UNIVERSITY OF CAMBRIDGE
DEPARTMENT OF ENGINEERING

Use of Speech Recognition in Computer-assisted Language Learning

Silke Maren Witt

Newnham College

November 18, 1999

A dissertation submitted to the University of Cambridge
for the degree of Doctor of Philosophy.

Forever is only what we lost.
Henrik Ibsen

To the memory of my sister Kristina.
* 25. April 1974 — † 29. August 1996

Summary

Computer-assisted language learning (CALL) systems which are able to listen to a student's speech and to judge its quality would be very valuable for foreign language teaching. However, currently it is difficult to integrate pronunciation teaching and assessment in computer-assisted language learning systems. Two major problems need to be addressed. Firstly, without improved acoustic modelling of non-native speech, oral interaction between a language student and a CALL system is unduly limited by poor recognition performance. Secondly, there are no reliable methods to automatically score the pronunciation of a student and to localise pronunciation errors. The research presented in this thesis investigates solutions to these problems within the framework of hidden Markov model based automatic speech recognition.

The thesis begins by outlining those aspects of pronunciation teaching that are important for computer-assisted language learning. This helps understanding of what types of pronunciation exist and how they could be taught in an automated system. Next the characteristics of non-native speech that degrade speech recognition accuracy are discussed.

In order to improve the acoustic modelling of non-native speech, two adaptation algorithms have been developed called *Linear Model Combination* and *Model Merging*. These algorithms are based on the assumption that the mother-tongue of a non-native speaker is known. The basic idea underlying most findings of this thesis is that non-native speech can be modeled with a mixture of sounds of a speaker's native language and the target language. The newly developed speaker adaptation algorithms combine the acoustic models of the source and target language of a non-native speaker. The algorithms only differ with regard to the details how the model sets are combined.

A database of non-native English was recorded for the purpose of testing these adaptation algorithms. This database mostly consists of utterances of Japanese and Latin-American Spanish accented English. The recordings were transcribed by trained phoneticians to obtain transcriptions corresponding to the actual phoneme sequence uttered by the student as opposed to canonical transcriptions obtained

from a standard pronunciation dictionary. Evaluation of the two new speaker adaptation techniques based on speech from this database demonstrates the algorithms' capability to reduce the recognition error rate by up to 40% relative to the baseline and by up to 20% in comparison with standard adaptation methods.

Minimising the required amount of adaptation data is highly desirable in CALL. Indeed the shorter the enrollment task, the more advantageous this is for commercial applications. For this reason, we developed a set of three algorithms to improve recognition of non-native speech. These algorithms are also based on combining the acoustic models of the source and target language of a foreign speaker, but they do not require any adaptation material. Evaluation of these algorithms with Japanese and Latin-American Spanish accented English proved their capability to improve recognition accuracy of non-native speech by up to 27% relative to a speaker-independent system.

The second part of this thesis analyses the problem of automatic pronunciation assessment. Because pronunciation assessment is highly subjective, it was necessary to develop a set of four performance measures which compare human or computer-based judgments of pronunciation on a phone-by-phone basis. These measures were applied to compare the manually edited transcriptions of the non-native database done by six different phoneticians. This analysis of human labelling characteristics showed that even though human assessment of pronunciation can vary considerably, there exists a common level of judgments. Therefore, the averaged assessment similarity of different human judges is used as a benchmark against which the performance of any automatic assessment method is measured.

Based on this analysis of how phoneticians assess pronunciation, an automatic method of assessing pronunciation which we call *Goodness of Pronunciation* (*GOP*) was developed. This method calculates a score for each phone in an utterance to localise pronunciation errors. The baseline algorithm was refined in several ways, with the result that the optimal set of refinements yields an algorithm whose assessment capability is comparable to human assessment.

The research findings of this thesis add a new direction to the future developments in research to improve CALL systems. The conclusions outline how these findings could be developed further. For example, the techniques could be applied to other languages and other types of recognition systems. The thesis concludes with a discussion of possible ways of integrating these new algorithms into a computer-assisted language learning system.

Acknowledgements

First, my thanks go to the Engineering and Physical Research Council and to the Marie-Curie Fellowship program of the European Union. The financial support of these two institutions both enabled my studies in Cambridge and my visits conferences in the Netherlands, Greece, Sweden, Australia and Hungary. These visits broadened my horizon in many way as well as providing me with the most useful feedback for my research.

Being teachers themselves, my parents taught me the importance of pedagogy, which made the topic of foreign language teaching so fascinating to me. But above all, their love and encouragement enabled me to study in three countries and to study for a PhD.

My friends Gamila, Natasha, Xiang, Balaji, Stephan, Knut, Katherine and Sabine where wonderful friends especially in sad times, always willing to cheer me up and make me laugh again.

I'm grateful for my friends in the lab, Klaus Reinhard, Rob Rohling, Harriet Nock, Philip Clarkson, Ed Whittaker, Jason Humphries, Graham Trent, Nando Freitas, Andy Tuerck, Jonathan Carr for their constructive discussion and proof-reading efforts as well as for the countless tea-time conversations which I will miss greatly.

Thanks also go to Patrick Gosling for his superb maintainance of the lab's computing facilities and his help with any computer woes.

Finally, I want to thank my supervisor Steve Young for his guidance throughout my time in Cambridge and also for the freedom he gave me to pursue my ideas, while keeping me on the right path.

Declaration

This thesis is the result of my own original work, and where it draws on the work of others, this is acknowledged at the appropriate points in the text. Some of this work has been published previously in conference proceedings [93, 95, 98, 96, 97, 99]. The length of this thesis, including appendices and footnotes is approximately 40,000 words.

Abbrevations

CALL Computer Assisted Language Learning

CAPT Computer Assisted Pronunciation Teaching

GOP Goodness of Pronunciation

HMM Hidden Markov Model

LMC Linear Model Combination

MLE Maximum Likelihood Estimation

MLLR Maximum Likelihood Linear Regression

MAP Maximum A-priori Probability

MM Model Merging

PBM Parallel Bilingual Models

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Prolog

Imagine it is the year 2020. You have been told that you will have to head a project in a far-eastern company. This requires learning the basics of the language of this country within the next few weeks. Unfortunately, in the town where you live, nobody can teach you the language. Fortunately however, there exists a software package for computer-assisted learning of this particular language. So, you get the software, install it and off you go! ... Or imagine that you've always wanted to learn Spanish, but your working hours vary and tend to clash with the time of evening classes. But there is a solution: You simply buy the Spanish language learning software and go ahead ...

These two fictional scenarios give a good idea of the range of situations where a good language learning software will be immensely useful. The research presented in this thesis will attempt to contribute to the development of such software by developing tools for pronunciation teaching employing speech recognition technology. The newly developed algorithms address the two main challenges of pronunciation teaching. Firstly, improved acoustic modeling of non-native speech and secondly, assessment of the pronunciation quality of a student's speech.

1.2 Outline

1.2.1 Problem

The rapid progress in communication and information technology during the past decade combined with rising market globalisation has led to a steady increase in the demand for foreign language teaching. Also, since the teaching paradigm within language teaching has shifted towards more emphasis on the ability to communicate orally. Therefore, language teaching nowadays increasingly focuses on pronunciation skills. However, within a traditional classroom environment it is difficult to focus on the needs of individual students. Conventional language teaching is also expensive and facilities for teaching every language will not always be available in any particular place. Often classes for different levels of fluency are also required, making it even more difficult to find the right class for one's needs. Self-study on the other hand is currently limited to studying a book with accompanying tapes or basic computer-based learning systems.

The performance of both computer hardware and speech processing technology has improved rapidly over the last few years. These technological advances make it possible and desirable to design computer-aided teaching software which exploits speech recognition technology in order to either assist language teachers or to enable self-study.

There already exist various commercial foreign language learning systems utilising hyper-text and multi-media, but within those systems interaction is generally limited to written text or recorded speech output. Although considerable research effort has been invested in the development of computer-assisted language learning (CALL) systems, little attention has been paid to pronunciation teaching. The oral component of such language teaching systems has the disadvantage that it is not possible to process and evaluate any oral response of a student using the standard means of interaction such as keyboard and mouse. Additionally, unlike grammar or vocabulary exercises where there exist clearly defined wrong or right answers, pronunciation exercises have no such yes/no answers. A large number of different factors contribute to the overall pronunciation quality which are difficult to measure. Hence, the transition from poor to good pronunciation is gradual, and any assessment of it must also be presented on a continuous scale. Thus, discourse and pronunciation training are only rudimentarily possible in current CALL systems, similarly limited are methods to monitor a student's oral performance.

Another difficulty with the development of computer-assisted pronunciation teaching is its location on the border of two very different research disciplines. On the one hand, there are the research findings regarding pedagogical and phonological aspects of pronunciation teaching, on the other hand, there is the knowledge of automatic speech processing, in particular speech recognition. These two distinct research disciplines need to be combined in order to develop systems which meet user requirements.

1.2.2 Goal

As outlined above, CALL systems which are able to listen to a student's speech, to judge its quality, and to train students individually would be highly useful tools for language teaching. The goal of this thesis is to develop techniques based on automatic speech recognition in order to enable or enhance computer-assisted pronunciation teaching. This general goal can be divided into the following components which will be addressed in this work:

1. Obtaining an understanding of those aspects of pronunciation teaching which are important for CALL, i.e. the types of pronunciation which exist and how pronunciation could be taught in an automatic system.

2. Improving the acoustic modeling of non-native speech in order to enable text-independent recognition of foreign accented speech. Such improvement would enable a dialogue between student and computer which again would create a more natural learning experience.

3. Obtaining an understanding of how teachers assess pronunciation: This requires an analysis of human assessment criteria based on human judgment of non-native data.

4. Developing performance measures to compare the assessment of different human judges.

5. Developing computer-assisted pronunciation assessment methods on sub-word level in order to localise and diagnose individual pronunciation errors.

1.2.3 Thesis contributions

The overall contributions of this thesis can be summarized as:

- Development of several algorithms for improved acoustic modeling of non-native speech. Two of these algorithms are adaptation techniques especially designed for non-native speech. Another three algorithms have been developed which improve recognition performance of accented speech without requiring adaptation data. The only requirement for these algorithms is a knowledge of the mother tongue of the non-native speaker. This knowledge has to consist of acoustic models of the native language as well as of typical mispronunciation characteristics for the given source language.

- Recording of a database of heavily accented non-native speech spoken by students of English as a second language. This database also has been hand-transcribed by trained phoneticians in order to annotate all mispronunciations at the phoneme level.
- Development of a set of four performance measures in order to analyse and compare human pronunciation assessment characteristics.
- Development of a computer-based scoring method of pronunciation, which calculates a score for each phoneme in an utterance. These scores can be used for assessment or as a basis for more detailed feedback on pronunciation mistakes.

1.2.4 Thesis structure

In Figure 1.1, the structure of this thesis is shown in the form of a block-diagram. Chapter 2 discusses design aspects of CALL systems especially with regard to pronunciation teaching. This discussion is followed by an analysis of how speech recognition could be integrated into such systems. Examples of existing integration efforts are given with the review of the current state-of-the-art of computer-assisted language teaching. The understanding obtain in this chapter influences the design of the non-native database (Chapter 5) and the design of the pronunciation scoring methods (Chapter 8 and 9).

After the discussion of the main pedagogic and phonetic issues regarding pronunciation teaching, Chapter 3 provides an introduction to the other main branch in this thesis, i.e. speech recognition technology. Furthermore, the characteristics of non-native speech, which can severely degrade the accuracy of continuous speech recognition systems are discussed. Such performance degradation due to accented speech recognition can render unconstrained recognition within a dialogue almost impossible. Therefore, the following chapter, Chapter 4, presents two adaptation algorithms, called *Linear Model Combination* and *Model Merging* in order to improve recognition of foreign accented speech. The main new aspect of these algorithms is that knowledge of the mother-tongue of the speaker is exploited in order to improve the acoustic modeling of accented speech.

Chapter 5 describes the database of non-native speech which has been recorded in order to test these new techniques of non-native speech recognition. This database consists of recordings of students of English as a second language. These recordings were then transcribed by trained phoneticians in order to obtain transcriptions corresponding to the actual phoneme sequence uttered by the student as compared

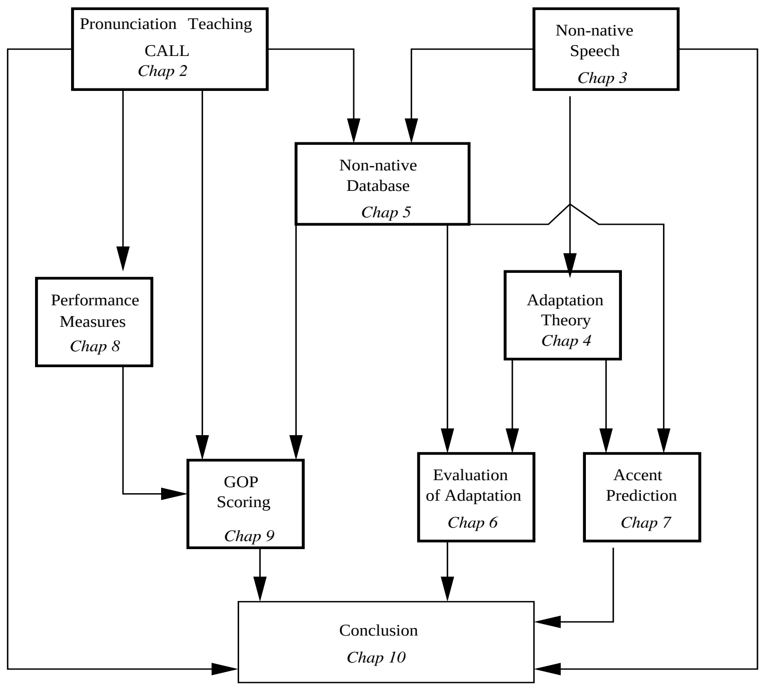

Figure 1.1: Block-diagram of how this thesis is organised. The arrows indicate how the different chapters are linked by using knowledge from previous chapters.

to canonical transcriptions obtained from a dictionary containing standard British-English pronunciations.

