Greece, 2000.

[29] Sun Microsystems. Java. `http://www.java.com/`, 2008.

[30] Jan-Torsten Milde and Ulrike Gut. A prosodic corpus of non-native speech. In *Speech Prosody*, Aix-en-Provence, France, April 2002.

[31] J. Mostow, G. Aist, P. Burkhead, A. Corbett, A. Cuneo, S. Eitleman, C. Huang, B. Junker, M. B. Sklar, and B. Tobin. Evaluation of an automated reading tutor that listens: Comparison to human tutoring and classroom instruction. *Journal of Educational Computing Research*, 29(1):61–117, 2003.

[32] J. Mostow, J. Beck, S. V. Winter, S. Wang, and B. Tobin. Predicting oral reading miscues. In *Seventh International Conference on Spoken Language Processing (ICSLP-02)*, Denver, CO, September 2002.

[33] J. Mostow, S. Roth, A. G. Hauptmann, and M. Kane. A prototype reading coach that listens. In *Proceedings of the Twelfth National Conference on Artificial Intelligence (AAAI-94)*, pages 785–792, Seattle, WA, August 1994. American Association for Artificial Intelligence.

[34] mygwt. `http://www.mygwt.net`, 2008.

[35] Ambra Neri, Catia Cucchiarini, and Wilhelmus Strik. Automatic speech recognition for second language learning: How and why it actually works. In *15th ICPhS*, pages 1157–1160, Barcelona, 2003.

[36] P. Pimsleur. A memory schedule. *Modern Language Journal*, 51(2):73–75, 1967.

[37] P. J. Price. Evaluation of spoken language systems: the atis domain. In *Proceedings of the workshop on Speech and Natural Language*, pages 91–95, 1990.

[38] Katharina Rohlfing, Daniel Loehr, Susan Duncan, Amanda Brown, Amy Franklin, Irene Kimbara, Jan-Torsten Milde, Fey Parrill, Travis Rose, Thomas Schmidt, Han Sloetjes, Alexandra Thies, and Sandra Wellinghoff. Comparison of multimodal annotation tools. *Gesprächsforschung - Online-Zeitschrift zur verbalen Interaktion*, 7:99–123, 2006.

[39] S. Seneff. Web-based dialogue and translation games for spoken language learning. In *Proceedings of the Speech and Language Technology in Education (SLaTE) Workshop*, Farmington, Pennsylvania, October 2007.

[40] S. Seneff, E. Hurley, R. Lau, C. Pao, P. Schmid, and V. Zue. Galaxy-ii: A reference architecture for conversational system development. In *ICSLP 1998*, pages 931–934, Sydney, Australia, December 1998.

[41] S. Seneff, C. Wang, M. Peabody, and V. Zue. Second language acquisition through human computer dialogue. In *Proceedings of ISCSLP*, Hong Kong, China, 2004.

[42] sourceforge.net. phppgadmin. `http://phppgadmin.sourceforge.net/`, 2008.

[43] Y. C. Tam, J. Mostow, J. Beck, and S. Banerjee. Training a confidence measure for a reading tutor that listens. In *Proceedings 8th European Conference on Speech Communication and Technology (Eurospeech 2003)*, pages 3161–3164, Geneva, Switzerland, 2003.

[44] L. von Ahn and L. Dabbish. Labeling images with a computer game. *CHI 2004*, 6(1), April 2004.

[45] L. von Ahn, S. Ginosar, M. Kedia, R. Liu, and M. Blum. Improving accessibility of the web with a computer game. In *CHI 2006*, Montreal, Quebec, Canada, April 2006.

[46] C. Wang, J. Glass, H. Meng, J. Polifroni, S. Seneff, and V. Zue. Yinhe: A mandarin chinese version of the galaxy system. In *Proceedings Eurospeech*, 1997.

[47] C. Wang and S. Seneff. A spoken translation game for second language learning. In *Proc. AIED*, Marina del Rey, California, July 2007.

[48] V. Zue, S. Seneff, J. Polifroni, H. Meng, and J. Glass. Multilingual human-computer interactions: From information access to language learning. In *Proceedings ICSLP*, 1996.