Methods for Pronunciation Assessment in Computer Aided Language Learning

by

Mitchell A. Peabody

M.S., Drexel University, Philadelphia, PA (2002)
B.S., Drexel University, Philadelphia, PA (2002)

Submitted to the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2011

Author . . . .
Department of Electrical Engineering and Computer Science
September 2011

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

Accepted by . . . .
Professor Leslie A. Kolodziejski
Chair, Department Committee on Graduate Students

Methods for Pronunciation Assessment in Computer Aided Language Learning

by

Mitchell A. Peabody

Submitted to the Department of Electrical Engineering and Computer Science
on September 2011, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

Learning a foreign language is a challenging endeavor that entails acquiring a wide range of new knowledge including words, grammar, gestures, sounds, etc. Mastering these skills all require extensive practice by the learner and opportunities may not always be available. Computer Aided Language Learning (CALL) systems provide non-threatening environments where foreign language skills can be practiced where ever and whenever a student desires. These systems often have several technologies to identify the different types of errors made by a student.

This thesis focuses on the problem of identifying mispronunciations made by a foreign language student using a CALL system. We make several assumptions about the nature of the learning activity: it takes place using a dialogue system, it is a task- or game-oriented activity, the student should not be interrupted by the pronunciation feedback system, and that the goal of the feedback system is to identify severe mispronunciations with high reliability.

Detecting mispronunciations requires a corpus of speech with human judgements of pronunciation quality. Typical approaches to collecting such a corpus use an expert phonetician to both phonetically transcribe and assign judgements of quality to each phone in a corpus. This is time consuming and expensive. It also places an extra burden on the transcriber. We describe a novel method for obtaining phone level judgements of pronunciation quality by utilizing non-expert, crowd-sourced, word level judgements of pronunciation.

Foreign language learners typically exhibit high variation and pronunciation shapes distinct from native speakers that make analysis for mispronunciation difficult. We detail a simple, but effective method for transforming the vowel space of non-native speakers to make mispronunciation detection more robust and accurate. We show that this transformation not only enhances performance on a simple classification task, but also results in distributions that can be better exploited for mispronunciation detection.

This transformation of the vowel is exploited to train a mispronunciation detector using a variety of features derived from acoustic model scores and vowel class distributions. We confirm that the transformation technique results in a more robust and accurate identification of mispronunciations than traditional acoustic models.

Thesis Supervisor: Stephanie Seneff
Title: Senior Research Scientist

Acknowledgments

This work would have not been possible without the support of many people:

Stephanie Seneff *for her guidance and patience during my long and circuitous program*.

Victor Zue and John Guttag,*for sitting on my committee.*

Natalija Jovanovic and Cordelia Zorana *for unwavering support and daddy hugs.*

Family Carol, Mat, Lisa, John, Zoran, Zorica, Natalie, Mike, Jason, Mandy, Karla, and Stephen,*for keeping me humbled.*

All of SLS but especially: Scott Cyphers, Jim Glass, Alex Gruenstein, Lee Hetherington, Ian McGraw, and Chao Wang*for technical assistance, guidance, and advice*. Marcia Davidson, *for years of witty banter about nothing in particular and keeping me in check, literally and figuratively.*

Friends Lisa Anthony, Joe Beatty, Mark Bellew, Nadya Belov, Michael Bernstein, Sylvain Bruni, Chris Cera, Chih-yu Chao, Ghinwa Choueiter, Christopher Dahn, Ajit Dash, Leeland Ekstrom, Suzanne Flynn, Michael Anthony Fowler, Abigail Francis, Tyrone Hill, Melva James, Amber Johnson, Fadi Kanaan, Alexandra Kern, Joe Kopena, Shawn Kuo, Rob Lass, Karen Lee, Vivian Lei, Hong Ma, Lisa Marshall, Gregory (grem) Marton, Amy McCreath, Ali Mohammad, Ali Motamedi, Song-hee Paik, Katrina Panovich, Anna Poukchanski, Bill Regli, Tom Robinson, Sarah Rodriguez, Micah Romer, Joseph Rumpler, Katie Ryan, Chris Rycroft, Josh Schanker, Yuiko Shibamoto, Ali Shokoufandeh, Ross Snyder, Susan Song, Evan Sultanik, Ryan Tam, William Tsu, Bob Yang, Stan Zanorotti, and Vera Zaychik,*for the stress relief and sanity checks.*

Mentors COL Joe Follansbee, MAJ Eric Schaertl, and CPT Vikas Nagardeolekar, US Army,*for showing me that Warrior-Scholars are not imaginary.*

Ops Brothers MAJ Jeffrey Rector, US Army, LT Carlis Brown and LT Jimmy Wang, US Navy, *MND-B G9 Ops: CA4Life.*

To all of the above and to anyone I missed who has touched my life and brought me to this point: **Thank you.**

This research was supported in part by the National Science Foundation (NSF) Graduate Research Fellowship Program in the United States, and by the Information Technology Research Institute (ITRI) in Taiwan.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Learning to speak a foreign language as an adult is difficult. It involves the learning of unfamiliar sounds, vocabulary, syntax, gestures, and dialogue structures such that the student can quickly understand and appropriately respond to sentences directed at them. The one method that works the most consistently in developing fluent communicative competence is extensive practice with native speakers by the person learning the foreign language. This repeated communication allows the student to make mistakes, receive feedback, and make corrections.

Computer Aided Language Learning (CALL) systems can be used to provide interactive, non-threatening, and fun opportunities for individual foreign language study outside the classroom. In particular, existing dialogue systems that are used to complete tasks through natural language can be adapted for CALL. This enables students to practice speaking a target foreign language in dynamic conversations without the need for a human partner. Systems that are adapted to this purpose have components that detect mispronunciations, grammatical errors, and provide feedback to students.

This thesis addresses the problem of detecting mispronunciation in the speech of foreign language students. This chapter explains our motivations, identifies our broad assumptions, states the contributions, and outlines the structure of the remaining thesis.

1.1 Motivations

Correct pronunciation is an important skill for foreign language students to successfully acquire. While virtually all non-native speakers of a foreign language learned as an adult have some sort of identifiable accent, possessing an accent does not necessarily imply poor pronunciation. Native speakers can be forgiving of slight deviations from native-like speech. The boundary between merely accented and mispronounced is fuzzy, and native speakers do not always agree on what is mispronounced. Most people *do* agree that mispronunciation exists—the question is where to draw the line.

Good teachers are skilled at choosing which mistakes to overlook and which to point out. If a teacher were to identify every single pronunciation mistake, a student could quickly become overwhelmed and discouraged in their learning endeavors. In addition to being selective in which mistakes to highlight, their perception prevents them from misidentifying examples of good pronunciation as mispronunciation, which would serve only to confuse the student.

Like a human teacher, a Computer Aided Pronunciation Training (CAPT) system must strike a balance between identifying egregious pronunciation errors and letting some—or most—mistakes slide. CAPT systems typically analyze speech at four distinct levels: speaker, sentence, word, and phonetic. A system that evaluates pronunciation at the speaker level seeks to evaluate the overall quality of an individual's pronunciation over a pooled set of sentences. Analogously, systems that evaluate at the sentence, word, and phoneme levels analyze individual sentences, words, and phonemes.

Modern CAPT systems are comparable to human speakers in assessing non-native speech at the speaker and sentence level. However, at the phonetic level, CAPT systems perform at levels that are far worse. A system that is too eager to point out a student's mistakes would be, at best discouraging to a student, and at worst even confusing and misleading because of misjudged errors. Systems typically utilize some sort of statistical model to render judgements on pronunciation quality. This is a difficult task because non-native speech is characterized by a higher degree of variation at the phonetic level than native speech. Some form of model adaptation or normalization is typically employed to account for this variation.

The basis for the CAPT statistics models is a labeled corpus of non-native speech. These corpora must be labeled both for phonetic accuracy—the transcription labels match the actual sounds in the utterance—and for pronunciation quality. Phoneticians often spend considerable time transcribing the exact sounds produced in an utterance. The additional task of deciding whether or not a sound was actually mispronounced places a substantial extra burden on the transcriber, and there is significant disagreement among different transcribers. Obtaining such a corpus of non-native speech is thus, costly and time-consuming.

1.2 Contributions

The research described in this thesis makes three main contributions with novel methods for: (1) crowd-sourcing of pronunciation labels; (2) acoustic feature representation; and (3) mispronunciation detection.

To cheaply obtain phonetic level judgements of pronunciation quality, a novel, crowd-sourced method for obtaining these labels is invented. This method allows anonymous non-experts using a web-based interface to collaboratively label whether words have been mispronounced or not. These judgements are used to identify incorrectly pronounced phones. This method is fast and cost effective when compared with a similar task.

A novel method for representing the acoustic features of vowels is proposed to account for non-native variation in vowel production. These features respresent sounds in relation to a speaker's measured anchor point. We argue that this method for representing sounds enables a more direct comparison of vowel quality. We demonstrate that a relative increase of between 1.8% to 8.4% in classification performance can be realized, if the acoustic space location of voiced regions of speech is measured.

These labels and anchor methods are incorporated into a method for mispronunciation detection based on probabilistic classification scores from parallel *Gaussian Mixture Models* (GMMs) and a novel set of acoustic features. We demonstrate that scores based on the anchored version of the vowels allow mispronunciations to be detected with higher precision and more robustness than traditional acoustic features.

1.3 Assumptions

This research makes a number of assumptions to constrain the scope of the problem. First, we assume a particular structure for the *Computer Aided Language Learning* (CALL) system and how the students interact with the system. The CALL system will be based around unscripted dialogues involving small domain activities such as making flight reservations, or playing simple web-based games.

Second, we assume that evaluation and feedback of students' speech does not occur during the activity. All evaluation and feedback is performed after the conclusion of the activity. This sequencing has the benefits of allowing the students to focus on using language while they complete the tasks, and providing access to all the speech recorded during the dialogue for an evaluation module to analyze. Post-session, the student can examine any mistakes that were made and learn from the computer's provided error feedback.

Third, we assume that the student speech has been correctly recognized and that a correct orthography has been provided by the speech recognition engine. This is a large assumption, but is common for CAPT systems. CAPT systems typically either constrain students to read sentences that have been previously scripted, or the dialogues allow only very limited sentences. We opt for the latter approach, as this provides students practice composing their own sentences. Our dialogue systems are, in fact, much less restrictive than most.

Finally, we assume that precision in identifying some mispronunciations is more important than identifying all mispronunciations. That is, we are willing to miss quite a few sounds that would be considered mispronounced in favor of being very confident that the sounds that are identified as mispronounced by the machine are truly mispronounced.

1.4 Terminology and Conventions

This thesis adopts the following definitions and conventions:

L1 A person's native or primary language. The research in this thesis specifically uses Cantonese as the L1 language.

L2 A person's second language. English is the L2 for the purposes of this thesis.

phoneme A segment of sound that results in a change to the meaning of the word when it is changed. The phonemes are realized as phones when speech is actually produced. These phones are subject to phonological rules which may alter the allowable sequence of phones.

phone A unit of speech that represents the actual sound produced by a speaker.

/*phone*/ Indicates the phonetic symbol under the *International Phonetic Alphabet* (IPA) standard.

[*phone*] Indicates the phonetic symbol under the ARPABET standard of ASCII phonetic notation.

1.5 Thesis Structure

This thesis is organized as follows:

Chapter 2 gives a broad and comprehensive introduction to Computer Aided Language Learning. It discusses general paradigms, systems of historical note, and computer aided pronunciation training (CAPT). CAPT is given special emphasis so that the contributions of this thesis can be placed into context.

Chapter 3 presents an algorithm to cheaply and quickly obtain a labeled corpus of phonetic pronunciation errors using Amazon Mechanical Turk.

Chapter 4 presents an algorithm to normalize vowel acoustic representations so that non-native speaker pronunciation can be directly compared with native speaker pronunciation. We argue that, by preprocessing the speech prior to classification, we can create pronunciation models that are more suited to mispronunciation detection.

Chapter 5 presents a classification algorithm that utilizes previously investigated and novel statistical features to detect mispronunciations with high precision. This chapter brings together the ideas presented in chapters 3 and 4. We use the algorithm presented in this chapter to demonstrate that the normalization algorithm presented in Chapter 4 is more robust than standard acoustic representations of vowels labeled using the technique presented in chapter 3.

Chapter 6 summarizes the research and contributions presented in this thesis and suggests future directions for research.

Chapter 2

Background

Computer Aided Language Learning (CALL) is a cross-disciplinary field that includes the subfields *Foreign Language Learning* (FLL), *Foreign Language Teaching* (FLT), Linguistics, and *Human Language Technologies* (HLT). FLL research typically focuses on topics such as learning strategies employed by students and effectiveness of environments designed to support learning. FLT focuses on discovering and employing effective pedagogies to facilitate learning as well as meaningful performance measurements. Linguistics, specifically the subfield of *Second Language Learning* (SLA), focuses on the process of learning a second language by investigating common patterns of mistakes and progression in competence. Finally, Human Language Technologies encompasses the full-range of technologies, from audio recordings to dialogue systems, used to facilitate learning.

A thorough discussion of all these topics would take many volumes, so this chapter restricts itself to a small subset. Specifically, this chapter briefly discusses pronunciation as it relates to foreign language teaching and learning. It then provides an overview of Computer Aided Language Learning, with a specific focus on dialogue systems for CALL. Finally, it provides an in depth overview of *Computer Aided Pronunciation Training* (CAPT). A more extensive survey of these topics and some of the fields cited above can be found in Appendix A.

2.1 Pronunciation

Intelligible pronunciation is only one of the needed skills for speaking a foreign language, and it is often not emphasized in the classroom. There has been some renewed interest in teaching pronunciation explicitly [87] because of studies that show that pronunciation quality below a certain level of proficiency places additional stress on the listener and seriously degrades the ability of native speakers to understand what is being said [98, 251].

Most adult learners of a foreign language, and even those as young as 6 years old [244], retain some artifacts in their pronunciation that identify them as non-native speakers. Despite the presence of an accent, native speakers will not necessarily identify speech as mispronounced if the quality is above some subjective level.

Improvements in the pronunciation of learners whose pronunciation has plateaued at a less than desirable level are possible through pronunciation training [52]. Native-like intonation can also be learned [153]; however, this is extremely difficult for even advanced language learners. In addition to requiring lots of output [220] to improve pronunciation, students cannot attend to all aspects of pronunciation at the same time [53], e.g. attending to phonetic accuracy takes processing time away from attending to intonation.

A foreign language learner will make a number of pronunciation errors at the phonetic (segmental) and prosodic levels when producing speech in a target language. Errors at the segmental level can be generally classified as substitution, insertion, deletion, and duration errors. Errors at the prosodic level are more difficult to categorize. There is some debate over whether phonetic or prosodic aspects of pronunciation have more impact on perceived pronunciation quality [165]. While the sources of these errors are a topic of research in the linguistic community, there seems to be a consensus that the phonetic inventory of the native language interferes to a certain extent with the production of sounds in the foreign language [72].

A well-known example of a substitution error caused by native language interference is the difficulty native Japanese speakers have with the /l/–/r/ contrast in English [27]. Another example of native language interference is the devoicing of word-final obstruents in Cantonese speakers of English [185]. More detailed discussion of second language pronunciation can be found in [134].

Another source of error is the inability of non-native speakers to become attuned to critical acoustic features in the target language. For tonal languages, such as Chinese, students arriving from a non-tonal language often have difficulty even perceiving changes in the pitch indicating the presence of a lexical tone. This has an impact on their ability to produce these tones correctly [234]. For example, Japanese learners of Korean have difficulty discriminating between lenis (weakly aspirated) and aspirated alveolar stops [123]. Careful analysis of perceptual differences between Japanese and native Korean speakers showed that Japanese learners of Korean placed more emphasis on *Voice Onset Time* (VOT) than on f_0 (the fundamental frequency of a voiced segment) when discriminating between the lenis and aspirated stop; however, native Korean speakers were able to use both acoustic features to successfully discriminate between the sounds. This suggests that students sometimes have incomplete or confused models of the speech sounds in the language.

2.2 Computer Aided Language Learning

Researchers have investigated the use of computers for language learning since the 1960s [227]. The field of CALL has seen an explosion of research over the past decade, and it would be impractical to include every piece of research in this thesis. This section will discuss representative examples of CALL. A further review of the history, key developments, and major paradigms in Spoken CALL can be found in [67].

CALL research, from a purely technical standpoint, can be divided into roughly two areas: research focused on whole systems and research focused on specific technologies to be integrated into whole systems. This section deals with whole systems, and specifically highlights modern, dialogue-based systems. The next section will go into depth on the subsystem that is the focus of this thesis, *Computer Aided Pronunciation Training* (CAPT).

CALL systems are numerous with diverse system configurations. On the simple end of the spectrum, the systems can take the form of web pages with fill-in forms [200, 135], online chat rooms, static multimedia programs, modifications to popular games [189], or even simply a set of digital music files for playback purposes. On the complex end, systems can have automatic speech recognition, voice synthesis, and highly interactive 3D environments that teach cultural norms as well as language.

Modern systems tend to be much richer language learning environments that incorporate high quality audio, graphics, and automated feedback. The content of the lessons is usually not static, and is generated randomly or adaptively, in response to student actions. Many systems use some form of *Automatic Speech Recognition* (ASR), speech synthesis, natural language understanding, or natural language generation.

2.2.1 Dialogue-based Systems

Dialogue systems can be used to create immersive environments in which students hold dynamic, fairly natural conversations [96, 132, 17, 231, 63]. Instead of being given a specific sentence or a limited script to follow, which can lead to memorization and plateauing [79] in learning, students can hold conversations that are varied between practice sessions. Since speech recognition technology is imperfect, there is constant tension in dialogue systems between allowing freedom in conversation and sufficiently constraining the domain to maintain acceptable performance. Dialogue systems adopt different strategies to strike an appropriate balance.

Subarashii [60, 19] was a dialogue system that advanced the conversation using a predefined set of responses in a sort of choose-your-own-adventure style of dialogue. Later research crafted the dialogues to elicit a limited set of responses without explicitly stating them.

Subarashii was specifically designed for language education. In contrast, a prototype system by Lau [133] was created by adapting an existing dialogue system capable of conversing in both English and Chinese. It allowed for simple, unstructured conversations about families, but the architecture allowed for adaptation to new domains. Students would conduct conversations in Chinese, or ask for translation help in English.

The *Tactical Language Tutoring System* (TLTS) [115, 112, 114, 113] is an example of a rich, multimedia system for language learning. The student is immersed in a 3D world using the *Unreal Tournament 2003* [62] game engine where he is instructed to accomplish missions—the system was developed for military use—by interacting with characters in the environment using Arabic speech and non-verbal communication. Speech recognition is performed using the *Hidden Markov Model Toolkit* (HTK) [248] augmented with noisy-channel models to capture mispronunciations [161].

Raux and Eskenazi [195] adapted an existing spoken dialogue system [196] to handle non-native speech [194] using a generic task-based dialogue manager [23]. Another key feature of the system was the use of clarification statements to provide implicit feedback through emphasis on certain parts of a student's utterance [193].

Chao et al. [32] created a web-based translation game for learning Chinese with repetitive exercises for acquiring vocabulary and grammar. This system was later adapted to create a simple dialogue game in [208, 207]. McGraw et al. [149, 150, 151, 246] created multiplayer web-based games focused on vocabulary acquisition. Students used natural speech in a highly constrained domain to manipulate cards representing new vocabulary items in competitive games.

The *Development and Integration of Speech technology into COurseware for language learning* (DISCO) system [47] is a Dutch system for providing feedback on pronunciation, morphology, and syntax. The system exploits morphology and syntax errors common in learners of Dutch as a foreign language. The DISCO system conducts dialogues by eliciting very constrained responses to questions; it uses a two step process for recognizing speech in a constrained domain. In the first step, it determines the content of a learner response, by augmenting an *Finite State Transducer* (FST) language model. In the second step, it then analyzes that response for correctness with stricter constraints [228].

The SayBot Player is a system for teaching English to native Chinese speakers [35]. It maintains a teacher designed dialogue flow using a Finite State Machine architecture. Pronunciation is scored using *Hidden Markov Model* (HMM) [11, 12] log-likelihood scores and duration measurements. Errors during the dialogue are classified into four categories: Correct (all words are correct and the pronunciation score is good), Pre-defined Error (pronunciation score is good, but sentence is recognized among a set of predefined errors), Mispronunciation (recognized words are produced poorly), and General (the system could not understand the student speech at all).

2.3 Computer Aided Pronunciation Training

CAPT systems are specifically designed to evaluate and improve pronunciation in foreign languages. A CAPT system can be considered to have an evaluation component and a feedback component. Pronunciation evaluation can take place at two general levels: holistic and pinpoint error detection. A holistic evaluation examines a large sample of speech and provides an overall assessment of a speaker's proficiency. Pinpoint error detection attempts to identify specific pronunciation mistakes at the word or subword level.

2.3.1 Holistic Pronunciation Evaluation

Several methods have been proposed for holistic pronunciation evaluation. Most involve the correlation of subjective human assessments with machine-based measures. Acoustic and probabilistic measurements include total duration of read speech with no pauses, total duration of speech with pauses, mean segment duration, rate of speech, and log likelihood measurements. Human ratings include global pronunciation quality, segmental quality, fluency, and speech rate.

Early work on pronunciation evaluation was performed by Wohlert [243, 242]. In his research, Wohlert selected 160 of the most commonly used, strong German verbs, and divided them up into 16 categories with 10 words each. The system used a template based on the average of five pronunciations for each German verb.

A series of five exercises, such as fill-in-the-blank and translation, were created for each group of verbs. During the tutoring session, the student is presented with a score from 500 to 1000, 1000 being a perfect match. The score is based on how closely the speech produced by the student matches the template stored in the database. One shortcoming of this research was that the correlation of the scores to human rater evaluations was not performed. Still, after a semester of work, with one group of students learning German using the new system compared to a control group, he found a significant increase in the number of verbs the students in the former group mastered (87% of the presented vocabulary) versus the number mastered by students in the latter (67%).

Early research by Bernstein et al. [16, 14] investigated methods for accurately predicting scores similar to those given in *Oral Proficiency Interviews* (OPI). The PhonePass system, which grew out of this research, was developed to assess non-native English proficiency [222]. The researchers gathered telephone quality data from a large number of responses to five different types of questions that reflected conversational speech. Correct and incorrect responses were combined with HMM scores and used as inputs into a function that produced a score correlated with expert human judgements of proficiency.

Later research validated the scores against the Common European Framework of Reference [177] for assessing language proficiency [15]. A version of the algorithm was developed to assess non-native Spanish and validated against the *American Council on the Teaching of Foreign Languages* (ACTFL), *Interagency Language Roundtable* (ILR), and *Spanish Proficiency Test* (SPT) OPIs [18], and later adapted to Modern Standard Arabic [20].

Cucchiarini et al. developed similar methods for assessing the proficiency of non-native speakers of Dutch [42, 41]. In contrast to other assessment methods, which examined pronunciation errors from speakers with a common native language, they investigated the assessment of speakers with many different language backgrounds. Subjects were asked to read two sets of five phonetically rich sentences. Human judgements on overall pronunciation, segment quality, fluency, and speech rate were gathered from three expert phoneticians.

They found that machine generated measures such as duration and rate of speech scores were highly correlated with human judgements of pronunciation quality. They discovered that using rate of speech or duration measurements also permitted students to ``cheat" by speaking very rapidly. Subsequent research found that the use of log-likelihood scores could mitigate this problem [48, 44, 69].

Later work expanded the research to include spontaneous speech as well as read speech [46, 40, 216, 45, 43]. In addition to adding spontaneous speech they added two groups of human raters, both consisting of speech therapists. They also modified the set of machine scores to: rate of speech, phonation-time ratio, articulation rate, pauses per unit of time, mean length of pauses, and mean length of runs. Test data measurements were divided into 7 classifications: three proficiency levels of read speech plus a combined measurement of all three, and two proficiency levels of spontaneous speech plus a combined measurement of both.

Correlations that were found between human ratings and machine measurements in read speech were almost halved when spontaneous speech was used. For example, the correlation of machine measured rate of speech with human judgement of overall pronunciation decreased from 0.75 to 0.46 when spontaneous speech was used. A drop in the correlations between machine scores and the human ratings for the high proficiency spontaneous speakers was attributed to the more difficult nature of the high proficiency material. The conclusion was that the optimal predictors of proficiency for read speech and spontaneous speech were different. In the case of read speech, the rate at which sounds were articulated and the frequency of pauses were strongly related. In spontaneous speech, they found that the mean length of the runs between pauses was a better predictor of pronunciation quality. Additional analysis comparing the rate of errors between read and spontaneous speech revealed the surprising result that the phonetic errors of substitution and deletion were more prevalent in read speech than in spontaneous speech [56]. The authors hypothesize that this may be due to interference of the orthographic representation of the language and the student's understanding of the writing system.

Neumeyer et al. [173] investigated the evaluation of French as spoken by Americans. In these studies, the researchers collected read and spontaneous speech samples from 100 native French speakers and 100 Americans. They investigated four separate methods for scoring pronunciation at two levels: the sentence level and the speaker level. Correlations were computed between various machine scores and human ratings, which included HMM log-likelihood, segment classification, segment duration, and timing scores.

Initially, they found that the HMM scores—average log-likelihood and posterior probability—did not correlate well with human expert pronunciation ratings on a Likert scale from 1 to 5 (1 was unintelligible, 5 was native-like). All of the scores, except for those based on timing, resulted in what they felt were unacceptable correlations at both the sentential level and the speaker level. They later improved the speaker level correlation of the HMM based scores by using the average of the log-posterior probability scores instead of the log-likelihood scores [74].

In other experiments, the researchers concentrated on sentential and speaker level pronunciation evaluation [202, 77, 75] using scores for specific phones. Additional methodology was introduced for detecting mispronunciation in which they compared a log-posterior probability from pure native models method with a dual model approach in which one phone model represented the correct pronunciation and the other represented the incorrect pronunciation.

Rhee and Park [181] describe a system that makes use of parallel native and non-native models to assign grades to student utterances at the sentential level. SpeechRater™is a program for rating the TOEFL iBT Practice Online product that also uses native and non-native models to generate features that are later used to score a speaker's overall perceived fluency [249, 250]. The authors found that the machine was able to assess a student's style or manner of delivery, even if recognition accuracy was not good. A system for evaluating spontaneous non-native Greek speech was developed using parallel native and non-native models [164]. The authors demonstrated that a system using parallel models outperformed a system using a single set of native models for evaluation.

The research cited above utilized many of the same features, such as duration, rate of speech, confidence scores, log-likelihood, and log-posteriors from HMM lattices to create regression functions to score speech. Research by Minematsu et al. takes a fundamentally different approach by modeling the pronunciation of sounds as distributions in frequency space relative to the other sound distributions in the language [156]. This was conducted in the spirit of work by Jakobson [107] who argued that the study of the sounds of a language must consider the structure of the sound system as a whole.

The structure defined by Minematsu et al. was then used to define a distortion metric that measured the difference between the phonetic structures of two populations of speakers, native American English speakers and Japanese learners of English [155]. This distortion metric was found to correlate with assessments of pronunciation proficiency [7, 157, 218], and this correlation held even when the non-native speech model was compared against multiple models of native speech (representing more than one teacher) [219].

The authors in [34] combine scores derived from HMM log-probabilities and *Gaussian Mixture Model* (GMM) [84] scores by using a non-linear regression to mimic the scoring function of a human rater on non-native Mandarin speech. In this research, the log-probabilities are not used directly in the scoring function; rather, the log-probabilities are used to rank order the correct syllable against 410 other syllables in the Chinese language. The rank of the syllable is then used to compute a syllable score. The GMM scores are used in a similar way. A non-linear regression is used to optimize several parameters to combine these scores into one that mimics a human rater.

An approach described in [83] used the log-posterior probabilities from forced alignment with HMM to classify the quality of syllables using *Support Vector Machines* (SVMs) [38]. The classification results over a large number of syllables produce a final score of speaker pronunciation ability. This score is correlated with the 普通话水平考试 (Putonghua Shuiping Kaoshi, PSK) corpus scores, which is a corpus of Chinese speakers from different dialect backgrounds.

Another example of a scoring method that does not make explicit use of HMM derived features is found in [124]. The authors found positive correlation between measures of pruned syllables per second, the ratio of the difference between total number of syllables and unnecessary syllables to total duration, and the ratio of unaccented syllables to accented syllables. A unique aspect to this study is that the authors were careful to gather human ratings from teachers who had been specifically trained in the Common European Framework of Reference [177] for assessing pronunciation. This included many specific evaluation items of loudness, sound pitch, quality of vowels, quality of consonants, epenthesis, elision, word stress, sentence stress, rhythm, intonation, speech rate, fluency, place of pause, and frequency of pause.

2.3.2 Pinpoint Error Detection

Pinpoint error detection is the identification of specific instances of pronunciation mistakes. Most modern pronunciation evaluation systems use log-posterior probability or log-likelihood scores produced by HMMs to evaluate foreign speech. These are then used to select word or subword units (syllables or phones) as mispronounced for later feedback to the student.

Word and phone level human assessments were found to be correlated with parallel HMMs trained on native and non-native speech [86, 210]. Posterior probabilities, followed by log-likelihood scores, were found to be most highly-correlated with human assessments of pronunciation quality [122]. Interestingly, the authors found that measurements of duration were almost uncorrelated with assessments of individual phone quality. This is in contrast to work described in the previous section that found temporal based measurements to be highly correlated with overall assessment of speaker pronunciation. This may be due to humans paying attention to different aspects of pronunciation when asked to assess proficiency at the speaker or sentence level versus proficiency at word or phonetic level.

The FLUENCY project is one of the earliest examples of a system that was able to detect pronunciation problems at the phonetic and prosodic levels [66]. *Carnegie Mellon University* (CMU) SPHINX-II [104] speech recognition system was used to measure prosodic information and detect phone errors from speech spoken by non-native speakers of English with French, German, Hebrew, Hindi, Italian, Mandarin, Portuguese, Russian, and Spanish as the native languages [65, 63].

This research was used to create a prototype language tutor [64] that was based on 5 principles articulated by [120]: production of large quantities of speech, reception of relevant corrective feedback, exposure to many examples of native speech, early emphasis on prosodic factors, and feeling of ease in learning environment. A key part of the system was the use of elicitation techniques in order to predict sentences that could be used for forced alignment recognition, in contrast to other systems, such as [224], which use completely scripted dialogues in their lessons.

Similarly, [111] examined the ability of HMMs to detect mispronunciations. In this study, tolerance levels were established for the scores of native speakers. When a non-native speaker produced a phone which generated a score that was at least one standard deviation away from the mean, feedback was given in the form of an illustrative diagram of proper articulation spots. HMMs were used by [118] to evaluate foreign speakers of Japanese on phonetic quality, but only for the quality of Japanese *tokushuhaku* (phones contrasted only by duration). Another system was implemented [119] to detect phone insertion, deletion and substitution using parallel phone models.

Witt et al. [239, 240] used HMM models to define a *Goodness of Pronunciation* (GOP) score, which was based on the log-likelihood of each phone segment in an HMM lattice, normalized by the number of frames in the segment. Phone dependent thresholds were defined to indicate the presence of mispronunciation. These were empirically derived based on hand analysis. Using results from forced alignment recognition, the most common substitution errors were discovered and the phone models augmented to allow for additional paths through the lattice during decoding. An evaluation of GOP [117] compared thresholds optimized for either artificially produced errors derived from linguistic knowledge or real errors, and found no significant difference in the performance of the algorithm. This was important to the authors as it validated the use of artificial errors. Speaker dependent phone thresholds also yielded slightly better performance.

Similar to Wohlert's work, [50] used template-based discrete word recognition to evaluate learners of Spanish and Mandarin Chinese. A segmental analysis was performed to tabulate pronunciation errors for specific phones. These were then used to create and a system for weighting the importance of various errors. Eventually, a game-like interface was added [49] to provide feedback on pronunciation exercises. An interesting aspect of this research is the comparison of HMM based recognition with the template method. The authors found that, while the HMM recognizer was better at overall recognition accuracy, the template recognizer was better at distinguishing between minimal pairs.

An approach in Kim et al. [121] combined the results of a forced-alignment of accented English spoken by Korean English language learners, with the hand phonetic transcriptions of an expert phonetician. A detailed phonological analysis was performed to obtain a set of augmentation rules that modeled common pronunciation phenomena exhibited by the students. These rules tagged phonetic mispronunciations in an utterances and triggered feedback messages for the students. This approach was later extended by Harrison et al [93].

A CAPT that is too harsh on a student is likely to leave them feeling frustrated and dissatisfied with the system. Achieving native-like pronunciation is probably an unrealistic goal, especially with older students, so some research tries to identify high priority phones that should be assessed and corrected. In [171], a data driven approach was introduced to establish priorities for certain segmental errors. This helped establish which phones were (1) mispronounced often or (2) resulted in misunderstanding or unintelligibility. In [223], these results were used to identify three of the phones commonly found to be mispronounced by non-native speakers. Classifiers were trained for these phones to decide if they were acceptable or not, using features selected through an analysis of the difference between native and non-native productions.

A novel approach by the authors in [179, 180] combined the frame log-posterior probability, phone log-posterior probability, and formant classification score derived from image feature extraction using the Gabor function to grade vowel quality in Mandarin spoken by Hong Kong residents. Three techniques were experimented with to combine the scores: linear regression to approximate a human rating, joint probability estimation, and a neural network. The neural network using all three features achieved a 9.7% higher correlation with human graders than the baseline using only frame-based log-posterior probabilities.

Finally, SVMs with linear kernels were used to detect phone-level mispronunciations in Mandarin Chinese using the log-likelihood ratios produced by an HMM lattice [235]. A phone-dependent ratio was set to balance precision and recall of mispronunciations. In contrast to most other HMM based methods which use GMMs to model phone pronunciations, this research used a model called a *Pronunciation Space Model* (PSM). The authors were motivated by the observation that many phone substitutions are not complete substitutions of one phone for another, but are substitutions of a partially changed phone for a sound that may not appear in the target language.

2.4 Summary

This chapter introduced several key ideas in Foreign Language Learning, briefly discussed related fields of research, and specifically highlighted foreign language pronunciation. It presented a discussion of general CALL highlighting existing systems, discussed some of the research questions, and finally focused on a detailed discussion of CAPT. The following chapters detail the research contributions of this dissertation for pronunciation assessment of foreign languages.

Chapter 3

Crowd-sourced phonetic labeling

This chapter outlines a novel algorithm for labeling a corpus of non-native speech for phonetic pronunciation quality when substitutions have occurred. Our method combines the results of crowd-sourced word level judgment of pronunciation quality with the results of aligning machine generated phonetic transcriptions and hand phonetic transcriptions. We justify this algorithm with measures of word level agreement among anonymous annotators, and provide an analysis of the nature of phonetic insertions, deletions, and substitutions.

3.1 Motivation

A labeled corpus of non-native speech is required for developing algorithms capable of detecting mispronunciations. Obtaining such a corpus is time-consuming and costly. This is due to the fact that two phonetic level labelings are required for every utterance in the corpus: the transcription of the phones produced and the judgment of quality for each phone produced.

When transcribing utterances, phoneticians try to precisely transcribe the sound that was actually produced. This task can be challenging in its own right. In addition to L2 phones, non-native speakers will also produce L1 sounds, and intermediate sounds that are between L1 and L2 sounds. Because the inventory of sounds is larger and non-standard for a given L2, phoneticians must decide on a set of standards for when to use one sound label over another.

In addition to this phonetic transcription task, a labeling of pronunciation quality must also be obtained. In the corpus used in this research (described later), there are an average of 38 phones per sentence and a total of 1,385,234 phones throughout the corpus. Assuming that an annotator was able to label 1,000 utterances, or 38,000 phones, a day, the entire process would take over a month—37 days. Even assuming an 8 hour work day at minimum wage ($8.00 (USD) in Massachusetts), this would be $2,368.00. In reality, the hourly rate would probably be double this amount, as this is a skilled task.

Additionally, humans do not always agree on what constitutes a mispronunciation. Some humans are more forgiving than others of deviations from canonical pronunciations in non-native speech. A useful labeling of pronunciation quality must include multiple annotators for every phone. A common number of annotators sought is 3. Thus, a full annotation of pronunciation quality on this corpus would probably cost as much as $15,000.00 (USD) and would take over a month's worth of time.

3.2 Related Work

Labeling a non-native corpus for pronunciation quality is critical for research on pronunciation evaluation. A variety of techniques have been reported in the literature. These techniques include using Likert scales to rate speech on a scale of accentedness or intelligibility, using a binary classification based on mismatch between gold-standard transcriptions and automatic transcriptions, and labeling phonetic transcriptions for insertions, deletions, and substitutions according to a canonical phonetic labeling. We focus on those techniques that resulted in corpora of speech data labeled at the word and sub-word levels.

Researchers in [86] labeled a corpus of 10 words spoken by 53 native and 49 non-native speakers of Dutch by asking a Dutch language teacher to decide whether each word token was produced by a native speaker or non-native. These judgments were used to test word-level mispronunciation detection by HMMs.

Phonetically labeled non-native French speech for experiments conducted by Kim et al. [122] was collected by asking a panel of five teachers of French to score individual phone segments on a 5-point Likert scale (1 being unintelligible, 5 being native-like). They listened to full sentences from each speaker with instructions to pay attention to only one phone segment at a time. A total of 4,656 scores were obtained using this method.

Errors were labeled in non-native English speech based on agreement between an automatic transcription obtained through forced-alignment and the assessment of expert tutors in English [66]. The tutors listened to sentences spoken by non-native English speakers and were instructed to annotate where mispronunciations occurred in the utterances, what the mistake was, and how they would correct it.