This database is then used to test the algorithms derived in Chapter 4. Chapter 6 presents experimental results for the two adaptation techniques and compares their performance with the performance of standard adaptation methods. The target of this work is to keep the required amount of adaptation data to a minimum, in the best case to zero, because the shorter the enrollment task, the more advantageous it is for commercial applications. Such considerations motivated the development of three further algorithms presented in Chapter 7. These algorithms are capable of improving non-native speech recognition without requiring any adaptation material provided that the accent of the test subject is known beforehand, i.e. there is access to acoustic models of the native language and typical mispronunciation characteristics are known.

Having addressed the problem of modeling and recognising non-native speech, the thesis turns towards analysing the question of pronunciation assessment. Pronunciation assessment is a highly subjective task, thus standard recognition error measurements cannot be applied to compare the assessment of different judges. For this reason, a set of performance measures has been derived in Chapter 8 which compares two sets of judgments with regard to different aspects of pronunciation assessment. These measures have been applied to the transcriptions of the non-native database in order to analyse human labeling characteristics.

In Chapter 9 an automatic method of assessing pronunciation which we call *Goodness of Pronunciation* (*GOP*) is developed. This method calculates a score for each phone [1] in an utterance. Different modifications of the baseline algorithm are introduced. Comparison with the human labeling based on the performance measures of Chapter 8 shows that the results of the automatic assessment technique are comparable to human assessment.

Finally, the conclusions discuss possible designs of pronunciation teaching systems which would incorporate the results of this research about pronunciation assessment and correction as well as the algorithms for modeling non-native speech. Additionally, possible directions of future work are outlined.

[1] In this work a “phone” denotes a sound unit used to model speech with HMMs, which roughly corresponds to a phoneme as defined by linguists.

Chapter 2

Pronunciation teaching and oral interactivity in CALL

This thesis addresses two major challenges which arise when exploiting speech recognition technology in a CALL system. The first challenge is to recognise a learner's speech within simulated dialogues in order to train conversation skills. The second challenge concerns Computer-Assisted Pronunciation Teaching (CAPT), where the learner's speech is assessed and corrected. In order to comprehend the main issues and challenges of both CAPT and CALL with regard to oral interaction, it is necessary to understand the main issues in CALL and pronunciation teaching.

The next section discusses several important aspects of CALL systems with emphasis on how speech recognition technology can be incorporated into such systems. Having discussed general CALL principles, Section 2.2 summarises the most important characteristics and problems of pronunciation teaching. Then, the knowledge from these two sections is used to categorise the existing state-of-the-art in both CALL with oral interaction and in CAPT, see Section 2.3.

Before discussing the details of CALL and pronunciation teaching, one definition has to be made, which will hold true throughout this thesis. The **target language** of a language student is defined as the language he or she is trying to learn, whereas the **source language** denotes his or her mother tongue.

2.1 Computer-assisted language-learning systems

2.1.1 Advantages and disadvantages of CALL

What is the motivation behind using CALL systems to teach foreign languages? At the very beginning of this thesis, some example applications of CALL were mentioned. A more detailed understanding of the motivation behind CALL systems can be obtained by analysing the advantages and disadvantages of CALL systems. For a more detailed discussion see also [72].

The advantages of a CALL system are that such a system can provide undivided attention to the user as opposed to a classroom environment where a teacher has to divide his attention among all students. CALL systems can also be designed for automatic language assessment. In this case, such systems can administer language tests which are more objective, cheaper and less time-consuming. Moreover, computers as opposed to human teachers are infinitely patient and usually available without any time constraints. Additionally, computer-based systems permit a high degree of individuality regarding the choice of material which is studied and the speed of progress. Finally, computer-based learning can be less threatening for self-conscious learners, who might be afraid of "losing face" in a classroom environment.

These collective advantages have led to an active interest in CALL by the language teaching community. However, the high expectations were not met by the performance of initial CALL systems, which often were only prototypes. Together with the inherent disadvantages of CALL, these experiences led to some disappointment of language teachers in this new technology. Such disadvantages of CALL are, for example, that human teachers will always be able to provide more motivation and feedback than a computer system. Another disadvantage is that speaking a language is a highly social process which can be best learnt through social interaction. Additionally, a good teacher will be able to offer a larger variety of learning activities and will be able to optimise which approach to use with which type of student, at least for the foreseeable future.

Currently, commercially available CALL systems make no or only very limited use of speech recognition. Mostly, these systems are limited to more exhortative types of exercises such as vocabulary and grammar drills. CALL systems increasingly incorporate multimedia, but predominantly show pictures and video clips or play pre-recorded speech of native speakers. Because of the limited technology and system complexity used in current commercial systems, most systems are aimed at beginners. Only few systems are sophisticated enough to teach writing and speaking to more advanced students.

One of the main reasons why the development of pedagogically sound CALL software is still fairly rudimentary is the lack of a unified theoretical framework for designing and evaluating such systems. It is therefore important to discuss how the plethora of teaching approaches which already exists for second language teaching can be applied to CALL. Chapelle, [17] discusses which type of research on second language acquisition is relevant for multimedia CALL design. She proposes seven criteria which should be fulfilled if a teaching approach is to be implemented in

CALL.

1. The linguistic characteristics of target language input need to be made salient.
2. Learners should receive help in comprehending semantic and syntactic aspects of linguistics input.
3. Learners need to have the opportunities to produce target language output.
4. Learners need to notice errors in their own output.
5. Learners need to correct their linguistic output.
6. Learners need to engage in target language interactions whose structure can be modified for different contents.
7. Learners should engage in source language tasks designed to maximize opportunities for good interaction.

Relating these teaching principles to the task of incorporating speech technology into CALL, the following classification can be made: Whereas issues 1, 2 and 5 have to be addressed by research in the area of computational linguistics and natural language understanding, issues 3 and 4 can be addressed by advances in CAPT. Finally, issues 3, 6 and 7 can profit from progress in non-native recognition and dialogue systems.

This classification indicates the important role speech recognition can and should play in CALL systems. This classification also provides a way to combine both the technical and pedagogical side when designing a CALL system. Given the current state-of-the-art, the most urgent requirement for progress in incorporating speech technology in a well-rounded CALL system is integration of language teaching and speech technology.

2.2 Pronunciation teaching

2.2.1 What is pronunciation?

Before starting to derive statistical methods of pronunciation assessment and correction based on speech technology, it is necessary to obtain a basic understanding of the main aspects and components of pronunciation teaching. This requires both the discussion of what is to be understood by "correct" pronunciation in the content of pronunciation teaching and the definition of all components of pronunciation. Moreover, it is important to understand the pedagogics of how to teach pronunciation.

'Correct' pronunciation

There exists no absolutely "correct" pronunciation. A wide variety of pronunciations can be accepted by native speakers as being correct, for instance, consider the multitude of dialects within most languages. Likewise, how accent is perceived depends on a given situation and on the previous exposure of listeners to people with accent. A wide range of psycholinguistic experiments has been executed in order to investigate how listeners perceive foreign accents, see for instance [38].

While pronunciation can vary considerably within a language and each pronunciation variant is accepted as "native", language teaching should be restricted to one type of pronunciation only, e.g. Standard Southern British English, so that a language student will learn to speak in a consistent manner. Teaching a single variant of a language is justified, because generally native speakers only speak one variant of a language too, and in most cases they can still communicate with other speakers of the same mother-tongue.

Apart from regional accents, further variations in the pronunciation and speaking characteristics of any individual depend heavily on the physiology of the speaker. For example, different lengths of the vocal tract can change the characteristics of a person's voice considerably.

Components of pronunciation

The accuracy of pronunciation is determined by both *segmental* and *suprasegmental* features. The segmental features are concerned with the distinguishable sound units of speech, i.e. phonemes. A phoneme is also defined as "the smallest unit which can make a difference in meaning" [90]. The set of phonemes of one language can be classified into broad phonetic subclasses; for example, the most general classification would be to separate vowels and consonants. Each language is characterised by its distinctive set of phonemes. When learning a new language, foreign students can divide the phonemes of the target language into two groups. The first group contains those phonemes which are similar to the ones in his or her source language. The second group contains those phonemes which do not exist in the source language. Teaching the pronunciation of segmental features thus means teaching the correct pronunciation of the target language phonemes, both in isolation and in context with other phonemes, i.e. within words or sentences. Even if a phoneme is known in isolation, new languages often contain clusters of phonemes which do not exist in the source language, so that the pronunciation of such clusters might require training as well.

The suprasegmental features of speech comprise of intonation, pitch, rhythm and stress and require a different teaching approach than teaching the phonetics of a foreign language. The work presented in this thesis will concentrate on the segmental features of pronunciation.

2.2.2 How to teach pronunciation

Before automatic pronunciation methods can be developed, the main issues of conventional pronunciation teaching need to be discussed. These are

1. **Goals of pronunciation**
2. **Requirements of language teachers**
3. **Teaching techniques**
4. **Feasibility of automatic teaching**

Goals of pronunciation

Depending on the reason why somebody is studying a foreign language, the learning goals can vary considerably. Someone might want to learn the basics of a language in order to be able to communicate during a vacation in a foreign country. Somebody else might want to learn a language in order to be able to negotiate business deals. Altogether, the desired level of fluency can vary to a large degree between bare communication and native-like fluency. However, since it is well known within the psycholinguistic community that achieving native-like pronunciation requires enormous learning efforts of adult learners, a goal common to most learners of a foreign language is to achieve *comfortable intelligibility*, [53, 2, 1]. Comfortable intelligibility is defined as a level of pronunciation quality, where words are correctly pronounced according to their phonetic transcription, but there are still subtle differences in how these phonemes sound in comparison with native speakers. Moreover, the speech of comfortably-intelligible non-native speakers might differ from native speakers with regard to intonation and rhythm, but on overall their speech is understandable without requiring too much effort from a listener.

As an aside, another aim should also be to teach a version of the target language which is socially acceptable to the students. For example, in countries where English is a language of business and administration, the aim would be to teach the local variant of English and not the standard Southern British English.

Finally, any teaching goal has to be fine-tuned to the individual needs of the student.

Concluding the discussion about pronunciation teaching goals, the aim of this thesis is to teach the correct pronunciation of words of the foreign language according to a previously decided language variant at a level of comfortable intelligibility.

Requirements for a language teacher

If we understand what is required of a language teacher in order to be able teach pronunciation efficiently, it will help us to define what is required in an automatic teaching system. In Abercrombie, [2], a teacher's requirements have been classified into theoretical and practical requirements. As theoretical requirements, Abercrombie lists an understanding of the vocal organs and an understanding how a spoken utterance can be analysed and described for teaching purposes. Also, a language teacher should be acquainted with the phonetic structure of both the source and the target language of a language student. On the practical side, a language teacher needs an ear to distinguish pronunciation mistakes. He also should be able to produce the sounds of the target language in isolation as well as mispronounced variations of these target language sounds in order to illustrate the difference. Next, a language teacher should be acquainted with an armory of teaching exercises in order to match them to the individual requirements of a learner. For a more detailed discussion see also [86].

Teaching techniques

Foreign language teaching represents an important research area within pedagogics. Thus, literature on pronunciation teaching methods abounds. However, for this thesis it will suffice to mention only a few of the main principles. With the intention to structure existing teaching approaches, Strevens, [86], introduces a rationale for teaching pronunciation, which contains three basic and distinct teaching techniques, covering the majority of existing exercise types. These are 1) exhortation, i.e. instructions to imitate and mimic; 2) speech training, which comprises of exercises to practice particular sounds and sound sequences or prosodic features; and 3) practical phonetics, which comprises of ear training, production exercises and the description of speech organs and articulation. In general, most existing teaching techniques fall into one of these categories. By classifying all exercises according to these principles it can be assured that all areas of teaching are equally covered.

Feasibility

Before the development of techniques for computer assisted pronunciation teaching, it has to be to be ensured that a computerised teaching approach is indeed capable of teaching pronunciation effectively. In a preliminary study Rogers et al., [78], showed that computer-based pronunciation training improved the pronunciation of Chinese subjects. Moreover, the authors observed that the subjects were capable of generalising the pronunciations of words they had learnt to unseen words.

Accent and pronunciation mistakes can only be corrected if the difference can be perceived by a student. Several studies, e.g. [55], have investigated the differences in the perception of target language sounds by speakers with different source languages. For instance, native Spanish speakers often cannot distinguish between /v/ and /b/ as an English speaker does. Therefore, teaching such sounds will require more effort, than other sounds which are common in both languages. It might be necessary to derive special listening training sessions to train the ear to distinguish unknown sounds of the target language from similar sounds in the source language.

Anderson et al., [8], presented one of the first evaluations of speech recognisers for speech training applications with special emphasis on utterance identification and speech quality assessment. In particular, it was found that it is possible for speech recognisers to identify disordered (i.e. also accented) speech as correctly or incorrectly produced. Additionally, judgment scores from recognisers were found to correlate well with human judgments of speech quality. These findings and others indicate that speech recognition technology can indeed be employed successfully for pronunciation teaching.

2.3 State-of-the-art in CAPT

The previous two sections described the main issues concerning both CALL systems and pronunciation teaching. Combining the knowledge of these two sections will help to understand the specific aspects of incorporating pronunciation teaching in a CALL system.

Generally speaking, any automatic teaching system should mimic the tasks of a language teacher. In addition to this, it should also exploit the advantages a computerised system has to offer with regard to particular training types which can be difficult for a human teacher to present. Speech recognition offers a powerful tool for analysing the speech of a language student and for providing detailed feedback. However, for this purpose, it is necessary to modify and enhance existing speech recognition algorithms, because currently these are predominantly geared towards the recognition of continuous, native speech. This situation yields the following research and development challenges:

1. Need for improved recognition performance of non-native speech

2. Development of robust language assessment tools:

   Tools for segmental features

   Tools for suprasegmental features

3. Development of stand-alone systems teaching all aspects of pronunciation

4. Development of methods to evaluate the effectiveness of CAPT systems

In this section, the current state-of-the-art of research according to these four categories will be discussed with reference to the principles of CALL and pronunciation teaching which have been outlined in Sections 2.1 and 2.2.