In Witt and Young [239], a database of 2,040 utterances was rated on a 4-point Likert scale by expert phoneticians. These ratings were assigned at the sentence and word levels. A phonetic analysis was used to determine the locations of insertions, deletions, and substitutions according to a canonical dictionary of native British English pronunciation. A similar procedure was used in [171] to mark the presence of pronunciation errors in a corpus of Dutch speech.

These techniques all share the same characteristic of utilizing expert annotators and requiring large amounts of time (and money) to label relatively small amounts of speech. Crowd-sourcing [102] has become a popular technique in recent years for rapidly obtaining large amounts of data at substantially lowered costs by using groups of anonymous workers to perform tasks over the Internet. *Amazon Mechanical Turk* (AMT) is a service provided by *Amazon.com, Inc.* that allows requesters to post *Human Intelligence Tasks* (HITs) for anonymous workers (Turkers) to complete for monetary compensation. This service has become popular for research in a variety of natural language tasks.

In [213], researchers evaluated the quality of AMT supplied annotations for five natural language tasks: affect recognition, word similarity, recognizing textual entailment, event temporal ordering, and word sense disambiguation. They found that AMT supplied annotations had a high correlation with gold-standard expert ratings, an encouraging result.

In [89], a corpus of 30,938 utterances was transcribed at near expert level quality using AMT. Researchers in [29] used AMT to evaluate the quality of machine translation and found that the non-expert Turkers achieved equivalent correlation with expert judges on the same task.

AMT was used in [103] to label political blog posts according to sentiment regarding

United States presidential candidates, John McCain and Barack Obama. They found that the correlation between expert labelers and aggregated Turkers was comparable. In [22], AMT was used to build evaluation test sets for machine translation tasks—the quality of these test sets was comparable to the quality of professionally developed test sets at a fraction of the cost.

Finally, AMT was used to collect human assessments of speech accentedness [129]. In this study, the authors presented Turkers with several utterances read by non-native speakers of English from three language groups: Arabic, Mandarin, and Russian. After listening to each utterance, Turkers were asked to rate the entire utterance on a 5-point Likert scale (1 being native-like accent, 5 being heavily-accented). As of the time of this writing, detailed analysis is being conducted on the results, but the authors reported that preliminary tests showed consistent correlation between phonological patterns and ratings of accentedness.

3.3 Approach

We propose a labeling method that takes advantage of crowd-sourced labor from AMT. The use of AMT to label phones for pronunciation quality is attractive because it potentially allows relatively simple tasks to be farmed out to hundreds of workers to produce near-expert quality labels for little money.

Unfortunately, asking non-expert labelers to provide a judgment on *phone*-level quality of pronunciation is unrealistic. The general population doesn't possess the expertise of a phonetician in identifying sub-syllable level units of sound, nor do they possess the knowledge to provide an assessment of pronunciation quality. Asking a layperson to mark whether the /æ/ phone in ``bat'' is mispronounced or not is impractical—this is a difficult task even for a phonetician. On the other hand, most native speakers of a language can tell if a *word* is mispronounced. We were encourage to explore this approach because AMT has been shown by other related work to produce acceptable results for natural language tasks.

Our technique labels phones for pronunciation quality by asking Mechanical Turk workers (Turkers) to provide judgments of the pronunciation quality of each *word* in our corpus. These word-level judgments of quality are combined with the lowest edit distance alignment of a machine-generated, forced-path phonetic transcription of our data and a hand generated phonetic transcription to produce a corpus of phone level judgments of pronunciation quality. We justify our technique based on an analysis of the types of alignment errors present in words that have been annotated as mispronounced.

3.3.1 Data

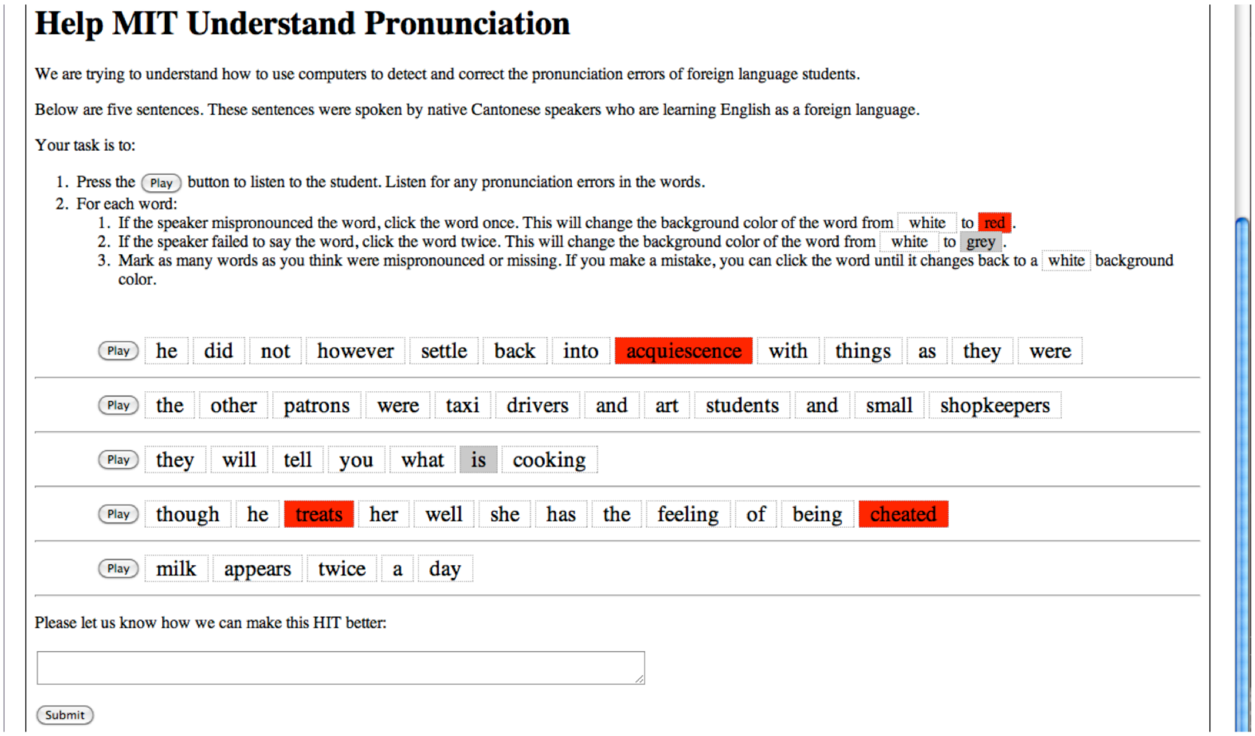
Our experiments made use of the *Chinese University Chinese Learners of English* (CU-CHLOE) corpus [152]. The CU-CHLOE corpus is part of the *Asian English Speech cOrpus Project* (AESOP) initiative, and is the result of an ongoing effort to create a corpus of English spoken by native speakers of Cantonese. It consists of 36,696 English utterances spoken by 100 (50 male, 50 female) non-native speakers of English. Each speaker read a series of 367 prompts that consisted of minimal word pairs (4 were discarded because of file corruption), TIMIT [81] prompts, and passages from the Aesop Fable ``The North Wind and the Sun." Recordings were sampled at 16kHz using close-talking microphones. Of these utterances, 5,597 (across all speakers) were phonetically hand transcribed. The entire corpus contains 306,752 words; the portion that was hand transcribed contains 36,874 words.

3.3.2 Annotation Task

We used the AMT service to collect word level judgments of pronunciation quality for each utterance in the CU-CHLOE corpus. The unit of work in an AMT task is called a *Human Intelligence Task* (HIT). AMT allows the requester to design a web-based HIT using HTML and simple template tags. Once the interface is finalized, the data for the HIT are formatted into a *Comma Separated Value* (CSV) file and uploaded to the AMT servers. In this way, the same interface can be used for any number of HITs.

Figure 3-1 is a screen capture of the interface we used to collect these annotations. The top part of the interface gave the Turker instructions about how to complete the task. Each HIT consisted of five utterances from the CU-CHLOE corpus. For each utterance, a *Play* button was presented alongside the prompt text of the utterance.

Each word in the utterance was made clickable using the mouse. One click changed the



Help MIT Understand Pronunciation

We are trying to understand how to use computers to detect and correct the pronunciation errors of foreign language students.

Below are five sentences. These sentences were spoken by native Cantonese speakers who are learning English as a foreign language.

Your task is to:

1. Press the Play button to listen to the student. Listen for any pronunciation errors in the words.
2. For each word:
   1. If the speaker mispronounced the word, click the word once. This will change the background color of the word from white to red.
   2. If the speaker failed to say the word, click the word twice. This will change the background color of the word from white to grey.
   3. Mark as many words as you think were mispronounced or missing. If you make a mistake, you can click the word until it changes back to a white background color.

Play he did not however settle back into acquiescence with things as they were

Play the other patrons were taxi drivers and art students and small shopkeepers

Play they will tell you what is cooking

Play though he treats her well she has the feeling of being cheated

Play milk appears twice a day

Please let us know how we can make this HIT better:

Figure 3-1: Interface presented to Turkers during labeling task.

background of the word to a *red* color, and signified that the Turker felt the word had been mispronounced. A second click changed the color to *gray*, and signified that the Turker felt the word had been omitted by the speaker. Finally, a third click changed the color back to *transparent*, and offered the Turker a chance to remove a judgment of mispronounced or missing.

Requesters must take into consideration the complexity, amount of work, and the wage they are willing to pay for each HIT. A complex HIT that requires a long time and offers a small reward will probably not have many Turkers willing to complete it. On the other hand, a simple HIT that requires little time and offers a substantial reward will be expensive to the requester when there are a lot of data to process. The key to a successful hit is to balance these constraints. We found through small trial runs that a reward of $0.05 (USD) per HIT was sufficient to entice Turkers to work on our HITs.

Our interface sought to simplify the annotation task to the greatest extent possible by obtaining judgements at the word-level. As noted in the previous section, labeling for pronunciation quality usually involves obtaining expert annotations at the phonetic level. These are either judgements placed on a Likert scale, or annotations of insertions, deletions, and substitutions. It would be difficult to guarantee that Turkers possess the level of skill required to complete this sort of task.

On the other hand, it would not be difficult to ask non-expert Turkers, most likely fluent in English, to provide ``gut-level" reactions at the word-level. AMT provides requesters the ability to specify a number of parameters for a HIT. Among these parameters are the approval rate of the Turker and the Turker's geographic location. We restricted the Turkers who were qualified to work on these HITs to those with a 95% HIT approval rate and who were located in the United States.

Finally, AMT allows requesters to specify that multiple Turkers complete each HIT. Three is a common number of annotators to use in this sort of task, so we specified that each HIT would be available for completion by three different Turkers. Since each HIT consisted of 5 utterances and each HIT was completed by 3 Turkers, 22,020 HITs were required to label the entire corpus of 36,696 utterances.

The approach of asking three Turkers to provide a binary judgement of pronunciation quality has the benefits of allowing an inter-rater agreement score to be computed and allows a Likert-like rating scale from 0-3 to be computed for each word. We considered a word *mispronounced* if all three Turkers marked the word as mispronounced. If all three Turkers felt the word was mispronounced, this is a pretty good indication that the word has some serious problems. In contrast, if no Turkers felt a word was mispronounced, then we felt this was a pretty good indication that the word was considered *good*. Words that were marked by at least one Turker, but not all three were considered *ugly* words. It's not clear that they were definitely mispronounced, but because not all Turkers agreed that they were well-pronounced, we can't necessarily considered them good words.

3.4 Annotation Results

We will now discuss the results of the annotation in terms of cost, efficiency, and inter-rater agreement. We will also discuss the correlation of patterns of phonetic insertion, deletion and substitution when combined with the Turker annotations. The results indicate that this annotation method is efficient, inexpensive, and sufficiently reliable. Our results also lead to a simple algorithm that can be used to phonetically label pronunciation quality.

3.4.1 Efficiency

The CU-CHLOE corpus has 36,696 utterances and contains a total of 306,752 words. This data was divided among 22,020 HITs with 5 utterances per HIT. Each HIT was completed by 3 different Turkers. This resulted in 920,256 judgements of pronunciation quality.

The data were published to AMT for assignment to Turkers on Oct 1, 2010 at 19:28. The final datum was submitted by a Turker on Oct 2, 16:28. Thus, all the data were annotated in 21 hours—less than a single day. In contrast, a similar task on a corpus of only approximately 1,700 utterances (about 17,000 syllables) was annotated by 6 expert annotators using a similar web-based interface over the course of about 2 months [183].

3.4.2 Cost

We offered a reward of \$0.05 (USD) per accepted HIT. Amazon also charges a small commission for providing the AMT service. The grand total for annotating 22,020 HITs was 22,020 × \$0.05 + \$110.10 = \$1,211.10 (USD). As noted at the start of this chapter, a very conservative estimate of the cost of annotating this same set of utterances by experts would be about \$15,000.00. The cost of using AMT is about 8.1% of this estimate, which is a substantial savings.

3.4.3 Agreement among raters

An important consideration for annotations performed by multiple people is whether or not they agree with each other. A high degree of agreement indicates that the task is both fair and consistent. One method for measuring the agreement among raters is to compute the percentage agreement between pairs of raters—that is, the proportion of the time a pair of annotators agreed that a word was well-pronounced or mispronounced.

Percentage agreement does not give a full picture of the extent of agreement and will give a false impression on highly skewed data. For example, when 80% of the words are

Turkers	Entire corpus		Hand labeled corpus	
	# Words	% Total	# Words	% Total
0	255,679	83.4%	29,706	80.6%
1	26,281	8.6%	3,796	10.3%
2	12,141	3.9%	1,638	4.4%
3	12,651	4.1%	1,735	4.7%
Total	306,752 (36,696 utterances)		36,874 (5,597 utterances)	

Table 3.1: This table shows a break down of how many Turkers thought each word was mispronounced. The left column indicates the number of Turkers who marked the word as mispronounced, the remaining columns indicate the number of words that fall into each category and relative distribution among all the words in the corpus. These numbers were computed for the entire corpus and over the portion of the corpus that had been hand-transcribed.

marked as well-pronounced by each annotator, the agreement due to chance is much greater than if the data were more balanced. Additionally, if one annotator marked 80% of the words as well-pronounced, then the other annotator could mark 100% of the words as well-pronounced and achieve an 80% agreement.

The Turker annotations are strongly skewed towards marking most words as well-pronounced. Table 3.1 shows a breakdown of the words as annotated by the Turkers. The left-most column of the table is how many Turkers felt that a word was mispronounced. The second column is the number of words that fell into each category. One way of reading this table, for example, is that there were 26,281 words that only one Turker felt were mispronounced. There are two breakdowns shown in the table. The left breakdown is for the entire corpus. The right breakdown is for only those words for which a hand phonetic transcription is available. As Table 3.1 demonstrates, the annotations received for the CU-CHLOE corpus are highly skewed, so another measure of agreement is required.

$$\kappa = \frac{P(a) - P(e)}{1 - P(e)} \tag{3.1}$$

One such measurement is the Cohen Pairwise κ [37]. Kappa attempts to account for the amount of agreement that occurred through chance. In Equation 3.1, $P(a)$ is the proportion of the time two annotators agreed, $P(e)$ is the estimated probability of agreement due to chance, and the denominator is the estimated probability that agreement was not due to

		Turker 1	
		Good	MP
Turker 2	Good	A	B
	MP	C	D

Table 3.2: Example confusion matrix. A is the number of times Turker 1 agreed with Turker 2 that a word was well-pronounced, B is the number of times Turker 1 said a word was mispronounced and Turker 2 said the word was good.

	Turker 1	Turker 2	Turker 3
Turker 1	1.0 (100%)	0.514 (91.5%)	0.525 (91.8%)
Turker 2		1.0 (100%)	0.520 (91.7%)
Turker 3			1.0 (100%)

Table 3.3: Table of κ-scores as if computed from only 3 Turkers. Numbers in parenthesis are percent agreement.

chance. An intuitive interpretation of κ is that it is the difference between the proportion of agreements minus the estimated probability of chance agreement, both normalized by the probability that agreement was not due to chance, and estimated from the data.

The estimation of $P(a)$ and $P(e)$ is performed using a confusion matrix. An example matrix is shown in Table 3.2. In this example, $P(a)$ is given by $P(a) = \frac{A+D}{A+B+C+D}$, or the number of times the Turkers agreed over the total number of judgements.

The estimated chance of agreement, $P(e)$, is computed from the sum of the probabilities, $P(e_g)$ and $P(e_b)$. $P(e_g)$ is the estimated joint probability that both Turkers said a word was good, and $P(e_b)$ is the estimated joint probability that both Turkers said a word was mispronounced. In this example, $P(e_g) = \frac{A+C}{A+B+C+D}\frac{A+B}{A+B+C+D}$, or the proportion of times Turker 1 said words were good times the proportion of times Turker 2 said words were good; analogously, $P(e_b) = \frac{B+D}{A+B+C+D}\frac{C+D}{A+B+C+D}$.

Cohen Kappa assumes that the same 2 raters are used for each of the items under consideration. Table 3.3 shows the κ scores for the CU-CHLOE corpus under the assumption that the first annotator for each utterance is the same person, the second annotator for each utterance is the same person, and so on. A κ score in the 0.4-0.6 indicates a moderate level of agreement that is not due to chance. A κ of 0.0 indicates no agreement above a chance level.

The way that Amazon Mechanical Turk records HIT results means that we cannot assume that all of the first annotators, second annotators, and third annotators are the same. That is, we know that three Turkers annotated each datum, but we can't guarantee that the same three Turkers annotated all the data. This means that the κ statistic computed in this way may not capture an accurate picture of agreement. The next section derives an extension of the κ statistic that is more principled and well-defined.

3.4.4 Aggregated κ

Our approach solves the problem of unmatched annotators by grouping the words into sets associated with unique Turker pairs, averaging the κ values computed from subsets with a common number of overlapping utterances, and then taking a weighted average of all these groups. This method is preferable to computing the κ as we did above because it takes into account the fact that different Turkers actually labeled each utterance.

AMT assigns a unique *TurkerID* to every Turker. When the Turker completes a HIT, their TurkerID is recorded with their work. We can use this information to determine all the unique pairs of TurkerIDs from the data. Each pair of Turkers will have annotated a common subset of the data. Additionally, these pairs of Turkers can be grouped into subsets of Turkers who annotated the same number of utterances (although they won't necessarily be the same utterances). We'll call this number the *annotation overlap*. A κ can be computed for each subset of Turker pairs that have the same annotation overlap.

The annotation of the CU-CHLOE corpus was performed by 463 unique Turkers. There were 10,511 Turker pairs, or pairs of Turkers who annotated the same utterances. The annotation overlap ranges from a minimum of 1 utterance to a maximum of 390 utterances. The mean annotation overlap was 10.5.

As a simplified example, consider that our task is to annotate a corpus of 20 words. If Turkers A, B, and C annotated words 1 through 10, and Turkers B, C, and D annotated words 11 through 20, then we can identify five unique Turker pairs: (A,B), (B,C), (A,C), (C,D), and (B,D). The possible Turker pair (A,D) is not included because it has an annotator overlap of 0. Four of these pairs annotated 10 words—an annotation overlap of 10. One pair annotated 20 words and has an annotation overlap of 20.

Each of their individual κ values can be computed. The aggregated κ is the weighted mean of all these κ values, where the weight is the number of Turker pairs for a particular annotation overlap divided by the total number of Turker pairs.

$$\kappa = \frac{1}{\sum\limits_{s \in S} |T_s|} \sum_{s \in S} |T_s| \sum_{t \in T_s} \frac{P(a|t) - P(e|t)}{1 - P(e|t)} \tag{3.2}$$

This computation is shown in Equation 3.2, where $P(a|t)$ is the proportion of words for which the annotators agreed, $P(e|t)$ is the estimated probability of chance agreement, and T_s is the set of Turker pairs that have an annotator overlap of s. The outer summation in Equation 3.2 weights each κ by the number of Turker pairs who share a given annotator overlap. This is to account for the fact that there are not equal numbers of Turker pairs for each possible annotator overlap in the corpus. Intuitively, we trust the mean κ more if more Turker pairs contributed to it.

Most research that employs κ to measure agreement records a large number of labeler judgements—the labelers have a large annotation overlap. These large sample sizes provide more stable estimates of $P(a)$ and $P(e)$. When computing the aggregated κ from Turker data, we can no longer be assured that a Turker pair has a high annotation overlap.

Figure 3-2 shows a histogram of the frequency of groups with common annotation overlaps. As can be seen from the plot, Turker pairs with low annotator overlap—5 or 10 utterances—make up the majority of the subsets in the CU-CHLOE corpus. There are special considerations that must be given for those subsets where there were only a small number of overlapping utterances.

It is not clear that computing an aggregate κ from subsets with small annotation overlap gives an accurate estimation. A Turker pair that annotates a small number of utterances has a higher chance of both marking every word as well-pronounced or mispronounced. The effect of this on κ is that $P(e)$ is computed to be 1, making the denominator in the κ equation 0, and hence undefined. To handle these cases, we chose the convention that the value of κ would be 0 when it is undefined, indicating that the agreement was all due to chance.

Figures 3-3a and 3-3b show histograms of κ values for Turker pairs with annotation

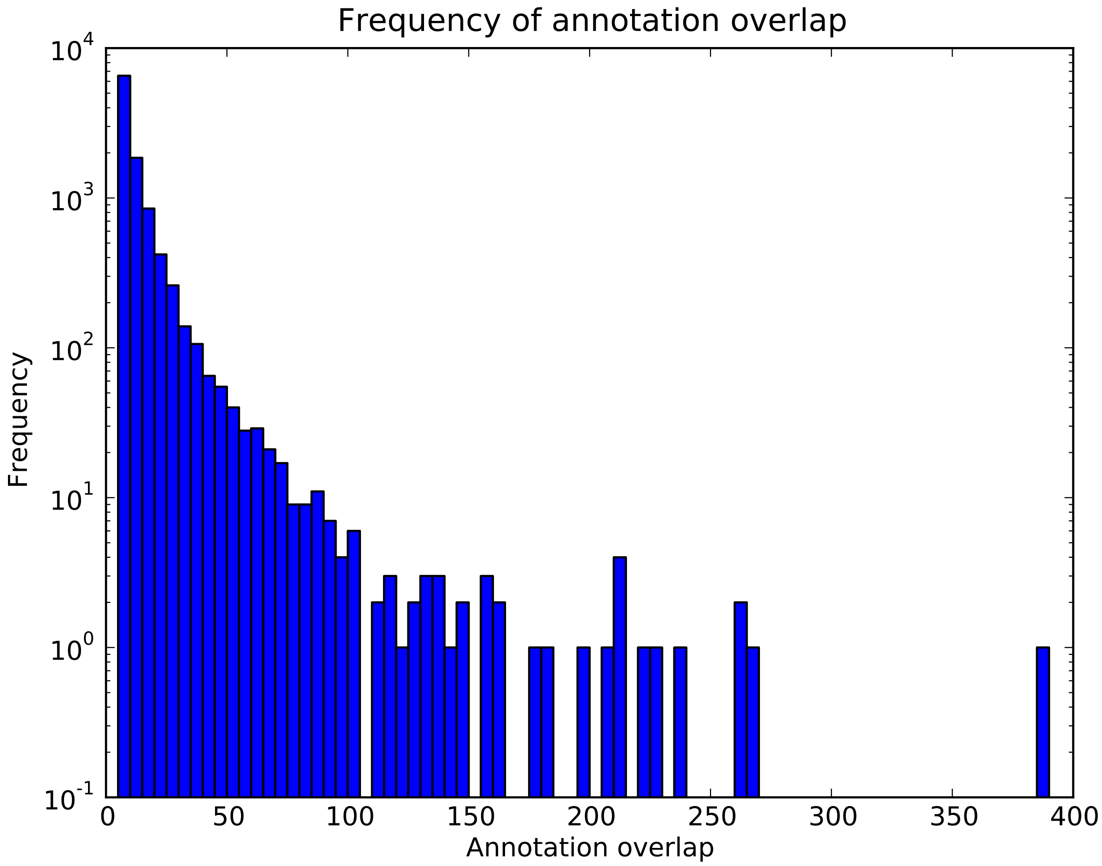
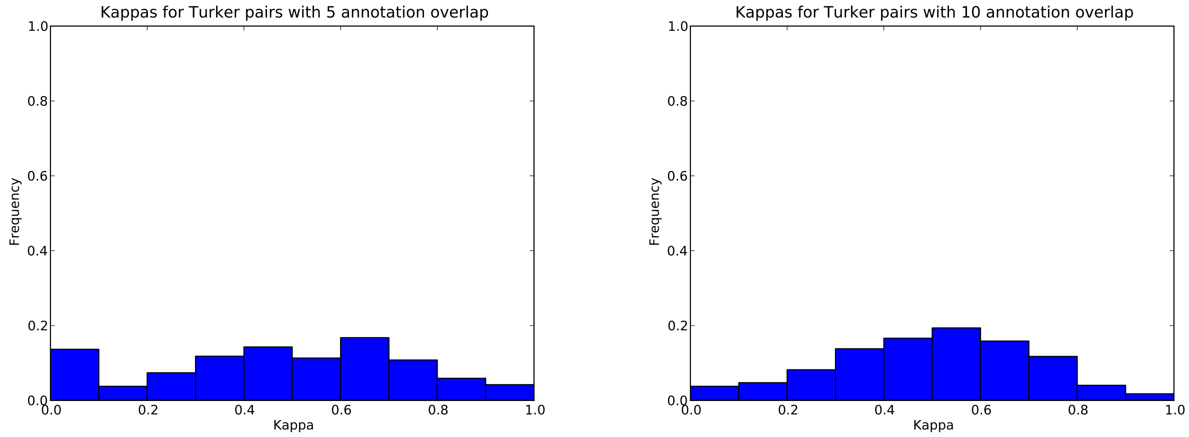


Figure 3-2: This shows the number of Turker pairs that annotated some common number of utterances. This plot shows that most pairs of Turkers overlap on a small number of utterances.



(a) Among Turker pairs that only annotated 5 utterances, there were 852 pairs that had no measurable agreement above chance ($\kappa = 0$). This represents about 13.0% of those pairs, which indicates that five utterances is too small an overlap to accurately gauge agreement.

(b) Among Turker pairs that annotated 10 or more utterances, there are only 61 pairs that had no measureable agreement above chance. These pairs are all in the set of Turker pairs that annotated 10 common utterances and represent only 3.3% of the data in that group. All other sets of Turker pairs with larger numbers of common utterances had no such problems.

Figure 3-3: Comparison of k for groups of Turkers with 5 and 10 annotation overlaps.

overlaps of 5 and 10, respectively. As can be seen from Figure 3-3a, there were 852 Turker pairs who were in complete agreement. In contrast, for an annotator overlap of 10, shown in Figure 3-3b, there were only 56 such instances. For larger annotator overlaps, there were no instances—all Turker pairs had some amount of disagreement.

In the final analysis, we chose to only compute the aggregated κ by considering those Turker pairs with an annotation overlap of 10 or more. We ignored Turker pairs with annotation overlap of 5 because computing a κ value for such a small number of utterances tended to produce a large number of undefined κ values.

We want to establish that computing aggregated κ produces reasonable results. The blue line (and gray errorbars) in Figure 3-4 show how the value of κ varies with the value of the annotator overlap. The red line at 0.51 is the aggregated κ for the CU-CHLOE dataset. The green dashed lines represent the κ values from Table 3.3.

At small values of annotator overlap, the computed κ mean is more stable, although it displays higher deviation. As annotator overlap increases, the value of the κ mean becomes more erratic due to the fact that there are not as many Turker pairs who share the same high

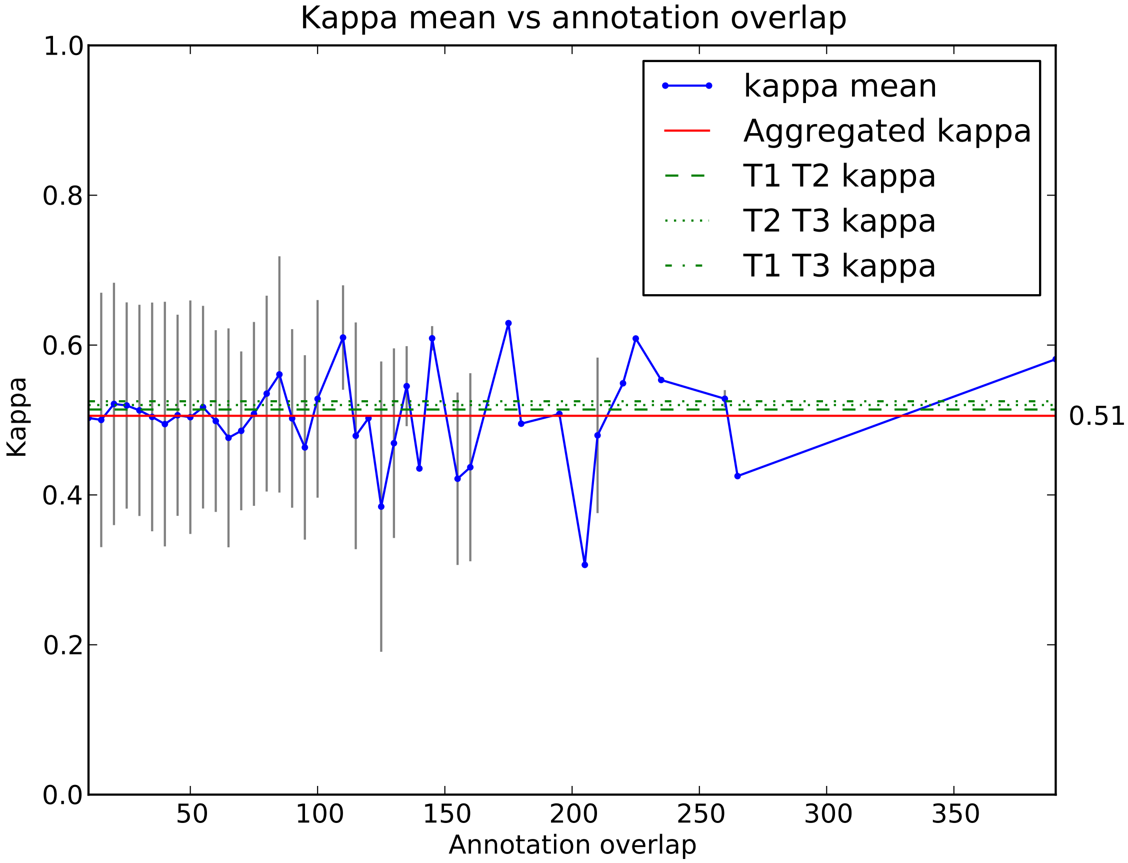


Figure 3-4: Plot of the mean κ value and standard deviation of κ values for Turker pairs plotted against the number of utterances the Turker pairs annotated together.

Prompt	worth	thing	thick	wrath	myth
Machine Transcription	w **er** th	th ih ng	th ih k	**r ae** th	m ih th
Human Transcription	w **ee** th	th ih ng	th ih k	**w ao** th	m ih th
Mispronunciation (diff)	worth	thing	thick	wrath	myth
Mispronunciation (Turker)	worth	thing	thick	wrath	myth

Table 3.4: This table illustrates that differences between a canonical labeling using machine transcription and human transcription do not indicate that humans would perceive the word as mispronounced. The phones on the **bold** font are those phones that differed between the machine and human transcriptions. The words in red indicate words that would be considered mispronounced.

annotator overlap. While having more high annotator overlap subsets is preferable because each Turker pair provides more data to compute κ, the majority of the Turker pairs have low annotator overlap. The computed mean is thus more stable. This is why the aggregated κ weights annotator overlap subsets with more Turker pairs higher than those with low annotator overlap.

The fact that the aggregated κ is comparable in value to the other κ values is encouraging, but it should be considered more trustworthy due to the fact that it was arrived at in a principled manner. It also indicates that we have a moderate amount of agreement across all subsets of Turkers. We will now turn our attention to how this can be used to derive a phonetic labeling algorithm for a mispronunciation corpus.

3.4.5 Pronunciation Deviation and Mispronunciation

A common assumption when constructing a corpus for mispronunciation detection algorithms is to assume that a difference between a canonical phonetic transcription and a hand phonetic transcription is equivalent to a mispronunciation. This canonical transcription can be produced from a baseform dictionary or from a forced alignment through a native language recognizer. We will use the annotations collected from Turkers to show that this assumption does not necessarily hold.

The first row in Table 3.4 shows an English language prompt from the CU-CHLOE corpus. Rows 2 and 3 show two phonetic transcriptions of an audio recording for the same prompt. The first transcription was produced from a forced alignment by the SUMMIT [252]

Annotation Class	# Words	# Substitutions	# Insertions	# Deletions
Good	29,706	14,073 (0.47)	3,413 (0.11)	2,253 (0.07)
Ugly	5,433	4,413 (0.81)	1,362 (0.25)	757 (0.14)
Mispronounced	1,735	2,160 (1.24)	434 (0.25)	523 (0.30)

Table 3.5: This table shows that as more Turkers felt the words were mispronounced, the rate (per word) of substitutions, insertions, and deletions increase. The total numbers of substitutions, insertions, and deletions are shown in the final row.

landmark based recognizer with American English acoustic models. The forced alignment will constrain the recognizer to choose the best acoustic labels that are allowed by the standard phonological rules for American English for the prompt words. The second transcription is a hand transcription by an expert phonetician participating in the AESOP initiative.

An alignment using the least-cost edit distance was performed between the two transcriptions. This procedure identified pronunciation variations in terms of the edit operations of substitution, insertion, and deletion. Differences in the phonetic transcriptions are marked in **bold** font. In this example, only substitution of phonetic labels were found, though insertions and deletions were found in other data.

The fourth row in Table 3.4 shows the words in a font that would be marked as mispronounced if only differences in the transcriptions were used to identify mispronunciations. The fifth row shows the words that were marked as mispronounced by the Turkers. As the final two rows show, mispronunciations cannot always be determined solely from differences in phonetic transcriptions.

3.5 Labeling Algorithm

Although differences in transcriptions cannot be used alone to indicate mispronunciations, the word-level Turker annotations and these differences can be combined to provide phone-level annotations of mispronunciations where substitution has occurred. We will start by examining the types and rates of phonetic differences—substitutions, insertions, and deletions—that exist for words that are considered *good*, *ugly*, and *mispronounced* (annotation class). We will then focus on substitutions and examine statistics from two directions:

Annotation class	Matched	Didn't Match
Good	15,429 (92.7%)	14,277 (70.5%)
Ugly	1,152 (7.0%)	4,282 (21.1%)
Mispronounced	56 (0.3%)	1,679 (8.3%)

Table 3.6: The columns in this table indicate whether or not the machine phonetic transcriptions matched the hand phonetic transcriptions. The rows indicate the class of pronunciation quality determined by the number of Turkers who felt the words were mispronounced. For example, 92.7% of the words where the transcriptions matched fell into the good class (i.e. no Turkers felt the word was mispronounced.)

the annotation class when the hand and phonetic transcriptions match vs when they do not match, and the number of substitutions in a word when it falls into one of the annotation classes.

Table 3.5 shows the annotation classes, number of substitutions, insertions, and deletions, and associated rates (per word) for each type of edit operation. For example, there were an average of 1.24 substitutions, 0.25 insertions, and 0.30 deletions for words that were judged to be mispronounced. There are clear relationships between the annotation class and the rates of substitutions, insertions, and deletions. The rate of substitution for words judged as mispronounced is 2.63 times the rate of substitution for words judged to have good pronunciation. Further, the most common edit operation—in terms of both rate and raw number—is substitution, indicating that substitution errors contribute the most to mispronunciation in this corpus.

We can now look at what characterizes mispronunciation in two directions. In one direction, we can look at what annotation class a word falls into if the machine and human transcriptions match. In the other direction, we'll look at how many of the words have substitution errors if we first look at the annotation class the word was assigned by the Turkers.

Table 3.6 shows that when the machine transcription and hand transcriptions match, almost none (0.3%) of the words were considered by Turkers to by mispronounced, and only 7.0% were considered *ugly*. Thus, 92.7% of these words were considered good. When the transcriptions didn't match, there is a change in the distribution, and 29.4% of these words were considered either ugly or mispronounced by the Turkers. This information alone is encouraging, but it is not enough to devise an algorithm.

	Good	Ugly	Mispronounced
With Substitution	7,487 (25.2%)	2,532 (46.6%)	1,445 (83.3%)
Without Substitution	22,219 (74.8%)	2,902 (53.4%)	290 (16.7%)

Table 3.7: The columns in this table indicate the number of words with substitutions when the machine phonetic transcriptions are aligned with the hand phonetic transcriptions. The rows indicate how many Turkers thought the words were mispronounced. For example, 25.5% words that no Turkers thought were mispronounced contain substitutions.

Table 3.7 shows that 83.3% of the words that Turkers annotated as mispronounced contained one or more substitutions. In contrast, only 25.2% of the words Turkers felt were good contained substitutions. There is a direct relationship between the annotation class and the proportion of the words that contain substitutions. When combined with the information that matched transcriptions are overwhelmingly words that fell into the good annotation class, this suggests that an algorithm can be devised to label phones that appear in words that are mispronounced.