2.3.1 Improved recognition of non-native speech

The teaching principles 3, 6 and 7, from Section 2.1, can only be implemented if recognition of non-native speech and efficient design of spoken dialogues are possible. This requires improvements in the acoustic modeling of non-native speech and advances in the design of spoken dialogues in the area of language teaching. However, so far little work has been carried out in this field. Ehsani et al., [31], showed that retraining speaker-independent models with non-native speech can improve recognition significantly. However, in this case the need for non-native data represents a bottleneck. Less training data is needed when applying speaker adaptation to improve the acoustic modeling. In Thelen et al., [89], and Abrash et al., [3], speaker adaptation has fairly successfully been applied to non-native speech, as well as in Leggetter et al., [60], [103] and [29].

2.3.2 Tools for segmental features

Firstly, two text-dependent (i.e. any new teaching material requires additional data recordings by a native speaker) tools will be discussed which train the pronunciation of isolated and hand-selected words. One of the earliest developments of a tool for teaching segmental features is presented by Hamada et al., [45]. Here, the pronunciation of isolated words is scored by calculating the spectral distances between a student's utterance and a pre-recorded native utterance of the same word, using vector quantisation and dynamic time warping.

Research on computer-based speech training for speech-impaired children and adults has been shown to decrease the pronunciation errors caused by speech disorders. This knowledge was utilised in the development of the *HearSay* system, [25]. For this text-dependent tool, lists of minimal pairs for typical pronunciation mistakes for a given language pair were collected and recorded by native speakers.

This material was then used to train isolated words and phrases with a system that adapts automatically to a student's performance by constantly recording successes and failures of the subject.

Next, text-independent tools have been built which score the overall pronunciation quality of larger speech segments, i.e. sentences. Such tools have been presented in a number of studies, see [68, 40, 24, 23, 84]. All these tools calculate a single score per sentence using averaged a-posteriori log-likelihoods. The reliability of this approach has been proven by the high correlation of these automatic sentence scores with the scores by human judges, which has been measured in several independent studies. However, computing a single, overall score for a long segment of speech has the drawback that it is not possible to localise individual mispronunciations or to provide a detailed error feedback.

This problem has partially been addressed by Kim et al., [54]. Here phone-specific scores based on likelihood-ratios, were introduced. This approach is similar to the work of this thesis. However, each phone-specific score is based on an average of the a-posteriori score of a large number of phone utterances. Thus, consistently mispronounced phonemes can be detected, but individual mispronunciations cannot be localised.

Whereas it has been shown that it is possible to develop text-independent tools for sentence level scores, recent tools to detect phone-level errors have gone back to text-dependent systems to make the task more tractable. Such tools have been presented in [5, 49]. All these methods calculate log-likelihood based scores for individual phonemes in order to measure the intelligibility and non-nativeness of a phoneme. These tools vary in their definition and calibration of the pronunciation scores as well as in the hidden Markov model sets used. In both approaches the error is localised within a word or sentence, but both approaches have the disadvantage of being text-dependent.

Another approach to localise and correct pronunciation errors at the phoneme level is to incorporate networks with alternative pronunciations, [47, 40, 52]. In the first case, alternative pronunciations were derived by hand for each taught word. In the second case, the alternative for each phone consists of an acoustic model for the same phone but trained on non-native speech. In the third case, networks based on typical substitutions by speakers with the same source language were implemented successfully.

Yet another possibility for teaching pronunciation at the phoneme level is to develop a tool for teaching just one specific group of phonemes. In [50, 51], durational information about models trained on non-native speech has been employed to teach the contrast of phonemes which only differ in their duration by measuring phoneme durations. This approach again is text-independent, rendering it more applicable to different teaching materials.

2.3.3 Tools for suprasegmental features

Tools for suprasegmental features generally deploy speech processing and visualisation technology in order to train a student so that he or she learns how to produce a target intonation or stress pattern. Examples of such tools can be found in [47, 64, 85, 70]. Visualisation examples are display of the F0 contour or the spectrum.

Among all suprasegmental features, fluency has attracted most attention. Tools to train fluency have been presented in [24, 32]. In the first case, fluency is defined as uttering all words within a pre-defined range of average native durations, whereas in the second case, nine different features including segment durations have been combined in order to measure fluency. Correlation of these automatic fluency scores with fluency scores by expert judges proved that fluency can be reliably predicted by computer-based scores.

2.3.4 CAPT systems

All the tools discussed so far will typically be useful as a supplement to classroom teaching rather than as a direct replacement. In order to fully benefit from these tools, they need to be integrated into an overall system which provides exercises for all types of pronunciation errors, and which operates according to pedagogical principles as discussed in Sections 2.1 and 2.2.

One of the first projects towards such a well-rounded system was presented by Hiller et al., [47]. This system provides exercises for rhythm and intonation using pitch contours and speech recognition as well as vowel training exercises using formant tracking. Similar systems have been presented by Murakawa et al., [64] and by Yoram et al.,[100]. These systems, too, employ various speech processing techniques such as, speech synthesis, to produce modified utterance of a student's speech with the purpose of highlighting intonational mistakes. Other speech processing techniques are spectrograms and F0 contours, which can also be used to visualise speech.

All the systems mentioned so far might combine several pronunciation teaching tasks. However, these tasks are usually not integrated into a well-rounded systems which has been designed according to pedagogic guidelines as discussed in Section 2.2.2. Development of pedagogically more sophisticated systems which fulfill the criteria discussed in Sections 2.1 and 2.2 are only slowly emerging. Some prototype developments which attempt to teach linguistic structure and limited conversation are VILTS [68] and Subarashii [31]. The former aims at teaching French in the form of engaging, flexible and user-centered lessons, where speech recognition is used to provide feedback on a student's fluency. The latter is a dialogue system aimed at teaching conversational Japanese, where speech recognition is used to analyse the student's answer at each stage of the dialogue.

2.3.5 System Validation and Effectiveness measurement

In section 2.3.2, a range of tools to score segmental features of pronunciation have been described. It is important to validate such automatic pronunciation scores with scores taken from human judges. This has successfully been done by [68, 23]. Additionally, in [39], methods have been presented to calibrate different automatic pronunciation scores in order to maximise their correlation with human scores.

The effectiveness of CAPT was formally evaluated in two cases: Murakawa et al., [65], and Simoes et al., [85]. Here attempts to teach the fricative /s/ to Japanese students of English demonstrated that visualising information of the phonetics and the articulation of a particular sound as well as visualising an utterance of this sound by both native speakers and students can help to improve the student's pronunciation. In [85], a study has been carried out which measures the contribution of computer-assisted language learning software in the teaching of pronunciation. The pronunciation of students has been evaluated before and after having been exposed to these computer-assisted prosody exercises. Similar assessment can be done for any type of planned exercises in order to determine their practicality and usefulness.

2.4 Summary

This chapter has discussed the main principles of CALL and pronunciation teaching as well as their influence on the design of computer-assisted pronunciation teaching methods. Moreover, the importance of the integration of technical and pedagogical principles in order to create well-rounded systems has been highlighted. The final section of this chapter discussed the current state-of-the-art in the field of computer-assisted pronunciation teaching, organised according to the main areas of development. The above information about current research challenges in CAPT provides the framework for the development of the algorithms presented in this thesis.

Chapter 3

Recognising Non-native Speech

3.1 Introduction

This chapter provides an introduction to the state-of-the-art and the challenges regarding non-native speech recognition. Since all algorithms presented in this thesis employ hidden Markov models (HMMs), the basics of automatic speech recognition using HMMs are outlined in Section 3.2. Next, in Section 3.3, the characteristics of non-native speech in comparison with native speech are discussed with a focus on phonetic, spectral and temporal differences. Building on the understanding of foreign accented speech as obtained in Section 3.3, the final section of this chapter analyses typical mispronunciations of Spanish and Japanese accented English. These accent types are used for all the experimental work described in this thesis.

3.2 HMM based speech recognition systems

All automatic speech recognition systems consist of the components outlined in Figure 3.1. Firstly, the acoustic waveform is converted into feature vectors in the signal processing front-end. Then, the decoder combines the hidden Markov model set with the language model or grammar network and the dictionary into a recognition network. Finally, the most likely path through the network yields the most likely word sequence for the given input speech.

3.2.1 Front-end signal processing

The first step in the front-end unit of a recognition system is to sample the incoming acoustic waveform at twice the maximum frequency which is desired for processing. The raw digitised speech signal needs to be parameterised in order to extract that information which has been found to be most useful for the statistical modelling.

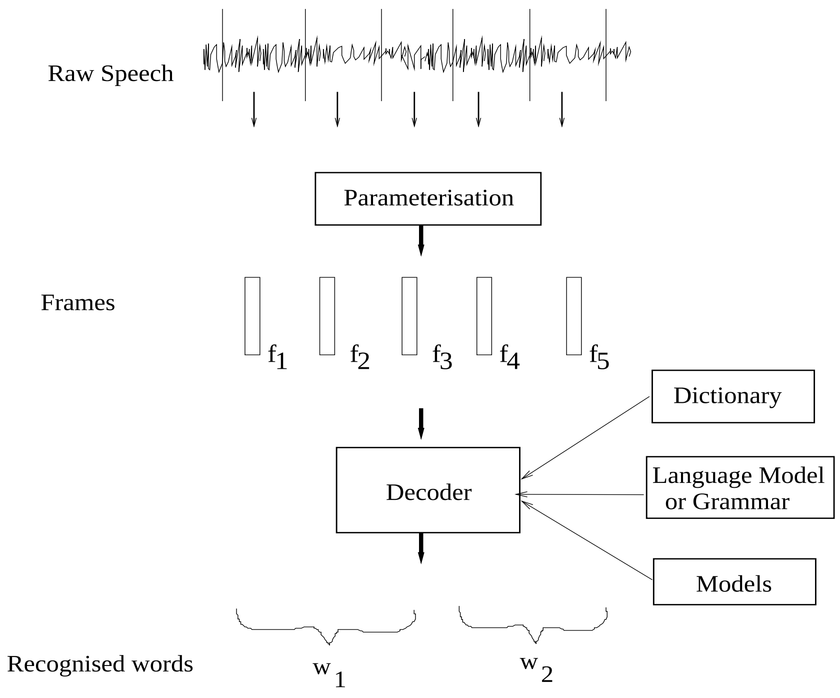

Figure 3.1: Block diagram of a typical recognition system

Methods have been devised which map a segment of 16bit speech samples, called a frame, onto one feature vector. Typical values for frame length are in the range of 10-25 ms. For any frame length in this interval the effect of edge discontinuities when calculating the spectrum of the speech segment is limited because the speech in a frame usually covers at least one pitch period. A longer window length would risk smoothing out rapid spectral changes which are relevant for the statistical modelling. Additionally, in order to avoid discontinuities in the digital speech processing, each frame is filtered with a window function, typically a Hamming window, [74], which weights the samples at the centre of the window more than at the edges.

Given the windowed frames, the next step is to find a parameterisation of a frame which represents the characteristics of sounds which are consistent over many speakers, while throwing away redundant information. Speech parameterisation algorithms are designed to extract phonetic information but not speaker-specific acoustic characteristics. However, this goal can only be approximated. Therefore, current speaker independent models can be seen as a generalisation of the acoustics of all the native speakers which have been used in the training material. This has the effect that non-native speech even if only slightly accented and perfectly intelligible, can yield a considerably worse recognition performance, because the acoustics of this type of speech differs considerably.

Current speech parameterisation methods are mostly based in the spectral domain. The most common approach is to use Mel-frequency cepstral coefficients (MFCCs) to represent the static information of the speech signal, see [74]. The Mel-scale has been designed to model the higher sensitivity of the human ear to absolute changes in the lower frequencies of the speech signal. Thus, the Mel-scale is roughly linear until 1 kHz and it is logarithmic for higher frequencies.

MFCC parameterisation consists of the following steps: Assuming that the speech waveform is stationary for the duration of a frame, each frame is transformed into the spectral domain with the Fourier transform. Then, triangular filters are spaced along the Mel-frequency scale. The output of this filter-bank is then transformed into MFCCs by applying a discrete cosine transform to the logarithm of each filter's output. In order to model the speech dynamics, the first and second order derivatives of the MFCCs are computed over several frames. Then they are appended to the static MFCC features. Finally, subtracting the cepstral mean from the MFCCs leads to cepstral mean normalization, see also [101]. This attempts to reduce the acoustic differences among speakers as well as to compensate long-term spectral effects caused by different microphones and audio channels.

3.2.2 Hidden Markov Models

Over the past decades a number of statistical modeling techniques have been applied to speech recognition systems. However, during the last ten years hidden Markov models have become a *de facto* standard on which most current recognition systems are based. The work presented in this thesis is also based on a speech recognition system consisting of HMMs. In this section, a short overview of the theory behind HMMs is given, for a more detailed description see [74]. Additional technical details about the models used in this thesis can be found in [101].

A Markov model is a finite state machine which makes a state transition once every time unit. Depending on the systems used, observation vectors are emitted from each state or with each transition. The system used in this thesis uses a model where one vector is emitted from each state. If the emission of an observation for a given state is a probabilistic function, then the observation sequence is generated by a statistical process and therefore, any generated state sequence is not directly visible, i.e. it is hidden.

Let a HMM consist of N states, (see for example Figure 3.2), then the HMMs used in this thesis have the following properties:

- The entry state S_1 and the exit state S_N are non-emitting.

- The transition probabilities are defined by a transition matrix $A = [a_{ij}]$, with a_{ij} being the probability of moving from state i to state j

$$a_{ij} = p(s_{t+1} = j | s_t = i) \tag{3.1}$$

(s_t denotes the state occupied at time t) with the following 2 constraints

- The transition probability from S_1 to S_2 is $a_{12} = 1$, i.e. state S_1 is only occupied at time 0. Likewise, the transition probability from S_N to S_N, is $a_{N,N} = 0$.