Algorithm 1 The algorithm used to label phones for pronunciation quality.

```
for all Utterances with hand transcription do
    Compute forced alignment using native recognizer
    Align machine phonetic transcription with hand phonetic transcription
    for all Phones in utterance do
        if aligned phones do not match then
            if one more more Turkers said the word the phone belongs to was mispronounced
            then
                Mark the phone as ``ugly"
            else if all the Turkers said the word the phone belongs to was mispronounced
            then
                Mark the phone as ``mispronounced"
            end if
        else
            Label the phone as ``good"
        end if
    end for
end for
```

Algorithm 1 takes advantage of these properties of the annotated CU-CHLOE corpus. It iterates through all of the 5,597 utterances with hand transcriptions and aligns them with the machine transcription using the least cost edit distance. It then iterates through all the phones in the utterance. For all the phones that don't match in the aligned transcriptions, it labels the phone as mispronounced, ugly, or good, depending on how the Turkers labeled the word to which the phone belongs.

While the algorithm will miss some substitution errors and all insertion and deletion errors, it has the nice property that no phones will be mislabeled as mispronounced. Phones are only labeled mispronounced or ugly when they are part of a word that was labeled mispronounced or ugly. Because this labeling will only be triggered when the phones are mismatched in the transcriptions, we will be able to capture mispronunciations due to phonetic substitution in 4,282 + 1,679 = 5,961 words out of a total of 4,282 + 1,679 + 1,152 + 53 = 7,169 words (see Table 3.6) labeled as mispronounced, or about 83.1% of the substitution errors present. We hypothesize that the remaining 16.9% of the words with substitution errors reflect variation that are acceptable as alternative pronunciations of the word. For example, English vowels are typically reduced towards the schwa (/ə/ [ax]) position and this would be reflected in mismatched transcriptions, but would not necessarily be considered mispronunciations.

3.6 Labeling Results

The algorithm was run on all 5,597 utterances in the CU-CHLOE corpus. For this analysis, we focus on vowels. although non-native speakers do exhibit mispronunciations for the non-vowels, these are often difficult to analyze for mispronunciation due to the extremely messy spectra of these sounds. In contrast, vowels have generally well defined formants and sound shapes.

Table 3.8 summarizes the results. Overall, a total of 41,677 phones were labeled with 37,691 (90.4%), 2,536 (6.1%), and 1,450 (3.5%) falling into the good, ugly, and mispronounced annotation classes, respectively. The first column shows the phone label. The total number of phones in the corpus is then listed along with the percentage of the total number of phones. For each annotation class (good, ugly, and mispronounced), the total number of instances for each phone label, the percentage of that phone label, and the percentage of

Label	All classes		Good			Ugly			Mispronounced		
	Tot	% Tot	Tot	% Tot	% Cls	Tot	% Tot	% Cls	Tot	%	% Cls
Overall	41,677	-	37,691	90.4	-	2,536	6.1	-	1,450	3.5	-
/ɑ/ [aa]	3,796	9.1	3,654	96.3	9.7	86	2.3	3.4	56	1.5	3.9
/æ/ [ae]	3,508	8.4	3,280	93.5	8.7	175	5.0	6.9	53	1.5	3.7
/ʌ/ [ah]	2,664	6.4	2,440	91.6	6.5	111	4.2	4.4	113	4.2	7.8
/ɔ/ [ao]	4,199	10.1	3,969	94.5	10.5	153	3.6	6.0	77	1.8	5.3
/ɑ^{w}/ [aw]	649	1.6	599	92.3	1.6	24	3.7	0.9	26	4.0	1.8
/ə/ [ax]	2,642	6.3	2,227	84.3	5.9	301	11.4	11.9	114	4.3	7.9
/ɑ^{y}/ [ay]	2,409	5.8	2,098	87.1	5.6	65	2.7	2.6	246	10.2	17.0
/ɛ/ [eh]	2,200	5.3	2,068	94.0	5.5	101	4.6	4.0	31	1.4	2.1
/ɚ/ [er]	3,800	9.1	3,015	79.3	8.0	503	13.2	19.8	282	7.4	19.4
/e/ [ey]	3,477	8.3	3,272	94.1	8.7	145	4.2	5.7	60	1.7	4.1
/ɪ/ [ih]	4,521	10.8	4,221	93.4	11.2	264	5.8	10.4	36	0.8	2.5
/i/ [iy]	1,917	4.6	1,759	91.8	4.7	117	6.1	4.6	41	2.1	2.8
/o/ [ow]	2,551	6.1	2,085	81.7	5.5	282	11.1	11.1	184	7.2	12.7
/ɔ^{y}/ [oy]	1,000	2.4	922	92.2	2.4	39	3.9	1.5	39	3.9	2.7
/Ʊ/ [uh]	773	1.9	682	88.2	1.8	80	10.3	3.2	11	1.4	0.8
/u/ [uw]	1,571	3.8	1,400	89.1	3.7	90	5.7	3.5	81	5.2	5.6

Table 3.8: Number of phones in corpus labeled as ``Good", ``Ugly", and ``Mispronounced".

that annotation class are listed.

For example, the phone /ɑ/ [aa] occurred a total of 3,796 times in the corpus, or 9.1% of the total phones in the corpus. Of the 3,796 instances of /ɑ/, 3,654 were marked good, 86 were marked ugly, and 56 were marked mispronounced. This corresponds to 96.3%, 2.3%, and 1.5% of the instances of the phone /ɑ/. Over the entire corpus, phones marked mispronounced comprised 3.5% of the corpus, but only 1.5% of the instances of /ɑ/ were marked mispronounced. This indicates that /ɑ/ is generally not a major source of mispronunciation.

Over all the phones marked good, instances of /ɑ/ that were marked good appeared 9.7% of the time, 3.4% of the time for phones marked ugly, and 3.9% of the time for phones marked mispronounced. Another way to view this is to note that while /ɑ/ comprises 9.1% of the corpus of vowels, it only accounts for 3.9% of the phones marked mispronounced.

Contrast this with the vowel /ɑ^{y}/ [ay]. Instances of /ɑ^{y}/ appear for 5.8% of the total corpus of vowels, yet it accounts for 17.0% of the vowels marked as mispronounced. This indicates that the speakers in the corpus had difficulty producing this vowel and it is a major source of mispronunciation. There are three vowels that stand out in this regard, /ɑ^{y}/ [ay], /ɚ/ [er], and /o/ [ow].

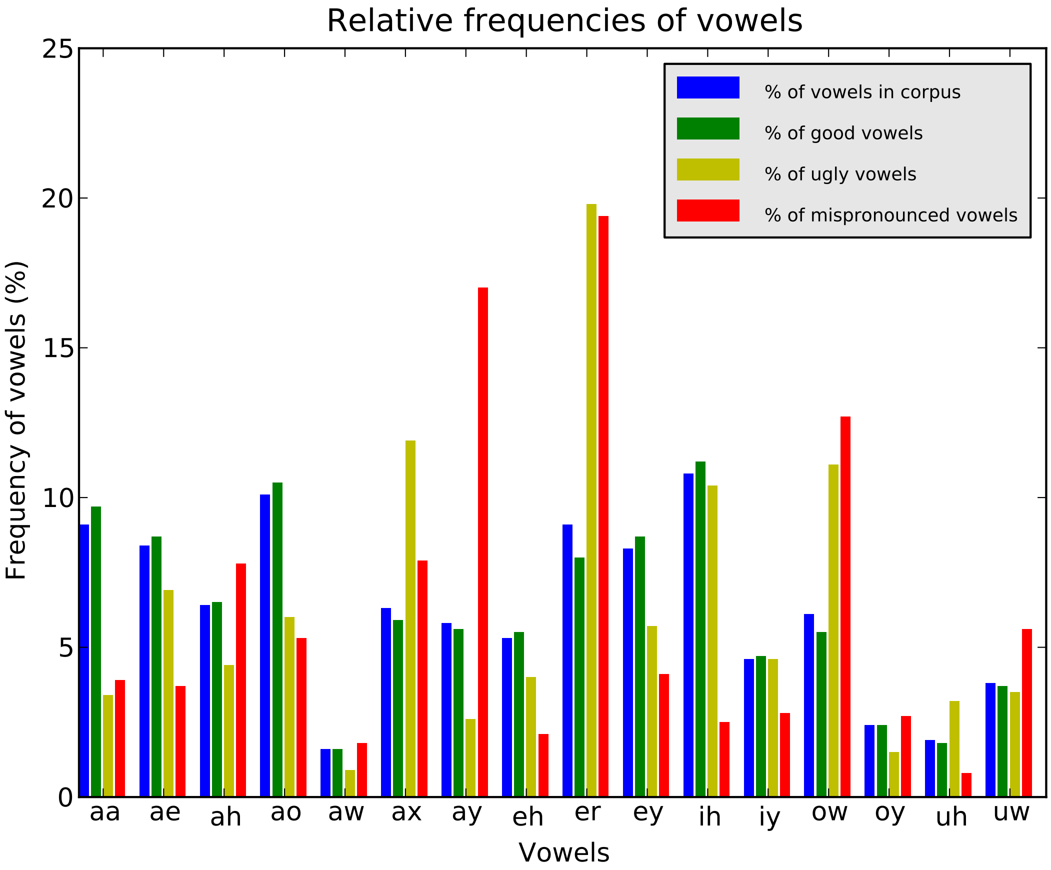


Figure 3-5: Relative frequencies of vowels in the corpus. Blue bars are the frequencies of the vowels in the corpus. The green, yellow, and red bars indicate the frequencies those vowels were labeled good, ugly, and mispronounced relative to the total numbers of good, ugly, and mispronounced vowels.

Another source of trouble is /ə/ [ax], a vowel produced when the vocal tract is in a relaxed state. Although it doesn't stand out as a vowel that was marked mispronounced, it was marked as ugly a disproportionate number of times. Vowels in English are often relaxed towards the schwa position. This data indicates that the CU-CHLOE speakers are performing this vowel reduction in a way that causes some native speakers to perceive it as a mispronunciation. Or it could be that another phoneme in the word was also mispronounced, and this phoneme was the major source influencing judgement.

An alternative way to view this data is shown in Figure 3-5. The blue bars indicate the frequency of the vowels within the corpus. The green, amber, and red bars indicate the relative frequencies of the vowels within their respective annotation classes. From this figure it is easy to see that the vowel /ɚ/ [er] is mispronounced way out of proportion to its relative frequency within the corpus.

3.7 Summary

This chapter presented a crowd-sourced labeling algorithm for creating phone-level labels of mispronunciation. These results were used to create the mispronunciation detector described later in this thesis. It combined the results of a hand transcription, machine transcription, and crowd-sourced word annotation of pronunciation quality. This algorithm is justified with the relative statistical properties of the phones found within the corpus and their relation to word marked as mispronounced in the CU-CHLOE corpus.

Chapter 4

Anchoring Vowels for phonetic assessment

This chapter proposes a novel method for transforming *Mel Frequency Cepstral Coefficients* (MFCCs) [154, 51], frequently used in speech recognition tasks, into a feature space that is more robust for computer aided pronunciation evaluation. Our method estimates the mean MFCCs of specific vowels that represent four key positions of English vowel production. Three positions represent the extremes of where vowels are produced in English and the fourth represents vowel production when the vowel tract is in a completely relaxed state. We show that, by representing speech sounds in relation to these positions, performance on a simple classification task can be significantly improved. We argue from a qualitative and quantitative perspective that this will improve performance in the task of detecting pronunciation errors, presented in the next chapter.

4.1 Motivation

CALL systems frequently employ statistical model scores to produce some measure of pronunciation quality. However, these scores can be very sensitive to intrinsic speaker differences that may not be the result of mispronunciations. Typically, the models that produce these scores are trained using MFCCs as feature vectors. MFCCs are compact representations of the acoustic signal associated with different speech sounds.

In native speech, a specific phone is generally located in a specific region of the MFCC feature space. This location can vary greatly due to the phonetic context in which the phone occurs, speaking conditions, speaker gender, vocal tract length, age, and many other factors. For example, the vowel /i/ [iy] may generally exist in one region of the MFCC space for speaker A and a slightly different region of the MFCC space for speaker B. These locations tend to have even greater variance for non-native speakers.

When training phone class models using MFCCs, the features intuitively specify a location in MFCC space without respect to other phones in the speaker's phonetic inventory. An alternative representation is to *anchor* the MFCCs in relation to another sound in the speaker's phonetic inventory. This brings all the other phones to a similar reference point in MFCC space, thus allowing a more direct comparison of sounds between speakers.

Native and non-native speakers exhibit systematic differences in pronunciation. By representing speech sounds in relation to a common anchor point, this representation takes advantage of the fact that speech sounds are typically differentiated by how the sound is perceived relative to other sounds, and it should allow a more robust assessment of pronunciation. An intuitive understanding of this can be summed up by a simple rephrasing of the statement ``This non-native /i/ [iy] does not sound as if it was produced in the same location as a native /i/ [iy]" to ``This non-native /i/ [iy] does not sound as if it was produced in the same location relative to the speaker's typical production of the sound /ə/ [ax] as a native speaker producing /i/ [iy] would produce it relative to their production of /ə/ [ax]."

4.2 Related Work

Numerous approaches have been proposed to normalize speech to account for speaker dependent variation. Vocal tract length normalization (VTLN) techniques model the length of the vocal tract and warp the acoustic signal to match a reference. In previous work, Nordström and Lindblom [175] scale the formants of the speech by a constant factor determined by an estimate of the vocal tract length from measurements of F_3. Fant [68] extended this by making the scale factor dependent on formant numbers and vowel class. These methods require knowledge of the formant number and frequencies. More recently, Umesh et al. [226, 128] introduced two automatic methods: one uses a frequency dependent scale factor that does not require knowledge of the formant number, and another is based on fitting a model relating the frequencies of a reference speaker to frequencies of a subject speaker.

In contrast to operating on the acoustic signal, Maximum Likelihood Linear Regression (MLLR) [78] attempts to accomodate speaker to speaker variation by adapting the means and variances of existing acoustic models given a relatively small amount of adaptation data. It accomplishes this by estimating linear transformations of model parameters to maximize the likelihood of the adaptation data. Some normalization approaches work directly on the MFCCs extracted as features for speech recognition. Cox [39] implements speaker normalization in the MFCC domain utilizing a filterbank approach to shift MFCCs up and down in the spectrum. He shows that this is a form of vocal tract normalization, and has similarities to a constrained MLLR. Pitz and Ney [187] showed that frequency warping vocal tract normalization can be implemented as linear transformations of MFCCs.

4.3 Approach

Our approach is inspired by the work presented in [156, 218], which used the Bhattacharyya Distance [21] to compute the overall structure of speakers' phonetic spaces. This was conducted in the spirit of work by Jakobson [107] who argued that the study of the sounds of a language must consider the structure of the sound system as a whole. Thus, the structure created by Minematsu et al. modeled a phonetic space in a holistic fashion, as opposed to the typical method for modeling acoustic spaces using MFCCs or other localized features.

This structure (see the graphical representation in Figure 4-1) was essentially a symmetric matrix of the pairwise Bhattacharyya Distances for all the phones in the phonetic space. This representation allowed them to model the pairwise distances for an individual speaker or a population of speakers. They defined a scalar distortion metric based on the normalized difference between the matrices of two phonetic spaces. They used this structure to measure the distortion between Japanese accented English and General American English

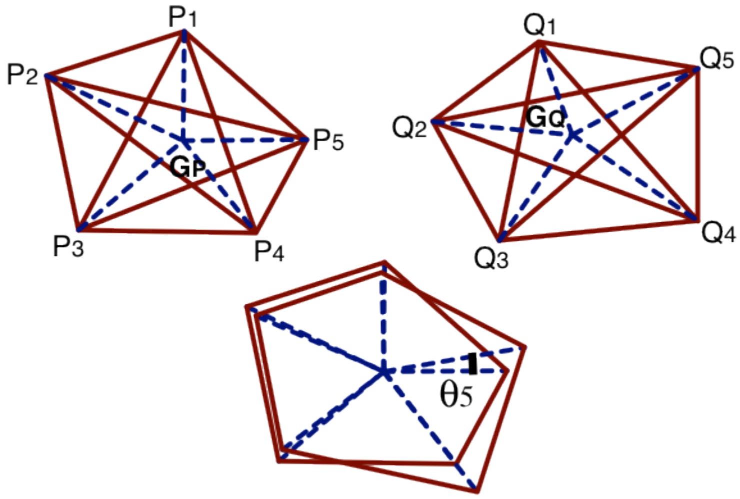


Figure 4-1: A graphical representation of the pronunciation structure defined by Minematsu et al [156].

and found a positive correlation with human assessments of pronunciation quality. One of the limitations of their technique was that it was unable to individually classify or assess sounds.

We hypothesize that vowels may be produced by humans via an internal relativistic model that attempts to maximize discriminability, akin to the principles in [188]. With this idea in mind, we decided to investigate a simple normalization method based on relativizing the Cepstral coefficients to those of a target reference vowel. We therefore propose a simple scheme that intuitively works by anchoring vowel spaces to a common reference point on a per speaker basis. Since speakers are using a common language, common phonetic inventory, and hence a similar vowel space shape, this anchoring should have the effect of shifting speaker vowel spaces into closer proximity.

4.3.1 Data

Our data come from two corpora. The first corpus is the TIMIT corpus [81], consisting of 6,300 (4,380 male, 1,920 female) utterances from native English speakers. The second corpus is the Chinese University Chinese Learners of English (CU-CHLOE) corpus [152] explained in the previous chapter. For the classification experiment, the TIMIT corpus was divided into a training set consisting of 4,620 utterances, and a test set consisting of 1,680 utterances. The CU-CHLOE corpus was divided into a training set of 33,026 utterances and a test set of 3,670 utterances.

The data were force-aligned using a standard SUMMIT [252] recognizer with native English landmark models to obtain a segmentation and assigned reference label for each target vowel. We averaged the MFCCs (14 dimensions) at five regions relative to the vowel endpoints for each segment of speech: 30ms-0ms before the segment (pre), at 0%-30% (start), 30%-70% (middle), and 70%-100% (end) through the segment, and to 30ms after the segment (post).

4.3.2 Anchoring

Anchoring the vowel space entails computing the difference between the mean MFCC values for each anchoring point and the MFCCs for a sample under consideration. For each MFCC measurement, we computed the difference between the measured MFCCs and the mean of a speaker's anchor vowel as shown in Equation 4.1 at corresponding parts of the segment. Mathematically,

$$AC_{i,v} = C_i - \overline{C_v} \tag{4.1}$$

where $AC_{i,v}$ is the normalized MFCC sample at phone segment i using v as the anchor vowel, C_i is the MFCC sample at phone segment i, and $\overline{C_v}$ is the mean MFCCs for a speaker's productions of vowel, v, where v is the anchor point of the transformation.

Anchor points are defined at the vowels /ɑ/ [aa], /i/ [iy], and /u/ [uw], since these quantal vowels [215] exist at relative extremes in the Universal Vowel Space [188], are found in nearly all languages, and should provide relatively stable points of reference. We also anchored points at /ə/, as Puppel and Jahr [188] argue that one of the forces acting on the location of /ɑ/ [aa], /i/ [iy], and /u/ [uw] is a thrust away from the neutral /ə/ in order to maximize discriminability, and Diehl [55] notes that in some respects, /ə/ [ax] is slightly more stable.

A final anchor point was the weighted mean of the speaker's vowels. This virtual anchor point was created to account for data sparseness issues. For example, when a speaker has not produced enough instances of any of the previously defined anchor points. This was especially true in the TIMIT corpus where each speaker recorded only 10 utterances. To mitigate the effects of sparse data, we constructed another anchor vowel, $\overline{C}$-anchor, that consisted of the weighted mean of all the vowels in the speaker's inventory. Mathematically, each anchored feature was computed using

$$AC_i = C_i - \overline{C} \tag{4.2}$$

where the weighted normalized MFCC sample is AC_i and the weighted mean of a speaker's vowels, represented by $\overline{C}$, is defined as:

$$\overline{C} \quad = \quad \frac{1}{\sum\limits_{v \in V} w_v} \sum_{v \in V} w_v \overline{C_v} \tag{4.3}$$

We created a number of different feature sets based on these measurements for use in our analysis. The MFCCs (baseline), /ɑ/-anchor, /i/, /u/, /ə/, and $\overline{C}$-anchor features (Table 4.1) were used to train Gaussian Mixture Model (GMM) classifiers using k-means clustering.

We validate this approach in three ways. First, we perform a simple classification task using a *Gaussian Mixture Models* (GMM) classifier with a maximum of 96 mixtures and trained using the k-means algorithm. We show significant improvements over MFCC based models in classification of the vowels under three conditions: native speakers with native trained models, non-native speakers with non-native trained models, and non-native speakers with native trained models. The improvements we achieve in the classification task indicate that the technique is effective at accounting for speaker differences.

Second, we perform a qualitative analysis where we examine the shape and location of the sample distributions for various phone classes before and after anchoring. Third, we quantitatively assess anchoring by computing statistical distance metrics for the phone classes. We correlate these distances with Amazon Mechanical Turk annotations of pronunciation quality.

4.4 Results

We analyzed the effect of anchoring from three perspectives: on classification performance relative to standard MFCC measurements, qualitatively on comparisons between native and non-native speech, and quantitatively based on correlations of the Bhattacharyya distance metric and Amazon Mechanical Turk annotations.

Classification

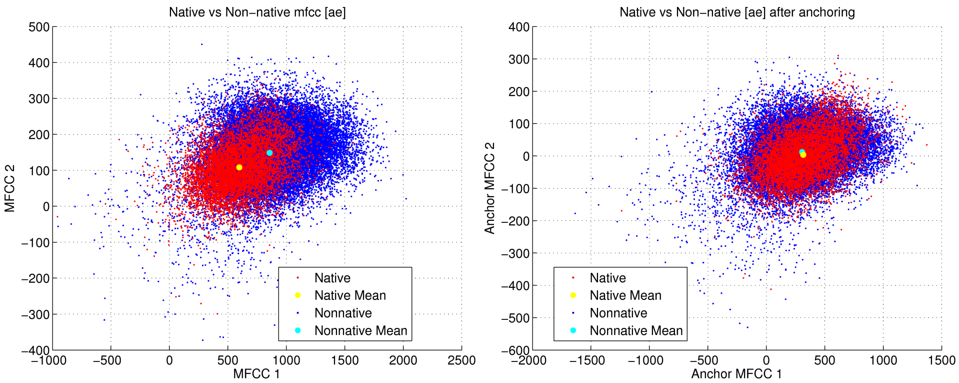
The results for our classification experiments are presented in Table 4.1. Our baselines for comparison are features from Table 4.1 row (a). These are standard sets of MFCCs used for segment models in our classifier. The poor performance for CHLOE, particularly when TIMIT is used for training, reflects the difficulty in pronouncing a non-native vowel.

		Training Data	TIMIT	CHLOE	TIMIT
		Test Data	TIMIT	CHLOE	CHLOE
	a	MFCCs	33.0%	38.3%	48.8%
Features	b	/ɑ/-anchor	31.4% (4.8%)	36.1% (5.7%)	45.4% (7.0%)
	c	/i/-anchor	31.4% (4.8%)	37.0% (3.4%)	45.4% (7.0%)
	d	/u/-anchor	32.4% (1.8%)	36.7% (4.2%)	45.8% (6.1%)
	e	/ə/-anchor	32.2% (2.4%)	36.5% (4.7%)	45.3% (7.2%)
	f	$\overline{C}$-anchor	30.8% (6.7%)	35.7% (6.8%)	44.7% (8.4%)

Table 4.1: Percent error vowel classification. The numbers in parenthesis represent relative error improvement. The classification error decreases significantly with normalization with respect to any vowel or with respect to the weighted average of the vowels.

Table 4.1 presents the error rates when the means of the anchor vowel MFCCs are computed from the labeled test data. The relative performance increases range from 1.8% to 6.6% for the native classifier with native speech, 3.4% to 6.8% for non-native speech with non-native classifier, and 6.1% to 8.4% for non-native speech with the native classifier. Of all the feature sets, the weighted anchor, $\overline{C}$-anchor set realizes the largest improvement across all three cases. This reflects the fact that this anchor generally has more data available to estimate the mean MFCC. We might also conclude that if more data were available for the other anchors, then the advantage of using $\overline{C}$-anchor would be diminished.

Qualitative Assessment



(a) MFCCs

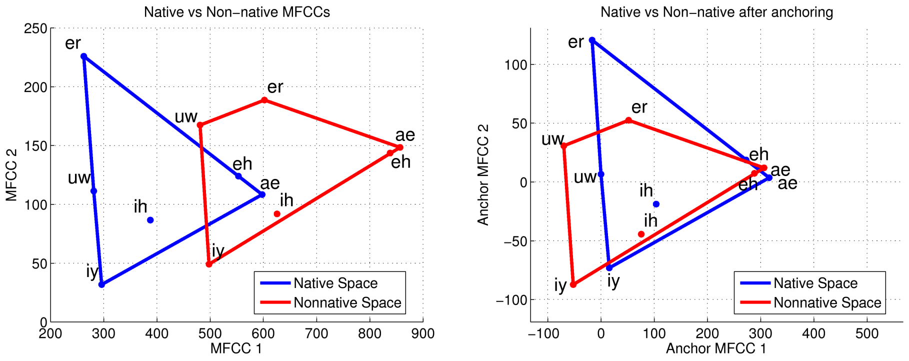
(b) /ə/-normalized

Figure 4-2: Distributions of the first two dimensions of the feature vectors for /æ/ spoken by native and non-native speakers.

To qualitatively understand why we see these performance improvements and why this scheme may be beneficial for assessment, it is helpful to visualize the transformation. Figure 4-2 depicts the effect of the transformation on the native and non-native data for MFCCs 1 and 2 for the vowel /æ/ [ae]. As can be seen from the figure, the mean of the non-native distribution is shifted closer to the native mean. This effect was seen for almost all pairs of vowel distributions (a comprehensive set of visualizations can be found in Appendix B). Note that MFCC 1 captures the total energy of the MFCC spectrum, so this normalization effectively corrects for differences in microphone gain as well.

By using only one point as the reference point, we are essentially shifting the entirety of the speaker's vowel space without affecting its shape. This creates a feature space in which the samples still exist in the same relative proximity to each other. This would be important for pronunciation assessment of individual vowels. Figure 4-3a depicts a representation of the MFCC vowel spaces of native and non-native speakers. The points represent the means of a subset of the vowel distributions for both sets of speakers. Figure 4-3b depicts the vowel spaces after they have been anchored by /ə/ [ax].

The overall shapes of the spaces have not been affected by the anchoring, but the spaces now directly overlap each other. The anchoring provides a direct comparison of the vowel



(a) MFCCs

(b) /ə/-normalized

Figure 4-3: Comparison of feature space for the first two dimensions. The large points represent the means of the features measured at the mid-point for the corresponding vowel. The outlined shapes (red and blue) form the convex hull of the space.

	[aa]	[ae]	[ah]	[ao]	[aw]	[ax]	[ay]	[eh]	[er]	[ey]	[ih]	[iy]	[ow]	[oy]	[uh]	[uw]
[aa]	-	-	64	671	-	7	2	3	-	-	1	-	63	1	5	7
[ae]	475	-	10	19	1	27	-	224	-	8	2	2	-	-	1	-
[ah]	36	9	-	124	4	173	-	-	1	-	2	1	4	1	29	74
[ao]	17	10	13	-	11	12	-	-	-	1	-	-	123	3	93	38
[aw]	26	-	12	2	-	4	-	-	-	-	-	-	17	1	-	1
[ax]	33	36	3	63	4	-	-	35	-	6	264	3	13	-	112	10
[ay]	25	1	-	1	5	2	-	24	-	32	110	46	-	-	-	2
[eh]	-	607	-	-	-	12	3	-	1	6	15	15	-	-	-	5
[er]	9	1	14	26	13	2361	15	6	-	10	14	7	9	1	7	11
[ey]	6	13	3	2	-	11	14	432	1	-	166	1	1	1	-	-
[ih]	-	-	1	1	1	966	4	11	-	2	-	259	-	-	-	5
[iy]	1	3	-	-	-	33	-	21	1	9	200	-	-	-	-	1
[ow]	3	-	44	332	38	14	1	1	-	-	-	2	-	1	5	43
[oy]	1	-	1	11	-	-	4	-	-	1	1	-	24	-	-	1
[uh]	-	-	-	-	-	3	-	-	-	-	-	-	1	-	-	145
[uw]	-	-	6	4	2	3	-	-	1	-	1	5	76	-	102	-

Table 4.2: Confusion matrix showing the number of times the vowels down the left column were substituted by the vowels along the top row.

spaces when the relative positions of the vowels are considered. For example, we can clearly see /ɚ/ [er], a sound that appears most often as mispronounced (Table 3.8), is located in very different relative positions between the native and non-native populations. It is, in fact, located towards the middle of the represented pronunciation space, where one would find the vowel /ə/ [ax]. Table 4.2 shows that /ɚ/ [er] is often confused with /ə/ [ax] when the canonical and hand transcriptions are aligned.

Additionally, /æ/ [ae] and /ɛ/ [eh] are all clustered together and the non-native /ɛ/ [eh] exists in a different position relative to the non-native /æ/ [ae] when compared with the relative positions of the native equivalents. Table 4.2 shows the number of times the vowels in the left-most column were substituted by vowels in the top row. We can see, for example, that the proximity of /æ/ [ae] and /ɛ/ [eh] to each other is a large source of confusion.

In interpreting this type of plot, we should be careful to note that there are other possible explanations for the shapes seen. For example, the vowel /ɚ/ [er] is interesting because it is also a vowel that is disproportionately (to the rest of the corpus) marked as ugly. It is marked ugly nearly twice as often as it is marked mispronounced, and this indicates ambivalence on the part of the Turkers when they marked words containing the /ɚ/ [er] vowel. We could interpret this to mean that Cantonese speakers have difficulty with the vowel, or it could be an artifact of the source for their English instruction. The non-native speakers were from Hong Kong, and it is more than likely that they have been instructed in British pronunciation. American English and British English have a number of differences, one of which is the difference in the phoneme [er] as in the word ``worth." It is unsurprising that there is such a relative difference in the locations of the phone and that it seems to be a controversial sound among the Turkers.

Quantitative Assessment

The anchoring method presented transforms individual phone instances, effectively altering the distributions for each phone class. Minematsu et al.'s work, which inspired our research, utilized the Bhattacharyya distance to assess pronunciation. The Bhattacharyya distance is defined for multivariate Gaussian distributions, $\mathcal{N}_1 = (\mu_1, \Sigma_1)$ and $\mathcal{N}_2 = (\mu_2, \Sigma_2)$, as follows:

$$BD(\mathcal{N}_1 \parallel \mathcal{N}_2) = \frac{1}{8}(\mu_1 - \mu_2)^T \Sigma^{-1} (\mu_1 - \mu_2) + \frac{1}{2} \ln(\frac{det\Sigma}{\sqrt{det\Sigma_1 det\Sigma_2}}) \tag{4.4}$$

where μ_i and Σ_i are the mean and covariance for the distribution $N(\mu_i, \Sigma_i)$ and $\Sigma = \frac{\Sigma_1 + \Sigma_2}{2}$. It is a measurement of the separability of two distributions.

The pronunciation structure Minematsu computed was a representation of the overall phonetic space of the speaker based on this distance, so the positive correlations they found applied to overall pronunciation quality. We want to assess whether this distance can be utilized at a phone class level. To this end, we compute correlations of the Bhattacharyya distances between the phone class distributions of native and non-native speakers with the

A Vowel	B MFCC	C $\overline{C}$-anchor	D Δ	E Good	F Ugly	G MP
/ɑ/ [aa]	227.78	194.53	-33.25	96.3	2.3	1.5
/æ/ [ae]	24.68	35.19	10.51	93.5	5	1.5
/ʌ/ [ah]	341.37	95.10	-246.27	91.6	4.2	4.2
/ɔ/ [ao]	198.93	96.13	-102.8	94.5	3.6	1.8
/ɑ^{w}/ [aw]	37.72	48.50	10.78	92.3	3.7	4.0
/ə/ [ax]	16.08	36.46	20.38	84.3	11.4	4.3
/ɑ^{y}/ [ay]	277.33	173.64	-103.69	87.1	2.7	10.2
/ɛ/ [eh]	169.22	40.64	-128.58	94	4.6	1.4
/ɚ/ [er]	129.32	94.35	-34.97	79.3	13.2	7.4
/e/ [ey]	409.52	88.25	-321.27	94.1	4.2	1.7
/ɪ/ [ih]	21.51	8.48	-13.03	93.4	5.8	0.8
/i/ [iy]	41.41	20.97	-20.45	91.8	6.1	2.1
/o/ [ow]	254.63	137.70	-116.92	81.7	11.1	7.2
/ɔ^{y}/ [oy]	643.41	87.69	-555.82	92.2	3.9	3.9
/ʊ/ [uh]	219.18	59.24	-159.94	88.2	10.3	1.4
/u/ [uw]	163.93	42.99	-120.94	89.1	5.7	5.2

Correlations	MFCC	0.077	-0.46*	0.16
	$\overline{C}$-anchor	-0.04	-0.46*	0.41**

Table 4.3: Bhattacharyya distances between native and non-native models trained on different feature sets and their correlations with pronunciation quality proportions for different vowels. The annotation classes are based on the labeling algorithm from Chapter 3. *Good* vowels are those vowels marked by no Turkers, *Ugly* vowels are those marked by at least one Turker as mispronounced, and *Mispronounced* (MP) vowels are those marked by all three Turkers as mispronounced. * $p < 0.1$, ** $p < 0.15$

proportion of each of the phone classes that were labeled *Good*, *Ugly*, and *Mispronounced* according to the algorithm from Chapter 3. This will also quantitatively confirm that the distributions between native and non-native speakers have moved closer together.

We measured the Bhattacharyya distance between native and non-native single Gaussian distributions of the MFCC values taken at the five regions specified in Section 4.3. These values are in Column B of Table 4.3. We also measured the Bhattacharyya distance of the Gaussian distributions trained from $\overline{C}$-anchors (see Table 4.3, Column C). Column D shows the change in distance from pre-anchored features to post-anchored features.

Table 4.3 also shows the proportion of each vowel marked *good* (Column E), *ugly* (Column F), and *mispronounced* (Column G) by Amazon Mechanical Turkers. We computed the Spearman Rho rank ordered correlation to determine if there exists any relationship between the Bhattacharyya Distance and the proportions phone instances from each annotation class. The correlation of the distance with each annotation class is shown in the bottom two rows for the anchored and unanchored versions of the vowels. For example, this table shows that the correlation of the distances for distributions trained using MFCC-based features is 0.077.

Overall, the distance is not correlated to both unanchored and anchored versions of the features for the *good* class, is negatively correlated to both feature versions for the *ugly* class at a 0.1 significance level, and is only positively correlated to the anchored version of the features for the *mispronounced* at a 0.15 significance level.

These results are difficult to interpret. First, the negative deltas on the Bhattacharyya distances quantitatively confirm the qualitative analysis that the anchoring moves the vowel classes to be in closer proximity. The distributions for every vowel class except /æ/ [ae], /ɑ^{w}/ [aw], and /ə/ [ax] show varying degrees of moving closer together. As a normalization method, we can conclude that it is having the desired effect of compensating for intrinsic speaker differences.

Second, the correlations show that Bhattacharyya is not necessarily a strong indicator of good pronunciation quality for either anchored or unanchored versions of the features. This could be due to the highly skewed distribution of the annotation classes—the vast majority of the vowels were marked with *good* pronunciation. This also indicates that there are other features that the Turkers paid attention to in order to arrive at the conclusion that the vowels in question were well-pronounced.

Third, the Bhattacharyya distance is negatively correlated, -0.46 ($p < 0.10$), to vowels in the *ugly* annotation class. This correlation is the same for both anchored and unanchored versions of the vowels and says that, as the distributions between the native and non-native speakers moved closer together, there was a greater chance that at least one Turker would indicate a mispronunciation occurred. This would be analogous to the situation seen with the spatial proximity of /æ/ [ae] and /ɛ/ [eh]—there is a high likelihood that these vowels are confused, and this happens irrespective of the anchoring. This is also supportive of the idea that the Turkers varied in their judgement of pronunciation if the vowels were close—

the same instance of pronunciation may be marked differently by individual Turkers. This is supported by the fact that both /æ/ [ae] and /ɛ/ [eh] have larger proportions of *uglies* than *mispronounced*.

Finally, the Bhattacharyya Distance is positively correlated, 0.41 ($p < 0.15$), to vowels in the *mispronounced* annotation class, but only after the vowels have been anchored. What this says is that, as the distributions are further and further separated under anchoring, then it is more likely that the vowels will be considered mispronounced by Turkers. Although the standard threshold of significance ($p < 0.05$) is not met, we still could consider applying the distance measure to pronunciation evaluation given the other results presented earlier. We shall see in the next chapter how this information can be exploited to detect mispronunciations. For example, the vowel /ɑ^y/ [ay] has a larger Bhattacharyya Distance, and a correspondingly larger proportion marked as mispronounced. This correlation is enhanced after anchoring.

4.5 Summary

This work introduced a simple feature normalization scheme for vowel classification and subsequent vowel assessment of non-native speakers. The MFCC features for particular speakers were transformed using simple operations into features anchored at a common reference point. We showed that this results in increased classifier performance. We qualitatively and quantitatively examined the effect of the transformation on the distributions of vowels between native and non-native speakers and the shape of the vowel space. Our quantitative analysis included a discussion on correlations with the Bhattacharyya distance —used in prior work for pronunciation assessment—and showed that anchoring improved correlation of the Bhattacharyya distance to *mispronounced* vowels. These results will be exploited in the next chapter on mispronunciation detection. The correlations do not support using Bhattacharyya Distance itself to detect mispronunciation, but when combined with other information, the distance measurement may enhance performance.