- Each state has an output probability density function $b_j(\mathbf{o}_t)$ consisting of a single Gaussian or a Gaussian mixture with M components. The output probability of the ith state, b_i, for a speech frame vector $\mathbf{o}$ is therefore given by

$$b_i(\mathbf{o}_t) = \sum_{k=1}^{M} w_{ik} b_{ik}(\mathbf{o}_t) \tag{3.2}$$

The output probability of each mixture component is given by

$$b_{ik}(\mathbf{o}_t) = \frac{1}{(2\pi)^{\frac{n}{2}} \mid C_{ik} \mid^{\frac{1}{2}}} e^{-\frac{1}{2}(\mathbf{o}_t - \mu_{ik})' C_{ik}^{-1} (\mathbf{o}_t - \mu_{ik})} \tag{3.3}$$

where μ_{ik} denotes the mean of mixture component k (vector of length n), w_{ik} the mixture weight and C_{ik} the $n \times n$ covariance matrix. A full covariance matrix models the correlations between elements of the feature vectors whereas in the case of a diagonal covariance matrix the feature vector elements are assumed independent.

Given these definitions, the probability of observing an HMM output string of T speech frame vectors, $\mathbf{O} = \{\mathbf{o}_1, \mathbf{o}_2, \ldots, \mathbf{o}_T\}$ following the state sequence $\theta = (\theta_0 \theta_1 \ldots \theta_T)$ is

$$P(\mathbf{O}, \theta) = a_{\theta_T N} \, \Pi_{t=1}^{T} \, a_{\theta_{t-1}\theta_t} b_{\theta_t}(\mathbf{o}_t) \tag{3.4}$$

Throughout this thesis each phone of a language is modeled by monophone HMMs with 3 emitting states and one multiple componentGaussian mixture per state.

3.2.3 Estimation of HMM parameters

The parameters of HMMs are estimated with the help of large amounts of training data, i.e. speech data for which the transcriptions are known. A standard technique of simultaneously estimating the transition probabilities and the output probability function parameters is Maximum Likelihood Estimation (MLE), a detailed description is given in [74]. The estimation task is equivalent to finding those HMM parameters which maximise the likelihood of the HMMs having generated the training data. MLE will typically find a good local but not necessarily global maximum. Generally, the ML function is very complex and has many local maxima, therefore, good initial model parameters are important. Typically, initial values are based on some globally averaged mean and variance. MLE is usually implemented with the help of the Expectation-Maximisation (EM) algorithm. New parameters are iteratively estimated to improve the likelihood of the training data, using alignments calculated with the current parameters. Then, the parameter set is updated with the new estimates and the re-estimation step is repeated until the change in the parameter estimates is less than some pre-defined threshold. The quality of the ML estimate depends on both the number of parameters which need to be estimated and the number of training data available. As discussed later in Chapter 4, adaptation algorithms are either based on reducing the amount of parameters to be estimated by applying estimation constraints to them, or they are based on introducing prior probabilities to improve the estimation.

3.2.4 Speech Recognition based on HMMs

Speech recognition with hidden Markov models is based on finding that sequence of units of speech which are most likely to have generated the given acoustics in a form of a sequence of observation frames. Denote each unit of speech, which could be a word, a phone or a syllable with w_i (henceforth simply called word) and a sequence of n units by $\mathbf{W} = w_1, w_2, \ldots, w_n$.

Then, a recognition system should decide on that word sequence $\hat{\mathbf{W}}$ which satisfies

$$\hat{\mathbf{W}} = arg \max_{\mathbf{W}} P(\mathbf{W}|\mathbf{O}) \tag{3.5}$$

Using Bayes' formula, the probability of a word sequence given the observations, $P(\mathbf{W}|\mathbf{O})$, can be written as:

$$P(\mathbf{W}|\mathbf{O}) = \frac{P(\mathbf{O}|\mathbf{W})P(\mathbf{W})}{P(\mathbf{O})} \tag{3.6}$$

where $P(\mathbf{W})$ is the probability that the word sequence $\mathbf{W}$ was uttered. This probability represents the language model, which is assumed to be independent from the observation vectors and which is solely based on prior knowledge. $P(\mathbf{O})$ is the average probability that $\mathbf{O}$ was observed. Since there is no other acoustic data beside the given observations, this probability is fixed. Thus, the recognition system has to find

$$\hat{\mathbf{W}} = arg \max_{\mathbf{W}} P(\mathbf{O}|\mathbf{W})P(\mathbf{W}) \tag{3.7}$$

If each distinct word w_i is modeled by a HMM, then a word sequence, $\mathbf{W}$, of length N can be modeled by a concatenation of the HMMs corresponding to the words in this sequence. The set Θ of all K possible states sequences, $\theta^{(k)}$ of length T is given by

$$\Theta = \{\theta^{(1)}, \dots, \theta^{(K)}\} \tag{3.8}$$

Then, recalling equation 3.4 in Section 3.2.2., $P(\mathbf{O}|\mathbf{W})$ is equivalent to the summed probability of all possible output states

$$P(\mathbf{O}|\mathbf{W}) = \sum_{\theta \in \Theta} P(\mathbf{O}|\theta, \mathbf{W})P(\theta|\mathbf{W}) \tag{3.9}$$

In practice this total maximum likelihood is approximated with the help of the Viterbi algorithm, see also [74]. In order to reduce the computational load, this algorithm calculates only the maximum likely state sequence. The likelihood of the best state path to state j at time t from the previous state i at time $t-1$ is recursively computed as

$$\gamma_j(t) = \max_i \gamma_i(t-1) a_{ij} b_j(\mathbf{o}_t) \tag{3.10}$$

where $\gamma_1(0) = 1$, and $\gamma_i(0) = 0$ if $j \neq 1$. The likelihood for a whole utterance of concatenated HMMs then becomes

$$\begin{aligned} \gamma_N(T+1) &= \max_i \gamma_t(T) a_{i,N} \\ &= \max_{\theta \in \Theta} P(\mathbf{O}|\theta, \mathbf{W})P(\theta|\mathbf{W}). \end{aligned} \tag{3.11}$$

This represents an approximation of equation 3.9 where the summation has been approximated by its maximum.
All recognition experiments in this thesis are based on the HTK Toolkit, [101], which provides tools for model parameter estimation as well as for recognition experiments.

This section gave an overview of the main principles of the acoustic modeling in HMM-based continuous speech recognition. One of the aims of this thesis is to investigate extensions of this technology to the challenge of recognising non-native speech. This target requires an understanding of the differences between native and non-native speech. Therefore, the following section discusses the characteristics of foreign-accented speech with regard to its temporal, spectral and phonetic components.

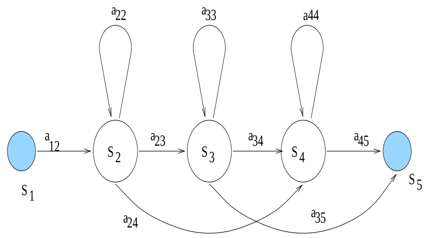

Figure 3.2: An example of a HMM with 5 states, 1 and 5 being non-emitting and with two skip states.

3.3 Characteristics of non-native speech

If asked, most people would be hard pressed to characterise non-native speech. However, any native speaker can easily tell whether somebody's speech is accented or not. Within the linguistic community it is known that in general learners of a second language tend to apply articulatory habits and phonological knowledge of their native language. A range of studies on foreign accented speech have been undertaken by phoneticians in order to analyse foreign-accented speech as opposed to native speech. Typical questions in such studies are for example, see also [77]:

1. What are the acoustical properties of foreign accented speech, i.e. what are the frequency and durational characteristics?

2. Which properties of target language speech by a learner are more or less difficult to understand by native listeners?

Answers to these questions are found by analysing non-native speech with regard to its spectral, temporal and phonetic characteristics.

3.3.1 Spectral characteristics

Comparing spectra of non-native with those of native speech provides a method of visualising differences with regard to intonation, formant structure and prosody. An example of such spectral differences is given in Figure 3.3, where spectrograms of the same word uttered by a native and by a non-native speaker are contrasted. In the word “I” especially the second formant differs between the non-native and native speaker. Whereas the non-native speaker has a fairly constant second formant, the same formant of the native speaker contains a sharp rise towards the end of the vowel. Similarly, in the second example, the second formant of the non-native speaker is constant as opposed to a falling formant by the native speaker.

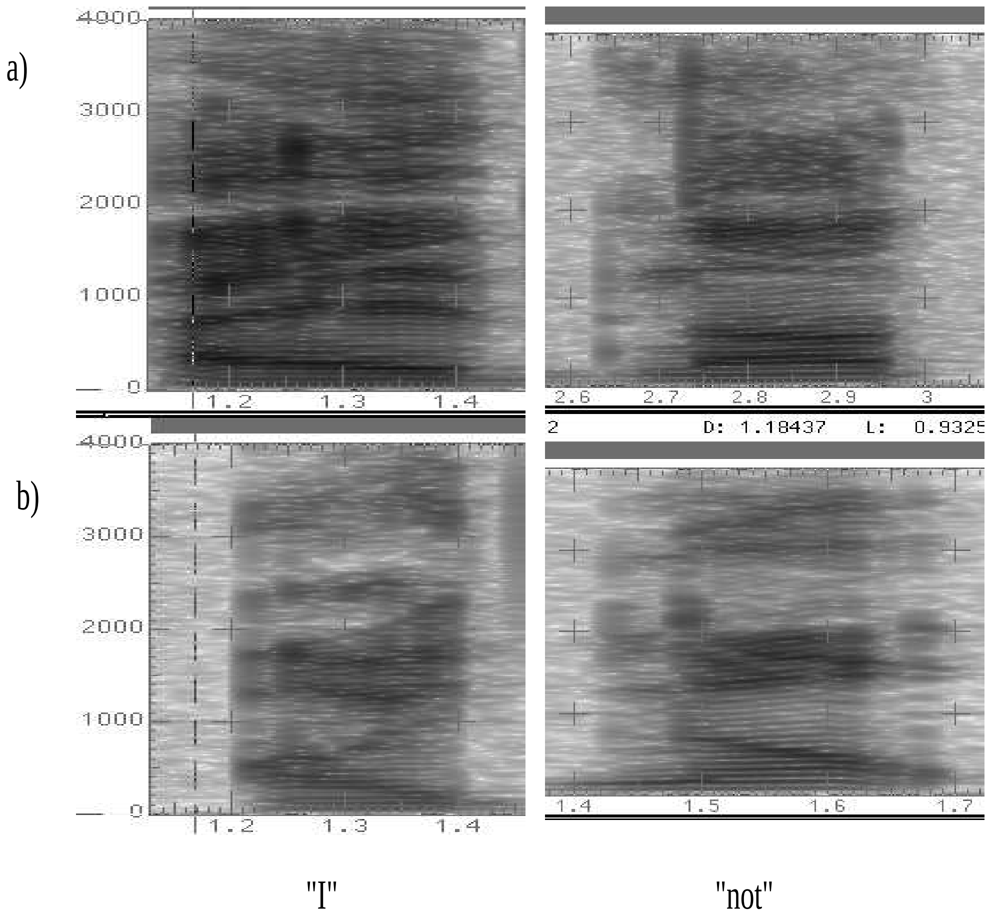

Figure 3.3: Spectral comparison between a) non-native and b) native speaker for the two example words “I” and “not”

This spectral difference between native and non-native speech has been investigated in more detail by Arslan, [10], and Flege, [35]. It has been found that one of the major differences between these two types of speech lies in a different pattern of the second and higher formants, even for fluent and intelligible non-native speech. In [9] it has been shown that the second formant is the most significant resonant frequency which aids the classification of foreign accent. In [35], an example of such frequency differences is given. Here native French speakers speaking English produced lower second formant frequencies for the sound /u/ than native English speakers, because in French /u/ is realised with a lower second formant than in English.

These formant differences can be explained through findings of Fant as described in [9]. Fant showed that small changes in the tongue constriction centre [1] can lead to large shifts in the frequency location F_2 and F_3, whereas the frequency location of F_1 only changes if the overall shape of the vocal tract changes. Generally, non-native speakers can master the overall shape of the vocal tract but they have difficulties with more precise tongue movements. Again this observation was confirmed by Flege, [35], where psycholinguistic experiments showed that phones of the target language are generally perceived as having been produced more authentically if these phones had similar sounding counterparts in the source language.

Given that these spectral differences exist, the question arises as to how important these different formant patterns are for intelligibility. In [87], the influence of spectral features as opposed to temporal features on the intelligibility of a foreign-accented utterance was measured. It was found that spectral characteristics have a significantly larger influence on intelligibility than temporal characteristics, which are discussed in the next section.

3.3.2 Temporal characteristics

Speech by beginners of a foreign language is generally characterised by a much lower rate of speech and by disfluencies such as stops and repetitions. Even speech by considerably more fluent non-native speakers tends to exhibit different temporal patterns than the speech produced by native speakers. For fluent non-native speakers, the main difference can be found in the voice onset time (VOT), [36], or word-final stop release time, [10], the latter usually being much longer for non-native speakers. Even if a similar sound exists in a speaker's source language, this speaker will — even after many years of speaking the target language — still produce the target language sounds with the VOT of his mother-tongue. For example, Flege,

[1] narrowing of the vocal tract due to the tongue

[36], reports that even highly experienced English learners of French produced a significantly longer VOT of /t/ than native French speakers.

On the other hand, temporal differences can influence the intelligibility of accented speech. For example, Rogers et al., [77], and Tajima, [87], showed that intelligibility correlates with vowel durations. In order to demonstrate this correlation, Tajima conducted the following experiment. Chinese accented English was modified with the help of LPC re-synthesis and dynamic time-warping in order to match the durational pattern of the same word sequence uttered by a native speaker. In the same manner, native English speech was modified to match the temporal pattern of the Chinese-accented speech. Then, the intelligibility of the time-wise manipulated speech utterances was judged by native speakers. It was observed that the intelligibility of the modified Chinese stimuli was significantly higher than for the original speech. Likewise, the time-warped version of the native English utterance was less intelligible than the original.

3.3.3 Phonetic characteristics

The last class of non-native speech characteristics concerns the phonetic inventory used by learners of a second language.