Chapter 5

Mispronunciation Detection

This chapter details the implementation of a method for accurately detecting pronunciation errors at a phonetic level. We use a decision tree classifier framework with parallel native and non-native models to precisely detect phonetic pronunciation errors. We also show that the anchoring method detailed in the previous chapter provides more stable features for detecting mispronunciations. Under the assumption that incorrectly labeling a phone as mispronounced is more damaging than incorrectly labeling a phone as well-pronounced, this system focuses only on detecting mispronunciations with high specificity. Therefore, we are willing to tolerate a number of false rejections (i.e. phones that were marked as mispronounced, but were not detected as mispronunciations by the system). We quantitatively analyze the performance of this system from a classification performance standpoint and qualitatively evaluate the decision tree rules.

5.1 Motivation

Pronunciation evaluation is an important component of Computer Aided Language Learning (CALL) systems. A common approach starts with training statistical acoustic models on native speech. These statistical models, typically *Gaussian Mixture Models* (GMMs), are often trained on absolute position of the acoustic features in the feature space. These models are used to produce scores such as *log-likelihood* or *log-posterior probabilities*. The model scores are then used in some combination, often with raw acoustic features, to train a classifier to detect mispronunciations.

These scores, and thus the mispronunciation detection, can be very sensitive to differences that may not be the result of mispronunciation. As a result of speaker variation, productions of vowels that would be accepted by native speakers as correct can prove troublesome as false errors in evaluation systems. The challenge for pronunciation evaluation systems is to pinpoint errors in pronunciation without overwhelming a student with negative feedback, especially when such negative feedback is wrong.

Because our focus is on being selective about which vowels to present to a student, we place high value on specificity—we want to be confident that a vowel that our system indicates is mispronounced is actually mispronounced. This can be challenging in corpora where only small numbers of vowels have been labeled as mispronounced. Chapter 3 showed that 1,450 out of 41,677 vowels were actually identified as mispronounced by the labeling algorithm. When the data are separated into training and testing data this further reduces the amount of available data.

5.2 Related Work

Several approaches to pinpoint mispronunciation detection were detailed in Section 2.3.2; the most relevant are discussed here. Techniques presented in [74, 122, 75] compute scores based on log-posterior probabilities, phone durations, log-likelihoods, and log-likelihood ratios from *Hidden Markov Models* (HMMs), using GMM distributions, and trained on native speech and non-native speech. The *Goodness of Pronunciation* (GOP) computed a single score value for each phone based on the average frame log-posterior probability in a forced alignment. Mispronunciations were determined by setting phone-specific scoring thresholds [239, 241, 240].

Support Vector Machines (SVMs) were used by [235] to detect mispronunciations based on log-likelihood ratios computed from HMMs. Feature vectors for phone productions were computed based on the log-likelihood ratio of the selected phone class to all other possible phone classes. SVMs for each phone were trained to differentiate between phones that were mispronounced and those that were not. A similar approach was used by [245] where the input feature vector was a confidence score computed from HMM scores.

A relativistic method for modeling pronunciation differences between native and non-native speakers was proposed by [156, 218]. This method used the Bhattacharyya Distance [21] to compute the overall structure of speakers' phonetic spaces, thus modeling the phonetic space in a holistic fashion. They used this structure to measure the distortion between Japanese accented English and General American English and found a positive correlation with human assessments of pronunciation quality. However, one of the limitations of their technique was that it was unable to individually classify or assess sounds.

These approaches all share the common characteristic that they assume the correct speech was recognized—a correct, word-level transcription of the speech has been provided. A forced path alignment through one HMM (trained on native or non-native data), or two HMMs (one trained on native data and the other trained on non-native data), produced scores that were later used to train a secondary classifier to detect mispronunciations. Differences chiefly include incorporating model adaptation to improve recognition performance, the number and types of features used for detection, and the type of classifier used to detect mispronunciation. One disadvantage they share is that the HMMs typically model phones as diphones, which typically require more training data.

5.3 Approach

The general principle our approach relies on is a multiway comparison of individual phonetic tokens scored against parallel sets of native acoustic models and non-native acoustic GMMs. It is a multistage process that assumes that a correct transcription is available. We assume that this method will be used as part of a system where the student will complete an entire dialogue or set of dialogues during their use of the CALL system. This assumption provides complete access to the recognition results for an individual user. The data can then be anchored using the procedure detailed in Chapter 4.

There are three major steps that are taken when detecting a mispronunciation. First, the utterance is force-aligned using the word transcription in order to find a canonical labeling and the end points of the phones. Second, two GMM classifiers, one using native acoustic models and the other using non-native acoustic models, are used to classify each segment in the utterance into a phone class. Finally, the classifier results, the scores for the classifiers, derived features, and raw acoustic features are passed to a decision tree classifier to obtain a judgement of pronunciation quality. We will now detail the corpora, segmentation of the utterances, features, and structure of the mispronunciation detector.

5.3.1 Corpora

We utilized two corpora in this research. Native data were provided by the TIMIT corpus [81], consisting of all 6,300 utterances (4,380 male and 1,920 female). Non-native data were provided by the Chinese University Chinese Learners of English (CU-CHLOE) corpus [152], consisting of 36,696 utterances (50 male speakers and 50 female speakers). In this approach, the native data serves only to train the acoustic models used to produce scores for the mispronunciation detector. The non-native data are used to train both the non-native acoustic models and the mispronunciation detector. We take care to separate the non-native data into two distinct sets for this purpose.

5.3.2 Segmentation

To evaluate the vowels in an utterance, we must determine where they begin and end. Since we have assumed that a correct transcription of the utterance has been obtained, a forced-path alignment through the SUMMIT [252] recognizer trained on American English is used to obtain a phonetic level canonical labeling of the speech. This identifies the best segmentation of the sounds in the utterance as determined by native acoustic models as well as the best phonetic labeling according to the word transcription. We will regard each phone, v_t, in this phonetic labeling as the *canonical labeling*. For the purposes of this research, we only investigated detecting mispronunciations in vowels. After the data were segmented, we labeled each of the vowels in the CU-CHLOE (non-native) corpus according to the algorithm detailed in Chapter 3.

5.3.3 Features

A unique aspect of this research is the number and type of features we incorporate into the decision tree classifier. Most mispronunciation detection systems use log-likelihood, log-posterior probabilities, or some variation of scores computed from GMMs as input features into a second classifier for mispronunciation detection.

Our system uses some of these these same scores; however, it is unique in three respects. First, we use multiple phonetic labels to extract an extensive variety of scores from parallel GMMs trained on non-native and native acoustic data. These scores cover a number of permutations for cross-comparison of scores between the GMMs. Second, we incorporate raw acoustic features into the mispronunciation detection system. These raw acoustic features serve minor roles in final determination of mispronunciation. Finally, we exploit these labels to compare phone classes based on model divergences. We will find that these divergence measures are important features for mispronunciation detection.

The next few subsections detail these features and how they are generated. GMMs are used to generate classification labels and a number of scores.

Gaussian Mixture Models

A key part of this technique is the use of GMMs trained on acoustic features to generate classification labels and the scores used in the mispronunciation classifier. There are two sets of GMMs: a native set and a non-native set (represented as θ_n and θ_{nn}). These are used to produce two classification results for a canonical segment, v_t for which a judgement of mispronunciation is desired.

The feature vector, x, used to train the GMMs is the same as described in Chapter 4, Section 4.3.1, plus the log-duration of the phone segment. This results in a 71-dimension feature vector. A *principle components analysis* is performed and used to train a principle components matrix. The GMMs are then trained using the k-means algorithm. The maximum number of clusters is chosen to be 96, since this is a common limit set in the SUMMIT recognizer. Clusters that consist of single points are pruned; thus, some mixtures may contain fewer than 96 Gaussians at the conclusion of training.

After the utterances are segmented during the forced alignment step described above, θ_n is trained from all 6,300 utterances in the TIMIT corpus. The CU-CHLOE corpus is used to train θ_{nn} using 31,099 of the 36,696 utterances. The remaining utterances are used to train the mispronunciation detection classifier. This separation was necessary to avoid contamination of the classifier used for mispronunciation detection. That is, we did not want scores for already seen training data to influence the mispronunciation detector.

The choice of split was primarily based on which utterances had hand phonetic transcriptions. The labeling algorithm defined in Chapter 3 requires a hand phonetic transcription; therefore the data used to train the acoustic models come from utterances that do not have a hand phonetic transcription. The data used to train and test the mispronunciation detector come only from those utterances that have hand phonetic transcriptions. We did not need to split the native data, as the sole purpose of the TIMIT corpus will be to train native acoustic GMMs, θ_n, to generate model scores.

Each segment, represented by the feature vector x, in an utterance is classified by both θ_{nn} and θ_n. This produces two classification results, v_{nn} and v_n, the decisions of the non-native classifier and the native classifier, respectively. Mathematically, this decision is represented as the phone class that produces the max log-posterior probability:

$$v_m = \arg\max_{v \in V} \lg p(v|x;\theta_m) \tag{5.1}$$

where V is the phonetic inventory of the classifier, $m \in \{n, nn\}$. Thus, v_{nn} is the phone class that produces the maximum posterior probability in the non-native models, θ_{nn}. The actual posterior probability is defined as:

$$p(v|x;\theta_m) = \frac{p(x|v;\theta_m)p(v;\theta_m)}{p(x;\theta_m)} \tag{5.2}$$

where $p(v;\theta_m)$ is the prior probability of the phone class v. The likelihood portion of the equation, $p(x|v;\theta_m)$ is defined as:

$$p(x|v;\theta_m) = \sum_{k \in K_{v,\theta_m}} w_k \frac{1}{(2\pi)^{d/2}|\Sigma_k|^2} e^{-\frac{1}{2}(x-\mu_k)\Sigma_k^{-1}(x-\mu_k)} \tag{5.3}$$

where K_{v,θ_m} is the set of Gaussian mixtures for phone class v in model θ_m, w_k is the weight assigned to the k^{th} Gaussian, d is the dimensionality of the feature vector (71-dimensions), μ_k is the mean of the k^{th} Gaussian, and Σ_k is the diagonal covariance matrix of the k^{th} Gaussian. The prior probability, $p(x)$, is estimated by summing over the classes as in, $p(x;\theta_m) = \sum_{v \in V} p(x|v;\theta_m)p(v;\theta_m)$.

After the classification is performed on segment feature, x, there are three labels per segment: the canonical labeling (v_t), the label assigned by the θ_{nn} models (v_{nn}), and the label assigned by the θ_n models (v_n). Using these labels in conjunction with θ_{nn} and θ_n, we can derive scores to be used in the mispronunciation detector.

Posterior Probabilities

A common score used by pronunciation scoring algorithms is the posterior probability of a phone class being produced under a given set of models. Because of the classification step, the posterior probabilities for v_{nn} and v_n are already defined for the non-native and native models, θ_{nn} and θ_n. We can also ask what the posterior probability of v_n was under the non-native models, θ_{nn}. In other words, given what the native models, θ_n, chose as the correct classification for a feature vector, x, what was the score of v_n in the non-native models, θ_{nn}? This allows us to define six posterior probabilities to be used in the mispronunciation classifier (see Table 5.1).

$p(v_{nn}\|x;\theta_{nn})$	$p(v_n\|x;\theta_{nn})$	$p(v_t\|x;\theta_{nn})$
$p(v_{nn}\|x;\theta_n)$	$p(v_n\|x;\theta_n)$	$p(v_t\|x;\theta_n)$

Table 5.1: Posterior probabilities used as features.

In Section 5.3.2, we defined v_t as the phone class that was chosen as the canonical labeling during forced alignment. So $p(v_t|x;\theta_n)$ is the posterior probability of the phone class v_t scored by the native models, θ_n. It should be noted that in the actual classifier, the log-posterior probabilities are used, but for the sake of simplifying notation, we omit the log in the equations.

Posterior Probability Ratios

Another value that has been used in previous literature is the ratio of the posterior probability of the non-native class to other values in the non-native models and native models —mathematically, $\frac{p(v_{nn}|x;\theta_{nn})}{p(v_n|x;\theta_{nn})}$ (in the actual experiments, the operation is a subtraction as it is being performed in log-space). Intuitively, this is quantifying how much more the non-native models (θ_{nn}) prefer choosing v_{nn} over v_n during the classification step.

We expand on this and measure a number of other ratios. Specifically, we take the posterior probability of the non-native class under the non-native models to all other posterior probabilities. This provides information on how much more the non-native models preferred v_{nn} over v_t and v_n in the non-native models (θ_{nn}) and the native models (θ_n). We can compute the same ratios for the reverse case—that is, how much more the native models preferred v_n over the other cases. These ratios are summarized in Table 5.2.

$\frac{p(v_{nn}|x;\theta_{nn})}{p(v_n|x;\theta_{nn})}$	$\frac{p(v_{nn}|x;\theta_{nn})}{p(v_t|x;\theta_{nn})}$	$\frac{p(v_{nn}|x;\theta_{nn})}{p(v_{nn}|x;\theta_n)}$	$\frac{p(v_{nn}|x;\theta_{nn})}{p(v_n|x;\theta_n)}$	$\frac{p(v_{nn}|x;\theta_{nn})}{p(v_t|x;\theta_n)}$
$\frac{p(v_n|x;\theta_n)}{p(v_{nn}|x;\theta_n)}$	$\frac{p(v_t|x;\theta_n)}{p(v_{nn}|x;\theta_n)}$	$\frac{p(v_{nn}|x;\theta_n)}{p(v_{nn}|x;\theta_{nn})}$	$\frac{p(v_n|x;\theta_n)}{p(v_{nn}|x;\theta_{nn})}$	$\frac{p(v_t|x;\theta_n)}{p(v_{nn}|x;\theta_{nn})}$

Table 5.2: Posterior probability ratios used as features.

Divergence Measures

Minematsu et. al's [156, 218] research relied on measurements of statistical divergence using Bhattacharyya Distance to construct their pronunciation structure. As noted earlier, while this method allowed for holistic pronunciation assessment, it precluded mispronunciation detection at an individual phone level. We can still make use of this statistical measurement. We established in Chapter 4 that Bhattacharyya Distance is correlated with the proportion of vowels labeled mispronounced by the labeling algorithm. We will exploit this to generate additional features for our mispronunciation detector.

The classification step provided three, possibly different, labels for each segment classified—v_{nn}, v_n, and v_t. We can think of these labels as selecting statistical distributions in both the non-native models and native models, θ_{nn} or θ_n, respectively. Thus, one can imagine that v_t, the phonetic label assigned due to the forced alignment, selects two distributions: one in θ_n and one in θ_{nn}. The statistical distribution for v_t in θ_n is denoted as $\omega_{t,n}$, and the statistical distribution for v_t in θ_{nn} is denoted as $\omega_{t,nn}$. Between distributions in θ_{nn} and in θ_n there are 9 such possible distances.

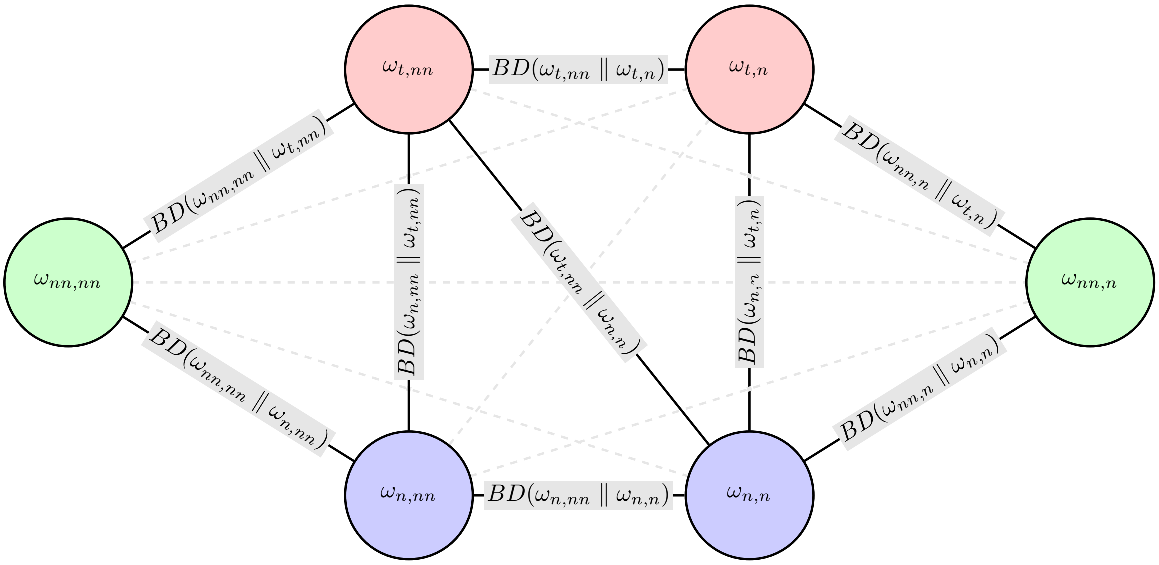

$\omega_{t,nn}$
$BD(\omega_{t,nn} \parallel \omega_{t,n})$
$\omega_{t,n}$
$BD(\omega_{nn,nn} \parallel \omega_{t,nn})$
$BD(\omega_{n,nn} \parallel \omega_{t,nn})$
$BD(\omega_{t,nn} \parallel \omega_{n,n})$
$BD(\omega_{n,n} \parallel \omega_{t,n})$
$BD(\omega_{nn,n} \parallel \omega_{t,n})$
$\omega_{nn,nn}$
$\omega_{nn,n}$
$BD(\omega_{nn,nn} \parallel \omega_{n,nn})$
$BD(\omega_{nn,n} \parallel \omega_{n,n})$
$\omega_{n,nn}$
$BD(\omega_{n,nn} \parallel \omega_{n,n})$
$\omega_{n,n}$

Figure 5-1: All the potential Bhattacharyya Distance measurements. For example, $BD(\omega_{t,nn} \parallel \omega_{t,n})$ is the Bhattacharyya Distance between the distribution of the canonical phone label in θ_{nn} and the distribution of the canonical phone label in θ_n.

Computing the distances *within* θ_{nn} and θ_n might also yield useful information. For example, it is possible that v_{nn}, v_n, and v_t are all different labels. That is, the canonical labeling, the native classifier, and the non-native classifier all disagreed on what the sound for that segment actually was. In the case where all three labels are different, it would be useful to measure how different those distributions are within the respective model sets—when the distributions are close together, one might expect that a judgement of mispronounced would be less likely. There are 6 such distances that can be computed, 3 in θ_{nn} and 3 in θ_n. All the potential distances are depicted in Figure 5-1. The Bhattacharyya-Distance is only well-defined for single multivariate Gaussians; in order to adapt the measure for Gaussian Mixture Models, we merged the Gaussian Mixtures into a single Gaussian prior to computing the Bhattacharyya Distance between two distributions.

In addition to the Bhattacharyya Distance, another divergence measure is the *Kullback-*

Leibler (KL) divergence [127]. This is a non-symmetric measure of divergence between two discrete probability distributions, P and Q:

$$KL(P \parallel Q) = \sum_i P(i) \lg \frac{P(i)}{Q(i)} \tag{5.4}$$

where i is set to the mean of each mixture in the GMM. The KL divergence is non-symmetric, that is $KL(P \parallel Q) \neq KL(Q \parallel P)$, therefore, when computing the divergence measurement for the feature, we compute the symmetric version of the divergence:

$$KD(P \parallel Q) = KL(P \parallel Q) + KL(Q \parallel P) \tag{5.5}$$

Similarly to the Bhattacharyya Distances, we compute parallel versions of the KL divergences. The rationale for including KL-divergence measures in addition to Bhattacharyya Distance measures has more to do with the differences in implementation in our system. The KL-divergence between two mixture distributions is computed by using the means of the mixtures as sample points. In contrast, when computing the Bhattacharyya Distance, the mixtures are merged into a single Gaussian prior to computing the distance. Using both types of measures allows the system to consider two slightly different models of divergence.

Divergence Delta Distances

Another potential source of information are the *changes* that occur in the divergences between the distributions as a result of anchoring. In Chapter 4, Table 4.3 showed that most of the vowel classes moved closer together after they had been anchored. For example, the native and non-native distributions of the vowel /ɑ/ [aa] moved closer together from a $BD = 227.78$ to a $BD = 194.53$ or a $\Delta BD = -33.25$.

To exploit this observation, we construct 9 delta measurements that measure the amount of change in the Bhattacharyya Distance measurements from non-native (θ_{nn}) models to native (θ_n) models. Recall that we are denoting the statistical distributions of a phone label, such as v_t, under a set of models, such as θ_n as $\omega_{t,n}$. To denote the *unanchored* distributions, we will use $\omega'_{t,n}$. The delta features measure the change in the Bhattacharyya Distance before and after anchoring the features. Mathematically, this is:

$$\Delta BD(\omega_{t,nn}, \omega_{n,n}) = BD(\omega_{t,nn} \parallel \omega_{n,n}) - BD(\omega'_{t,nn} \parallel \omega'_{n,n}) \tag{5.6}$$

Delta measures for KL-divergence are similarly defined.

Acoustic Features

The final set of features are simply the raw acoustic features from the middle third of the segment. For unanchored versions of the phones, this would correspond to 14 MFCCs. Analogously, this would correspond to 14 Anchored MFCCs in the anchored version of the features.

Feature Summary

We have described an extensive set of features that will be used in the mispronunciation detector. Some of these features are categorical (the phonetic labels of the segments), some of the features are provided directly by the GMM classifiers (for example, the posterior probability scores), other features were derived based on classifier results, and some features represent raw acoustic measurements. Altogether, there are 81 features. We shall later see that some feature prove more useful than others for mispronunciation detection. Table 5.3 summarizes all of these features. The next section details the structure and training of the classifier.

Type	Features			# Dims
Phone Label	v_{nn}	v_n	v_t	3
Posterior Probability	$p(v_{nn}\|x;\theta_{nn})$	$p(v_n\|x;\theta_{nn})$	$p(v_t\|x;\theta_{nn})$	6
	$p(v_{nn}\|x;\theta_n)$	$p(v_n\|x;\theta_n)$	$p(v_t\|x;\theta_n)$	
Posterior Probability Ratio	$\frac{p(v_{nn}\|x;\theta_{nn})}{p(v_n\|x;\theta_{nn})}$	$\frac{p(v_n\|x;\theta_n)}{p(v_{nn}\|x;\theta_{nn})}$		10
	$\frac{p(v_{nn}\|x;\theta_{nn})}{p(v_t\|x;\theta_{nn})}$	$\frac{p(v_n\|x;\theta_n)}{p(v_n\|x;\theta_{nn})}$		
	$\frac{p(v_{nn}\|x;\theta_{nn})}{p(v_{nn}\|x;\theta_n)}$	$\frac{p(v_n\|x;\theta_n)}{p(v_t\|x;\theta_{nn})}$		
	$\frac{p(v_{nn}\|x;\theta_{nn})}{p(v_n\|x;\theta_n)}$	$\frac{p(v_n\|x;\theta_n)}{p(v_{nn}\|x;\theta_n)}$		
	$\frac{p(v_{nn}\|x;\theta_{nn})}{p(v_t\|x;\theta_n)}$	$\frac{p(v_n\|x;\theta_n)}{p(v_t\|x;\theta_n)}$		
Bhattacharyya Distance	$BD(\omega_{nn,nn} \parallel \omega_{n,nn})$	$BD(\omega_{nn,n} \parallel \omega_{n,n})$	$BD(\omega_{t,nn} \parallel \omega_{nn,n})$	15
	$BD(\omega_{nn,nn} \parallel \omega_{t,nn})$	$BD(\omega_{nn,n} \parallel \omega_{t,n})$	$BD(\omega_{t,nn} \parallel \omega_{n,n})$	
	$BD(\omega_{n,nn} \parallel \omega_{t,nn})$	$BD(\omega_{n,n} \parallel \omega_{t,n})$	$BD(\omega_{t,nn} \parallel \omega_{t,n})$	
	$BD(\omega_{nn,nn} \parallel \omega_{nn,n})$	$BD(\omega_{n,nn} \parallel \omega_{nn,n})$	$BD(\omega_{nn,nn} \parallel \omega_{t,n})$	
	$BD(\omega_{nn,nn} \parallel \omega_{n,n})$	$BD(\omega_{n,nn} \parallel \omega_{n,n})$	$BD(\omega_{n,nn} \parallel \omega_{t,n})$	
Δ Bhattacharyya Distance	$\Delta BD(\omega_{nn,nn}, \omega_{nn,n})$	$\Delta BD(\omega_{n,nn}, \omega_{nn,n})$	$\Delta BD(\omega_{t,nn}, \omega_{nn,n})$	9
	$\Delta BD(\omega_{nn,nn}, \omega_{n,n})$	$\Delta BD(\omega_{n,nn}, \omega_{n,n})$	$\Delta BD(\omega_{t,nn}, \omega_{n,n})$	
	$\Delta BD(\omega_{nn,nn}, \omega_{t,n})$	$\Delta BD(\omega_{n,nn}, \omega_{t,n})$	$\Delta BD(\omega_{t,nn}, \omega_{t,n})$	
KL Divergence	$KD(\omega_{nn,nn} \parallel \omega_{n,nn})$	$KD(\omega_{nn,n} \parallel \omega_{n,n})$	$KD(\omega_{t,nn} \parallel \omega_{nn,n})$	15
	$KD(\omega_{nn,nn} \parallel \omega_{t,nn})$	$KD(\omega_{nn,n} \parallel \omega_{t,n})$	$KD(\omega_{t,nn} \parallel \omega_{n,n})$	
	$KD(\omega_{n,nn} \parallel \omega_{t,nn})$	$KD(\omega_{n,n} \parallel \omega_{t,n})$	$KD(\omega_{t,nn} \parallel \omega_{t,n})$	
	$KD(\omega_{nn,nn} \parallel \omega_{nn,n})$	$KD(\omega_{n,nn} \parallel \omega_{nn,n})$	$KD(\omega_{nn,nn} \parallel \omega_{t,n})$	
	$KD(\omega_{nn,nn} \parallel \omega_{n,n})$	$KD(\omega_{n,nn} \parallel \omega_{n,n})$	$KD(\omega_{n,nn} \parallel \omega_{t,n})$	
Δ KL Divergence	$\Delta KD(\omega_{nn,nn}, \omega_{nn,n})$	$\Delta KD(\omega_{n,nn}, \omega_{nn,n})$	$\Delta KD(\omega_{t,nn}, \omega_{nn,n})$	9
	$\Delta KD(\omega_{nn,nn}, \omega_{n,n})$	$\Delta KD(\omega_{n,nn}, \omega_{n,n})$	$\Delta KD(\omega_{t,nn}, \omega_{n,n})$	
	$\Delta KD(\omega_{nn,nn}, \omega_{t,n})$	$\Delta KD(\omega_{n,nn}, \omega_{t,n})$	$\Delta KD(\omega_{t,nn}, \omega_{t,n})$	
Acoustic Features	14 MFCCs or Anchored MFCCs			14
			Total number of features	81

Table 5.3: A summary of the features used in the mispronunciation detector.

5.3.4 Decision Tree Classifier

The mispronunciation detector is a decision tree classifier trained using the c4.5 algorithm [192] and incorporating a weighted cost matrix. It was implemented using the WEKA Datamining Toolkit from the University of Waikato [91]. The choice of a decision tree classifier over other types of classifier was made for a few reasons. First, decision trees produce rules that can be reasoned about by a human wishing to understand how the classifier arrives at decisions for a given datum. Second, decision trees have relatively few parameters to adjust before acceptable results can be obtained. It can be contrasted with a *Support Vector Machine* (SVM), where the kernel type alters the number and type of parameters that must be optimized, or an *Artificial Neural Network* (ANN) [201], where the structure of the network has significant impacts on the classification results. Third, pruning methods can be automatically employed to reduce the size of the tree and remove rules which do not split the data well—effectively identifying which features are important or unimportant to classification.

The features detailed in the previous section were all combined into a single feature vector and paired with a pronunciation label provided from the Mechanical Turkers. The numeric features are normalized to a -1.0 to 1.0 range—the decision to normalize to this range instead of 0 to 1 was made in order to preserve sign information. The labels fall into three categories: *good* (no Turkers felt the vowel was mispronounced), *ugly* (at least one Turker felt the vowel was mispronounced), or *mispronounced* (all the Turkers felt the vowel was mispronounced). The cost matrix is used to reweight the mistake of classifying a phone labeled *good* as *mispronounced*, in order to bias it toward high precision in the classification assignments. Our results show the performance of the classifier as this parameter is adjusted to have cost values of 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, and 4.0. We expect the precision to rise as the cost of misclassifying a *good* vowel as *mispronounced* is raised.

Training and Testing

The training and test data come from the remainder of the CU-CHLOE corpus that was not used to train the acoustic GMM models. This set is comprised of 5,597 utterances and contains 41,677 vowels. Of these, 37,691 were labeled *good*, 2,536 labeled *ugly*, and 1,450 labeled *mispronounced*. For the purposes of this research, we focused only on separating *good* from *mispronounced*. We believe that the classifier could choose *good*, *ugly*, or *mispronounced* for a vowel marked as *ugly* and it would not be necessarily an incorrect judgement. Therefore, we removed those instances of vowels to produce a more pristine dataset. After removal, there were 39,141 vowels remaining, 37,691 *good* (96.3%) and 1,450 *mispronounced* (3.7%).

We conducted two tests using two methodologies. To gain an understanding of the average performance of the detectors, we performed a 10-fold cross-validation test. This will give us an understanding of how the features perform in average case scenarios. To perform a deeper analysis that examines the performance on a vowel class level, we fixed the training and test sets.

The data were split so that 80% of the vowels (31,313 total, 30,163 good and 1,150 mispronounced) were used to train the classifier, and the remaining 20% (7,828 total, 7,528 good, 300 mispronounced) were used as a test set. We elected to split the data as a whole which resulted in speakers appearing in both the training and test data sets. This decision was made because, on average, each speaker in the corpus mispronounced only 14 vowels and exhibited different mispronunciation profiles—some speakers mispronounced certain vowels more frequently than other speakers. This method of splitting the data is imperfect, but maintains relative distributions of *good* vs *mispronounced* vowels in the training and test datasets and ensures adequate coverage for each vowel.

Models

We wish to establish that the anchoring procedure outlined in Chapter 4 improves performance in detecting mispronunciations. To this end, we experimented with two different types of GMMs to produce the feature vectors required for the decision tree classifier. The first GMM was trained on unanchored vowels. The second GMM was trained on anchored vowels. The feature vectors from these two types of GMMs were used to train and test the decision tree classifiers. The below results are a comparison between the features from these two types of GMMs.

5.4 Results

A decision tree can be evaluated both in terms of its performance at the classification task, and the specific decisions it makes in determining the class of a given instance. The next two sections analyze the results of the decision tree classifier, first, in terms of the performance at the actual task of detecting mispronunciations, and second in terms of the size of the tree and the features selected for the decision nodes.

5.4.1 Performance

In this research, we were concerned only with accurately identifying mispronunciations; therefore, we were not interested in identifying vowels that would be considered *good* pronunciations. We are also not interested in identifying every single mispronunciation, only that the classifier is accurate when it identifies a vowel as mispronounced. Therefore, we are interested in *high-precision*, but not necessarily *high-recall*, rates for vowels marked as *mispronounced*.

The standard way to define precision is:

$$Pr = \frac{TM}{TM + FM} \tag{5.7}$$

where TM is the number of mispronunciations identified by the classifier that were actually mispronounced and FM is the number of mispronunciations that were actually not mispronunciations. Recall is defined as:

$$Re = \frac{TM}{TM + FG} \tag{5.8}$$

where FG are those vowels misclassified as *good* pronunciations when they are actually mispronunciations. We will see that these measures give poor assessments of classifier performance.

Table 5.4 summarizes the precision and recall rates for detecting mispronunciations using a 10-fold cross-validation testing strategy. The left column shows the performance of the decision tree that used GMMs trained on Mel-Frequency Cepstral Coefficients (MFCCs)

Cost	Feature Source	
	MFCC	$\overline{C}$-anchor
1.0	0.65 (0.30)	0.65 (0.33)
1.5	0.69 (0.26)	0.77 (0.26)
2.0	0.71 (0.21)	0.77 (0.27)
2.5	0.79 (0.16)	0.79 (0.22)
3.0	0.86 (0.13)	0.84 (0.18)
3.5	0.87 (0.13)	0.88 (0.13)
4.0	0.86 (0.13)	0.89 (0.13)

Table 5.4: Precision and recall rates computed using cross-validated results under default WEKA analysis for the *mispronounced* annotation class. Precision rate is the first number, with recall rate represented in parentheses following precision. The feature source refers to the feature type the GMMs were trained on.

as the feature source. The right column shows the performance of the tree that used GMMs trained on $\overline{C}$-anchors as the feature source. As can be seen from the results, the anchored version of the features outperforms the unanchored version of the features at almost every cost level except for 2.5 and 3.0. These results, however, only give a partial picture of the performance of the detectors. We have no idea what the performance breakdown among the different vowel classes is. We do not know, for example, how precise the system is at identifying mispronunciations of the vowel /ɑ^{y}/ [ay].

Cost	Feature Source	
	MFCC	$\overline{C}$-anchor
1.0	0.68 (0.31)	0.64 (0.31)
1.5	0.77 (0.26)	0.78 (0.27)
2.0	0.85 (0.19)	0.74 (0.26)
2.5	1.0 (0.14)	0.79 (0.24)
3.0	1.0 (0.14)	0.89 (0.15)
3.5	1.0 (0.14)	0.90 (0.15)
4.0	1.0 (0.14)	0.85 (0.15)

Table 5.5: Precision and recall rates computed using the default WEKA analysis for the *mispronounced* annotation class. Precision rate is the first number, with recall rate represented in parentheses following precision. The feature source refers to the feature type the GMMs were trained on.

In order to examine the results at a detailed level, we decided to fix the training and test sets to perform a deeper analysis. Table 5.5 summarizes the precision and recall rates for detecting mispronunciations. The left column shows the performance of the decision tree that used GMMs trained on Mel-Frequency Cepstral Coefficients (MFCCs) as the feature source. The right column shows the performance of the tree that used GMMs trained on $\overline{C}$-anchors as the feature source. Starting at a cost of 2.5, it appears that the MFCC based classifier achieves perfect performance identifying mispronunciations.

This result is misleading, however, because it turns out that the MFCC based classifier is good at detecting mispronunciations for only a single vowel class: the vowel /ɑ^{y}/ [ay]. For all other vowel classes, it identifies all instances of the vowels as *good*. The MFCC classifier is able to attain a *per vowel* precision of 1.0 for the /ɑ^{y}/ [ay]. Therefore, when WEKA computed the precision, it reported a precision of 1.0 for the entire classifier. This is clearly not a good mispronunciation detector if it is only able to detect mispronunciations for a single vowel class.

Cost	Feature Source	
	MFCC	$\overline{C}$-anchor
1.0	0.86	0.93
1.5	0.79	0.93
2.0	0.43	0.93
2.5	0.07	0.93
3.0	0.07	0.07
3.5	0.07	0.07
4.0	0.07	0.07

Table 5.6: Diversity of Recall for classification results using default WEKA analysis.

We can gain a more accurate assessment of decision tree performance by looking at the number of vowel classes for which each tree is capable of detecting mispronunciations. As a means of analyzing this, we will define a measurement called the *diversity of recall* (DOR) measurement. This measures the proportion of times the recall for each *vowel class* exceeded 0.0. Thus, if the results for a classifier have 12 out of 14 vowels with non-zero recalls, then the DOR is $12/14$=0.86. This gives an additional assessment of how flexible the mispronunciation detector is at detecting mispronunciation across all the vowel classes. This measurement is shown in Table 5.6. This table shows that at every cost until 3.0, the

$\overline{C}$-anchor features are able to identify a more diverse array of mispronunciations. Further, as the cost increases, MFCC features identify mispronunciations in a smaller and smaller fraction of the vowels.

Cost	Feature Source	
	MFCC	$\overline{C}$-anchor
1.0	0.59 (0.31)	0.59 (0.31)
1.5	0.55 (0.19)	0.64 (0.25)
2.0	0.37 (0.11)	0.64 (0.25)
2.5	0.07 (0.06)	0.67 (0.22)
3.0	0.07 (0.06)	0.10 (0.08)
3.5	0.07 (0.06)	0.06 (0.06)
4.0	0.07 (0.06)	0.06 (0.06)

Table 5.7: Aggregated precision and recall rates for the *mispronounced* annotation class. Precision rate is the first number, recall rate is the second number (in parentheses). The feature source refers to the feature type the GMMs were trained on.

This observation leads to a slightly different method for assessing performance. Instead of reporting precision and recall for the overall number of mispronunciations, we will report aggregate numbers for the precisions and recalls over the different vowel classes. These aggregate numbers are simply the arithmetic means of the precisions and recalls over all of the vowel classes. These numbers are reported in Table 5.7. These precision and recall values give a more accurate assessment of the classifier for identifying pronunciations. As can be seen from these results, the $\overline{C}$-anchor feature source outperforms the MFCC feature source both in precision *and* recall.

In fact, immediately upon increasing the cost of misdiagnosing a *good* pronunciation, the performance of the MFCC feature source begins to decline, and virtually collapses in performance when the cost reaches 2.5. In contrast, the performance of the $\overline{C}$-anchor feature source improves in performance (as measured by precision), achieving an aggregated precision of 0.67 and a recall of 0.22. It improves until a cost of 3.0, at which point it too collapses.

To better understand the aggregated precision and recall values, Table 5.8 breaks down the precision and recall values to the individual vowel level. As can be seen from the results, the $\overline{C}$-anchor feature source outperforms the MFCC feature source for all vowels except for

Vowel	Feature Source	
	MFCC	$\overline{C}$-anchor
/ɑ/ [aa]	0.0 (0.0)	1.0 (0.07)
/æ/ [ae]	0.0 (0.0)	0.5 (0.17)
/ʌ/ [ah]	0.0 (0.0)	1.0 (0.2)
/ɔ/ [ao]	0.0 (0.0)	0.5 (0.08)
/ɑ^{w}/ [aw]	0.0 (0.0)	0.8 (0.57)
/ə/ [ax]	0.0 (0.0)	0.0 (0.0)
/ɑ^{y}/ [ay]	1.0 (0.84)	0.88 (0.88)
/ɛ/ [eh]	0.0 (0.0)	0.6 (0.38)
/ɚ/ [er]	0.0 (0.0)	0.5 (0.04)
/e/ [ey]	0.0 (0.0)	1.0 (0.17)
/i/ [iy]	0.0 (0.0)	1.0 (0.07)
/o/ [ow]	0.0 (0.0)	0.67 (0.11)
/ɔ^{y}/ [oy]	0.0 (0.0)	0.5 (0.2)
/u/ [uw]	0.0 (0.0)	0.5 (0.12)

Table 5.8: Precision and recall rates for individual phone classes when cost is 2.5.

/ɑ^{y}/ [ay]. It achieves precisions of between 0.5 and 1.0 for all of the vowel classes. Various other research efforts report comparable results. Mispronunciations of the vowel /ɑ^{y}/ [ay] are easy for the classifier to detect. This also verifies the analysis performed earlier that showed the MFCC feature source decision tree is less capable in identifying mispronunciations from a diverse set of vowels. This lends strong support to anchoring the vowels for pronunciation assessment.

One vowel that should be pointed out in particular is /ɚ/ [er]. This vowel had the highest proportion of instances labeled as *ugly* or *mispronounced* in the results from Chapter 3. This vowel was also very prominently in a different relative location after the anchoring procedure from Chapter 4, and we noted that this may have been due to the fact that Hong Kong students were instructed in British English as opposed to American English. The Turkers seemed to be split on whether or not a British production of this phone constituted a mispronunciation, as evidenced by the rate of *ugly* annotations being twice the rate of *mispronounced* annotations. This could also account for the relatively low precision seen for this particular vowel.

5.4.2 Decision Tree Rules

The final analysis is to examine the actual decision trees produced by the two feature sources. We will focus only on the decision trees that resulted from a cost of 2.5. We will show that, while the decision tree for MFCC source features is smaller than the decision tree for $\overline{C}$-anchor source features, the difference in size reflects mostly finer grained distinctions in the terminal leaves of the $\overline{C}$-anchor based decision tree. We will also see that the divergence measurements, which were a unique feature for this mispronunciation detector, are important for the $\overline{C}$-anchor based decision tree.

The first point of comparison is the size of the trees. The MFCC feature source trains a decision tree that has 6 terminal leaves and 11 total nodes. The $\overline{C}$-anchor feature source results in a substantially larger tree with 186 terminal leaves and 221 total nodes. This difference is somewhat misleading. The entire decision tree for the MFCC feature source is presented below:

```
lpr_n_n_n_nn <= -0.374268: good (29619.49/307.12)
lpr_n_n_n_nn > -0.374268
|   div_nn_t_nn_n <= -0.626821: good (1219.04/177.65)
|   div_nn_t_nn_n > -0.626821
|   |   div_t_t_nn_n <= -0.066919: good (352.4/92.31)
|   |   div_t_t_nn_n > -0.066919
|   |   |   div_t_nn_nn_nn <= -0.888398
|   |   |   |   lpr_nn_t_nn_n <= -0.402055: mispronounced (95.36/7.11)
|   |   |   |   lpr_nn_t_nn_n > -0.402055: good (4.06)
|   |   |   div_t_nn_nn_nn > -0.888398: good (22.64/2.32)
```

As can be seen, the only features utilized are the $lpr\_n\_n\_n\_nn$ (in our mathematical notation this corresponds to $\frac{p(v_n|x;\theta_n)}{p(v_n|x;\theta_{nn})}$), $div\_nn\_t\_nn\_n$ ($BD(\omega_{nn,nn} \parallel \omega_{t,n})$), $div\_t\_t\_nn\_n$ ($BD(\omega_{t,nn} \parallel \omega_{t,n})$), $div\_t\_nn\_nn\_nn$ ($BD(\omega_{t,nn} \parallel \omega_{nn,nn})$), and $lpr\_nn\_t\_nn\_n$ ($\frac{p(v_{nn}|x;\theta_{nn})}{p(v_t|x;\theta_n)}$). This tree shows that the classifier is relying entirely on posterior probability ratios and Bhattacharyya divergence measurements.

The c4.5 algorithm selects features based on how well they split the data. This is measured by information gain, which is measured by KL-divergence. This means that features selected first, e.g. the first decision, could be interpreted as those features that are important for detecting mispronunciations.

This decision tree uses the feature $\frac{p(v_n|x;\theta_n)}{p(v_n|x;\theta_{nn})}$ for the first decision, and corresponds to a situation where the log-posterior probability of $p(v_n|x;\theta_n)$, or the score of the native phone class v_n under the native models is less than the score of the same phone class under θ_{nn}. This is seemingly counter-intuitive, because it essentially says that when the non-native models assign a stronger score than the native models, it is more likely that the phone was well-pronounced. On the other hand, when the native models and the non-native models have more comparable scores (i.e. the ratio increases), there is an entire decision subtree activated to make a final classification. This could be explained by noting that the ratio could be increased by either the native score increasing or the non-native score decreasing, and that the latter case indicates serious pronunciation problems that place the feature instance on the out bounds of the class distribution for the non-native models. The remainder of the tree is decided solely on Bhattacharyya distance measurements.

In contrast, the tree produced from the $\overline{C}$-anchor feature source is larger and uses a more diverse array of features. Due to its size, the decision tree for the $\overline{C}$-anchor feature is included as Appendix C. The first item to note is that a large portion of the leaves and internal nodes of the tree are decisions on the non-native (nn_result in the tree) or native (n_result) label assigned during classification. Of the 221 nodes, 154 are decisions about the label assignment. The vast majority of these decisions (138) are leaf decisions, where the label assignment determines the final judgement of pronunciation quality. Considered as a proportion of the tree, nodes involving the label assignment are $154/221 * 100 = 69.7$% of the total decision tree.

As an example, consider the classification assigned by the following rule chain (excerpted from the tree):

```
t_score_nn <= 0.638435
|   div_t_t_delta > 0.088494
|   |   t_score_nn <= 0.434373
|   |   |   div_t_t_delta > 0.219971
|   |   |   |   div_t_n_n_n > -0.974855
|   |   |   |   |   n_result = _
|   |   |   |   |   |   div_t_n_nn_n > -0.807392
|   |   |   |   |   |   |   div_t_nn_nn_nn <= -0.875521
|   |   |   |   |   |   |   |   nn_result = aa: mispronounced (0.58)
|   |   |   |   |   |   |   |   nn_result = ae: good (0.0)
|   |   |   |   |   |   |   div_t_nn_nn_nn > -0.875521: mispronounced (6.39)
```

In this example, when n_result is assigned the silence label ``_" by $\theta_n$ and $nn\_result$ is assigned the label ``aa" by θ_{nn}, the decision tree determines that this particular instance of a vowel (assuming the tree above these nodes had all been activated) was mispronounced. Note that, although every instance presented to the classifier will be some sort of a vowel, the classification results produce a number of labels that are not necessarily vowels. The interpretation of this is that, if the instance of the vowel produces a blank (``_") label in the native models and a non-blank response in the non-native models, that, depending on a threshold on the value of nn_result, the vowel could be considered good or mispronounced.

Another interesting aspect to the decision tree produced from the $\overline{C}$-anchors is the extensive use of the divergence measurements and their deltas. Altogether, KL divergence and Bhattacharyya distance features are used in 28 nodes of the tree and the delta divergence measurements are used in 18 nodes of the tree—or 46 total nodes in the tree. When the leaves involving n_result or nn_result are factored out, this constitutes $46/(221 - 154) * 100 = 46/67 * 100 = 68.7\%$ of the remaining decisions in the tree.

The unique set of divergence measurements used includes $div\_nn\_n\_nn\_nn$ $(BD(\omega_{nn,nn} \parallel \omega_{n,nn}))$, $div\_t\_n\_n\_n$ $(BD(\omega_{n,n} \parallel \omega_{t,n}))$, $div\_t\_n\_nn\_n$ $(BD(\omega_{t,nn} \parallel \omega_{n,n}))$, $div\_t\_nn\_n\_n$ $(BD(\omega_{nn,n} \parallel \omega_{t,n}))$, $div\_t\_nn\_nn\_nn$ $(BD(\omega_{nn,nn} \parallel \omega_{t,nn}))$, $kldiv\_nn\_t\_nn\_n$ $(KD(\omega_{nn,nn} \parallel \omega_{t,n}))$, $kldiv\_t\_n\_n\_n$ $(KD(\omega_{n,n} \parallel \omega_{t,n}))$, $kldiv\_t\_n\_nn\_n$ $(KD(\omega_{t,nn} \parallel \omega_{n,n}))$, $kldiv\_t\_nn\_n\_n$ $(KD(\omega_{nn,n} \parallel \omega_{t,n}))$, and $kldiv\_t\_nn\_nn\_n$ $(KD(\omega_{t,nn} \parallel \omega_{nn,n}))$. What is interesting about this set of features is that, aside from one case, all of the features are measuring the divergence of the v_t canonical label distribution under either θ_n or θ_{nn} to either v_n or v_{nn} in both θ_n and θ_{nn}. This indicates that, when the v_t is different from v_n or v_{nn} (i.e. the native and non-native classifiers disagreed with the phonetic label to assign a given segment), the divergence measurements play a significant role in determining whether or not a vowel would be labeled as mispronounced.

It is important to note that the first decision made by this tree is on the posterior probability (t_score_nn or $p(v_t|x;\theta_{nn})$) of the canonical label, v_t. It is only when this score is below a certain threshold that the rest of the decision tree is activated. When the score is above this threshold, the decision tree automatically assigned a classification of *good* to the vowel under consideration.

Finally, a difference between the MFCC feature decision tree and the $\overline{C}$-anchor feature decision tree is that the $\overline{C}$-anchor version only makes use of the log-posterior probability feature in four of the leaf nodes. This indicates that it plays a far less important role in the decision tree than it does in previous literature. The divergence measures and associated classifier results seem to be more important for determining pronunciation quality.

5.5 Summary

This chapter introduced a novel method for pronunciation evaluation. A set of Gaussian Mixture Models were utilized to provide statistical scores for vowels presented to the classifier. Using these scores and classification results, it established a number of unique features, a portion of which were derived from the Bhattacharyya Distance and Kullback-Leibler divergence measurements for statistical distributions.

These features were used to train and test a decision tree classifier to identify mispronounced vowels. The classification experiments compared the performance of features produced from GMMs trained using standard MFCC acoustic features with the performance of features produced from GMMs trained using $\overline{C}$-anchored features.

The results indicate that the anchored versions of the features are more robust and provide higher precision (0.67 when cost is at 2.5) than standard MFCCs (0.07) for determining pronunciation quality. Furthermore, at an overall recall rate of 0.22, $\overline{C}$-anchor finds mispronounced tokens for every vowel, except schwas, whereas the MFCC model identifies only mispronounced /ɑ^y/ [ay]. The decision trees produced confirm that the divergence measurements are important in determining pronunciation quality after the anchoring has been performed, as they constitute approximately 68.7% of the number of decisions in the tree, after the superficial decisions about classification labels have been removed.

Chapter 6

Summary & Future Work

This thesis explored pronunciation evaluation for *Computer Aided Language Learning* (CALL) systems. It focused on detecting vowel mispronunciations by Cantonese speaking learners of American English with high precision. To accomplish these tasks, our research made the following assumptions about the structure of the CALL system. It assumed that a correct word transcription of each utterance had been obtained and that mispronunciation detection would be performed as part of an offline operation run after a complete dialogue had been finished.

6.1 Contributions

This research invented three novel techniques that addressed different aspects of detecting mispronunciation. A labeling algorithm was developed that enables the use of cheap online labor to obtain phone-level labels of pronunciation quality from word-level non-expert annotations. An anchoring technique was developed to account for speaker intrinsic pronunciation differences and to allow for meaningful comparisons of vowel pronunciation. Finally, a mispronunciation detection technique was invented based on data labeled using the crowd-sourced algorithm and the anchoring method developed in the previous two chapters. The next three sections detail the significant findings of this research.

6.1.1 Crowd-sourced phonetic labeling

Chapter 3 presented an interface and methodology for collecting word-level judgements of pronunciation quality from anonymous English speakers using the Amazon Mechanical Turk service. A cost analysis showed that the methodology was extremely cheap—costing $1,211.10—and produced very rapid results by collecting 920,256 word level annotations in under 24 hours.

Novel methods for analyzing the quality and consistency of these annotations were developed, and they showed that the annotation quality was comparable to that of expert annotated corpora for similar annotation tasks. A statistical analysis of the data at the phonetic level showed that annotations and substitution rates between hand transcribed and machine transcribed utterances could be exploited to provide phone-level annotations of mispronunciation.

An algorithm was invented that combined the results of word level annotations collected using Amazon Mechanical Turk with alignments between hand transcribed utterances and machine transcribed utterances to produce phone level annotations of pronunciation quality. This algorithm was applied to a large corpus of non-native English speech data.

6.1.2 Anchoring for vowel normalization

Chapter 4 presented a novel method for normalizing vowel productions to account for individual speaker differences. This method relied on the estimation of the intrinsic vowel locations for individual speaker voices. We showed that, by normalizing acoustic features with this method, substantial performance increases in a simple classification task could be realized.

In particular, we showed that anchoring produced relative error improvements of between 1.8% to 6.7% for native speech classified with native acoustic models and 3.4% to 6.8% for non-native speech classified with non-native acoustic models. These improvements were seen regardless of the specific vowel or point of anchoring. Surprisingly, substantial increases in performance were realized for non-native data classified using native acoustic models after anchoring. These improvements ranged from 6.1% to 8.4% relative improvement. The most substantial improvement came from using a weighted mean of the entire vowel space of individual speakers.

We hypothesized that the improvements indicated that anchoring the vowels would enable more robust comparisons of non-native speech with native speech. A qualitative analysis showed that after anchoring, the vowel space of native speakers and non-native speakers were moved closer together. This was shown with a holistic convex hull representation of the vowel space as well as in individual vowel distributions.

A quantitative analysis comparing the Bhattacharyya distances between native and non-native distributions of the vowels showed that the distances between the distributions were negatively correlated with vowels that had been labeled *ugly* by the crowd-sourced labeling algorithm. Additionally, a positive correlation with vowels labeled *mispronounced* was found to exist for the Bhattacharyya distances between native and non-native distributions after the feature spaces had been anchored. This positive correlation did not exist for distributions that had not been anchored. This result lent further support to using anchoring in conjunction with statistical divergence measurements in a mispronunciation detector.

6.1.3 Mispronunciation detection

A mispronunciation detector, based on a decision tree classifier trained with the c4.5 algorithm and augmented with a cost matrix, was presented that used results from Chapters 3 and 4. A set of novel features were identified and developed to train and test the decision tree.

Features, such as the posterior probability and posterior probability ratios, have been used in previous research. This research introduced expanded versions of the pre-existing features as well as derived novel features for mispronunciation detection. A comprehensive set of features based on divergence measurements between statistical distributions of the vowel classes was incorporated into the mispronunciation detector feature set.

The performance of decision trees trained with features from two difference acoustic features, MFCCs and anchored MFCCs, showed that the anchored version of the features provided enhanced precision in identifying mispronunciations. Novel methods for analyzing the precision of the decision tree were developed; in particular, we accounted for the fact that some vowels are more easily evaluated for mispronunciation than other vowels, and quantified this measurement.

When a sufficient cost was applied to misclassifying *good* pronunciations as *mispronounced*, the $\overline{C}$-anchored version of the decision tree attained a precision of 0.67 compared to 0.07 for the MFCC version of the decision tree, which exhibited the peculiar property of zeroing in on a single vowel, /ɑ^y/ [ay]. This strengthens the findings and hypothesis from Chapter 4 that anchoring the vowel space enables more robust comparisons of pronunciation quality.

We also analyzed the performance of the mispronunciation detector in terms of the actual decision trees. In particular we found that, while the MFCC version of the tree was significantly smaller than the $\overline{C}$-anchor version of the tree, the differences could be accounted for by noting that much of the size increase was due to decisions involving the label assignment by both the native and non-native GMM classifiers. The $\overline{C}$-anchored version of the decision tree utilized more information about the divergences of the statistical distributions. In order do this, it needed to know what the n_result and nn_result of the GMM classification step was. This resulted in a bushy tree. When these were accounted for, we found that the divergence measurements comprised 68.7% of the decisions in the tree. We also found that the posterior probabilities and posterior probability ratios were seldom used in the the trees. This, again, supports the findings from Chapter 4 that showed correlations between the divergence measurements and assessments of mispronunciation.

6.2 Directions for Future Research

As with all thesis work, there are several aspects of this research that could be improved, expanded on, or further explored. This section will address each area separately and discuss potential directions for future work.

6.2.1 Crowd-sourced phonetic labeling

The use of crowds to perform tedious speech tasks is new, having only really taken off in the past two years. Therefore, the field is wide open for all manner of research. For the purposes of this discussion, we'll focus on the use of crowds for mispronunciation labeling.

One hazard of using anonymous, non-experts in a service such as Amazon Mechanical Turk is that verifying credentials can be tricky. In the research presented here, we used a crude system where we simply required that all Turkers be located in the United States and have a 95% accept rate on their HITs. We assumed that this would sufficiently restrict Turkers to be at least fluent in American English, if not native speakers of the language. This assumption is not necessarily a correct assumption; for example Gruenstein et al. [89] found that many participants in their language tasks had strong Indian accents. This could affect results, especially in a task where the question is a judgement of pronunciation quality.

We found in our research that the Kappa scores were consistent with other research conducted under more controlled circumstances, so we did not think it invalidated our approach. This, however, is a topic that should be explored. Verification strategies could range from requiring an audio recording of the Turker completing the task—which could discourage people from participating—to presenting the Turker with obviously mispronounced words and weeding out those Turkers who failed to correctly mark those words.

Another hazard with anonymous crowds is the quality of work. We found that a large portion of the utterances (2,734) had to be rejected and resubmitted from the original batch. The HITs were found to have been completed in fewer seconds than would have been required to listen to all the utterances—obviously, the work was not worth anything. The reject was performed manually, but it could have been easily automated with a little foresight. An interesting question might be to what extent bad responses affect agreement results.

In this research, we established that the level of agreement was within what could be considered a moderate amount of agreement. This conclusion, however, was reached based on superficial comparisons with other kappa values in the literature, as well as kappa values we obtained from a similar, though different study. These definitions of what constitute moderate levels of agreement are arbitrary, but generally accepted by the community. A more rigorous study of this would be to compare the level of agreement reached by a traditional controlled annotation of the corpus with that reached by anonymous crowds.

The agreement levels we studied were inter-rater agreement, or how much Turkers agreed with each other on the same utterances. Another source to quantify the quality of the ratings would be intra-rater agreements. This measurement would examine the self-consistency of the raters. In a traditional annotation scheme, this would involve presenting the rater with a few of the same utterances as they were performing the annotation, without informing them that this was occurring. This is easy in a situation where it is known how many utterances the rater will label, and randomization could be used to minimize the chance that they could simply copy their previous answers. This would not be so straightforward using anonymous crowds because there is no guarantee that the Turkers would take on another HIT. Further, simply batching the same utterance into the same HIT would not give an accurate assessment of intra-rater agreement, because it would be pretty easy for a Turker to figure out what the duplicate utterances were.

We used a simple annotation scheme for this study. We allowed Turkers to only mark words as mispronounced or missing. This simple system of categorization was intended to restrict annotators enough to facilitate agreement between annotators and to keep the task simple. Although our results indicate a moderate level of agreement among the annotators, it is possible that there is an inherent limitation of annotating non-native speech for pronunciation errors using such a simple scheme. The results may indicate that an additional category may be beneficial, for example a third category signifying that the rater felt the word was not mispronounced, but it wasn't necessarily pronounced well—instead of deriving the *ugly* category based on the number of mispronunciation markings, we push the decision to the rater. This would give them some flexibility when they aren't sure which of the two categories to choose.

Finally, each datum in our corpus was annotated by 3 Turkers. Recent work by Hönig et al. [90] attempted to answer the question of how many labelers are needed for a given annotation task. Although the investigators focused on labeling non-native prosody, it is conceivable that this could be extended to the task of annotating good and mispronounced words. This has potential impact on the quality of the annotations and the cost of annotations. A HIT that pays \$0.10 for the annotation of 5 utterances costs a total of \$720.00 to annotate 36,000 utterances with a single Turker. Additional annotators increase the cost linearly, with 3 Turkers costing \$2,160.00, 4 Turkers costing \$2,880.00. While still very cheap compared with annotation by experts, having an idea of the number of annotations required for a task would help further control costs.

6.2.2 Anchoring for vowel normalization

Anchoring is a simple method for removing speaker dependent differences in vowel pronunciation that translates MFCC features vectors. This translation is defined by an anchor point measured from many samples of a particular vowel, or derived from many samples of multiple vowels.

While this thesis only considered the language pair of English-Cantonese, there is no reason why a similar technique could not be employed for any other language pair. We started from the premise that anchoring on so-called universal vowels would put speakers on equal footing when performing mispronunciation detection, but we later showed that a weighted average of the speaker's vowels functioned more effectively as anchor points. There is no reason to think that this method would not be generalizable to other languages with different vowel inventories.

The corpus we used consisted of Cantonese speakers from Hong Kong. We have noted at a few points in this research that the accent of instruction, British English, could have affected Turker judgements of mispronunciation, as well as the mispronunciation detection algorithm. This was due to the differences in a few of the English phonemes as produced by American and British speakers. A further analysis of the techniques presented here, either utilizing a corpus of British English, or a corpus of English learners instructed using an American accent, is warranted.

We tried several anchor points for the vowels and found that the $\overline{C}$-anchor vowel was the best performing. This anchor point was a weighted mean of all the known vowel instances the speaker had uttered. The other anchor points were simply the means of the instances for a particular vowel class. In all of these cases, knowledge of what vowels had occurred and their quantity was required. This is in part why we required knowledge of the transcripts of the vowels to be assessed—we need to be able to measure the anchor points.

A potential direction for future research in this area would be to look at using a voiced-unvoiced classifier to determine points at which the speaker had any voicing. This could be used in place of a full blown forced path recognition or as a pre-processing step prior to recognition, thus enabling the use of the anchoring algorithm for typical recognition applications.

We also did not explore differences in genders. We've assumed that the anchoring transforms all features into the same feature space; however, it is possible that gender differences would result in slightly different feature shapes, particularly in the upper MFCCs. A study of the effects of gender on the anchoring algorithm would also be a potential area for research.

We only explored the effect of this anchoring technique on vowels. This restriction seemed logical as vowels have better defined formants than, for example, a fricative such as /s/ [s]. It is unclear if the same, or similar technique would be applicable for non-vowels.

The transformation performed is similar to the MLLR technique developed in [85], with the transformation matrix set to the identity matrix. The attraction of transforming the MFCCs using our technique is that it is simple to implement and only requires instances of a speaker's common anchor vowels in order to be applied. Future work could include comparing the performance of our transformation with the MLLR technique and exploring simple methods that account for variance in our technique. We should also compare our technique with VTLN; however, because VTLN shows the most significant gains when normalizing for child speech and between genders, we are not sure how it will perform when moving between native and non-native speakers.

Finally, while we did perform a good deal of analysis concerning the relation of the Bhattacharyya Distance measure to rates of *mispronunciation* labeling, we did not compare the technique against the work by Minematsu et al. [156, 218] In part, this was because their work assessed pronunciation holistically. It would still make an interesting study to see if the correlations they found with human assessments of pronunciation still held after the statistical distributions were anchored.

6.2.3 Mispronunciation Detection

We utilized c4.5 decision trees to perform the actual detection of mispronunciations. We simplified the analysis somewhat by excluding instances of *ugly* vowels. We made this decision in order to provide a sharper contrast in the training and testing data between *good* and *mispronounced* instances. One question we did not attempt to answer was how the highly skewed distribution of the mispronunciation data affected the results and whether another algorithm for training the decision trees would be more appropriate.

It would be worth exploring the question of how this technique performs when there is not such a sharply binary decision. Similar to the idea of expanding the number of annotation choices available to the Turker labelers, it would be interesting to examine if a similar multiple labeling system would work for the decision trees.

One potential analysis would be to regard the *ugly* category of the vowel labels as a fallback position. If, in the course of analyzing a speaker's performance in a dialogue, no vowels are identified as *mispronounced*, then the system could fallback to pointing out vowels identified as *ugly*.

This would be useful for learners who have pronunciation problems, but not severe problems. For example, in Chapter 3, we found that the vowel /ɚ/ [er] had substantially more instances of *ugly* judgements than the rest of the vowel classes. This indicated ambivalence on the part of the labelers, and a system that could mimic or detect that would be valuable.

Along these lines, another potential analysis would be to regard the *ugly* vowels as an explicit *don't care* class. We analyzed the detector only in terms of precision and recall for the *mispronounced* category. When a *good* vowel was misclassified as *mispronounced* we applied a severe penalty. But it is not necessarily the case that penalizing an *ugly* vowel should have a similar penalty. The reason is that *ugly* vowels have been marked as mispronounced by at least one Turker; thus, it wouldn't be incorrect for the system to flag it as *mispronounced* as opposed to *ugly*. In effect, one could regard the *ugly* and *mispronounced* as equivalent under a certain analysis.

Methods for optimizing the decision trees could be explored. This research trained a single decision tree for the task of determining mispronunciations. One avenue would be to train individual decision trees for every vowel class. Instead of relying on the algorithm to sort out the features applicable to mispronunciation detection for all vowel classes, we would instead train separate decision trees for every vowel class. For example, when deciding the label to assign to a vowel /ɑ^y/ [ay], instead of using the same tree as would be used for all other vowels, there would be a specialized mispronunciation detection tree for /ɑ^y/ [ay].

Finally, a tree pruning strategy should be explored. We are really only interested in those decisions where the resulting label is *mispronounced*. A potential method for optimizing the tree would be to discover those rules, or series of decisions, that are good at identifying *mispronounced* vowels. The tree could then prune away other branches to favor branches that are highly successful at identifying mispronunciations.

6.2.4 Application to other domains

This research focused solely on vowel mispronunciation detection. However, this step simply used probabilistic scores obtained from a GMM classifier to perform the detection. It should be easy to adapt to other domains where detection of pronunciation errors is desired. For example, the groundwork has already been laid in [184, 209, 182, 183] for automatic tone mispronunciation detection. In this research, tone classification was performed by GMMs after normalizing f_0 to account for speaker differences. The adaptation of the decision tree to detecting tone mispronunciations based on the model scores produced in this framework should be relatively straightforward.

Appendix A

A Comprehensive Overview of Computer Aided Language Learning

Computer Aided Language Learning (CALL) is a cross-disciplinary field that includes the subfields *Foreign Language Learning* (FLL), *Foreign Language Teaching* (FLT), Linguistics, and *Human Language Technologies* (HLT). FLL research typically focuses on topics such as learning strategies employed by students and effectiveness of environments designed to support learning. Closely related, FLT focuses on discovering and employing effective pedagogies to facilitate learning as well as meaningful performance measurements. Linguistics, specifically the subfield of *Second Language Learning* (SLA), focuses on the process of learning a second language by investigating common patterns of mistakes and progression in competence. Finally, Human Language Technologies encompasses the full-range of technologies, from audio recordings to dialogue systems, used to facilitate learning.

This chapter is divided into four sections. Section A.1 gives a brief overview of FLL. Section A.2 discusses some of the challenges of using technology for FLL. Section A.3 discusses general technological issues with CALL. And finally, Section A.4 goes in depth on the technologies and approaches used for *Computer Aided Pronunciation Training* (CAPT).

A.1 Foreign Language Learning

How people learn a language is a complex subject with several fields of related research. *Foreign Language Learning* (FLL) research is concerned with the investigation of successful and unsuccessful strategies employed by students to learn a foreign language in a directed learning setting. FLL is part of a broader field called *Second Language Acquisition* (SLA), which studies foreign language acquisition in all contexts. *Foreign Language Teaching* (FLT) studies strategies intended to help facilitate learning a foreign language. In contrast to FLL, which is student centered, FLT is teacher centered; attempting to discover and refine techniques to better instruct students (see [33] for a review of language teaching research in the 20th century).

These fields all interact to influence curriculums, teaching and learning strategies. For example, FLL research has identified the motivation of a student to learn a foreign language [80] as a strong predictor of successful foreign language learning [31]. FLT has responded with research on methods for motivating students in the classroom [57, 58].

FLL researchers have also found that language anxiety [100] is correlated with success in language learning [139, 142, 140, 141, 144, 143]. A comprehensive review of the literature on language anxiety by [247] found that there were six factors associated with language anxiety: personal and interpersonal anxieties, learner beliefs about language learning, instructor beliefs about language teaching, instructor-learner interactions, classroom procedures, and language testing. She proposed several methods for helping reduce langauge anxiety, among them planning language activities for small groups of students that involve roleplay or games.

A.1.1 Teaching Methodology

The complexity in language learning is compounded by the fact that the best method of instruction is still the subject of investigation. There are two broad categories of classroom instructional methods that are supported by contrasting views on foreign language acquisition: structural and interactive [197].

Teaching methods that fall into the structural category view language as a habit that is learned through repeated drill and knowledge of the rules of a language. After habitual knowledge of the structure and rules of a language has been established, the learner can communicate in the language [198]. Although structural teaching methods have fallen into disfavor in part due to Chomsky's criticisms of behavioralist views on language [36], significant elements of these types of methods remain in use.

Teaching methods in the interactive category view language as a communicative activity that should be practiced as such. One specific method for language instruction is the communicative method, which emphasizes interaction as the means and goal of foreign language learning. Syntax and pronunciation will be learned naturally through practice speaking and listening [126]. A succinct description of the differences between the structuralist and interactive views on language teaching is ``Function follows form; form follows function." More in depth discussion can be found in [13, 28].

The current trend in language teaching favors communicative methods. Conversational practice is emphasized and corrections are made judiciously. Modern communicative methods include task-based techniques, which use loosely defined scenarios to prompt dynamic conversation between students. Good discussions of the various forms and issues in task-based instruction can be found in [61, 212].

A.1.2 Measuring Language Performance

A core principle of communicative language learning is that knowledge of syntax and vocabulary form only a part of a larger hierarchy (Figure A-1 that collectively form an individual's communicative competence [148]. Assessing student communicative competence is a major research challenge for FLL and FLT.

Tests such as fill-in-the-blank, part-of-Speech quizzes, etc, measure a student's performance on a small subset of language related activities. This leads to situations where a student who does well on grammar tests fails to perform in real world situations. In a classroom patterned on communicative principles, more comprehensive examinations must be performed to measure student progress [236].

Foreign language tests to measure foreign language proficiency are quite numerous and

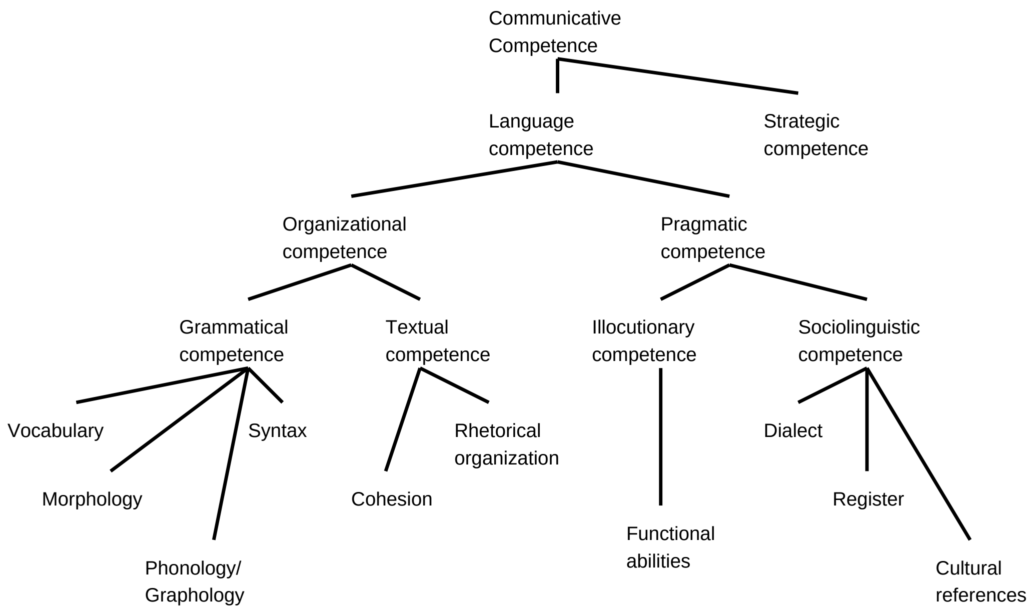


Figure A-1: A hierarchical breakdown of communicative competence, recreated from [178].

always under development [73]. Most tests take the format of an *Oral Proficiency Interview* (OPI), such as the *American Council on the Teaching of Foreign Languages* (ACTFL) OPIs [221].

In Oral Proficiency Interviews, a certified interviewer attempts to elicit speech by asking questions of varying difficulty. These questions guide an interviewer to one of four proficiency levels: novice, intermediate, advanced, and superior [5, 6]. Another standard set of speaker levels comes from the *Common European Framework* (CEF) [177]. *Simulated Oral Proficiency Interviews* (SOPI), are based on the same ACTFL *Proficiency Guidelines*, but are self-administered through carefully constructed tape interviews [130, 214]. Student responses are recorded on a blank tape for evaluation by a certified rater at a later time. *Computerized Oral Proficiency Interview* (COPI) [145] stores different levels of questions which are used to adapt the test according to the comfort level of the student during the interview.

A common denominator of all of these tests is that they attempt to measure overall language ability. They do not make use of any language technologies such as speech recognition or synthesis to automatically perform assessment. While the current state of the art in speech technology is not able to fully assess a student's language competence as well as a human, some systems can operate well enough at lower proficiency levels to be useful. Additionally, there are many systems that can assess small subsets of the language competence hierarchy (Figure A-1), such as phonology (pronunciation).

A.1.3 Pronunciation

Intelligible pronunciation is only one of the needed skills for speaking a foreign language, and it is often not emphasized in the classroom. There has been some renewed interest in teaching pronunciation explicitly [87] due to studies that show that pronunciation quality below a certain level of proficiency places additional stress on the listener and seriously degrades the ability of native speakers to understand what is being said [98, 251].

Most adult learners, and even those as young as 6 years old [244], of a foreign language retain some artifacts in their pronunciation that identify them as non-native speakers, although the attainment of native-like pronunciation has been observed [24]. Despite the presence of an accent, native speakers will not necessarily identify speech as mispronounced if the quality is above some subjective level.

Improvements in the pronunciation of learners whose pronunciation has plateaued at a less than desirable level are possible through pronunciation training [52]. Native-like intonation can also be learned [153]; however, this is extremely difficult for even advanced language learners. In addition to requiring lots of output [220] to improve pronunciation, students cannot attend to all aspects of pronunciation at the same time [53], e.g. attending to phonetic accuracy takes processing time away from attending to intonation.

A foreign language learner will make a number of pronunciation errors at the phonemic (segmental) and prosodic levels when producing speech in a target language. Errors at the segmental level can be generally classified as substitution, insertion, deletion, and duration errors. Errors at the prosodic level are more difficult to categorize. There is some debate over whether phonetic or prosodic aspects of pronunciation have more impact on perceived pronunciation quality [165]. While the sources of these errors are a topic of research in the linguistic community, there seems to be a consensus that the phonetic inventory of the native language interferes to a certain extent with the production of sounds in the foreign language [72].

A well-known example of a substition error caused by native language interference is the difficulty native Japanese speakers have with the `/l/-/r/` contrast in English [27]. Another example of native language interference is the devoicing of word-final obstruents in Cantonese speakers of English [185]. More detailed discussion of second language pronunciation can be found in [134].

Another source of error is the inability of non-native speakers to become attuned to critical acoustic features in the target language. For tonal languages, such as Chinese, students arriving from a non-tonal language often have difficulty even perceiving changes in the pitch indicating the presence of a lexical tone. This has an impact on their ability to produce these tones correctly [234]. For example, Japanese learners of Korean had difficulty discriminating between lenis (weakly aspirated) and aspirated alveolar stops [123]. Careful analysis of perceptual differences between Japanese and native Korean speakers showed that Japanese learners of Korean placed more emphasis on VOT than f_0 when discriminating between the lenis and aspirated stop; however, native Korean speakers were able to use both acoustic features to successfully discriminate between the sounds. This suggests that students sometimes have incomplete or confused models of the speech sounds in the language.

A.2 Technology in Foreign Language Learning

New technology always introduces challenges and controversy when applied to teaching. The previous section provided a brief overview of research in foreign language learning. This section summarizes some of the research on the challenges and benefits of integrating technology into the foreign language classroom.