In general, research in second language acquisition differentiates the task of learning the sound system of a new language into modifying previously established patterns of production and intonation and into learning how to produce new sounds in the target language, [35, 37]. The first case is much more likely to occur than the second. It usually requires a lot of time and/or effort until learners eventually add phonetic categories to their existing phonetic inventory in order to accommodate second language sounds that differ substantially from any sound in the native language.

An example of the difficulty of producing a sound which does not exist in the source language is given in [35]. The stop /p/ does not exist in Arabic whereas /t/ and /k/ do. A listening test showed that a /p/ spoken by Arabs was significantly less often identified as correct as /k/ or /t/ spoken by Arabs.

So far, only substitutions of inter-language or source language phones have been discussed. However, non-native speech is also characterised by insertions and deletions. Such patterns have been investigated in a study on inter-language phonology by Tarone, [88]. The author observed that depending on the source language of a speaker they tend to either insert vowels or delete consonants, in order to produce a syllable structure related to the source language. For example, Portuguese speakers of English tend to insert vowels whereas Cantonese or Korean speakers demonstrated a tendency for consonant deletion. Further analysis of the errors produced by the non-native speakers in this study showed that on average 70% of the errors can be regarded as being due to language transfer from the source language.

3.4 Typical error statistics of Latin-American Spanish and Japanese

The purpose of the preceding summary of non-native speech characteristics was to obtain an understanding of the characteristics and patterns of non-native speech. Common to all the sources cited so far is the observation that the source language of a language student significantly influences the types of mispronunciations that he or she will make. This observation serves as the basic idea for the non-native acoustic modeling techniques introduced in this thesis.

Because Spanish accented and Japanese accented English will be used for the experimental evaluation of the algorithms developed later in this thesis, this section analyses the patterns of typical mispronunciation patterns for these two accents. Three approaches are introduced which allow the systematic collection of typical mispronunciations in Spanish accented and Japanese accented English. Such lists of typical substitutions can be incorporated in techniques for the detection of mispronunciations or for improved acoustic modeling of non-native speech. The substitution lists based on these three approaches are summarised in Tables 3.1 and 3.2.

3.4.1 Linguistic knowledge

The first approach to find typical mispronunciations, which does not require speech data of a specific accent, is to search the linguistics literature on pronunciation teaching for listings of typical mistakes. For example, in Kenworthy, [53], complete listings of typical mispronunciations in English by speakers of various source languages can be found. These mispronunciations are described with statements like, (see Kenworthy, [53], p.153)

```
There is a sound in Spanish which is a kind of combination
of /b/ and /v/. Learners tend to substitute this sound for
the two English consonants.
```

All these listed typical mistakes can be translated into a table where the first column contains all target language phones, and where the second column contains all possible substitutions. However, such listings as in Kenworthy, [53], usually only refer to major mistakes. For all those target language sounds which are not mentioned, it can usually be assumed that the correct target phone is produced in some approximate way.

An example of the above analysis based on linguistic knowledge is shown in the first and second columns of Tables 3.1 and 3.2. For those fields containing a dash no typical pronunciation error is mentioned in the literature. In the case of the above example typical error description, the Spanish /b/, here denoted by $/b_s/$, has been listed as a possible substitution for the British phones /b/ and /v/.

3.4.2 Transcription analysis

Another method of obtaining listings of typical pronunciation mistakes is based on non-native recordings which have been hand-transcribed by trained phoneticians. Such transcriptions describe the phone sequence which has been uttered as opposed what should have been said according to transcriptions derived from a standard pronunciation dictionary. From the comparison of these two types of transcriptions, statistics can be collected about how often each individual phone has been substituted by another phone. That phone which has been substituted most often is then the one which will be listed as the most likely substitution.

The results of this method are listed in the third column of Tables 3.1 and 3.2. Again, in the fields containing a dash no statistically significant amount of corrections has been been made by the human judges. In the case of our example, the British English /v/ has been listed as the most frequent substitution for the British English /b/, whereas the British English /b/ and /f/ have been listed as frequent substitutions for the British English /v/.

3.4.3 Source language alignment

The final approach to obtaining substitution statistics utilises a small amount of accented speech data in order to calculate two recognition passes over the non-native speech data. In the first recognition pass, the non-native speech is recognised in forced alignment mode using transcriptions derived from a standard pronunciation dictionary. The second recognition pass applied to the same data uses a phone loop consisting only of the phone models of the mother-tongue of the speaker, i.e. in this example Latin-American Spanish or Japanese HMMs. Next, the two resulting alignments are compared in order to count how often each source phone has been aligned with a given target phone. For instance, if the forced alignment of an utterance identifies the phone /a/ between frame 0 and frame 10 and the phone loop then identifies the source phone $/o_s/$ from frame 0 to 8, then the source phone $/\mathrm{o}_s/$ is included in the substitution statistics for target phone /a/. An example of

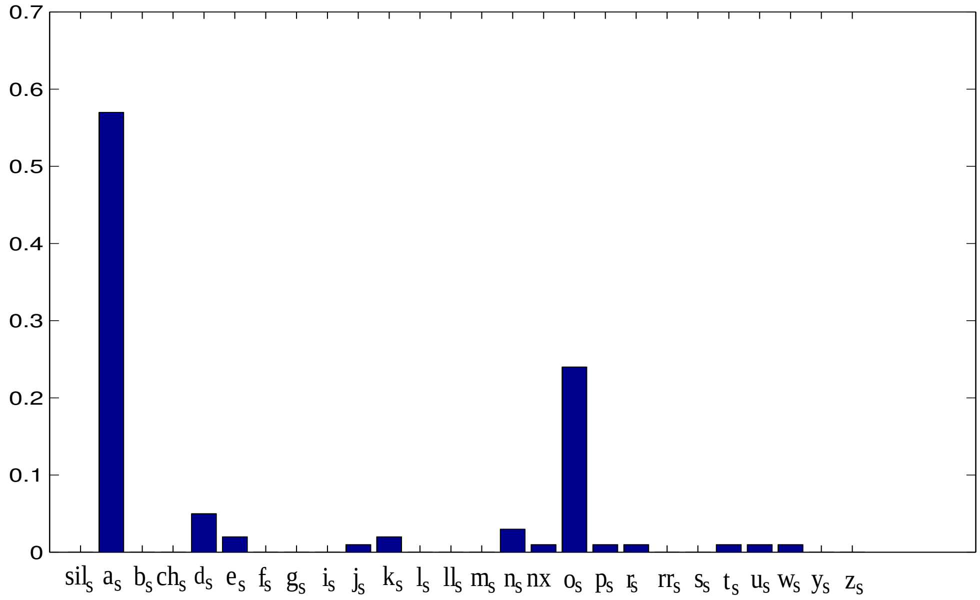

Figure 3.4: Example statistics of how often each Spanish phone was aligned with the British phone /a/

the resulting statistics for the British phone /a/ can be seen in Figure 3.4. Here, the Spanish source phone /a_s/ has been recognised most often in those time segments corresponding to occurrences of the British phone /a/ in the forced alignment pass. However, in about another 30% of the occurrences of the British /a/, the Spanish phone /o_s/ has been aligned with /a/ instead of the Spanish phone /a_s/. This indicates, that the Spanish phone /a_s/ is the most likely substitution for /a/, and that the next likely substitution would be the Spanish phone /o_s/.

Based on such statistics, the most frequently aligned source phones for each target phone are listed as the most likely substitutions in the fourth column of the two tables. With this approach one most likely substitution can be found for each target phone. Therefore, this column does not contain any dashes for missing information.

3.4.4 Automatic mapping derivation

Looking forward to the development of the accent adaptation techniques presented later, it is desirable to derive such substitution statistics automatically without requiring a large amount of time, effort and training data. For instance, the substitution patterns shown in Tables 3.1 and 3.2 are based on about 300 training sentences. Therefore, the third method based on alignment comparisons has been

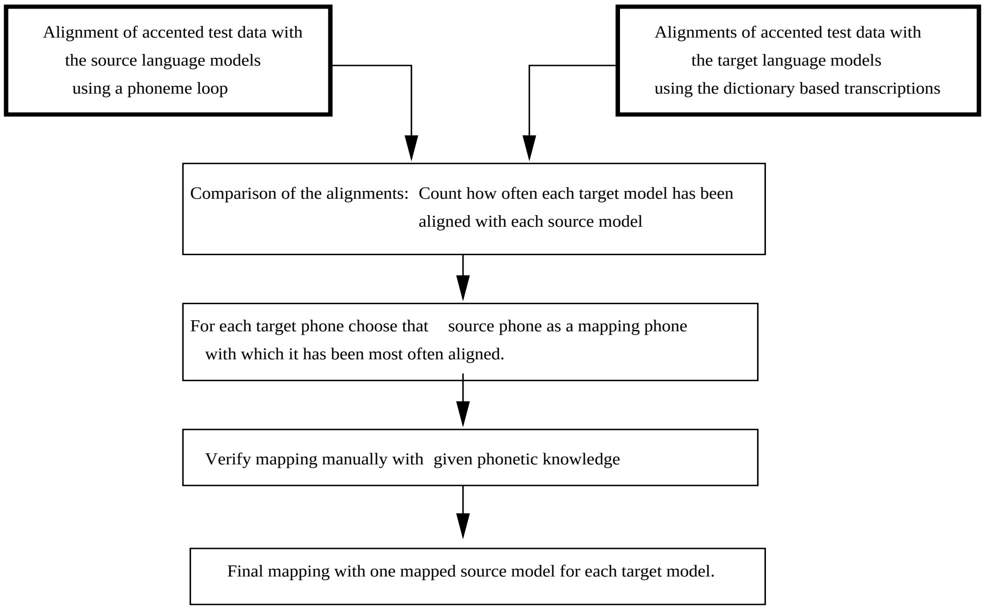

Figure 3.5: Block-diagram for automatic calculation of a mapping

converted into a program which — with a small set of accented speech data as input — automatically outputs a substitution pattern. The block-diagram in Figure 3.5 describes the structure of this program. After the calculation of the two recognition passes, comparison of the alignments yields the substitution statistics. Based on these statistics, the most likely mapping is chosen and optionally manually verified with phonetic knowledge. Thus, if a few hundred sentences of accented speech are given, a mapping can be calculated. Once a mapping exists, it is independent of the type of models used, and it is independent of the recording setup or the task. In other words, the mapping is independent of all those factors on which a general recognition system depends and which therefore make it quite difficult to transfer a recognition system from one task to another.

Summarising this section, the following two conclusions can be drawn. Firstly, the comparison of the predicted errors derived by these three different methods shows fairly high agreement among them. For example for Spanish accented British English, Kenworthy, [53], mentions that people with a Spanish accent tend to substitute the British /b/ with their native /b_s/ sound, which is somewhere between the English /b/ and /v/. In Table 3.1, the Spanish /b_s/ is listed as an error mentioned in the literature (second column), the human judges often marked the /b/ with /bv/, their made-up symbol for this in-between sound (third column) and finally, in the alignment with Spanish models, the Spanish /b/ was most often aligned for the English /b/ (forth column).

Secondly, there exist no absolute truth about which is the single the most likely substitution for each target phone. Therefore, none of the above methods can claim to yield more reliable substitution patterns than the others. Additionally, it can be the case that there is no single most likely substitution but that there might be several equally likely substitutions for a given target phone. Thus, if a mapping for a new language pair is needed, the necessary information can be found with the help of this automatic method, possibly augmented by linguistic knowledge. In the case that no accented data is available at all, only the method based on linguistics can be applied.

The same techniques as outlined above have been applied to obtain typical error statistics for Japanese accented speech. In Table 3.2 the results of such an analysis are shown. For example, it can be seen from the table that the Japanese /r-l/ sound is often substituted for the English /r/ sound. This is a very typical error for Japanese speakers of English.

3.5 Summary

This chapter presents the foundation for the acoustic modeling algorithms which will be presented in the following chapters. After a discussion of HMM-based continuous speech recognition, the main differences of non-native speech as opposed to native speech have been analysed with regard to its temporal, spectral and phonetic characteristics. It is the combination of these characteristics of non-native speech which causes a large degradation of recognition performance when using speech recognition systems trained on native speech. This observation motivated the development of the algorithms presented next in Chapter 4. In the last section of this chapter, three techniques have been developed which allow the tabulation of typical substitutions or mispronunciations of non-native speakers coming from the same source language.