> *``This new technology will ruin education."*
>
> *``No, it won't. It will make education much more efficient than it is now."*
>
> *``I see the problem as one of depersonalization! If this new technology is*

> *done well, it won't even be necessary to have teachers at all. Students will interact with technology rather than with human beings."*
>
> *``Not true! Teachers can permit students to learn basic information more efficiently from the new technology. Then the teachers will be able to use their own time to focus on individual needs. The result will be an increased quality of the interactions between students and teachers."*
>
> *``But almost no students or teachers know how to use the new technology. They'll be dependent on unseen technologists and mysterious forces to control their learning."*
>
> *``Then maybe students and teachers will have to acquire a certain degree of literacy. The benefits will be worth the effort."*

The above fictional dialogue from Vockell and Schwarz [230] is between two educators discussing the increasing availability of the book about 500 years ago. Many of the same concerns illustrated in the dialogue are applicable to CALL. Foreign Language Learning has endured and incorporated a number of technologies --- from books to tape recordings to video to full-fledged multimedia presentations --- amid healthy debates on their merits [205].

A primary concern about integrating computer technology in the language classroom is if it will actually help students [59, 105]. While controlled studies on integrating computer technology into the classroom are difficult to perform due to the large confluence of factors involved [82, 70], the results are generally positive with some caveats.

Some of the earliest results from IBM [2, 159] indicated improvement in German proficiency among college age students who completed fill-in-the-blank exercises paired with audio recordings. *English as a Second Language* (ESL) students improved their English language proficiency significantly utilizing the VOXBOX (now *Yo Hablo Español*) [147].

A comparison of computer-versus teacher-directed grammar instruction in [176] found that, on a test containing open-ended questions, students taught in a computer-based classroom scored significantly higher than students taught in a classroom without computers. However, the same study also found no significant differences between the groups of students on tests that were multiple-choice or fill-in-the blank.

Research at *Carnegie Mellon University* (CMU) found that students in a French class with a required, but independently completed, Technology Enhanced Language Learning component of instruction performed at equal or better levels than counterparts in classes without the component [1].

An *Automatic Speech Recognition* (ASR) based CAPT system was used to provide feedback on problematic sounds to learners of Dutch with varying native-language backgrounds [168]. The authors found that the performance of the speakers improved after using the system for four weeks as part of a standard language course at the university.

Computer technology must also be considered in the context of the student. Research in [95] attempts to answer the question of what types of students would benefit most from computer-aided pronunciation training by assessing performance on listening tests pre- and post-training. They found some correlation with syllable and word identification tasks, but did not find correlations with rate of learning measurements.

These results indicate that the contributions of technologies must be narrowly stated. The studies cited above assessed language ability for pronunciation, grammar, or communication ability, but not all at once. No single computer-based technology will be better than a live teacher at the whole process of foreign language instruction: ``the computer is a *medium for learning and not a method for L2 instruction*" [1]. Computers are prone to mistakes that human teachers do not necessarily make [160, 106], and are not yet able to adapt to the learning styles displayed by students. These issues aside, the results still indicate that computer technology can be successfully integrated in a FLL classroom, at least in a narrow sense.

A.3 Computer Aided Language Learning

Researchers have investigated the use of computers for language learning since the 1960s [227]. The field of *Computer Aided Language Learning* (CALL) has seen an explosion of research over the past decade, and it would be impossible to include every piece of research in this thesis. This section will discuss representative examples of CALL. A further review of the history, key developments, and major paradigms in Spoken CALL can be found in [67].

CALL research, from a purely technical standpoint, can be divided into roughly two areas: research focused on whole systems and research focused on specific technologies to be integrated into whole systems. This section deals with whole systems, and highlights three areas: early systems, modern systems with voice input, and dialogue-based systems. The next section will go into depth on the *Computer Aided Pronunciation Training* (CAPT) subsystem.

CALL systems are numerous and diverse. On the simple end of the spectrum, the systems can take the form of web pages with fill-in forms [200, 135], online chat rooms, static multimedia programs, modifications to popular games [189], or even simply a set of digital music files for playback purposes. On the complex end, systems can have automatic speech recognition, voice synthesis, and highly interactive 3D environments that teach cultural norms as well as language [115].

Systems can vary by intention. For example, some CALL systems are intended only for vocabulary acquisition [186, 88], and some software focuses on grammar instruction [166]. Software intended for pronunciation training can be broken down into even finer categories, such as those intended to train students on the segmental quality of speech, and those intended to teach intonation at the phrasal level.

A.3.1 Early Systems

The *Programmed Logic for Automatic Teaching Operations* (PLATO) [94] system was one of the earliest CALL systems that ran on a large and costly mainframe. PLATO and other similar systems were primarily text-based in which a student was presented with an exercise and told to fill in the appropriate word or some other similar exercise. If they were wrong, the program informed them, often times without a clue as to the nature of the error, and prompted them again. The pejorative monikers, ``drill-and-kill" or ``wrong-try-again" were used to describe the monotonous and unenjoyable aspect of systems of this type.

IBM also developed specialized hardware and programmed materials for teaching beginning German at the State University of New York at Stony Brook [2, 159, 203]. The exercises in this system were mainly fill-in-the-blank questions accompanied by pre-recorded audio and 35-mm still photos.

The *Computer-Assisted Review Lessons On Syntax* (CARLOS) [225, 3] system was another mainframe-based system developed at Dartmouth to help students learn Spanish grammar [26]. When desktop computers began appearing in the early 80s, DASHER [190] was developed with similar functionality to the mainframe based systems.

At the *Massachusetts Institute of Technology* (MIT), a sophisticated program for teaching scientific German was created [206]. A unique characteristic of this program is that students could interactively explore the meaning of words and phrases using German. The MIT Athena Language Learning Project [125, 158] utilized a large number of networked computers to deliver multimedia content and interactive typed-input language games.

Other early systems used graphical displays [229, 116, 54, 174] to aid in pronunciation training. The novelty in these systems is that a visual representation of the speech was used to provide objective feedback to the students. A limitation is that the technology did not provide guidance for correcting speech by indicating the precise nature of the errors, so a teacher had to be present to help the student interpret the results.

Key characteristics of these early systems are that they had a relatively small amount of material and they were mostly text-based with audio being available only in the form of pre-recorded phrases. They also tended to focus on one or two aspects of language learning, i.e. pronunciation or vocabulary acquisition. These systems also completely neglected the communicative aspects of language learning in that they required little output from the student.

A.3.2 Modern Systems

Modern systems tend to be much richer language learning environments that incorporate high quality audio, graphics, and automated feedback. The content of the lessons is usually not static, and is generated randomly or adaptively, in response to student actions. Many systems use some form of ASR, speech synthesis, natural language understanding, or natural language generation.

WebGrader™ [172] was a pronunciation tutoring tool that enabled students of French to obtain automatic assessments of their pronunciation qualities based on calibrated machine scores. One of the interesting findings was that students were frustrated that the scoring sometimes seemed inconsistent, felt the ability to break down the sentence into word level evaluations was helpful, and desired targeted feedback to help improve problem areas.

The *Voice Interactive Language Training System* (VILTS) [204] used a task-based language learning approach. Learning activities were divided into three separate levels with categories of activity (speaking, reading, and listening) dealing with several topics. A GUI suggested the order in which the lessons could be covered, but students were allowed to explore on their own in order to adapt to individual learning needs. The study found that students reacted positively to the system, finding that the freedom of navigation, speech recognition in interactive activities, and pronunciation feedback were all important factors in the positive reception of the program.

The *EduSpeak* system [76] was a toolkit that used ASR to implement pronunciation scoring for a variety of languages. Although not a complete system in and of itself, the toolkit is noteworthy because it was specifically designed for allowing different recognizers and models to be used as required by the specific language learning task.

The *Tactical Language Tutoring System* (TLTS) [115, 112, 114, 113] is an example of a rich, multimedia system for language learning. The student is immersed in a 3D world using the *Unreal Tournament 2003* [62] game engine where he is instructed to accomplish missions --- the system was developed for military use --- by interacting with characters in the environment using Arabic speech and non-verbal communication. Speech recognition is performed using the *Hidden Markov Model Toolkit* (HTK) [248] augmented with noisy-channel models to capture mispronunciations [161].

The CALLJ system [233] created dynamic practice questions based on teacher specified sentence patterns. Pictorial representations of the parts of the sentence to be practiced were shown to prompt the student, and an explicit target sentence was generated. A grammar network, is created based on a decision tree, attempts to capture potential errors according to greatest impact, where impact was defined as an increase in the error coverage of the grammar augmentation divided by the increase in perplexity of the model. This constrained the recognizer so that errors in grammar could be captured without too many recognition errors.

A.3.3 Dialogue-based Systems

Dialogue systems can be used to create immersive environments in which students hold dynamic, fairly natural conversations [96, 132, 17, 231, 63]. Instead of being given a specific sentence or a limited script to follow, which can lead to memorization and plateauing [79] in learning, students can hold conversations that are varied between practice sessions. Since speech recognition technology is imperfect, there is constant tension in dialogue systems between allowing freedom in conversation and sufficiently constraining the domain to maintain acceptable performance. Dialogue systems adopt different strategies to strike an appropriate balance.

Subarashii [60, 19] was a dialogue system that advanced the conversation using a predefined set of responses in a sort of choose-your-own-adventure style of dialogue. Later research crafted the dialogues to elicit a limited set of responses without explicitly stating them.

Subarashii was specifically designed for language education. In contrast, a prototype system by Lau [133] was created by adapting an existing dialogue system capable of conversing in both English and Chinese. It allowed for simple, unstructured conversations about families, but the architecture allowed for adaptation to new domains. Students would conduct conversations in Chinese, or ask for translation help in English.

Raux and Eskenazi [195] adapted an existing spoken dialogue system [196] to handle non-native speech [194] using a generic task-based dialogue manager [23]. Another key feature of the system was the use of clarification statements to provide implicit feedback through emphasis on certain parts of a student's utterance [193].

Another example dialogue system is the *Computer Simulator in Educational Communication* (CSIEC) [109]. The CSIEC is unique in that, although it does not use speech to carry on a dialogue, the dialogue is unconstrained. Instead of working towards the completion of a task, as in most other dialogue-based systems, the CSIEC envisions the interaction of the student and the computer as a friendly chat. Later versions of CSIEC added Microsoft Agents to function as avatars for the computerized chat partners [101], and constrained the chat to specific topics favored by a particular student student [108].

Chao et al. [32] created a web-based translation game for learning Chinese with repetitive exercises for acquiring vocabulary and grammar. This system was later adapted to create a simple dialogue game in [208, 207]. McGraw et al. [149, 150, 151, 246] created multiplayer web-based games focused on vocabulary acquisition. Students used natural speech in a highly constrained domain to manipulate cards representing new vocabulary items in competitive games.

The *Development and Integration of Speech technology into COurseware for language learning* (DISCO) system [47] is a Dutch system for providing feedback on pronunciation, morphology, and syntax. The system exploits morphology and syntax errors common in learners of Dutch as a foreign language. The DISCO system conducts dialogues by eliciting very constrained responses to questions; it uses a two step process for recognizing speech in a constrained domain. In the first step, it determines the content of a learner response, by augmenting an *Finite State Transducer* (FST) language model. In the second step, it then analyzes that response for correctness with stricter constraints [228].

The *SayBot* Player is a system for teaching English to native Chinese speakers [35]. It maintains a teacher designed dialogue flow using a Finite State Machine architecture. Pronunciation is scored using *Hidden Markov Model* (HMM) log-likelihood scores and duration measurements. Errors during the dialogue are classified into four categories: Correct (all words are correct and the pronunciation score is good), Pre-defined Error (pronunciation score is good, but sentence is recognized among a set of predefined errors), Mispronunciation (recognized words are produced poorly), and General (the system could not understand the student speech at all).

A.4 Computer Aided Pronunciation Training

Computer Aided Pronunciation Training (CAPT) systems are specifically designed to evaluate and improve pronunciation in foreign languages. A CAPT system can be considered to have an evaluation component and a feedback component. Pronunciation evaluation can take place at two general levels: holistic and pinpoint error detection. A holistic evaluation examines a large sample of speech and provides an overall assessment of a speaker's proficiency. Pinpoint error detection attempts to identify specific pronunciation mistakes at the word or subword level.

A.4.1 Holistic Pronunciation Evaluation

Several methods have been proposed for holistic pronunciation evaluation. Most involve the correlation of subjective human assessments with machine-based measures. Acoustic and probabilistic measurements include total duration of read speech with no pauses, total duration of speech with pauses, mean segment duration, rate of speech, and log likelihood measurements. Human ratings include global pronunciation quality, segmental quality, fluency, and speech rate.

The earliest work on pronunciation evaluation was performed by Wohlert [243, 242]. In his research, Wohlert selected 160 of the most commonly used, strong German verbs, and divided them up into 16 categories with 10 words each. The system used a template based on the average of five pronunciations for each German verb.

A series of five exercises, such as fill-in-the-blank and translation, were created for each group of verbs. During the tutoring session, the student is presented with a score from 500 to 1000, 1000 being a perfect match. The score is based on how closely the speech produced by the student matches the template stored in the database. One shortcoming of this research was that the correlation of the scores to human rater evaluations was not performed. Still, after a semester of work, with one group of students learning German using the new system compared to a control group, he found an increase in the number of verbs the students in the former group mastered (87% of the presented vocabulary) versus the number mastered by students in the latter (67%).

Early research by Bernstein et al. [16, 14] investigated methods for accurately predicting scores similar to those given in *Oral Proficiency Interviews* (OPI). The PhonePass system, which grew out of this research, was developed to assess non-native English proficiency [222]. The researchers gathered telephone quality data from a large number of responses to five different types of questions that reflected conversational speech. Correct and incorrect responses were combined with HMM scores and used as inputs into a function that produced a score correlated with expert human judgements of proficiency.

Later research validated the scores against the CEF [177] for assessing language proficiency [15]. A version of the algorithm was developed to assess non-native Spanish and validated against the ACTFL, *Interagency Language Roundtable* (ILR), and *Spanish Proficiency Test* (SPT) OPIs [18], and later adapted to Modern Standard Arabic [20].

Cucchiarini et al. developed similar methods for assessing the proficiency of non-native speakers of Dutch [42, 41]. In contrast to other assessment methods, which examined pronunciation errors from speakers with a common native language, they investigated the assessment of speakers with many different language backgrounds. Subjects were asked to read two sets of five phonetically rich sentences. Human judgements on overall pronunciation, segment quality, fluency, and speech rate were gathered from three expert phoneticians.

They found that machine generated measures such as duration, rate of speech, and log-likelihood scores were highly correlated with human judgements of pronunciation quality, though a caveat is that the log-likelihood scores are also highly correlated with duration measurements and might not be of any use. They also discovered that using rate of speech or duration measurements also permitted students to ``cheat" by speaking rapidly. Subsequent research found that the use of log-likelihood scores could mitigate this problem [48, 44, 69].

Subsequent research expanded the research to include spontaneous speech as well as read speech [46, 40, 216, 45, 43]. In addition to adding spontaneous speech they added two groups of human raters, both consisting of speech therapists. They also modified the set of machine scores to be: rate of speech, phonation-time ratio, articulation rate, pauses per unit of time, mean length of pauses, and mean length of runs. Test data measurements were divided into 7 classifications: three proficiency levels of read speech plus a combined measurement of all three, and two proficiency levels of spontaneous speech plus a combined measurement of both.

Correlations that were found between human ratings and machine measurements in read speech were almost halved when spontaneous speech was used, but the correlations were still relatively strong. A drop in the correlations between machine scores and the human ratings for the high proficiency spontaneous speakers was attributed to the more difficult nature of the high proficiency material. The conclusion was that the optimal predictors of proficiency for read speech and spontaneous speech were different. In the case of read speech, the rate at which sounds were articulated and the frequency of pauses were very strongly related. In spontaneous speech, they found that the mean length of the runs between pauses was a better predictor of pronunciation quality. Additional analysis comparing the rate of errors between read and spontaneous speech revealed the surprising result that the phonetic errors of substitution and deletion were more prevalent in read speech than in spontaneous speech [56]. The authors hypothesize that this may be due to interference of the orthographic representation of the language and the student's understanding of the writing system.

Neumeyer et al. [173] investigated the evaluation of French as spoken by Americans. In these studies, the researchers collected read and spontaneous speech samples from 100 native French speakers and 100 Americans. They investigated four separate methods for scoring pronunciation at two levels: the sentence level and the speaker level. Correlations were computed between various machine scores and human ratings, which included HMM log-likelihood, segment classification, segment duration, and timing scores.

Initially, they found that the HMM scores did not correlate well with human expert pronunciation ratings on a Likert scale from 1 to 5 (1 was unintelligible, 5 was native-like). In fact, all of the scores, except for those based on timing, resulted in what they felt were unacceptable correlations at both the sentential level and the speaker level. They later improved the speaker level correlation of the HMM based scores by using the average of the log-posterior probability scores instead of the log-likelihood scores [74].

In other experiments, the researchers concentrated on sentential and speaker level pronunciation evaluation [202, 77, 75] using scores for specific phones. Additional methodology was introduced for detecting mispronunciation in which they compared a log-posterior probability from pure native models method with a dual model approach in which one phone model represented the correct pronunciation and the other represented the incorrect pronunciation.

Rhee and Park describe a system that makes use of parallel native and non-native models to assign grades to student utterances at the sentential level [181]. SpeechRater™is a program for rating the *Test of English as a Foreign Language* (TOEFL) iBT Practice Online product that also uses native and non-native models to generate features that are later used to score a speaker's overall perceived fluency [249, 250]. The authors found that the machine was able to assess a student's style or manner of delivery, even if recognition accuracy was not good. A system for evaluating spontaneous non-native Greek speech was developed using parallel native and non-native models [164]. The authors demonstrated that a system using parallel models outperformed a system using a single set of native models for evaluation.

The research cited above utilized many of the same features, such as duration, rate of speech, confidence scores, log-likelihood, and log-posteriors from HMM lattices to create regression functions to score speech. Research by Minematsu et al. takes a fundamentally different approach by modeling the pronunciation of sounds as distributions in frequency space relative to the other sound distributions in the language [156]. This was conducted in the spirit of work by Jakobson [107] who argued that the study of the sounds of a language must consider the structure of the sound system as a whole.

The structure defined by Minematsu et al. was then used to define a distortion metric that measured the difference between the phonetic structures of two populations of speakers, native American English speakers and Japanese learners of English [155]. This distortion metric was found to correlate with assessments of pronunciation proficiency [7, 157, 218], and this correlation held even when the non-native speech model was compared against multiple models of native speech (representing more than one teacher) [219].

The authors in [34] combine scores derived from HMM log-probabilities and *Gaussian Mixture Model* (GMM) scores by using a non-linear regression to mimic the scoring function of a human rater on non-native Mandarin speech. In this research, the log-probabilities are not used directly in the scoring function; rather, the log-probabilities are used to rank order the correct syllable against 410 other syllables in the Chinese language. The rank of the syllable is then used to compute a syllable score. The GMM scores are used in a similar way. A non-linear regression is used to optimize several parameters to combine these scores into one that mimics a human rater.

An approach described in [83] used the log-posterior probabilities from forced alignment with HMM to classify the quality of syllables using *Support Vector Machines* (SVMs). The classification results over a large number of syllables produce a final score of speaker pronunciation ability. This score is correlated with the 普通话水平考试 (putonghua shuiping kaoshi, PSK) corpus scores, which is a corpus of Chinese speakers from different dialect backgrounds.

Another example of a scoring method that does not make explicit use of HMM derived features is found in [124]. The authors found positive correlation between measures of pruned syllables per second, the ratio of the difference between total number of syllables and unnecessary syllables to total duration, and the ratio of unaccented syllables to accented syllables. A unique aspect to this study is that the authors were careful to gather human ratings from teachers who had been specifically trained in the Common European Framework of Reference [177] for assessing pronunciation. This included many specific evaluation items of loudness, sound pitch, quality of vowels, quality of consonants, epenthesis, elision, word stress, sentence stress, rhythm, intonation, speech rate, fluency, place of pause, and frequency of pause.

A.4.2 Pinpoint Error Detection

Pinpoint error detection is the identification of specific instances of pronunciation mistakes. Most modern pronunciation evaluation systems use log-posterior probability or log-likelihood scores produced by HMMs to evaluate foreign speech. These are then used to select word or subword units (syllables or phones) as mispronounced for later feedback to the student.

Word and phone level human assessments were found to be correlated with parallel HMMs trained on native and non-native speech [86, 210]. Posterior probabilities, followed by log-likelihood scores, were found to be the most highly correlated with human assessments of pronunciation quality[122]. Interestingly, the authors found that measurements of duration were found to be almost uncorrelated with assessments of individual phone quality.

This is in contrast to work in the previous section that found temporal based measurements to be highly correlated with overall assessment of speaker pronunciation.

The `FLUENCY` project is one of the earliest examples of a system that was able to detect pronunciation problems at the phonetic and prosodic levels [66]. CMU's SPHINX-II [104] speech recognition system was used to accurately measure prosodic information and detect phone errors from speech spoken by non-native speakers of English with French, German, Hebrew, Hindi, Italian, Mandarin, Portuguese, Russian, and Spanish as the native languages [65, 63].

This research was used to create a prototype language tutor [64] that was based on 5 principles articulated by [120]: production of large quantities of speech, reception of relevant corrective feedback, exposure to many examples of native speech, early emphasis on prosodic factors, and feeling of ease in learning environment. A key part of the system was the use of elicitation techniques in order to predict sentences that could be used for forced alignment recognition, in contrast to other systems, such as [224], which use completely scripted dialogues in their lessons.

Similarly, [111] examined the ability of HMMs to detect mispronunciations. In this study, tolerance levels were established for the scores of native speakers. When a non-native speaker produced a phone which generated a score that was at least one standard deviation away from the mean, feedback was given in the form of an illustrative diagram of proper articulation spots. HMMs were used by [118] to evaluate foreign speakers of Japanese on phonetic quality, but only for the quality of Japanese *tokushuhaku* (phones contrasted only by duration). Another system was implemented [119] to detect phone insertion, deletion and substitution using parallel phone models.

Witt et al. [239, 240] used HMM models to define a *Goodness of Pronunciation* (GOP) score, which was based on the log-likelihood of each phone segment in an HMM lattice, normalized by the number of frames in the segment. Phone dependent thresholds were defined to indicate the presence of a mispronunciation. These were empirically derived based on hand analysis. Using results from forced alignment recognition, the most common substitution errors were discovered and the phone models augmented to allow for additional paths through the lattice during decoding. An evaluation of GOP [117] compared thresholds optimized for either artificially produced errors derived from linguistic knowledge or real errors, and found no significant difference in the performance of the algorithm. This was important to the authors as it validated the use of artificial errors. Speaker dependent phone thresholds also yielded slightly better performance.

Similar to Wohlert's work, [50] used template-based discrete word recognition to evaluate learners of Spanish and Mandarin Chinese. A segmental analysis was performed to tabulate pronunciation errors for specific phones. These were then used to create and a system for weighting the importance of various errors. Eventually, a game-like interface was added [49] to provide feedback on pronunciation exercises. An interesting aspect of this research is the comparison of HMM based recognition with the template method. The authors found that, while the HMM recognizer was better at overall recognition accuracy, the template recognizer was better at distinguishing between minimal pairs.

An approach in Kim et al. [121] combined the results of a forced-alignment of accented English spoken by Korean English language learners, with the hand phonetic transcriptions of an expert phonetician. A detailed phonological analysis was performed to obtain a set of augmentation rules that modeled common pronunciation phenomena exhibited by the students. These rules tagged phonetic mispronunciations in an utterances and triggered feedback messages for the students. This approach was later extended by Harrison et al [93].

A CAPT that is too harsh on a student is likely to leave them feeling frustrated and dissatisfied with the system. Achieving native-like pronunciation is probably an unrealistic goal, especially with older students, so some research tries to identify high priority phones that should be assessed and corrected. In [171], a data driven approach was introduced to establish priorities for certain segmental errors. This helped establish which phones were (1) mispronounced often or (2) resulted in misunderstanding or unintelligibility. In [223], these results were used to identify three of the phones commonly found to be mispronounced by non-native speakers. Classifiers were trained for these phones to decide if they were acceptable or not, using features selected through an analysis of the difference between native and non-native productions.

A novel approach by the authors in [179, 180] combined the the frame log-posterior probability, phone log-posterior probability, and formant classification score derived from image feature extraction using the Gabor function to grade vowel quality in Mandarin spoken by Hong Kong residents. Three techniques were experimented with to combine the scores: linear regression to approximate a human rating, joint probability estimation, and a neural network. The neural network using all three features achieved a 9.7% higher correlation with human graders than the baseline using only frame-based log-posterior probabilities.

Finally, SVMs with linear kernels were used to detect phone-level mispronunciations in Mandarin Chinese using the log-likelihood ratios produced by an HMM lattice [235]. A phone-dependent ratio was set to balance precision and recall of mispronunciations. In contrast to most other HMM based methods which use GMMs to model phone pronunciations, this research used a model called a *Pronunciation Space Model* (PSM). The authors were motivated by the observation that many phone substitutions are not complete substitutions of one phone for another, but are substitutions of a partially changed phone for a sound that may not appear in the target language.

A.4.3 Pronunciation Feedback

The techniques for pronunciation feedback can be rougly divided into six forms: explicit correction, recast, elicitation, meta-linguistic feedback, clarification request, and repetition [138]. The effectiveness of methods for providing feedback is a topic of active research. It is often a temptation for researchers on the technical side of the problem to create systems based around new technologies without consideration for pedagogical requirements in foreign language learning. Automatic CAPT systems occupy especially treacherous ground because of the novelty of the technology and because of the constant change in capabilities of computer systems.

In surveys of existing CALL systems, Neri et al. [167, 169] characterized this situation as ``technology push or demand pull," and concluded that while there are severe pedagogical deficiencies in many available CALL systems, CALL with ASR can be employed effectively as long some principles are adhered to.

Based on an extensive literature review, they concluded that errors to be addressed by

CAPT systems should be those that are frequent, persistent, perceptually important, and reliably detected with automatic techniques [170]. Their research also suggested that a system should not overwhelm the students with too many corrections and should provide corrections in a timely manner. Additionally, some researchers suggest that telling a student that the speech they have produce is incorrect when it is, in fact, correct (a false positive), is more detrimental to learning than simply letting minor errors slide [8].

Early examples of explicit pronunciation feedback were oscilloscope and spectrogram displays [229, 116, 54, 174] from the 1960s. The intuition was that, if the student could both see and hear a native speaker's voice, they could imitate the speech by attempting to match the display for their own speech with that of their teacher. These systems required the presence of a teacher.

In the SPELL system [97], a graphical representation of the vowel space was presented to the student. When students completed exercises, the ideal placement for a vowel in the vowel space was shown along with the student's actual pronunciation. A similar system was developed to teach students the correct articulatory motions of the tongue for Swedish vowels [238]. The target vowel was displayed in the space, the student would practice vowel production by altering their voice in real-time to move a ball representing their speech onto the target ball. The researchers timed the ability of Swedish and international students to move the student ball onto the target ball and found that international students improved their times between two separate sessions.

Video games are another method for providing pronunciation feedback. In [4], a student receives feedback in the form of a video game. A simple car driving game indicates to the student the quality of their pronunciation by how well the car remains in the center of a twisting and curving road.

Graphical representations of human heads provide pronunciation feedback by showing students the correct placement of tongue and lips in the mouth. For example, a web-based system for Japanese learners of English displayed static pictures of heads for sounds identified as incorrect by an HMM lattice [211].

Other systems try to reverse engineer the speech signal to display what the student's tongue, lips, and throat are actually doing during speech [9, 10]. An example of a talking head feedback system that operates in real-time is ARTICULA, a tool used for teaching Spanish vowels [199]. As students speak, the signal is reverse engineered to display a real-time graphical representation of articulator positions.

Another form of feedback used in pronunciation training is shadowing. In shadowing, a native voice is played to the students, who are expected to speak almost simultaneously along with the native speaker. Since a transcription is unavailable to the student, closer attention must be paid to pronunciation[99]. Positive correlations have been found between the *Test of English for International Communication* (TOEIC) scores of Japanese learners of English, the GOP scores, and the number of proficiently pronounced words [136, 137].

Simicry is another system for shadowing [237]. The authors conducted a comparative study of student reactions to a say-after exercise and a shadowing exercise. The authors found that, in a group of students who had performed both types of exercises, the students significantly preferred the say-after exercise to the shadowing exercise. A preliminary analysis of pre- and post-exercise data showed differences in individual performances, but no differences between the group who exclusively did the say-after exercise vs the group who exclusively did the shadowing exercise.

Another type of feedback that can be given to students is to repair the pronunciation mistakes using their own voice. This allows the student to hear constrasts in a voice with which they are intimately familiar: their own. Some research focuses on the relatively easier problem of converting the intonation of foreign accented speech by either modifying the fundamental frequencies, durations, or both of non-native speech segments.

In [110], the authors attempt to repair intonation structure while preserving phonetic quality through re-synthesis using a native f_0 contour. It is concluded that this re-synthesis for comparison playback helps students identify intonation errors, though the methodology for arriving at this conclusion is not mentioned. The technique in this research relies on a good understanding of the stress patterns of the languages in question (in this particular paper, American English and German) such that target intonation contours can be automatically generated by the linguistic rules of the language.

In [217], *Pitch Synchronous OverLap and Add* (PSOLA) [92, 162, 163] is used to repair the f_0 of non-native speech on isolated words and phrases. Reference pronunciations are provided by recorded teacher utterances or by *Kungliga Tekniska h ogskolan's* (KTH) text-to-speech system [30]. The re-synthesis of isolated words showed that the technique held promise, but there were issues with alignment between student speech and the reference speech.

Systems that allow for manual modification of the intonation of utterances operate on a student-centric premise. Practice utterances are spoken by the student, at which point an interface that allows for the interactive modification of the f_0 harmonic is displayed. In [25], ActiveX controls are developed to allow the use of signal editing functions in Win-Snoori [131]. In a similar vein, *WinPitch LTL* [146] provides students and instructors with an interactive environment with the principle that students who participate in the understanding of prosody will learn it better than those who merely receive instruction passively.

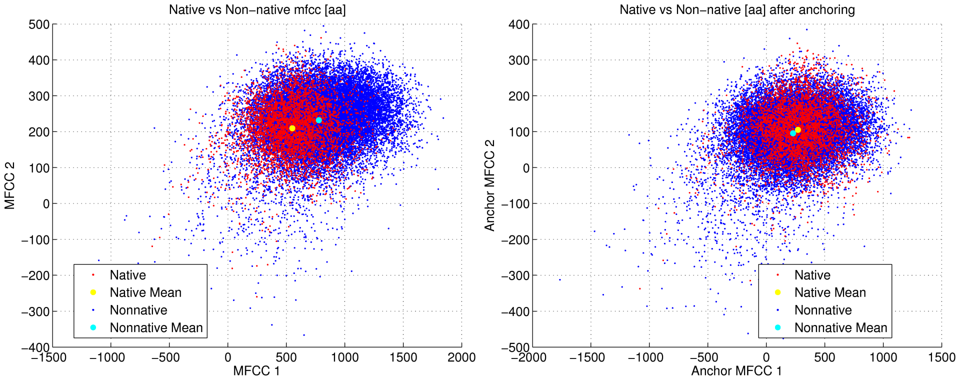
Some research in this area modify both the pitch and the phonetic aspects of speech. Felps et al. propose a system that modifies accented speech to have a more native-like quality [71]. Perceptual experiments confirmed that the technique made the speech seem more native-like while still preserving fundamental characteristics of the speakers' voices.

An interesting method for giving rhythmic feedback to students is *MusicSpeak* [232], a system created to address teaching stress-timed rhythm to students with a syllable-timed language background. In this research the authors developed a program that generated musical phrases according to the stress timing in a typed English sentence. Syllables occupied measures in a musical beat, with stress syllables as the first beat in a bar. Durations were modeled as different length notes in the phrase. Chinese students of English exhibited more variation in the rhythm of their English speech after using the system.

A similar style of feedback system was created to teach the correct pronunciation of Chinese lexical tones [191]. In this research, the author created a method for ``composing'' music using the four lexical tones of Mandarin Chinese. A music database was combined with instrument notes played at the relative frequency heights of the tones, plus a tone 3 modified through tone-sandhi. The system could produce feedback in the form of speech only, music only, or speech and music combined. In a comparison, the authors found significant differences in the use of one method over another.

Appendix B

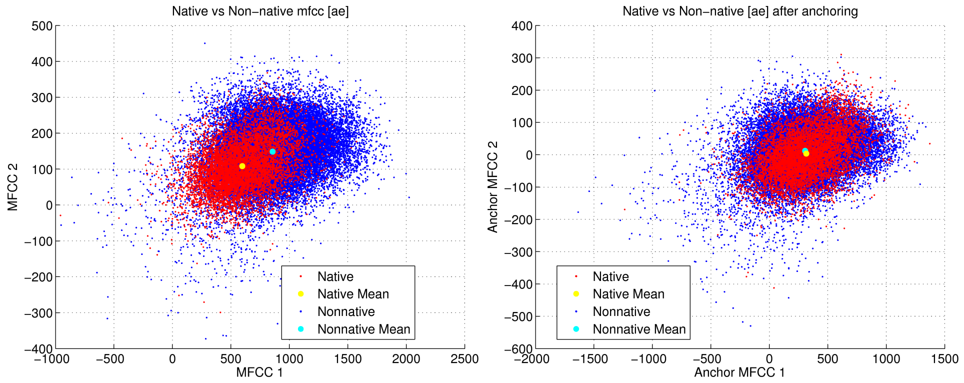
Comprehensive Listing of Anchoring Examples



(a) MFCCs

(b) /ə/-normalized

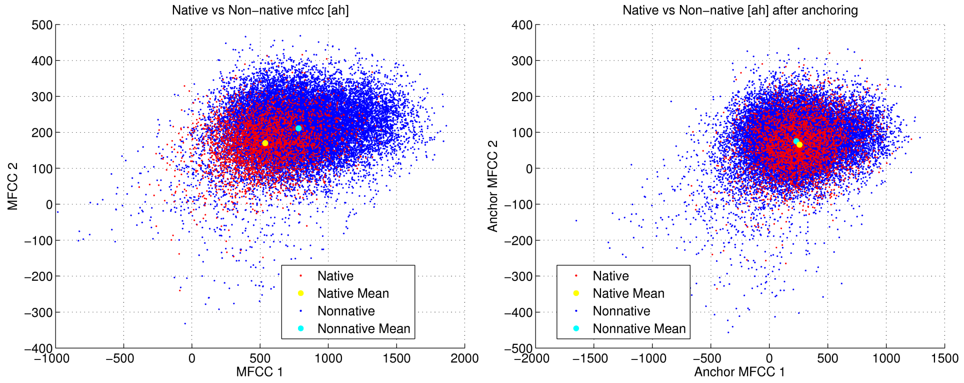
Figure B-1: Distributions of the first two dimensions of the feature vectors for /ɑ/ [aa] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

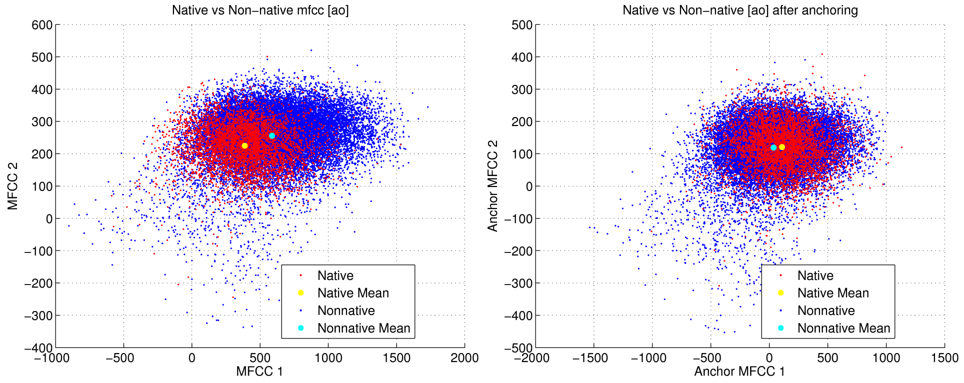
Figure B-2: Distributions of the first two dimensions of the feature vectors for /æ/ [ae] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

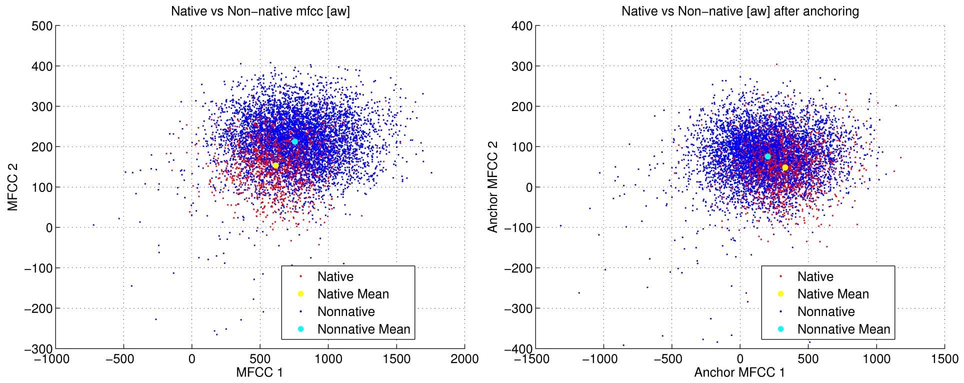
Figure B-3: Distributions of the first two dimensions of the feature vectors for /2/ [ah] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

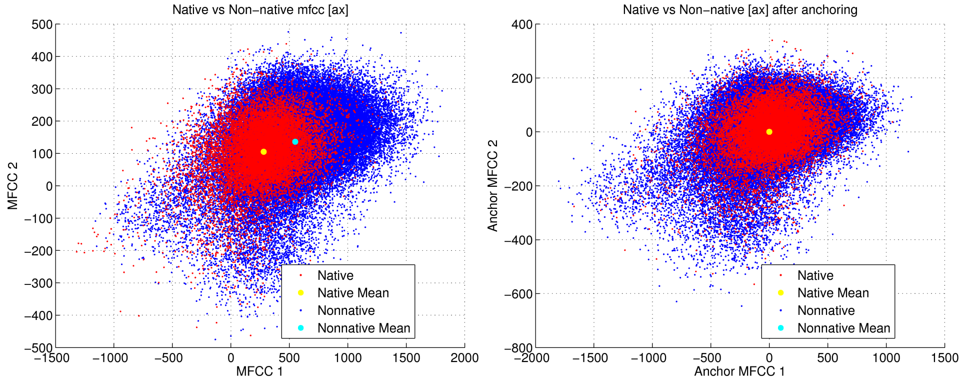
Figure B-4: Distributions of the first two dimensions of the feature vectors for /ɔ/ [ao] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

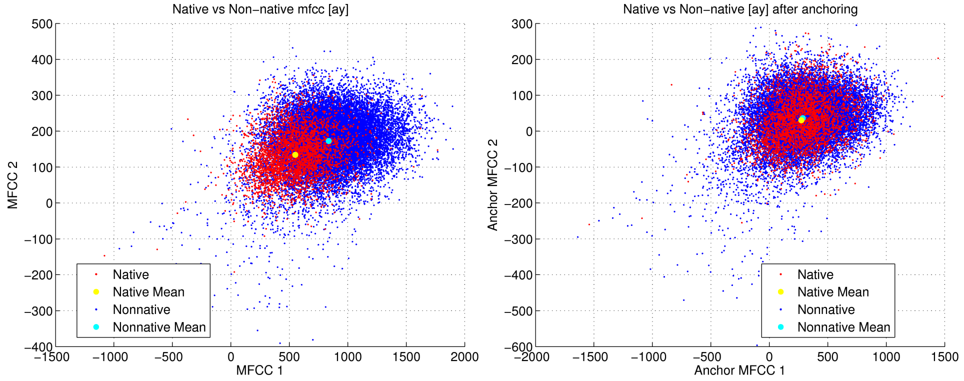
Figure B-5: Distributions of the first two dimensions of the feature vectors for /ɑ^{w}/ [aw] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

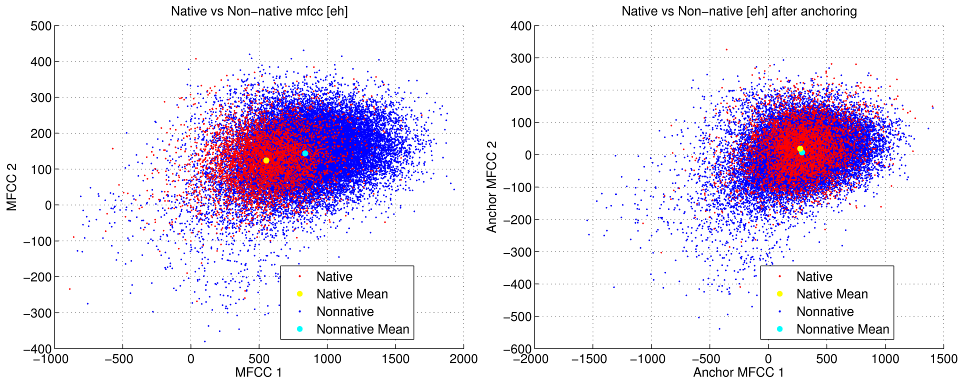
Figure B-6: Distributions of the first two dimensions of the feature vectors for /ə/ [ax] spoken by native and non-native speakers.