British Phone	Phonetics Literature	Correction based	Native Substitutions
ɑː	-	ae	a$_s$
ae	e	ɑː ʌ	a$_s$
ʌ	ə ɒ	ə ɒ	a$_s$ o$_s$
ɔː	-	ɒ r	o$_s$
aʊ	-	-	o$_s$
ə	-	ae e ɒ uː	$_s$a e$_s$ r$_s$
aɪ	-	ɪ	a$_s$
b	b$_s$	v	b$_s$
ʧ	-	-	tʃ$_s$
d	ð	ð del	d t
ð	-	d	d$_s$ t$_s$
eə		er	e$_s$ a$_s$
e	eɪ ae	-	e$_s$ a$_s$ d$_s$
ɜː		r	a$_s$ r$_s$ d$_s$ o$_s$
eɪ	e	aɪ e ə	e$_s$ ll$_s$
f	-	-	f$_s$
g	-	k	d$_s$ g$_s$
h	del x	x	x$_s$
ɪə		e ɪ	i$_s$ j$_s$ e$_s$
ɪ	iː	ə iː e	e$_s$ i$_s$
iː	ɪ	ə ɪ	e$_s$ i$_s$
ʤ	tʃ$_s$ y	-	tʃ$_s$
k	del	g del	k$_s$
l	-	-	l$_s$ o$_s$
m	-	-	m$_s$ n$_s$
n	-	del ŋ	n$_s$
ŋ	n	n	n$_s$
ɒ	-	ʌ ae ə	o$_s$
əʊ	-	ɔː ɒ ə	o$_s$
ɔɪ	-	-	o$_s$
p	p$_s$	-	p$_s$ t$_s$
r	r$_s$ rr$_s$	-	r$_s$ rr$_s$
s	s z	z del	s$_s$ z $_s$
ʃ	ʧ s tʃ$_s$	-	tʃ$_s$
t	del	d del ð t	tʃ$_s$ t$_s$
θ	f s	-	f$_s$ s $_s$
ɔə	-	-	u$_s$ w$_s$
ʊ	uː	uː	u$_s$ o$_s$
uː	ʊ	ɔə ə	u$_s$
v	b$_s$	b f	b$_s$ t$_s$
w	b g+w	-	w$_s$
y	-	ʤ del	i$_s$ ll$_s$
z	s	s del	s$_s$

Table 3.1: Typical error statistics for Spanish accented English, using three difference sources of knowledge

British Phone	Phonetics Literature	Correction based	Native Substitutions
ɑː	ɔː a_j	ae	a_j
ae	-	-	a_j
ʌ	a_j	ɑː	a_j
ɔː	a_j	ɒ	oo_j
aʊ	a + u	-	a_j
ə	?	ae	a_j
aɪ	-	-	a_j
b	b_j	-	p_j
ʧ	ʧ	-	ch_j
d	d_j	-	ch_j
ð	z d	z	t_j ch_j s_j
eə	-	-	ee_j
e	i eɪ	-	ee_j e_j
ɜː	ɑː aʊ	ɑː	aa_j
eɪ	eɪ	e	ee_j
f	h	-	s_j f_j
g	g_j	-	k_j
h	h_j	-	h_j
ɪə	-	ə	ii_j e_j
ɪ	i	e	i_j
iː	i	ɪ	ii_j
ʤ	ʤ	-	ch_j
k	k_j	-	k_j
l	l_j	r	r_j oo_j
m	m_j	-	m_j
n	n_j	-	nn_j
ŋ	ŋ	-	nn_j
ɒ	o_j	-	o_j
əʊ		ɒ	oo_j
ɔɪ	o_j	-	oo_j
p	p	-	p_j
r	l d	del	r_j a_j
s	s	-	s_j sh_j
ʃ	ʃ	s	sh_j qsh_j
t	t	-	ch_j
θ	s t	-	sh_j s_j
ɔə	-	-	uu_j
ʊ	u	-	uu_j
uː	uː	-	uu_j
v	b v	b	b_j d_j
w	w	-	w_j
y	y	-	y_j ii_j
z	z	ʤ	s_j

Table 3.2: Typical error statistics for Japanese accented English, using three difference sources of knowledge, "-" denotes that no info was available, $-j$ denotes Japanese sounds

Chapter 4

Adaptation to non-native speech

4.1 Introduction

Current speaker independent recognition systems are known to perform considerably worse when recognising non-native speech. The large performance drop in recognition accuracy of non-native speech in comparison with native speech, as shown in [44, 14] demonstrates the need to improve the acoustic modelling of non-native speech. In Chase [18], it was shown that such performance deterioration is due to bad acoustic modeling.

One way to improve non-native recognition accuracy is to retrain speaker-independent acoustic models of native speech with non-native speech. In [30], retraining speaker-independent models of American English with Japanese accented English yielded a large improvement in recognition performance, bringing it up to the level of native recognition. However, this approach still requires a fairly large amount of non-native speech, and this is usually not available.

If a recognition task is restricted to recognising a single speaker, recognition accuracy can be improved with speaker adaptation techniques. In this chapter, two adaptation techniques are developed which deploy additional information about a non-native speaker's source language. These techniques are based on speech recognition using hidden Markov models as explained in Chapter 3. For such HMM-based recognition, a number of speaker adaptation algorithms have been developed. In Section 4.2, an overview of the state-of-the-art in speaker adaptation is given in order to provide the framework within which the new algorithms can be located.

These two new adaptation techniques are based on the observation that non-native speech contains phones which represent a mixture of phones of the speaker's target language and source language. Therefore, these new algorithms combine the acoustic models of a speaker's source and target language. Their derivation is presented in Sections 4.3 and 4.4.

These algorithms require a mapping based on typical substitution patterns between the source and target language of a speaker. The final section of this chapter will describe how to obtain such mappings by exploiting the knowledge of substitutions typical for speakers with the same source language as discussed in Section 3.4. Performance results for these new adaptation techniques will be presented later in Chapter 6.

4.2 State-of-the-art of adaptation

Within the last decade a lot of research effort has been devoted to the development of adaptation algorithms. The driving force behind these efforts has been the well-known fact that speaker-dependent recognition systems tend to perform considerably better than speaker-independent systems. Furthermore, adaptation can be required for varying channels, environmental noise or varying speaker characteristics. This section outlines the main groups of adaptation algorithms in order to provide the theoretical framework for the novel adaptation algorithms of foreign-accented speech presented in this chapter. The summary below does not represent an exhaustive discussion of all approaches published; rather it attempts to provide an overview of the current state-of-the-art in speaker adaptation.

Because the work in this thesis uses speech recognition systems based on hidden Markov models with output probability functions modeled by continuous Gaussian density functions, the following discussion is mostly restricted to algorithms of speaker adaptation for systems with continuous density HMMs.

Approaches to speaker adaptation can be grouped into two major classes: *feature space* algorithms which apply a transform to feature vectors, often called speaker normalisation, and *model space* algorithms which transform HMM parameters. In a more detailed classification the latter group of algorithm can be divided into:

- Transformation based adaptation
- MAP (Maximum A Priori) adaptation
- Predictive adaptation

4.2.1 Feature space algorithms

Feature space algorithms have often been called *speaker normalisation*, because their target is to normalise physiological differences among speakers such as varying vocal tract lengths and the size of the mouth or the nasal cavity. Furui, [42], introduced cepstral-mean based normalisation, which is nowadays widely used. In this technique, inter-speaker variability in the logarithmic spectral domain is modeled an unconstrained parameter alteration. A related normalisation technique, [57], compensates variations in the length of the vocal tract by applying a linear frequency warping factor to rescale the Mel-frequency scale which is used to parameterise a speaker's speech. There also exist many other methods for computing spectral transformations. For example, Choi et al., [19] present three different techniques, minimum mean square error, canonical correlation analysis and multi-layer perceptrons to transform the feature vectors of the input speech to the spectral space of the training speakers. Most recently, a new normalisation algorithm has been introduced by [63]. This algorithm represents a generalisation of the vocal tract length normalisation technique mentioned above, see [57]. The main difference between this technique and the frequency warping technique is that speaker normalisation is realised with linear transforms in the logarithm spectral domain instead of a single warping factor.

In general, speaker normalisation has been found to yield only fairly small improvements in recognition accuracy. However, it has been shown, see for instance [33] or [73], that applying feature space and model space adaptation in sequence can yield higher performance than either method on its own.

4.2.2 Model space algorithms

Transformation based adaptation

The basic principle of transformation based adaptation is to use adaptation data to estimate transformation matrices. These matrices are applied to the codebook entries in vector quantisation or to HMM mean and variance vectors in order to move all these vectors to a new spectral space which is better suited to model the speech of a particular speaker. For global adaptation, a single matrix is used to transform all HMM parameter vectors or codebook entries. More detailed adaptation is possible if different transformation matrices are used for different groups of codebook entries or HMM parameter vectors. Existing transformation based adaptation schemes vary in the way that the transformation matrices are calculated as well as in the manner by which the parameter vectors are grouped for partial transformation matrices.

Two transformation based adaptation schemes for recognition systems using vector quantisation were introduced by Schwartz et al., [83], and Nakamura et al., [67]. A codebook entry of a vector quantisation codebook can be interpreted as a quantised spectrum. Thus, a transformation matrix can be regarded as a transformation from the quantised space of the input speech to the quantised space of the prototype speaker's spectrum. Different techniques can now be used to calculate a spectral mapping from the prototype codebook to the codebook of a new speaker. In [83], the matrix of probabilities that a new speaker will produce one quantised spectrum given a codebook entry of a prototype speaker is calculated. Additionally, the idea of using different transforms for different phones was introduced in this work, too. In [67], the mapping from input speaker to standard speaker has been computed using neural networks and fuzzy vector quantisation as opposed to a matrix of probabilities. Bellegarda et al., [12], introduced the following interesting notion: Rather than normalising a new input speaker to the training speakers, with the consequence that only one global transform can be estimated for each new speaker due to limited adaptation data, they propose the use of a phone-based piecewise linear mapping from the normalised speakers of the training data to the new speaker's spectral space.

The last paragraph described adaptation techniques for speech recognition systems using vector quantisation. All following references use recognisers based on continuous density HMMs. For example, in Zhao, [104], spectral variations for each phone are modeled with phone-dependent transforms which adapt Gaussian means and covariance matrices. Another method to calculate spectral transformations called Maximum Likelihood Linear Regression (MLLR), was proposed by Leggetter et al., [58]. Here, transforms are used to update the means of speaker-independent HMM Gaussian mixture components to make them speaker-dependent. This technique has been shown to significantly increase recognition accuracy even with relatively little adaptation data. The transformation matrices estimate the rotation and the shift of the speaker-independent spectral space to the speaker-dependent spectral space. With few adaptation sentences only a single global transformation matrix can be estimated. If more adaptation data are available, more transformations can be estimated; at most, one transformation for each Gaussian component. A very similar approach of reducing the mismatch between the acoustics of a new speaker and a HMM set was also introduced in [80].

MAP adaptation

Whereas MLLR has been shown to perform well for adaptation with relatively few adaptation sentences, Maximum a Posteriori (MAP) adaptation, as derived in [43], has been shown to slightly outperform MLLR for larger amounts of adaptation material. MAP adaptation uses the training data of the speaker-independent models to estimate the parameters of the prior densities. These are then used to estimate the newly adapted HMM parameters. The largest gain was obtained when updating the means of a HMM model set, additional adaptation of variances yielded only small improvements. Therefore, the algorithms presented in this chapter only update the means of HMMs, not the variances.

Because of these reciprocal adaptation characteristics of MAP and MLLR, both Thelen et al., [89] and Ishii et al., [48], proposed combining these two methods. By computing the MAP estimate after the MLLR transformation has been carried out, additional performance gains were obtained. A combination of MAP and MLLR can also improve rapid adaptation if a subset of MLLR adapted training speakers which are close to the test speaker is used to increase the amount of available adaptation material for MAP estimation, see [71].

Predictive adaptation

In recent years several publications introduced methods for rapid adaptation based on a combination of MAP and predictive techniques, see [21, 4]. Here linear regression models are built to model the relationship between different phones. These regression models are used to predict sounds which do not occur in the adaptation material. This scheme employs correlations to find the best estimate of an unheard sound, using either Maximum Likelihood or MAP estimation. These approaches are driven by the assumption that previously unheard sounds can be predicted with the help of models of related sounds. This assumption is the basis of the new algorithms introduced in the next sections, too.

4.2.3 Adaptation of non-native speech

Several of the above described adaptation techniques have been applied to the non-native speech recognition task, too. Firstly, the combination of MLLR and MAP adaptation by Thelen et al, [89], was tested on non-native speakers, and shown to give a significant reduction in the word error rate. On the other hand, in [91], adaptation based on a combination of MLLR and MAP yielded no significant recognition improvements for Italian-accented German, due to a large acoustic mismatch, whereas adaptation for native Italian or German speakers yielded significant improvements.

In [3], a nonlinear transformation was implemented using multi-layer perceptrons, in order to model the mismatch between native and non-native speech. This approach yielded only modest improvements over linear transformation techniques. Additionally, Neumeyer et al., [69], compared a range of feature and model space adaptation algorithms with regard to their performance on non-native data. The authors observed that using a full transformation matrix to adapt Gaussian mixture component means, followed by a stochastic update of the variance and finally, a

MAP estimation of all these parameters can reduce the non-native WER down to the level of native speech. However, the results required at least twenty adaptation sentences. Similarly, Leggetter et al., [60], Zavaliagkos et al., [103] and Digalakis et al. [29] reported significant improvements inrecognition accuracy onthe same Spoke 3 corpus of the DARPA 1994 evaluation. Also, the non-native speech in this experiment was taken from the Wall Street Journal Corpus and contained fairly fluent non-native speakers, not "beginner level" students of English.

The above overview highlighted several aspects which are important for the success of any new adaptation system:

- Capability for fast adaptation
- Use of phone-dependent transforms
- Combination of speaker normalisation followed by model-based adaptation

All these three aspects were taken into consideration in the development of two new adaptation algorithms: *Linear Model Combination* and *Model Merging*. These are both schemes to adapt the means vectors of HMMs. They are described in Sections 4.3 and 4.4, respectively. Both techniques use a mapping to combine each HMM of the target language with a HMM of the source language. The difference between these two techniques lies in the manner in which each pair of acoustic models of the target and source language are combined. Section 4.5 describes how to calculate different types of mappings. Both methods incur the same computational load as the standard MLLR algorithm once a mapping between the two respective languages has been found.

4.3 Linear Model Combination (LMC)

Transformation based adaptation schemes, such as for example MLLR, transform speaker-independent models into speaker-dependent models by using a transformation matrix which shifts and rotates HMM mean vectors. Usually, maximum likelihood estimation is employed to find a transformation matrix which transforms the speaker-independent space to a locally optimal space of the new speaker.

The approach presented here also updates HMM mean vectors, but constrains the transformation to a linear shift along the space which lies between the model sets of the two languages, see also Figure 4.1. Incorporating information from both languages provides more direction in the acoustic space, which might be especially helpful in the case of a bad acoustic match between the speaker-independent models and the non-native speech. It is hoped that this additional constraint helps to find better local maxima than in the case of an unconstrained transform. Additionally, due to the constrained search space, the LMC algorithm requires the estimation of fewer parameters which may allow faster adaptation with small amounts of data.

In addition to a speaker-independent model set of the target language, the LMC algorithm requires a speaker-independent model set of the source language and a mapping between the two languages. This mapping defines which HMM mean vector of the source language has to be combined with each HMM mean vector of the target language.

This combination of two mean vectors is expressed as a weighted linear combination of each vector element. The combination weights are estimated with the help of the adaptation data. Define $\mathbf{B_s}$ as a diagonal matrix for state s in order to map from target language mean vector μ_{T_s} of state s to source language mean vector μ_{S_s}. Then, the jth diagonal element $b_{s,jj}$ represents the linear combination weight of the combination between source and target mean vector. In Figure 4.1 the combination of mean vectors has been illustrated for the case of HMMs with a Gaussian mixture output density per state. It can be seen that the mixture component mean vectors of the newly combined mixture are located on a straight line which combines a mean vector of a target language mixture with a mean vector of a source language mixture.