(a) MFCCs

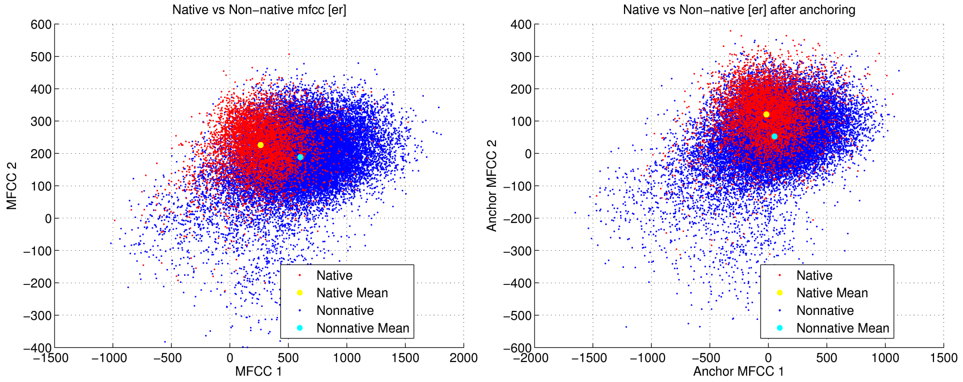
(b) /ə/-normalized

Figure B-7: Distributions of the first two dimensions of the feature vectors for /ɑ^{y}/ [ay] spoken by native and non-native speakers.



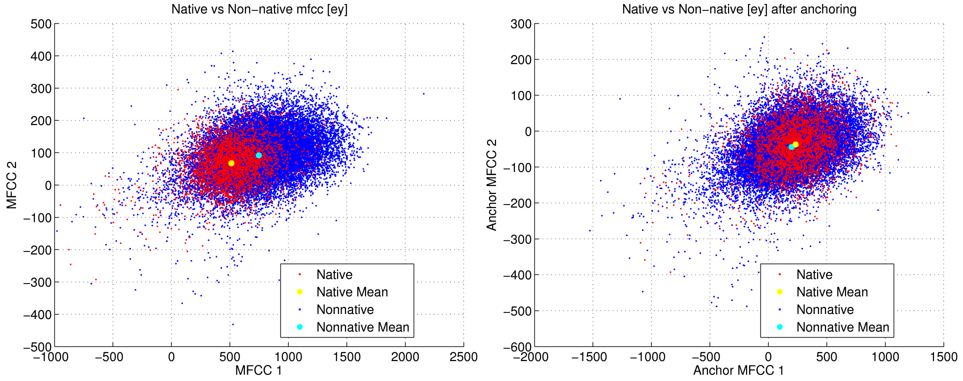
(a) MFCCs (b) /ə/-normalized

Figure B-8: Distributions of the first two dimensions of the feature vectors for /ε/ [eh] spoken by native and non-native speakers.



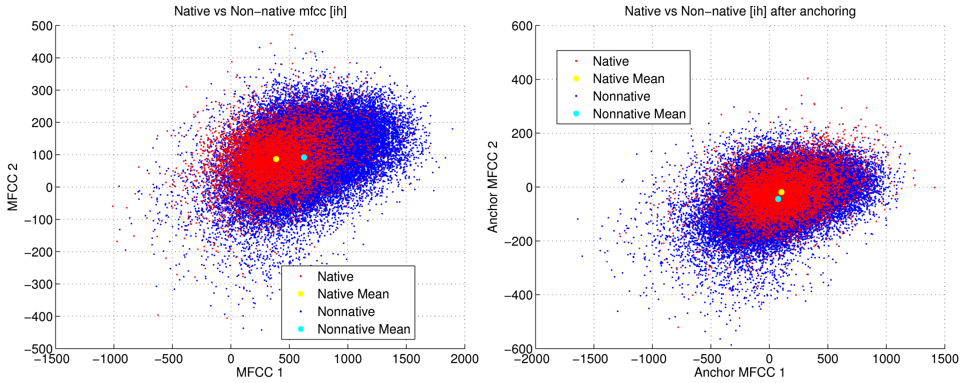
(a) MFCCs (b) /ə/-normalized

Figure B-9: Distributions of the first two dimensions of the feature vectors for /ɚ/ [er] spoken by native and non-native speakers.



(a) MFCCs (b) /ə/-normalized

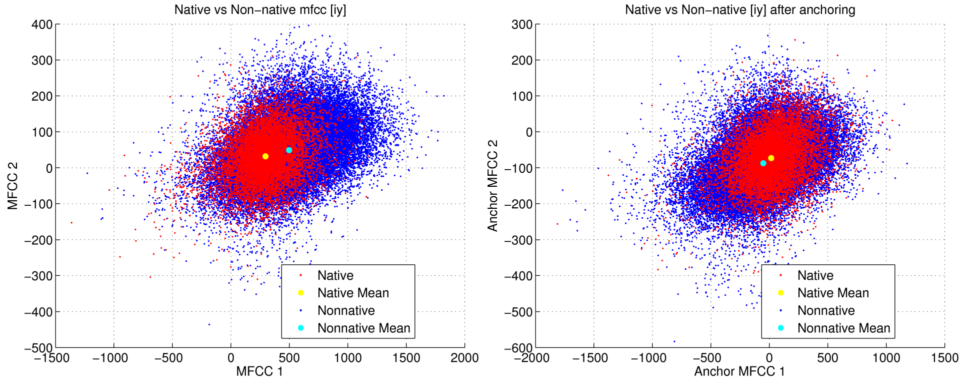
Figure B-10: Distributions of the first two dimensions of the feature vectors for /e/ [ey] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

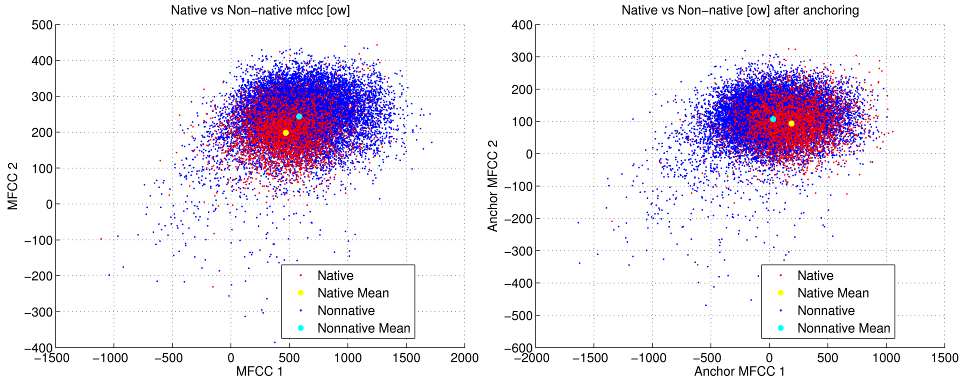
Figure B-11: Distributions of the first two dimensions of the feature vectors for /ɪ/ [ih] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

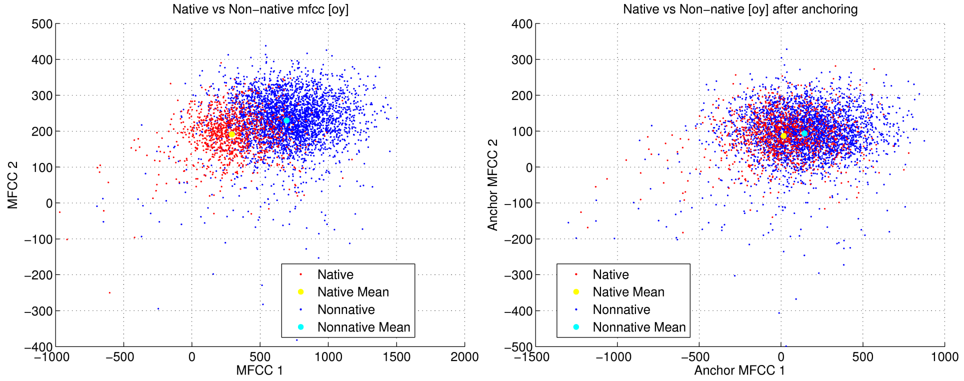
Figure B-12: Distributions of the first two dimensions of the feature vectors for /i/ [iy] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

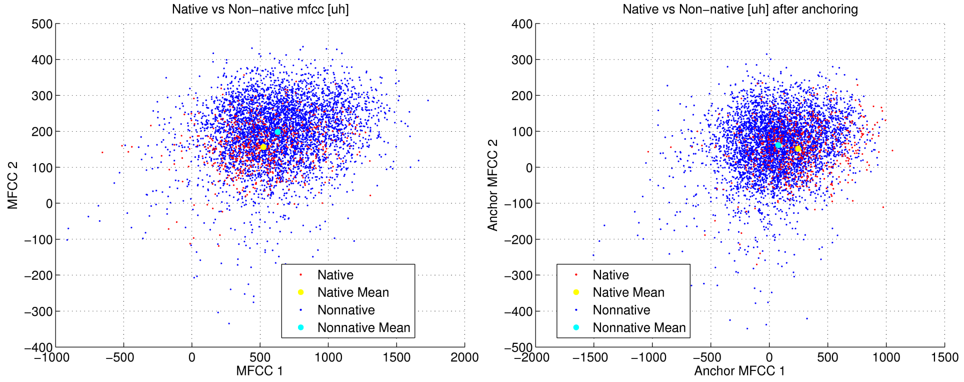
Figure B-13: Distributions of the first two dimensions of the feature vectors for /o/ [ow] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

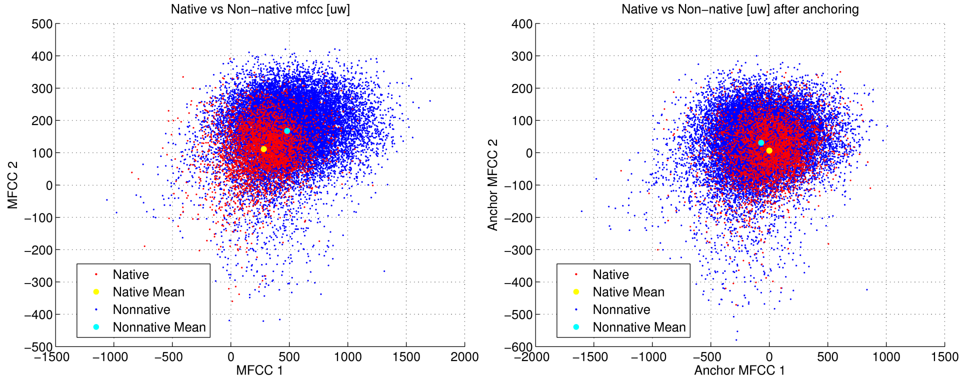
Figure B-14: Distributions of the first two dimensions of the feature vectors for /ɔ^{y}/ [oy] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

Figure B-15: Distributions of the first two dimensions of the feature vectors for /ʊ/ [uh] spoken by native and non-native speakers.



(a) MFCCs

(b) /ə/-normalized

Figure B-16: Distributions of the first two dimensions of the feature vectors for /u/ [uw] spoken by native and non-native speakers.

Appendix C

Decision Tree for $\overline{C}$-anchor feature source

```
t_score_nn <= 0.638435
|   div_t_t_delta <= 0.088494
|   |   div_t_nn_nn_nn <= -0.700277
|   |   |   kldiv_t_nn_delta <= 0.16964
|   |   |   |   div_nn_t_delta <= -0.276612: bad (71.55/3.05)
|   |   |   |   div_nn_t_delta > -0.276612
|   |   |   |   |   div_nn_n_nn_nn <= -0.919094: bad (36.0/8.13)
|   |   |   |   |   div_nn_n_nn_nn > -0.919094: good (3.05)
|   |   |   kldiv_t_nn_delta > 0.16964: good (10.89/1.74)
|   |   div_t_nn_nn_nn > -0.700277
|   |   |   kldiv_t_nn_delta <= -0.3397
|   |   |   |   kldiv_t_nn_n_n <= -0.147737: good (2.03)
|   |   |   |   kldiv_t_nn_n_n > -0.147737
|   |   |   |   |   kldiv_n_t_delta <= -0.112827: good (5.81/1.74)
|   |   |   |   |   kldiv_n_t_delta > -0.112827: bad (5.81)
|   |   |   kldiv_t_nn_delta > -0.3397
|   |   |   |   t_score_n <= 0.526404
|   |   |   |   |   kldiv_t_n_nn_n <= -0.249597: bad (4.64)
|   |   |   |   |   kldiv_t_n_nn_n > -0.249597: good (58.06/9.29)
|   |   |   |   t_score_n > 0.526404: good (81.28)
|   div_t_t_delta > 0.088494
|   |   t_score_nn <= 0.434373
|   |   |   div_t_t_delta <= 0.219971: good (79.25/4.06)
|   |   |   div_t_t_delta > 0.219971
|   |   |   |   div_t_n_n_n <= -0.974855: good (34.54)
|   |   |   |   div_t_n_n_n > -0.974855
|   |   |   |   |   n_result = -: good (3.63/0.58)
|   |   |   |   |   n_result = _
|   |   |   |   |   |   div_t_n_nn_n <= -0.807392: good (106.24/20.9)
|   |   |   |   |   |   div_t_n_nn_n > -0.807392
|   |   |   |   |   |   |   div_t_nn_nn_nn <= -0.875521
|   |   |   |   |   |   |   |   nn_result = -: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _
|   |   |   |   |   |   |   |   |   mfcc0 <= -0.415879: bad (5.23)
|   |   |   |   |   |   |   |   |   mfcc0 > -0.415879: good (8.85/1.74)
|   |   |   |   |   |   |   |   nn_result = _b1: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _b2: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _b3: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _b4: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _c1: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _c2: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _c3: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _c4: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _h1: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _h2: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _h3: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _l1: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _l2: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _l3: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _l4: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _n1: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _n2: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _n3: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _n4: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _n5: good (0.0)
|   |   |   |   |   |   |   |   nn_result = _n6: good (0.0)
|   |   |   |   |   |   |   |   nn_result = aa: bad (0.58)
|   |   |   |   |   |   |   |   nn_result = ae: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ah: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ah_fp: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ao: good (0.0)
|   |   |   |   |   |   |   |   nn_result = aw: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ax: good (0.0)
|   |   |   |   |   |   |   |   nn_result = axr: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ay: good (0.0)
|   |   |   |   |   |   |   |   nn_result = b: good (0.0)
|   |   |   |   |   |   |   |   nn_result = bcl: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ch: good (0.0)
|   |   |   |   |   |   |   |   nn_result = d: good (0.0)
|   |   |   |   |   |   |   |   nn_result = dcl: good (0.0)
|   |   |   |   |   |   |   |   nn_result = dh: good (0.0)
|   |   |   |   |   |   |   |   nn_result = dx: good (0.0)
|   |   |   |   |   |   |   |   nn_result = eh: good (0.0)
|   |   |   |   |   |   |   |   nn_result = el: good (0.0)
|   |   |   |   |   |   |   |   nn_result = em: good (0.0)
|   |   |   |   |   |   |   |   nn_result = en: good (0.0)
|   |   |   |   |   |   |   |   nn_result = epi: good (0.0)
|   |   |   |   |   |   |   |   nn_result = er: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ey: good (0.0)
|   |   |   |   |   |   |   |   nn_result = f: good (0.0)
|   |   |   |   |   |   |   |   nn_result = g: good (0.0)
|   |   |   |   |   |   |   |   nn_result = gcl: good (0.0)
|   |   |   |   |   |   |   |   nn_result = hh: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ih: good (0.0)
|   |   |   |   |   |   |   |   nn_result = iy: good (0.0)
|   |   |   |   |   |   |   |   nn_result = jh: good (0.0)
|   |   |   |   |   |   |   |   nn_result = k: good (0.0)
|   |   |   |   |   |   |   |   nn_result = kcl: good (0.0)
|   |   |   |   |   |   |   |   nn_result = l: good (2.61/0.58)
|   |   |   |   |   |   |   |   nn_result = m: good (0.0)
|   |   |   |   |   |   |   |   nn_result = n: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ng: good (0.0)
|   |   |   |   |   |   |   |   nn_result = not: good (0.0)
|   |   |   |   |   |   |   |   nn_result = ow: good (0.0)
|   |   |   |   |   |   |   |   nn_result = oy: good (0.0)
|   |   |   |   |   |   |   |   nn_result = p: good (0.0)
|   |   |   |   |   |   |   |   nn_result = pcl: good (0.0)
|   |   |   |   |   |   |   |   nn_result = r: good (0.0)
|   |   |   |   |   |   |   |   nn_result = s: good (0.0)
|   |   |   |   |   |   |   |   nn_result = sh: good (0.0)
|   |   |   |   |   |   |   |   nn_result = t: good (0.0)
|   |   |   |   |   |   |   |   nn_result = tcl: good (0.0)
|   |   |   |   |   |   |   |   nn_result = th: good (0.0)
|   |   |   |   |   |   |   |   nn_result = uh: good (0.0)
|   |   |   |   |   |   |   |   nn_result = uw: good (0.0)
|   |   |   |   |   |   |   |   nn_result = v: good (0.0)
|   |   |   |   |   |   |   |   nn_result = w: good (0.0)
|   |   |   |   |   |   |   |   nn_result = y: good (0.0)
|   |   |   |   |   |   |   |   nn_result = z: good (0.0)
|   |   |   |   |   |   |   |   nn_result = zh: good (0.0)
|   |   |   |   |   |   |   div_t_nn_nn_nn > -0.875521: bad (6.39)
|   |   |   |   |   n_result = _b1: good (0.0)
|   |   |   |   |   n_result = _b2: good (0.0)
|   |   |   |   |   n_result = _b3: good (0.0)
|   |   |   |   |   n_result = _b4: good (0.0)
|   |   |   |   |   n_result = _c1: good (0.0)
|   |   |   |   |   n_result = _c2: good (0.0)
|   |   |   |   |   n_result = _c3: good (0.0)
|   |   |   |   |   n_result = _c4: good (0.0)
|   |   |   |   |   n_result = _h1: good (0.0)
|   |   |   |   |   n_result = _h2: good (0.0)
|   |   |   |   |   n_result = _h3: good (0.0)
|   |   |   |   |   n_result = _l1: good (0.0)
|   |   |   |   |   n_result = _l2: good (0.0)
|   |   |   |   |   n_result = _l3: good (0.0)
|   |   |   |   |   n_result = _l4: good (0.0)
|   |   |   |   |   n_result = _n1: good (0.0)
|   |   |   |   |   n_result = _n2: good (0.0)
|   |   |   |   |   n_result = _n3: good (0.0)
|   |   |   |   |   n_result = _n4: good (0.0)
|   |   |   |   |   n_result = _n5: good (0.0)
|   |   |   |   |   n_result = _n6: good (0.0)
|   |   |   |   |   n_result = aa
|   |   |   |   |   |   div_t_n_n_n <= -0.818116: good (3.05)
|   |   |   |   |   |   div_t_n_n_n > -0.818116: bad (4.06)
|   |   |   |   |   n_result = ae
|   |   |   |   |   |   kldiv_nn_t_nn_n <= -0.732452: good (7.11)
|   |   |   |   |   |   kldiv_nn_t_nn_n > -0.732452: bad (6.39)
|   |   |   |   |   n_result = ah
|   |   |   |   |   |   div_t_nn_n_n <= -0.912046: good (3.05)
|   |   |   |   |   |   div_t_nn_n_n > -0.912046: bad (2.9)
|   |   |   |   |   n_result = ah_fp: good (0.0)
|   |   |   |   |   n_result = ao
|   |   |   |   |   |   kldiv_t_nn_nn_n <= -0.177437: good (8.13)
|   |   |   |   |   |   kldiv_t_nn_nn_n > -0.177437: bad (2.9)
|   |   |   |   |   n_result = aw: good (2.03)
|   |   |   |   |   n_result = ax
|   |   |   |   |   |   kldiv_n_nn_delta <= -0.109643
|   |   |   |   |   |   |   mfcc5 <= 0.352335: good (20.32/4.06)
|   |   |   |   |   |   |   mfcc5 > 0.352335: bad (3.48)
|   |   |   |   |   |   kldiv_n_nn_delta > -0.109643: bad (2.9)
|   |   |   |   |   n_result = axr: good (0.0)
|   |   |   |   |   n_result = ay
|   |   |   |   |   |   lpr_nn_n_nn_n <= -0.459214: good (4.06)
|   |   |   |   |   |   lpr_nn_n_nn_n > -0.459214: bad (4.06)
|   |   |   |   |   n_result = b: good (3.19/1.16)
|   |   |   |   |   n_result = bcl: good (1.02)
|   |   |   |   |   n_result = ch: good (0.0)
|   |   |   |   |   n_result = d
|   |   |   |   |   |   kldiv_t_n_n_n <= -0.331752: good (15.24)
|   |   |   |   |   |   kldiv_t_n_n_n > -0.331752: bad (2.32)
|   |   |   |   |   n_result = dcl: good (1.02)
|   |   |   |   |   n_result = dh: good (2.61/0.58)
|   |   |   |   |   n_result = dx: good (4.06)
|   |   |   |   |   n_result = eh: good (7.26/1.16)
|   |   |   |   |   n_result = el
|   |   |   |   |   |   mfcc8 <= 0.243649: bad (2.9)
|   |   |   |   |   |   mfcc8 > 0.243649: good (9.29/1.16)
|   |   |   |   |   n_result = em: good (0.0)
|   |   |   |   |   n_result = en: bad (0.58)
|   |   |   |   |   n_result = epi: good (5.08)
|   |   |   |   |   n_result = er: good (3.19/1.16)
|   |   |   |   |   n_result = ey
|   |   |   |   |   |   kldiv_t_nn_n_n <= -0.375819: good (2.03)
|   |   |   |   |   |   kldiv_t_nn_n_n > -0.375819: bad (4.06)
|   |   |   |   |   n_result = f: good (3.63/0.58)
|   |   |   |   |   n_result = g: good (6.1)
|   |   |   |   |   n_result = gcl: good (7.11)
|   |   |   |   |   n_result = hh: good (3.19/1.16)
|   |   |   |   |   n_result = ih
|   |   |   |   |   |   div_t_n_delta <= 0.538505: good (4.64/0.58)
|   |   |   |   |   |   div_t_n_delta > 0.538505: bad (4.64)
|   |   |   |   |   n_result = iy
|   |   |   |   |   |   mfcc5 <= -0.150369: good (6.68/0.58)
|   |   |   |   |   |   mfcc5 > -0.150369: bad (7.98/1.02)
|   |   |   |   |   n_result = jh: good (0.0)
|   |   |   |   |   n_result = k: good (3.63/0.58)
|   |   |   |   |   n_result = kcl: good (2.03)
|   |   |   |   |   n_result = l: good (25.84/11.61)
|   |   |   |   |   n_result = m: bad (1.16)
|   |   |   |   |   n_result = n: good (6.24/1.16)
|   |   |   |   |   n_result = ng: bad (1.16)
|   |   |   |   |   n_result = not: good (0.0)
|   |   |   |   |   n_result = ow: bad (8.56/1.02)
|   |   |   |   |   n_result = oy
|   |   |   |   |   |   lpr_nn_t_nn_nn <= -0.304711: good (6.1)
|   |   |   |   |   |   lpr_nn_t_nn_nn > -0.304711: bad (2.9)
|   |   |   |   |   n_result = p: good (2.03)
|   |   |   |   |   n_result = pcl: bad (0.58)
|   |   |   |   |   n_result = r
|   |   |   |   |   |   mfcc13 <= 0.239278: bad (3.48)
|   |   |   |   |   |   mfcc13 > 0.239278: good (4.06)
|   |   |   |   |   n_result = s: good (3.19/1.16)
|   |   |   |   |   n_result = sh: good (1.02)
|   |   |   |   |   n_result = t: good (3.77/1.74)
|   |   |   |   |   n_result = tcl: good (22.06/1.74)
|   |   |   |   |   n_result = th: bad (0.58)
|   |   |   |   |   n_result = uh: bad (3.92/1.02)
|   |   |   |   |   n_result = uw
|   |   |   |   |   |   div_t_n_n_n <= -0.903611: good (4.06)
|   |   |   |   |   |   div_t_n_n_n > -0.903611
|   |   |   |   |   |   |   kldiv_t_nn_delta <= -0.173946: good (2.03)
|   |   |   |   |   |   |   kldiv_t_nn_delta > -0.173946: bad (12.77)
|   |   |   |   |   n_result = v: good (0.0)
|   |   |   |   |   n_result = w: bad (7.4/1.02)
|   |   |   |   |   n_result = y: bad (2.76/1.02)
|   |   |   |   |   n_result = z: good (4.64/0.58)
|   |   |   |   |   n_result = zh: good (1.02)
|   |   t_score_nn > 0.434373: good (1575.66/145.14)
t_score_nn > 0.638435: good (28891.6/242.68)
```

Bibliography

[1] Bonnie Adair-Hauck, Laurel Willingham-McLain, and Bonnie Earnest Youngs. Evaluating the Integration of Technology and Second Language Learning. *CALICO journal*, 17(2):269--306, 2000.

[2] EN Adams, HW Morrison, and JM Reddy. Conversation with a Computer as a Technique of Language Instruction. *The Modern Language Journal*, 1968.

[3] John R Allen. Individualizing foreign language instruction with computers at Dartmouth. *Foreign Language Annals*, 5(3):348--349, 1972.

[4] A Álvarez, R Martínez, P Gómez, and J L Domínguez. A Signal Processing Technique for Speech Visualization. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 33--36. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[5] American Council on the Teachings of Foreign Languages, Hastings-on-Hudson, NY. *Proficiency Guidelines*, 1986.

[6] American Council on the Teachings of Foreign Languages, Hastings-on-Hudson, NY. *ACTFL Proficiency Guidelines*, 1999.

[7] Satoshi Asakawa, Nobuaki Minematsu, T Isei-Jaakkola, and Keikichi Hirose. Structural representation of the non-native pronunciations. In *Ninth European Conference on Speech Communication and Technology*, pages 165--168, 2005.

[8] L bachman. *Fundamental Considerations in language testing*. Oxford University Press, oxford applied linguistics edition, 1990.

[9] Pierre Badin, Gèrard Bailly, and Louis-Jean Boë. Towards the use of a Virtual Talking Head and of Speech Mapping tools for pronunciation training. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[10] Pierre Badin, Atef Ben Youssef, Gérard Bailly, Frédéric Elisei, Thomas Hueber, Houille Blanche, and F-Saint Martin. Visual articulatory feedback for phonetic correction in second language learning. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 2--5, Tokyo, Japan, 2010.

[11] Leonard E Baum and Ted Petrie. Statistical inference for probabilistic functions of finite state Markov chains. *The Annals of Mathematical Statistics*, 1966.

[12] Leonard E Baum, Ted Petrie, George Soules, and Norman Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The annals of mathematical ...*, 41(1):164--171, 1970.

[13] Roger T Bell. *An Introduction to Applied Linguistics: Approaches and Methods in Language Teaching*. St. Martin's Press, New York, 1981.

[14] J Bernstein, Michael Cohen, Hy Murveit, Dimitry Rtischev, and Mitchel Weintraub. Automatic evaluation and training in English pronunciation. In *Proceedings of ICSLP*, 1990.

[15] J Bernstein, J De Jong, D Pisoni, and Brent Townshend. Two experiments on automatic scoring of spoken language proficiency. *STILL2000*, 2000.

[16] Jared Bernstein. Automatic evaluation of English spoken by Japanese students. *The Journal of the Acoustical Society of America*, 86(S1):S77, 1989.

[17] Jared Bernstein. New Uses for Speech Technology in Language Education. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 175--177. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[18] Jared Bernstein, Isabella Barbier, Elizabeth Rosenfeld, and John De Jong. Development and Validation of an Automatic Spoken Spanish Test. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, pages 143--146, 2004. www.ordinate.com.

[19] Jared Bernstein, A. Najmi, and F. Ehsani. Subarashii: Encounters in Japanese spoken language education. *CALICO journal*, 16(3):361--384, 1999.

[20] Jared Bernstein, Masanori Suzuki, Jian Cheng, and Ulrike Pado. Evaluating Diglossic Aspects of an Automated Test of Spoken Modern Standard Arabic. In *SLaTE 2009 - 2009 ISCA Workshop on Speech and Language Technology in Education*, 2009.

[21] A Bhattacharyya. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*, 35:99--109, 1943.

[22] Michael Bloodgood and Chris Callison-Burch. Using mechanical turk to build machine translation evaluation sets. *... with Amazon's Mechanical Turk*, 2010.

[23] D Bohus and A Rudnicky. RavenClaw: Dialog management using hierarchical task decomposition and an expectation agenda. In *Proceedings of EUROSPEECH 2003*, Geneva, Switzerland, 2003.

[24] T. Bongaerts, C. Van Summeren, B. Planken, and E. Schils. Age and ultimate attainment in the pronunciation of a foreign language. *Studies in Second Language Acquisition*, 19(04):447--465, 1997.

[25] Anne Bonneau, Matthieu Camus, Yves Laprie, and Vincent Colotte. A computer-assisted learning of English prosody for French students. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, 2004.

[26] T.A. Boyle, W.F. Smith, and R.G. Eckert. Computer mediated testing: A branched program achievement test. *Modern Language Journal*, 60(8):428--440, 1976.

[27] Ann R. Bradlow and David B Pisoni. Training Japanese listeners to identify English /r/ and /l/: IV. Some effects of perceptual learning on speech production. *Journal of the Acoustical Society of America*, 101(4):2299--2310, 1997.

[28] H Douglas Brown. *Principles of Language Learning and Teaching*. Prentice Hall Regents, 3rd editio edition, 1994. ISBN: 0-13-191966-0.

[29] Chris Callison-Burch. Fast, cheap, and creative: evaluating translation quality using Amazon's Mechanical Turk. In *EMNLP '09: Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, August 2009.

[30] R Carlson, B Granström, and S Hunnicutt. Multilingual text-to-speech development and applications. In A W Ainsworth, editor, *Advances in speech, hearing and language processing*, pages 269--296. JAI Press, London, 1990.

[31] J B Carroll. The Prediction of Success in Intensive Foreign Language Training. In *Training and research in Education*, pages 87--136. University of Pittsburgh Press, Pittsburgh, PA, 1962.

[32] Chih-yu Chao, Stephanie Seneff, and Chao Wang. An Interactive Interpretation Game for Learning Chinese. In *Proceedings of ISCA ITRW SLaTE07*, Farmington, PA, 2007.

[33] C. Chaudron. Progress in Language Classroom Research: Evidence from The Modern Language Journal, 1916-2000. *The Modern Language Journal*, 85(1):57--76, 2001.

[34] Jiang-Chun Chen, Jyh-Shing Roger Jang, Jun-Yi Li, and Ming-Chun Wu. Automatic pronunciation assessment for Mandarin Chinese. *Multimedia and Expo, 2004. ICME '04. 2004 IEEE International Conference on*, 3:1979--1982 Vol.3, 2004.

[35] Sylvain Chevalier and Zhenhai Cao. Application and evaluation of speech technologies in language learning: experiments with the Saybot Player. In *Proceedings of Interspeech*, pages 2811--2814, 2008.

[36] Noam Chomsky. *Aspects of the Theory of Syntax*. The MIT press, 1965.

[37] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37--46, 1960.

[38] Corinna Cortes and V.N. Vapnik. Support-vector networks. *Machine Learning*, 20(3):273--297, 1995.

[39] Stephen Cox. Speaker normalization in the MFCC domain. In *Sixth International Conference on Spoken Language Processing*, pages 4--7, 2000.

[40] C. Cucchiarini, H. Strik, D Binnenpoorte, and L. Boves. Towards an Automatic Oral Proficiency Test for Dutch as a Second Language: Automatic Pronunciation Assessment in Read and Spontaneous Speech. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, 2000.

[41] C. Cucchiarini, H. Strik, and L. Boves. Automatic evaluation of Dutch pronunciation by using speech recognition technology. In *1997 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '97*, pages 622--629, 1997.

[42] C. Cucchiarini, H. Strik, and L. Boves. Using speech recognition technology to assess foreign speakers' pronunciation of Dutch. In *Proceedings of the Third International Symposium on the Acquisition of Second Language Speech: NEW SOUNDS 97*, 1997.

[43] Catia Cucchiarini, Helmer Strik, Diana Binnenpoorte, and Lou Boves. Pronunciation evaluation in read and spontaneous speech: A comparison between human ratings and automatic scores. In *Proceedings of the Fourth International Symposium on the Acquisition of Second-Language Speech*, pages 72--79. Citeseer, 2002.

[44] Catia Cucchiarini, Helmer Strik, and Lou Boves. Automatic Pronunciation Grading For Dutch. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 95--98. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[45] Catia Cucchiarini, Helmer Strik, and Lou Boves. Different aspects of expert pronunciation quality ratings and their relation to scores produced by speech recognition algorithms. *Speech Communication*, 30:109--119, 2000.

[46] Catia Cucchiarini, Helmer Strik, and Lou Boves. Quantitative assessment of second language learners' fluency by means of automatic speech recognition technology. *Journal of the Acoustical Society of America*, 107(2):989--999, 2000.

[47] Catia Cucchiarini, J. van Doremalen, and Helmer Strik. DISCO: Development and Integration of Speech technology into Courseware for language learning. In *Proceedings of Interspeech*, page 2791, Brisbane, Australia, 2008. Bonn, Germany: ISCA.

[48] Catia Cucchiarini, F.D. Wet, Helmer Strik, and Lou Boves. Assessment of Dutch pronunciation by means of automatic speech recognition technology. In *Fifth International Conference on Spoken Language Processing*, pages 2--5. Citeseer, 1998.

[49] Jonathan Dalby and Diane Kewley-Port. Explicit Pronunciation Training Using Automatic Speech Recognition Technology. *CALICO journal*, 16(3):425--445, 1999.

[50] Jonathan Dalby, Idane Kewley-Port, and Roy Sillings. Language-Specific Pronunciation Training Using the HearSay System. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 25--28. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[51] S.B. Davis and P Mermelstein. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4):357--366, 1980.