With this definition of the combination matrix, $\mathbf{B_s}$, a new estimate of a mean vector is defined by

$$\tilde{\mu_s} = \mathbf{B_s}(\mu_{S_s} - \mu_{T_s}) + \mu_{T_s} \tag{4.1}$$

Given this definition of a new mean vector estimate, the remainder of this section will derive the estimation formula of $\mathbf{B_s}$ for HMMs with a single Gaussian output density as well as for HMMs, where the output probability function of each state is modeled with a Gaussian mixture.

4.3.1 Derivation of the combination matrix for single Gaussians

The mean vector of a single Gaussian output probability function will be adapted by a linear combination of a mean vector of the source language and a mean vector of the target language. It is assumed that the mapping of target language means to source language means is calculated according to the methods discussed in Section 4.5.

Let M_T be a model set of the target language which contains Q_T models, and M_S a model set of the source language with Q_S models. Assume a continuous density HMM with N states, each state having a single Gaussian output probability function.

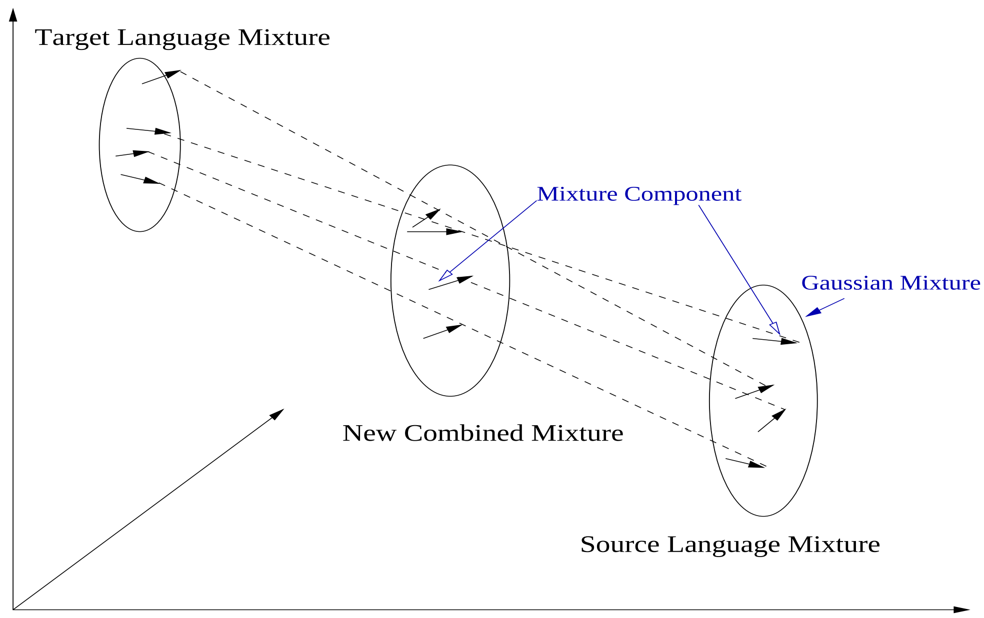

Figure 4.1: Linear combination of mixture component mean vectors

The transition probability between two states i and j is a_{ij} and the Gaussian output probability is $b_i(\mathbf{o})$, see also Section 3.2.2.

Denote the current set of HMM model parameters by λ and a re-estimated set of model parameters as $\bar{\lambda}$. Thus, the re-estimated transition probabilities and output distributions are defined as $\bar{a_{ij}}$ and $b_s(\bar{\mathbf{o}}_t)$ respectively. The sequence of states used to generate $\mathbf{O}$ is given by

$$\theta = (\theta_0 \theta_1 \dots \theta_T) \tag{4.2}$$

where $\theta_0 = 1$. The probability $P(\mathbf{O}, \theta|\lambda)$ of generating the observed speech frame sequence $\mathbf{O}$ while following the state sequence θ is

$$P(\mathbf{O}, \theta|\lambda) = a_{\theta_T N} \prod_{t=1}^{T} a_{\theta_{t-1}\theta} b_{\theta_t}(\mathbf{o}_t) \tag{4.3}$$

If all possible state sequences of length T are denoted by the set Θ, the total probability of the model set generating the observation sequence is

$$P(\mathbf{O}|\lambda) = \sum_{\theta \in \Theta} P(\mathbf{O}, \theta|\lambda) \tag{4.4}$$

This is the objective function to be maximised during adaptation. This maximisation can be achieved using an iterative procedure such as the Baum-Welch algorithm, a specific instance of the Estimation Maximisation (EM) algorithm, for a more detailed description see [74]. It is helpful to define an auxiliary function $Q(\lambda, \bar{\lambda})$, [28]:

$$Q(\lambda, \bar{\lambda}) = \sum_{\theta \in \Theta} P(\mathbf{O}, \theta|\lambda) log(P(\mathbf{O}, \theta|\bar{\lambda})) \tag{4.5}$$

Estimating the diagonal elements of the combination matrix $\mathbf{B_s}$ is now based on iteratively maximising this auxiliary function by improved estimates of $\mathbf{B_s}$ and forming a new auxiliary function with the improved estimates of $\mathbf{B_s}$.

A re-estimation expression for the diagonal elements of $\mathbf{B_s}$ is found through differentiating $Q(\lambda, \bar{\lambda})$ with respect to $\mathbf{B_s}$ using equation 4.1. To do so, define the probability of occupying state s as:

$$\gamma_s(t) = \frac{1}{P(\mathbf{O}|\lambda)} \sum_{\theta \in \Theta} P(\mathbf{O}, \theta_t = s|\lambda) \tag{4.6}$$

In [58], it has been shown that the auxiliary function can be rewritten as:

$$Q(\lambda, \bar{\lambda}) = \sum_{i=1}^{N} Q_{a_i}[\lambda, \{\bar{a}_{ij}\}_{j=1}^{N}] + \sum_{j=1}^{N} P(\mathbf{O}|\lambda) \sum_{t=1}^{T} \gamma_j(t) log\ \bar{b_j}(\mathbf{o}_t) \tag{4.7}$$

where

$$\begin{aligned} &Q_{a_i}[\lambda, \{\bar{a}_{ij}\}_{j=1}^{N}] = \\ &\sum_{\theta \in \Theta} P(\mathbf{O}, \theta_T = i|\lambda) log\ \bar{a}_{iN} + \sum_{\theta \in \Theta} \sum_{t=1}^{T} \sum_{j=1}^{N} P(\mathbf{O}, \theta_{t-1} = i, \theta_t = j|\lambda) log\ \bar{a}_{ij} \end{aligned} \tag{4.8}$$

To estimate the matrix $\mathbf{B_s}$, it is necessary to differentiate $Q(\lambda, \bar{\lambda})$ with respect to $\mathbf{B_s}$ and equate it to zero (μ_s and $\mathbf{C_s}$ denote the mean and covariance of the output density function $b_s(\mathbf{o_t})$). Because only the second part of equation 4.7 is a function of the output probability function and thus of the combination matrix, the derivative of the auxiliary function has the following form

$$\begin{aligned} \frac{dQ(\lambda, \bar{\lambda})}{d\mathbf{B_s}} &= \frac{d}{d\mathbf{B_s}} P(\mathbf{O}|\lambda) \sum_{t=1}^{T} \gamma_s(t)\ log\ \bar{b_s}(\mathbf{o}_t) \\ &= \frac{d}{d\mathbf{B_s}} P(\mathbf{O}|\lambda) \sum_{t=1}^{T} \gamma_s(t)[n\ log(2\pi) + log|\mathbf{C}_s| + h(\mathbf{o}_t, s)] \end{aligned} \tag{4.9}$$

with the definition of $h(\mathbf{o}_t, s)$ as

$$h(\mathbf{o}_t, s) = (\mathbf{o}_t - \tilde{\mu}_s)' \mathbf{C_s}^{-1} (\mathbf{o}_t - \tilde{\mu}_s) \tag{4.10}$$

This scalar is the only part of the derivative which contains the re-estimated mean, $\tilde{\mu_s}$. Thus, $h(\mathbf{o}_t, s)$ is the only component of the derivative which is a function of the combination matrix $\mathbf{B_s}$. Assuming a diagonal covariance matrix, the derivative becomes

$$\frac{dh(\mathbf{o}_t, s)}{d\mathbf{B_s}} = \frac{d}{d\mathbf{B_s}} \left[\sum_{i=1}^{N} c_{ii}(o_t(i) - \tilde{\mu_s}(i))^2 \right] \tag{4.11}$$

Substituting equation 4.1 for $\tilde{\mu_s}$ in equation 4.11 gives the derivative of this with respect to each diagonal element of $\mathbf{B_s}$ (c_{jj} denotes the diagonal elements of the inverse covariance matrix $\mathbf{C_s^{-1}}$)

$$\frac{\delta h}{\delta b_{jj}} = c_{jj} \left[o_t(j) - b_{jj}(\mu_S(j) - \mu_T(j)) + \mu_T(j) \right] (\mu_S(j) - \mu_T(j)) \tag{4.12}$$

Plugging equation 4.12 into equation 4.9 yields a set of N equation for the N diagonal elements of $\mathbf{B_s}$. The equation for the j-th element is

$$\sum_{t=1}^{T} \gamma_s(t) \left[o_t(j) - b_{jj} \left\{ \mu_S(j) - \mu_T(j) \right\} - \mu_T(j) \right] \left\{ \mu_S(j) - \mu_T(j) \right\} c_{jj} = 0$$

$$\sum_{t=1}^{T} \gamma_s(t) \left[o_t(j) - \mu_T(j) \right] \left\{ \mu_S(j) - \mu_T(j) \right\} = \sum_{t=1}^{T} \gamma_s(t) b_{jj} \left\{ \mu_S(j) - \mu_T(j) \right\}^2 \tag{4.13}$$

This yields the re-estimation formula for each diagonal element of $\mathbf{B_s}$

$$b_{jj} = \frac{\sum_{t=1}^{T} \gamma_s(t) \left[o_t(j) - \mu_T(j) \right]}{\sum_{t=1}^{T} \gamma_s(t)(\mu_S(j) - \mu_T(j))} \tag{4.14}$$

Once $\mathbf{B_s}$ has been estimated using the adaptation data, the new estimates of the HMM mean vectors are found using equation 4.1.

Tied combination matrices

Given the problem of data sparseness, it is desirable to extend the above derivation to the case of tied combination matrices.
If a combination matrix $\mathbf{B_s}$ is shared by R states $\{s_1, s_2 \ldots s_R\}$ equation 4.13 becomes:

$$\sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_{s_r}(t) \left[o_t(j) - b_{jj} \left\{ \mu_{S_r}(j) - \mu_{T_r}(j) \right\} - \mu_{T_r}(j) \right] \left\{ \mu_{S_r}(j) - \mu_{T_r}(j) \right\} c_{r_{jj}} = 0$$

Solving for b_{jj}

$$\sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_{s_r}(t) c_{r_{jj}} \left[o_t(j) - \mu_{T_r}(j) \right] (\mu_{S_r}(j) - \mu_{T_r}(j)) = \sum_{r=1}^{R} \sum_{t=1}^{T} \gamma_{s_r}(t) c_{r_{jj}} (\mu_{S_r}(j) - \mu_{T_r}(j))^2 b_{jj} \tag{4.15}$$

Therefore

$$b_{jj} = \frac{\sum_{r=1}^{R}\sum_{t=1}^{T}\gamma_{s_r}(t)c_{r_{jj}}\left[o_t(j) - \mu_{T_r}(j)\right]\left(\mu_{S_r}(j) - \mu_{T_r}(j)\right)}{\sum_{r=1}^{R}\sum_{t=1}^{T}\gamma_{s_r}(t)c_{r_{jj}}\left(\mu_{S_r}(j) - \mu_{T_r}(j)\right)^2} \tag{4.16}$$

4.3.2 Extension to Gaussian mixtures

The above derivation of the estimation of the combination matrix can be extended to the case of Gaussian mixtures in a straightforward way, because each weighted mixture component can be pictured as one weighted state, where all these weighted states are connected in parallel. For an equivalent derivation see [58]. The change affects only the indices in equation 4.16.

4.4 Model Merging (MM)

This section proposes a second adaptation technique which is also based on the idea of combining the model sets of source and target language of a non-native speaker. Similar to LMC, the additional requirements of this algorithm are a speaker-independent model set of the source language and a mapping between the two languages. Instead of combining HMM mean vectors to obtain better acoustic models as it is done in LMC, in this method the Gaussian mixture of each state of the target language is merged with a state of the corresponding model of the source language according to the given mapping. This merging of two Gaussian mixtures yields a new mixture with twice as many components as in the original mixtures.

A related adaptation technique is described in [66]. Here the size of a Gaussian mixture associated with a state is inflated by copying mixture components from other mixtures. The choice of which mixture components to use for the inflation is based on a criterion of minimised frame-level errors. Finally, the weights of all mixture components are re-estimated with Baum-Welch re-training. Recently, similar work has been carried out by Saraclar et al., [81]. There, too, two model sets were combined to form a new model set by merging Gaussian mixtures. However, in this case the two model sets which were used for merging had been trained for the same language. The difference between the two model sets was that each set was trained with a different phone-level transcription of the training material.

This MM algorithm consists of the following steps:

1. Combine each adapted target mixture m_t with M_t components with its corresponding Gaussian mixture m_s containing M_s components according to a mapping. Then, the new output probability density function for state i is

given as

$$b_i(\mathbf{o_t}) = \alpha \sum_{k=1}^{M_t} w_{m_t,k} b_{m_t,k}(\mathbf{o}_t) + (1-\alpha) \sum_{l=1}^{M_s} w_{m_s,l} b_{m_s,l}(\mathbf{o_t}) \tag{4.17}$$

with the initial value for α as $\alpha_{ini} = 0.5$. Experiments in Section 6.5.3 will show that the initial value of α does not influence the recognition performance significantly.