[52] Tracey M Derwing, Murray J Munro, and Grace Wiebe. Pronunciation Instruction for ``Fossilized'' Learners: Can it Help? *Applied Language Learning*, 8(2):217--235, 1997.

[53] Tracey M Derwing and Marian J Rossiter. The Effects of Pronunciation Instruction on the Accuracy, Fluency, and Complexity of L2 Accented Speech. *Applied Language Learning*, 13(1):1--17, 2003.

[54] F Destombes. The development and application of the IBM speech viewer. In A Brekelmans, Ben A.G. Elsendoorn, and Frans Coninx, editors, *Interactive Learning Technology for the Deaf*. Springer, 1993.

[55] Randy L Diehl. Acoustic and auditory phonetics: the adaptive design of speech sound systems. *Philosophical transactions of the Royal Society of London. Series B, Biological sciences*, 363(1493):965--978, 2008.

[56] Joost Van Doremalen, Catia Cucchiarini, and Helmer Strik. Phoneme Errors in Read and Spontaneous Non-Native Speech : Relevance for CAPT System Development. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 7--10, Tokyo, Japan, 2010.

[57] Z. Dörnyei. Motivation and motivating in the foreign language classroom. *Modern Language Journal*, pages 273--284, 1994.

[58] Z. Dörnyei and K. Csizér. Ten commandments for motivating language learners: Results of an empirical study. *Language Teaching Research*, 2(3):203, 1998.

[59] P Dunkel. The effectiveness of research on computer-assisted instruction and computer- assisted language learning. In P Dunkel, editor, *Computer-assisted language learning and testing*, pages 5--36. Newbury House, New York, 1991.

[60] F. Ehsani, J Bernstein, and O Todic. Subarashii: Japanese interactive spoken language education. In *Proceedings of EUROSPEECH 1997*, Rhodes, Greece, 1997.

[61] R Ellis. *Task-based language learning and teaching*. Oxford University Presss, 2003.

[62] Unreal Tournament, 2003.

[63] M Eskenazi. Using a Computer in Foreign Language Pronunciation Training: What Advantages? *CALICO journal*, 16(3):447--469, 1999.

[64] M Eskenazi. Using Automatic Speech Processing for Foreign Language Pronunciation Tutoring: Some Issues and a Prototype. *Language, Learning & Technology*, 2(2):62--76, 1999.

[65] M Eskenazi and S Hansma. The Fluency Pronunciation Trainer. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 77--80. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[66] Maxine Eskenazi. Detection of foreign speakers' pronunciation errors for second language training-preliminary results. In *Proceedings of ICSLP*. IEEE, 1996.

[67] Maxine Eskenazi. An overview of spoken language technology for education. *Speech Communication*, 51(10):832--844, 2009.

[68] G. Fant. Non-uniform vowel normalization. *Speech Trans. Lab. Q. Prog. Stat. Rep*, pages 2--3, 1975.

[69] Catia Cucchiarini Helmer Strik Lou Boves Febe de Wet. Using Likelihood Ratios To Perform Utterance Verification In Automatic Pronunciation Assessment. In *Proceedings of EUROSPEECH 1999*, pages 173--176, 1999.

[70] Uschi Felix. Analysing Recent CALL Effectiveness Research---Towards a Common Agenda. *Computer Assisted Language Learning*, 18(1-2):1--32, February 2005.

[71] Daniel Felps, Heather Bortfeld, and Ricardo Gutierrez-Osuna. Foreign accent conversion in computer assisted pronunciation training. *Speech Communication*, 51(10):920--932, 2009.

[72] James Emil Flege. Second-language learning: The Role of Subject and Phonetic Variables. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 1--8. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[73] Foreign Language Assessment Directory.

[74] H. Franco, L. Neumeyer, Yoon Kim, and O. Ronen. Automatic Pronunciation Scoring for Language Instruction. In *1997 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '97*, pages 1471--1474. IEEE Comput. Soc. Press, 1997.

[75] H. Franco, L. Neumeyer, M Ramos, and H Bratt. Automatic Detection of Phone-Level Mispronunciation for Language Learning. In *Proceedings of EUROSPEECH 1999*, 1999.

[76] Horacio Franco, Victor Abrash, Kristin Precoda, Harry Bratt, Ramana Rao, John Butzberger, Romain Rossier, and Federico Cesari. The SRI EduSpeak System: Recognition and Pronunciation Scoring for Language Learning. In *Proceedings of ESCA ETRW INSTiL 2000*, pages 123--128, Dundee, Scotland, 2000.

[77] Horacio Franco and Leonardo Neumeyer. Calibration of Machine Scores for Pronunciation Grading. In *Proceedings of ICSLP*, 1998.

[78] M. Gales, D. Pye, and P. Woodland. Variance compensation within the mllr framework for robust speech recognition and speaker adaptation. In *Proc. ICSLP '96*, volume 3, pages 1832--1835, Philadelphia, PA, USA, October 1996.

[79] David Galloway and Kristin Peterson-Bidoshi. The case for dynamic exercise systems in language learning. *Computer Assisted Language Learning*, 21(1):1--8, February 2008.

[80] R Gardner and W Lamber. Motivational variables in second language acquisition. *Canadian Journal of Psychology*, 13:266--272, 1959.

[81] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, D. S. Pallett, and N. L. Dahlgren. DARPA TIMIT acoustic-phonetic continuous speech corpus CD-ROM, 1993. National Institute of Standards and Technology, NISTIR 4930.

[82] N. Garrett. Technology in the service of language learning: Trends and issues. *Modern Language Journal*, 75(1):74--101, 1991.

[83] Fengpei Ge, Fuping Pan, Changliang Liu, Bin Dong, Shui-duen Chan, X. Zhu, and Y. Yan. An SVM-Based Mandarin Pronunciation Quality Assessment System. *The Sixth International Symposium on Neural Networks (ISNN 2009)*, pages 255--265, 2009.

[84] H Gish, M Krasner, W Russell, and J Wolf. Methods and experiments for text-independent speaker recognition over telephone channels. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '86*, pages 865--868, 1986.

[85] D Giuliani, M Gerosa, and F Brugnara. Speaker normalization through constrained MLLR based transforms. In *Eighth International Conference on Spoken Language Processing*, page 3, 2004.

[86] Simo M. A. Goddijn and Guus de Krom. Evaluation of second language learners' pronunciation using Hidden Markov Models. In *Proceedings of EUROSPEECH 1997*, pages 2331--2334, 1997.

[87] Manuela Gonz a lez Bueno. Pronunciation Teaching Component in SL/FL Education Programs: Training Teachers to Teach Pronunciation. *Applied Language Learning*, 12(2):133--146, 2001.

[88] Peter J M Groot. Computer Assisted Second Language Vocabulary Acquisition. *Language, Learning & Technology*, 4(1):60--81, 2000.

[89] Alexander Gruenstein, Ian Mcgraw, and Andrew Sutherland. A Self-Transcribing Speech Corpus : Collecting Continuous Speech with an Online Educational Game. In *SLaTE 2009 - 2009 ISCA Workshop on Speech and Language Technology in Education*, 2009.

[90] Florian H, Anton Batliner, Karl Weilhammer, and Elmar N. How Many Labellers? Modelling Inter-Labeller Agreement and System Performance for the Automatic Assessment of Non-Native Prosody. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 6--9, Tokyo, Japan, 2010.

[91] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. The WEKA Data Mining Software: An Update. *SIGKDD Explorations*, 11(1), 2009.

[92] C. Hamon, E. Moulines, and F. Charpentier. A diphone synthesis system based on time-domain prosodic modifications of speech. In *Proc. ICASSP '89*, pages 238--241, Glasgow, Scotland, May 1989.

[93] Alissa M Harrison, Wai-kit Lo, Xiao-jun Qian, Helen Meng, The Chinese, and Hong Kong. Implementation of an Extended Recognition Network for Mispronunciation Detection and Diagnosis in Computer-Assisted Pronunciation Training. In *SLaTE 2009 - 2009 ISCA Workshop on Speech and Language Technology in Education*, 2009.

[94] R.S. Hart. The Illinois PLATO foreign languages project. *CALICO journal*, 12(4):15--37, 1995.

[95] Valerie Hazan, Yoon Hyun Kim, and Phonetic Sciences. Can we predict who will benefit from computer-based phonetic training ? In *Second Language Studies: Acquisition, Learning, Education and Technology*, Tokyo, Japan, 2010.

[96] J Higgins. *Language, learners, and computers: Human intelligence and artificial unintelligence*. Longman, London, 1988.

[97] Steven Hiller, Edmund Rooney, John Laver, and Mervyn Jack. SPELL: An automated system for computer-aided pronunciation teaching. *Speech Communication*, 13(3-4):463--473, December 1993.

[98] F Hinofotis and K Bailey. American undergraduates' reactions to the cumminication skills of foreign teaching assistants. In J C Fisher, M A Clarke, and J Schacter, editors, *On TESOL '80*, pages 120--133, Washington, DC, 1980.

[99] T Hori. *Exploring Shadowing as a Method of English Pronunciation Training*. PhD thesis, Graduate School of Language Communication and Culture, Kwansei Gakuin University, 2008.

[100] Elaine K Horwitz, Michael B Horwitz, and Joann Cope. Foreign Language Classroom Anxiety. *The Modern Language Journal*, 70(2):132--152, 1986.

[101] J Jia S Hou and W Chen. Improving the CSIEC Project and Adapting It to the English Teaching and Learning in China. *ArXiv Computer Science e-prints*, 2006.

[102] Jeff Howe. *Crowdsourcing: Why the Power of the Crowd Is Driving the Future of Business*. Crown Business, 2008.

[103] Pei-Yun Hsueh, Prem Melville, and Vikas Sindhwani. Data quality from crowdsourcing: a study of annotation selection criteria. In *HLT '09: Proceedings of the NAACL HLT 2009 Workshop on Active Learning for Natural Language Processing*. Association for Computational Linguistics, June 2009.

[104] X. Huang, F. Alleva, M.-Y. Hwang, and R. Rosenfeld. An overview of the sphinx-ii speech recognition system. In *Proc. ARPA Human Language Technology Workshop '93*, pages 81--86, Princeton, NJ, March 1994. distributed as *Human Language Technology* by San Mateo, CA: Morgan Kaufmann Publishers.

[105] Philip Hubbard. A Survey of Unanswered Questions in CALL. *Computer Assisted Language Learning*, 16(2-3):141--154, July 2003.

[106] Gabriel Jacobs and Catherine Rodgers. Treacherous Allies: Foreign Language Grammar Checkers. *CALICO journal*, 16(4):509--530, 1999.

[107] Roman Jakobson and L.R. Waugh. *The sound shape of language*. Mouton de Gruyter, 1987.

[108] J Jia and Weichao Chen. Motivate the Learners to Practice English through Playing with Chatbot CSIEC. *Technologies for E-Learning and Digital Entertainment*, 2008.

[109] Jiyou Jia. CSIEC (Computer Simulator in Educational Communication): A Virtual Context-Adaptive Chatting Partner for Foreign Language Learners. In *Proceedings of ICALT 04*, pages 690--692. IEEE, 2004.

[110] Matthias Jilka and Gregor Möhler. Intonational Foreign Accent: Speech Technology and Foreign Language Testing. In *Proceedings of ESCA Workshop on Speech*

Technology in Language Learning, pages 115--118. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[111] C H Jo, T Kawahara, S Doshita, and M Dantsuji. Automatic Pronunciation Error Detection and Guidance for Foreign Language Learning. In *Proceedings of ICSLP*, pages 2639--2642, 1998.

[112] Lewis Johnson, Carole R Beal, Anna Fowles-Winkler, Ursula Lauper, Stacy Marsella, Shrikanth Narayanan, Dimitra Papachristou, and Hannes Vilhj a lmsson. Tactical Language Training System: An Interim Report. In *Intelligent Tutoring Systems*, pages 336--345, 2004.

[113] W. Johnson and A Valente. Tactical language and culture training systems: using artificial intelligence to teach foreign languages and cultures. In *Proceedings of IAAI*, 2008.

[114] W.L. Johnson, S. Marsella, and H. Vilhjálmsson. The DARWARS tactical language training system. In *Proceedings of I/ITSEC*, 2004.

[115] W.L. Johnson, Stacy Marsella, N. Mote, H. Viljh a lmsson, S. Narayanan, and S. Choi. Tactical Language Training System: Supporting the rapid acquisition of foreign language and cultural skills. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*. Citeseer, 2004.

[116] Daniel N Kalikow and John A Swets. Experiments with Computer-Controlled Displays in Second-Language Learning. *IEEE Transactions on Audio and Electroacoustics*, 20(1):23--28, 1972.

[117] Sandra Kanters, Catia Cucchiarini, and Helmer Strik. The Goodness of Pronunciation Algorithm : a Detailed Performance Study. In *SLaTE 2009 - 2009 ISCA Workshop on Speech and Language Technology in Education*, pages 2--5, 2009.

[118] Goh Kawai and Keikichi Hirose. A CALL System Using Speech Recognition to Teach the Pronunciation of Japanese Tokushumaku. In *Proceedings of ESCA Work-*

shop on Speech Technology in Language Learning, pages 73--76. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[119] Goh Kawai and Keikichi Hirose. A method for measuring the intelligibility and nonnativeness of phone quality in foreign language pronunciation training. In *Proceedings of ICSLP*, 1998.

[120] J Kenworthy. *Teaching English Pronunciation*. Longman, New York, 1995.

[121] Jong-mi Kim, Chao Wang, Mitchell Peabody, and Stephanie Seneff. An interactive English pronunciation dictionary for Korean learners. In *Proceedings of ICSLP*, 2004.

[122] Y. Kim, H. Franco, and L. Neumeyer. Automatic pronunciation scoring of specific phone segments for language instruction. In *Fifth European Conference on Speech Communication and Technology*. Citeseer, 1997.

[123] Yoon Hyun Kim and Jung-oh Kim. Attention to Critical Acoustic Features for L2 Phonemic Identification and its Implication on L2 Perceptual Training Interdisciplinary Program in Cognitive Science , Seoul National University , Seoul , Korea Department of Psychology , Seoul National Unive. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 1--4, Tokyo, Japan, 2010.

[124] Yusuke Kondo, Eiichiro Tsutsui, and Michiko Nakano. Bridging the Gap between L2 Research and Classroom Practice (2): Evaluation of Automatic Scoring System for L2 Speech. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 2--5, Tokyo, Japan, 2010.

[125] C. Kramsch, D. Morgenstern, and J. Murray. An Overview of the Mit Athena Language Learning Project. *CALICO journal*, 2(4):31--34, 1985.

[126] S.D. Krashen and T.D. Terrell. *The Natural Approach: Language Acquisition in the classroom*. Language Teaching methodology series. Phoenix ELT, 1988.

[127] S Kullback and R A Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79--86, March 1951.

[128] S.V.B. Kumar and S. Umesh. Non-Uniform Speaker Normalization Using Frequency-Dependent Scaling Function. In *Proceedings of ICSLP*, Bangalore, 2004.

[129] Stephen A Kunath and Steven H Weinberger. The wisdom of the crowd's ear: speech accent rating and annotation with Amazon Mechanical Turk. In *CSLDAMT '10: Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*. Association for Computational Linguistics, June 2010.

[130] J Kuo and X Jiang. Assessing the assessments: The OPI and the SOPI. *Foreign Language Annals*, 30(4):503--512, 1997.

[131] Y Laprie. Snorri, a software for speech sciences. *MATISSE*, 1999.

[132] P LaReau and E Vockell. *The computer in the foreign language curriculum*. Mitchell Publishing, Inc, Santa Cruz, CA, 1989.

[133] Tien-Lok Jonathan Lau. SLLS: An Online Conversational Spoken Language Learning System. Master's thesis, Massachusetts Institute of Technology, 2003.

[134] Jonathan Leather and Allan James. Second Language Speech. In C Doughty and M Long, editors, *Handbook of second language acquisition*. Blackwell, Oxford, 2 edition, 2002.

[135] Jean W LeLoup and Robert Ponterio. On The Net: Interactive and Multimedia Techniques in ONline Language Lessons: A Sampler. *Language, Learning & Technology*, 7(3):4--17, 2003.

[136] Dean Luo, Naoya Shimomura, Nobuaki Minematsu, Yutaka Yamauchi, and Keikichi Hirose. Automatic pronunciation evaluation of language learners' utterances generated through shadowing. In *Proceedings of Interspeech*, pages 2807--2810, 2008.

[137] Dean Luo, Yutaka Yamauchi, and Nobuaki Minematsu. Speech Analysis for Automatic Evaluation of Shadowing. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 1--4, Tokyo, Japan, 2010.

[138] R Lyster and L Ranta. Corrective feedback and learner uptake. *Studies in Second Language Acquisition*, 19:37--66, 1997.

[139] P. D. MacIntyre and R. C. Gardner. Anxiety and second language learning: toward a theoretical clarification. *Language Learning*, 32:251--275, 1989.

[140] P. D. MacIntyre and R. C. Gardner. Investigating language class anxiety using the focused essay technique. *The Modern Language Journal*, 75:290--304, 1991.

[141] P. D. MacIntyre and R. C. Gardner. Language Anxiety: Its relationship to other anxieties and to processing in native and second languages. *Language Learning*, 41:513--534, 1991.

[142] P. D. MacIntyre and R. C. Gardner. Methods and results in the study of foreign language anxiety: A review of the literature. *Language Learning*, 41:85--117, 1991.

[143] P. D. MacIntyre and R. C. Gardner. The effects of induced anxiety on three stages of cognitive processing in computerized vocabulary learning. *Studies in Second Language Acquisition*, 16:1--17, 1994.

[144] P. D. MacIntyre and R. C. Gardner. The subtle effects of language anxiety on congnitive processing in the second language. *Language Learning*, 44:283--305, 1994.

[145] V Malabonga. Computers and language testing: The Computerized Oral Proficiency Interview. *Language Testing Update*, 24:29, 1998.

[146] Philippe Martin. WinPitch LTL II, a Multimodel Pronunciation Software. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, 2004.

[147] Jr. Martin J Petersen. *An Evaluation of Voxbox, A Computer-based Voice-interactive Language Learning System for Teaching English as a Second Language*. PhD thesis, United States International University, San Diego, CA, 1990. Doctor of Education.

[148] P H Matthews. *Concise Dictionary of Linguistics*. Oxford University Press, 1997. ISBN: 0-19-280008-6.

[149] Ian Mcgraw and Stephanie Seneff. Immersive second language acquisition in narrow domains: a prototype ISLAND dialogue system. In *Proceedings of ISCA ITRW SLaTE07*, Farmington, PA, 2007.

[150] Ian Mcgraw and Stephanie Seneff. Speech-enabled Card Games for Language Learners. In *Proceedings of AAAI*, Chicago, IL, July 2008.

[151] Ian Mcgraw, Brandon Yoshimoto, and Stephanie Seneff. Speech-enabled Card Games for Incidental Vocabulary Acquisition in a Foreign Language. *Speech Communication*, 2008.

[152] H Meng, Y Y Lo, L Wang, and W.Y. Lau. Deriving salient learners' mispronunciations from cross-language phonological comparisons. In *Proceedings of Automatic Speech Recognition and Understanding (ASRU)*, 2007.

[153] Ineke Mennen. Can language learners ever acquire the intonation of a second language? In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 17--19. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[154] P Mermelstein. Distance measures for speech recognition, psychological and instrumental. In C. H. Chen, editor, *Pattern Recognition and Artificial Intelligence*, pages 374--388, Hyannis, Massachusetts, June 1976.

[155] N Minematsu. Pronunciation assessment based upon the phonological distortions observed in language learners' utterances. In *Eighth International Conference on Spoken Language Processing*, pages 1669--1672, 2004.

[156] N Minematsu. Yet another acoustic representation of speech sounds. *2004 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '04*, pages I--585--8, 2004.

[157] N Minematsu, K Kamata, S Asakawa, T Makino, and K. HIROSE. Structural Representation of pronunciation and its application for classifying Japanese learners of English. In *Proceedings of ISCA ITRW SLaTE07*, Farmington, PA, 2007.

[158] D. Morgenstern. The Athena Language Learning Project. *Hispania*, 69(3):740--745, 1986.

[159] H. Morrison and E. Adams. Pilot study of CAI laboratory in German. *Modern Language Journal*, 52(5):279--287, 1968.

[160] Jack Mostow and G. Aist. Giving Help and Praise in a Reading Tutor with Imperfect Listening-Because Automated Speech Recognition Means Never Being Able to Say You're Certain. *CALICO journal*, 16(3):407--424, 1999.

[161] N. Mote, L. Johnson, Abhinav Sethy, Jorge Silva, and S. Narayanan. Tactical language detection and modeling of learner speech errors: The case of Arabic tactical language training for American English speakers. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, page 19, 2004.

[162] E Moulines and F Charpentier. Pitch synchronous waveform processing techniques for text-to-speech conversion using diphones. *Speech Communication*, 9:453--467, 1990.

[163] E. Moulines and J. Laroche. Non-parametric techniques for pitch-scale and time-scale modification of speech. *Speech Communication*, 16:175--206, February 1995.

[164] N Moustroufas and V Digalakis. Automatic pronunciation evaluation of foreign speakers using unknown text. *Computer Speech & Language*, 21(1):219--230, January 2007.

[165] Murray J Munro and Tracey M Derwing. Foreign Accent, Comprehensibility, and Intelligibility in the Speech of Second Language Learners. *Language Learning*, 49(S1):285--310, 1999.

[166] Noriko Nagata. Computer vs. Workbook Instruction in Second Language Acquisition. *CALICO journal*, 14(1):53--75, 1996.

[167] A. Neri and C. Cucchiarini. Feedback in computer assisted pronunciation training: technology push or demand pull? In *Proceedings of ICSLP*, 2002.

[168] A. Neri, C. Cucchiarini, and H. Strik. ASR-based corrective feedback on pronunciation: does it really work. In *Ninth International Conference on Spoken Language Processing*. Citeseer, 2006.

[169] A. Neri, C. Cucchiarini, H. Strik, and L. Boves. The pedagogy-technology interface in Computer Assisted Pronunciation Training. *Computer Assisted Language Learning*, 15(5):441--467, 2002.

[170] Ambra Neri, Catia Cucchiarini, and Helmer Strik. Effective feedback on L2 pronunciation in ASR-based CALL. In *Proceedings of the workshop on Computer Assisted Language Learning*, pages 40--48. Citeseer, 2001.

[171] Ambra Neri, Catia Cucchiarini, and Helmer Strik. Segmental errors in Dutch as a second language: how to establish priorities for CAPT. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, 2004.

[172] L. Neumeyer, H. Franco, V Abrash, and L Julia. WebgraderTM: a multilingual pronunciation practice tool. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, 1998.

[173] L. Neumeyer, H. Franco, M. Weintraub, and P. Price. Automatic text-independent pronunciation scoring of foreign language student speech. *Proceedings of ICSLP*, pages 1457--1460, 1996.

[174] R.S. Nickerson and K N Stevens. An experimental computer-based system of speech training aids for the deaf. In *Proceedings of the Conference on Speech Communication and Processing*. Institute of Electrical and Electronics Engineers and Air Force Cambridge Research Laboratories, 1974.

[175] PIE. NORDSTROM and B. LINDBLOM. A Normalization Procedure For Vowel Formant Data. In *The International Congress Of Phonetic Sciences*, Leeds, 1975.

[176] Joyce Nutta. Is Computer-Based Grammar Instruction as Effective as Teacher-Directed Grammar Instruction for Teaching L2 Structures? *CALICO journal*, 16(1):49--62, 1998.

[177] Council of Europe. *Common European Framework of Reference for Languages: Learning, Teaching, Assessment.* Cambridge University Press, 2001. ISBN: 0521005310.

[178] William O'Grady, John Archibald, Mark Aronoff, and Janie Rees-Miller. *Contemporary Linguistics: An Introduction*. Bedford/St.Martin's, 4 edition, 2001. ISBN: 0-312-24738-9.

[179] Fuping Pan, Qingwei Zhao, and Yonghong Yan. *New machine scores and their combinations for automatic Mandarin phonetic pronunciation quality assessment*. Springer-Verlag, September 2007.

[180] Fuping Pan, Qingwei Zhao, and Yonghong Yan. Mandarin vowel pronunciation quality evaluation by a novel formant classification method and its combination with traditional algorithms. *2008 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '08*, pages 5061--5064, 2008.

[181] Jeon G Park and Seok-Chae Rhee. Development of the knowledge-based spoken English evaluation system and its application. In *Proceedings of ISCA INTERSPEECH 2004*, 2004.

[182] Mitchell Peabody and Stephanie Seneff. Towards automatic tone correction in non-native Mandarin. *Chinese Spoken Language Processing*, 4274:602--613, 2006.

[183] Mitchell Peabody and Stephanie Seneff. Annotation and Features of Non-native Mandarin Tone Quality. *Tenth Annual Conference of the International ...*, 2009.

[184] Mitchell Peabody, Stephanie Seneff, and Chao Wang. Mandarin tone acquisition through typed interactions. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, 2004.

[185] L Peng. Obstruent voicing and devoicing in the English of Cantonese speakers from Hong Kong. *World Englishes*, 2004.

[186] Martin J. Petersen Jr. *SPLASH: The Computer Program*. United States International University, San Diego, CA, 1989.

[187] Michael Pitz and Hermann Ney. Vocal tract normalization as linear transformation of MFCC. In *Proceedings of EUROSPEECH 2003*. Citeseer, 2003.

[188] Stanisław Puppel and Ernst Hɑkon Jahr. The theory of universal vowel space and the Norwegian and Polish vowel systems. In Raymond Hickey and Stanisław Puppel, editors, *Language History and Linguistic Modelling*, volume 2, pages 1301----1324. Mouton de Gruyter, Berlin, 1997.

[189] Ravi Purushotma. Commentary: You're not studying, you're just ... *Language, Learning & Technology*, 9(1):80--96, 2005.

[190] James P Pusack. *DASHER: An Answer Processor for Language Study*. CONDUIT, Iowa City, IA, 1983.

[191] Siwei Qin, Satoru Fukayama, Takuya Nishimoto, and Shigeki Sagayama. Lexical Tones Learning with Automatic Music Composition System Considering Prosody of Mandarin Chinese. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 3--6, Tokyo, Japan, 2010.

[192] J. R. Quinlan. Learning decision tree classifiers. *ACM Computing Surveys*, 28(1):71--72, 1996.

[193] A Raux and A Black. A Unit Selection Approach to F0 Modeling and its Application to Emphasis. In *Proceedings of Automatic Speech Recognition and Understanding (ASRU)*, St Thomas, US Virgin Islands, 2003.

[194] A Raux and M Eskenazi. Non-Native Users in the Let's Go!! Spoken Dialogue System: Dealing with Linguistic Mismatch. In *HLT/NAACL 2004*, Boston, MA, 2004.

[195] A Raux and M Eskenazi. Using Task-Oriented Spoken Dialogue Systems for Language Learning: Potential, Practical Applications and Challenges. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, 2004.

[196] A Raux, B Langner, M Eskenazi, and A Black. LET'S GO: Improving Spoken Dialog Systems for the Elderly and Non-natives. In *Proceedings of EUROSPEECH 2003*, Geneva, Switzerland, 2003.

[197] Jack C. Richards and Theodore S. Rodgers. Communicative language teaching. In Jack C. Richards, editor, *Approaches and Methods in Language Teaching*, pages 153--177. Cambridge University Press, 2001.

[198] Wilga M Rivers. *Teaching Foreign Language Skills*. University of Chicago Press, 2nd editio edition, 1981.

[199] William R Rodr. ARTICULA - A tool for Spanish Vowel Training in Real Time. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 2--5, Tokyo, Japan, 2010.

[200] Carsten Roever. Web-based Language Testing. *Language, Learning & Technology*, 5(2):84--94, 2001.

[201] Raul Rojas. *Neural Networks: A Systematic Introduction*. Springer-Verlag, New York, 1996.

[202] Orith Ronen, Leonardo Neumeyer, and Horacio Franco. Automatic detection of mispronunciation for language instruction. In *Proceedings of EUROSPEECH 1997*. Citeseer, 1997.

[203] Peter S Rosenbaum. The computer as a learning environment for foreign language instruction. *Foreign Language Annals*, 2(4):457--465, 1969.

[204] Marikka Elizabeth Rypa and Patti Price. VILTS: A Tale of Two Technologies. *CALICO journal*, 16(3):385--404, 1999.

[205] MR Salaberry. The use of technology for second language learning and teaching: A retrospective. *The Modern Language Journal*, 2001.

[206] Jr. Samuel H Desch. *An Interactive Computer Aid to reading scientific German*. Massachusetts Institute of Technology Press, Cambridge, MA, 1973.

[207] S. Seneff. Web-based dialogue and translation games for spoken language learning. In *Proceedings of ISCA ITRW SLaTE07*, pages 9--16, 2007.

[208] Stephanie Seneff, Chao Wang, and Chih-yu Chao. Spoken dialogue systems for language learning. In *Proceedings of NAACL HLT07*, Rochester, NY, 2007.

[209] Stephanie Seneff, Chao Wang, Mitchell Peabody, and Victor Zue. Second Language Acquisition through Human Computer Dialogue. In *4th International Symposium on Chinese Spoken Language Processing, 2004. ISCSLP'04.*, 2004.

[210] Bob Sevenster, Guus de Krom, and Gerrit Bloothooft. Evaluation and training of second-language learners' pronunciation using phoneme-based HMMs. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 91--94, 1998.

[211] Wang Shudong and Michael Higgins. Training English Pronunciation for Japanese Learners of English Online. *JALT CALL Journal*, 1(1):39--47, 2005.

[212] Peter Skehan. Task-based instruction. *Language Teaching*, 36(1):1--14, 2003.

[213] Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. Cheap and fast---but is it good?: evaluating non-expert annotations for natural language tasks. *... in Natural Language ...*, 2008.

[214] C W Stansfield. An evaluation of simulated oral proficiency interviews as measures of oral proficiency. In J E Alatis, editor, *Georgetown University Roundtable on Languages and Linguistics*, pages 228--234, Washington, DC, 1990. Georgetown University Press.

[215] K. N. Stevens. The quantal nature of speech: Evidence from articulatory-acoustic data. In E. E. David, Jr. and P. B. Denes, editors, *Human Communication: A Unified View*. McGraw-Hill, New York, 1972.

[216] Helmer Strik, Catia Cucchiarini, and Diana Binnenpoorte. L2 Pronunciation Quality In Read And Spontaneous Speech. In *Proceedings of ICSLP*, 2000.

[217] Anna Sundstr o m. Automatic prosody modification as a means for foreign language pronunciation training. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 49--52. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[218] Masayuki Suzuki, Luo Dean, Nobuaki Minematsu, and Keikichi Hirose. Improved Structure-based Automatic Estimation of Pronunciation Proficiency. In *SLaTE 2009 - 2009 ISCA Workshop on Speech and Language Technology in Education*, 2009.

[219] Masayuki Suzuki, Yu Qiao, Nobuaki Minematsu, and Keikichi Hirose. Pronunciation Proficiency Estimation Based on Multilayer Regression Analysis Using Speaker-independent Structural Features. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 2--5, 2010.

[220] M Swain and S Lapkin. Problems in Output and the Cognitive Processes They Generate: A Step Towards Second Language Learning. *Applied Linguistics*, 16(3):371--391, September 1995.

[221] E Swender, editor. *ACTFL Oral Proficiency Interview Tester Training Manual*. American Council on the Teaching of Foreign Languages, Yonkers, NY, 1999.

[222] Brent Townshend, Jared Bernstein, Ognjen Todic, and Eryk Warren. Estimation of Spoken Language Proficiency. In *Proceedings of ESCA Workshop on Speech Tech-*

nology in Language Learning, pages 179--182. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[223] Khiet Truong, Ambra Neri, Catia Cucchiarini, and Helmer Strik. Automatic pronunciation error detection: an acoustic-phonetic approach. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, 2004.

[224] Yasushi Tsubota, Masatake Dantsuji, and Tatsuya Kawahara. Practical Use of Autonomous English Pronunciation Learning System for Japanese Students. In *Proceedings of InSTIL/ICALL Symposium: NLP and Speech Technologies in Advanced Language Learning Systems*, 2004.

[225] R.C. Turner. CARLOS: Computer-assisted instruction in Spanish. *Hispania*, 53(2):249--252, 1970.

[226] S. Umesh, S.V.B. Kumar, MK Vinay, Rajesh Sharma, and Rohit Sinha. A Simple Approach to Non-Uniform Vowel Normalization. In *IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS SPEECH AND SIGNAL PROCESSING*. Citeseer, 2002.

[227] John H Underwood. *Linguistics, Computers and the LanguageTeacher: A communicative Approach*. Newbury House Publishers, Inc., Rowley, MA, 1984.

[228] J. van Doremalen, Helmer Strik, and Catia Cucchiarini. Optimizing non-native speech recognition for CALL applications. In *Proceedings of Interspeech*, pages 592--595, Brighton, UK, 2009.

[229] R M Vardanian. Teaching English through oscilloscope displays. *Languate Learning*, 3(4):109--118, 1964.

[230] Edward Vockell and Eileen Schwartz. *The Computer in the Classroom*. Mcgraw-Hill, Santa Cruz, CA, 1988.

[231] Krystyna A Wachowicz and Brian Scott. Software That Listens: It's Not a Question of Whether, It's a Question of How. *CALICO journal*, 16(3):253--276, 1999.

[232] Hao Wang, Peggy Mok, and Helen Meng. MusicSpeak : Capitalizing on Musical Rhythm for Prosodic Training in Computer-Aided Language Learning. In *Second Language Studies: Acquisition, Learning, Education and Technology*, pages 2--5, Tokyo, Japan, 2010.

[233] Hongcui Wang and Tatsuya Kawahara. A Japanese CALL System based on Dynamic Question Generation and Error Prediction for ASR. In *Proceedings of Interspeech*, 2008.

[234] Yue Wang, Allard Jongman, and Joan A Sereno. Acoustic and perceptual evaluation of Mandarin tone productions before and after perceptual training. *The Journal of the Acoustical Society of America*, 113(2):1033, 2003.

[235] Si Wei, Guoping Hu, Yu Hu, and Ren-Hua Wang. A new method for mispronunciation detection using Support Vector Machine based on Pronunciation Space Models. *Speech Communication*, 51(10):896--905, 2009.

[236] M.B. Wesche. Communicative testing in a second language. *Modern Language Journal*, 67(1):41--55, 1983.

[237] Preben Wik. Simicry - A mimicry-feedback loop for second language learning. In *Second Language Studies: Acquisition, Learning, Education and Technology*, Tokyo, Japan, 2010.

[238] Preben Wik and David Lucas Escribano. Say ` Aaaaa ' Interactive Vowel Practice for Second Language Learning. In *SLaTE 2009 - 2009 ISCA Workshop on Speech and Language Technology in Education*, 2009.

[239] Silke Witt and Steve Young. Language Learning Based on Non-Native Speech Recognition. In *Proceedings of EUROSPEECH 1997*, pages 633--636, Rhodes, Greece, 1997.

[240] S.M. Witt. *Use of Speech Recognition in Computer-assisted Language Learning*. PhD thesis, University of Cambridge, 1999.

[241] S.M. Witt and S J Young. Performance Measures for Phone-Level Pronunciation Teaching in CALL. In *Proceedings of ESCA Workshop on Speech Technology in Language Learning*, pages 99--102. ESCA, ESCA and Department of Speech, Music and Hearing KTH, 1998.

[242] H Wohlert. German by Satellite. *Annals of the American Academy of Political and Social Sciences*, 1991.

[243] H.S. Wohlert. Voice input/output speech technologies for German language learning. *Die Unterrichtspraxis/Teaching German*, pages 76--84, 1984.

[244] Grace H. Yeni-Komshian, James E. Flege, and Serena Liu. Pronunciation proficiency in the first and second languages of Korean--English bilinguals. *Bilingualism: Language and Cognition*, 3(2):131--149, 2000.

[245] Su-youn Yoon, Mark Hasegawa-johnson, and Richard Sproat. Automated Pronunciation Scoring using Confidence Scoring and Landmark-based SVM. In *Proceedings of Interspeech*, pages 1903--1906, Brighton, UK, 2009.

[246] Brandon Yoshimoto, Ian Mcgraw, and Stephanie Seneff. Rainbow Rummy : A Web-based Game for Vocabulary Acquisition using Computer-directed Speech. In *SLaTE 2009 - 2009 ISCA Workshop on Speech and Language Technology in Education*, 2009.

[247] Dolly Jesusita Young. Creating a Low-Anxiety Classroom Environment: What Does Language Anxiety Research Suggest? *The Modern Language Journal*, 75(4):426--439, 1991.

[248] S. Young, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book*. Cambridge University, Cambridge, UK, 1997.

[249] K Zechner and D Higgins. Speechrater: A construct-driven approach to scoring spontaneous non-native speech. In *Proceedings of ISCA ITRW SLaTE07*, Farmington, PA, 2007.

[250] Klaus Zechner, Derrick Higgins, Xiaoming Xi, and David M. Williamson. Automatic scoring of non-native spontaneous speech in tests of spoken English. *Speech Communication*, 51(10):883--895, 2009.

[251] F Zhang. Exploring computer-based browsing systems in the teaching of pronunciation. In *Applied Languages Curriculum Design Conference for the 2001 4th Southern Technical Institutes and Schools of Taiwan, Republic of China.*, KaoHsiung, 2001. Fortune Institute of Technology.

[252] V. Zue, J. R. Glass, D. Goodine, M. Phillips, and S. Seneff. The SUMMIT speech recognition system: Phonological modeling and lexical access. In *Proc. ICASSP*, pages 49--52, 1990.