2. Use MLLR adaptation to adapt the mixture component mean vectors.

3. Use the same adaptation sentences to re-estimate the weights of the new mixture with the standard re-estimation expression for Gaussian mixture weights ([74], p. 351):

$$\tilde{w_{jk}} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T} \sum_{k=1}^{M} \gamma_t(j,k)} \tag{4.18}$$

where $\gamma_t(j,k)$ denotes the probability of occupying mixture k of state j at time t.

4. Cut off all mixture components whose weight is below a given threshold.

The reason for taking out all mixture component with a weight below a given threshold is that the re-estimated weights of these mixture components can vary by several orders of magnitude. In Figure 4.4 an example distribution is given of the re-estimated weights for state 2 of the HMM modeling the British-English phone /b/. It can be seen that the weights vary within four orders of magnitude whereas the original weights typically only vary within one order of magnitude. Given this large range of the re-estimated mixture component weights, it is likely that reducing the number of mixture components by cutting of the components with minimal weight will not decrease the performance significantly. The performance might even increase slightly because the resulting output probability functions become narrower due to the reduced number of mixture components. Thus, the discrimination ability of the newly merged models can be increased. Note that in the case of extending the model merging technique to triphones, weight re-estimation can be difficult due to data sparseness.

4.5 Mapping from target to source language

Both algorithms, LMC and MM, as derived in the past two sections, require a mapping between the target and source language. This section discusses how to

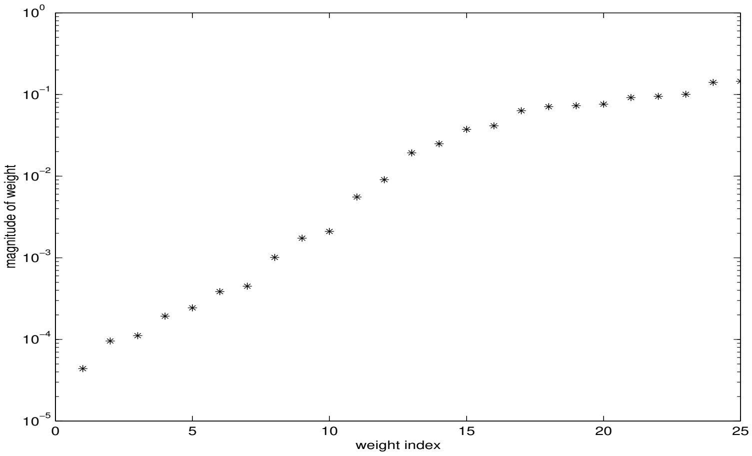

Figure 4.2: Distribution of re-estimation mixture component weights

obtain these mappings. Altogether, three different levels of mapping have been tested. Each of these three levels combines different components of the model sets. The first level, *mixture mapping*, finds for each target language mixture component mean vector the corresponding mean vector from the source language. On the next higher level, *state mapping*, a relationship is found between two states; i.e. for each state of the target language model set a corresponding source state is found. Finally, on the third level, *model mapping*, a source model is found for each target language model. In the following, the calculation of mappings on all three levels is described.

4.5.1 Mixture mapping based on acoustic distance

A mapping on the mixture component level is calculated by finding for each mixture component mean vector of the target system that mixture component mean vector of the source system, which has minimal acoustic distance to the target mean vector. Note that the approach using minimal acoustic distance is problematic, as it might cause a distorted mapping between the wo mixture distributions. In this work, mappings based on two different acoustic distance measures have been compared. These are the standard Euclidean distance measure and a divergence measure, which measures the separability of two Gaussians $\mathcal{N}(\mu_A, C_A)$ and $\mathcal{N}(\mu_B, C_B)$.

$$D_{div} = \frac{1}{2}trace\left(C_B^{-1}C_A - I\right) + \frac{1}{2}(\mu_A - \mu_B)^T C_B^{-1}(\mu_A - \mu_B) + \frac{1}{2}ln\frac{|C_B|}{|C_A|} \tag{4.19}$$

Such a mapping of mean vectors is solely based on acoustic information and does not contain any phonetic knowledge.

4.5.2 State mapping based on acoustic distance

The state mapping method maps each state of the target language to the closest state of all the states in the model set of the source language, irrespective of which model a state belongs to. A state distance measure, see equation 4.20, which measures the distance between state i and state j, is employed in order to find the nearest state of the source language for each state of the target language (S denotes the set of all states, M_S the total number of mixture components).

$$d(i,j) = \frac{1}{M} \sum_{m=1}^{M} log[b_j(\mu_{im})] + log[b_i(\mu_{jm})] \tag{4.20}$$

Some examples of a state mapping between Latin-American Spanish as the source language and British English as the target language are listed in Table 4.1. Once a state-to-state mapping has been found, all mixture components of the target state are mapped to the closest mixture component mean of the corresponding source state using the divergence distance measure.

B Phone State	Sp Phone State	B Phone State	Sp Phone State
ae 2	a_s 3	ae 4	a_s 4
ʌ 2	o_s 2	ʌ 3	a_s 4
ə 2	e_s 3	ə 4	n_s 3
ð 4	$ð_s$ 4	h 3	j_s 3

Table 4.1: State mapping from British English (B) models to Spanish (Sp) models using the state distance measure. The subscript 's' denotes a Spanish model. IPA symbols are used to describe the phones.

4.5.3 Model mapping based on phonetic knowledge

The last mapping approach finds a model-to-model mapping. The two previous mappings are solely based on acoustic distance measurements and do not include phonetic knowledge about likely mispronunciations of a language student. On the other hand, this third mapping approach is based on the idea that either a given target model is likely to be substituted by a source model, or that the models of an accented model set can be considered to model a sound somewhere in-between a source and a target model, see also the discussion in Section 3.3. Consequently, the task is to find a model of the source language corresponding to each model of the target language, see also Figure 4.3. Such a mapping does not need to include all source models nor is it a one-to-one mapping. A source model can be a substitute

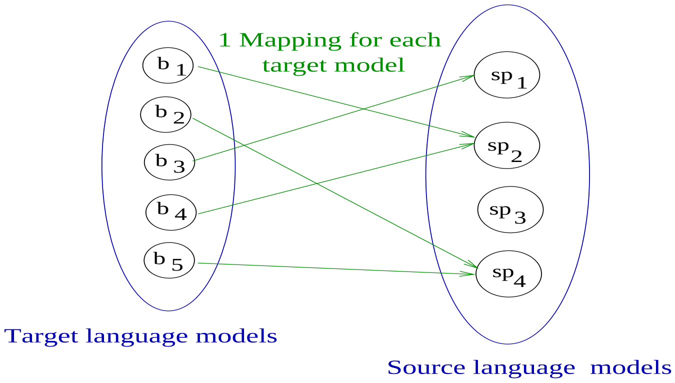

Figure 4.3: Model mapping from target to source language

sound for several target sounds. In Section 3.4 three methods for obtaining a set of substituted source language phones for each target language phone were described based on knowledge about likely substitutions. For the model mapping, the most likely phone of each substitution pattern is taken as the source language phone. Such a mapping represents how a foreign language student moves between the phonetic space of his mother tongue and the phonetic space of the target language.

Once a source model has been found for each model of the target language, a strategy for mapping between the states and mixture components of these two models is needed. The states are mapped sequentially, i.e. the first state of the target model maps to the first state of the source model. Within each state the closest source mixture component is found for each target mixture component using the divergence measure, see equation 4.19, in the same way as when calculating the mixture-level mapping.

4.6 Extension to triphone models

Both algorithms developed in this chapter can also be applied to recognition systems using triphones instead of monophones. This requires that the mapping strategy is extended to deal with triphones.

Let the name of a target language triphone, ph_{tr}, be described by the sequence of the names of three monophones of which the triphone consists, see [101] for a more detailed description of the notation:

$$ph_{tr} = ph_1 - ph_2 + ph_3$$

For each target language triphone, the mapped source language triphone is found by translating each of the three monophone names of the target language triphone to the name of it's mapped monophone in the source language. Combining these three monophone names to the mapped source triphone name yields the respective source triphone.

No other adjustment for the transition to triphones is necessary. For instance, each triphone consists of a Gaussian mixture, too. Thus, the re-estimation equation for LMC for tied Gaussian mixtures, equation 4.16 can be applied in the same way as for monophones. However, since all the work presented in this thesis is based on Gaussian mixture monophones, this approach was not tested.

4.7 Summary

In the first section of this chapter an overview of the current state-of-the-art in speaker adaptation was presented with emphasis on adaption to non-native speakers. Then, two new adaptation techniques for non-native speech, Linear Model Combination (LMC) and Model Merging (MM) were derived.

Both of these schemes are based on the assumption that speaker-dependent models of non-native speech can be found through the combination and adaptation of models from the source and the target language of a non-native speaker. However, the two methods differ in the way the models are combined. LMC assumes the mean vector of a mixture component of a model for accented speech can be found as a linear combination of the corresponding mean vectors of the source and the target language. The adaptation data are used to estimate the combination weights. On the other hand, MM assumes that the output probability function of a state of the accented model can be modeled by merging all mixture components of one state of a target model with all mixture components of the corresponding state from a model of the source language.

The final section of this chapter presented a set of different levels of mappings between source and target language models. Such mappings are necessary for both novel adaptation algorithms presented here. The performance of these algorithms is evaluated in 6. Moreover, experiments will be conducted to find out which type of mapping is the most suitable.

Chapter 5

A Non-native Database

5.1 Introduction

The research presented in this thesis is concerned with recognising non-native speech spoken by beginners of English. However, since there is little corpus data available for this domain, a database of hand-transcribed non-native speech has been collected. This chapter describes in detail the collection and transcription of this database. The database was recorded for the purpose of developing methods to detect mispronunciations in non-native speech for interactive spoken language education, and for improving acoustic modeling techniques of such speech. Therefore, the target was to collect recordings of fairly heavily accented speech.

The creation of the accented database consisted of the following three steps.

1. Design of the test material
2. Recording of the data
3. Annotation of the data by phoneticians

The actual recording of non-native data can be administered by non-experts in phonetics and can be completed in a comparatively short period of time. The bottleneck in creating such a database lies in the annotation of the data by trained phoneticians, because for each recorded utterance a detailed phonetic transcription is required. This transcription should correspond to the actual sequence of phones spoken as opposed to the transcriptions of the utterance according to a pronunciation dictionary. Additionally, each word and sentence should be rated according to its pronunciation quality. These annotation types are required in order to be able to calibrate and validate the pronunciation scoring methods presented in Chapter 9.

Section 5.2 describes the methodology for collecting the database with regard to the choice of prompting material, speakers and recording setup. The majority of

Mother-tongue	No. of Female Speakers	No. of Male Speakers
Latin American Spanish	1	2
Japanese	3	0
Korean	2	1
Italian	0	1

Table 5.1: Origin and gender distribution of the subjects in the database

the technical details about recording equipment, data structures etc. can be found in Appendix A. In Section 5.3, the transcription of the data by trained phoneticians is discussed with emphasis on the difficulties of transcribing non-native speech.

5.2 Database design and collection

5.2.1 Subjects

Given the target to record heavily accented speech, the criteria for choosing recording subjects was that they had to speak English as a foreign language at beginner and intermediate level. This ensured that on the one hand the subjects were able to understand the prompting texts and instructions, and that they were able to read the sentences aloud with a limited amount of hesitation and repetition. On the other hand, the spoken English of the subjects had to be elementary enough so that they were likely to produce a large number of easily detectable phonetic mispronunciations.

The subjects were recruited from several language schools in the Cambridge area. There were 10 speakers; 6 females and 4 males in an age range of 17 to 34 years. Between them, these speakers spoke Latin-American Spanish, Japanese, Korean and Italian as their mother tongue. The exact distribution of origin and gender of the subjects is given in Table 5.1.

5.2.2 Prompting material

In order to obtain recordings of speech data as it would typically be produced in a language learning environment, the prompting texts were extracted from Penguin readers [34, 16]. These are books especially written for the purpose of teaching English as a foreign language. They use a simplified grammar, simplified sentence structures and a limited vocabulary. For instance, in [34] the vocabulary has been restricted to 1050 words.

The prompting material for each reading session consisted of the following three groups of sentences:

- **discard**: 3 sentences for warming up and for training the subjects on the reading task. These sentences were discarded and not used in the final database.
- **adapt**: 41 sentences for adaptation or training. The same 41 sentences were read by each subject. The selection of these sentences was such that they contained phonetically balanced text.
- **test**: 80 test sentences, which were different for each speaker.

For example, a prompting file would look like:

```
## discard                                       (sentence label)
I originally wanted to be a mechanic,            (actual prompt)

   ⋮

## adapt1
I did not want to be employed in a company and be indoors all day.

   ⋮

## test80
Redhead laughed happily. "Any time, Mr Marlowe."
## last
```

Since all these prompting texts consist of stories, they contain sentences associated with all the types of intonation patterns, such as declarations, questions, exclamations etc., which are likely to occur in language education.

5.2.3 Recording setup and equipment

All recordings were made in a quiet room measuring $2m \times 4m$ with a low noise level, located in the speech laboratory of Cambridge University Engineering Department.

The subjects sat on a stable chair in front of a desk with a computer monitor. On the monitor screen, a window displayed the prompts. In order to coordinate the display of the prompting text and the actual recording, the same recording software tool was used as for the WSJCAM0 corpus, see [41].

The recording equipment consisted of the following components (a more detailed description is given in Appendix A.1):

- A head-mounted Sennheiser HMD 414 microphone
- A SX202 Dual Mic Preamp from Symetric
- The SGI line-level analogue input with sampling rate of 16 kHz and A/D Converter.

The recordings were made with a Sennheiser microphone which was attached to a pair of headphones, worn by the subject. Then, the acoustic signal was passed from the microphone, through the pre-amplifier, to the SGI analogue line input and the SGI A/D converter. Finally, the signal was stored in the HTK format on the hard disk of the SGI machine.

5.2.4 Recording session

After the arrival of the subject the recording administrator would explain to the subject what he or she was expected to do. Then, the subject had the opportunity to practise before starting the actual recording session. After the recording the speaker was asked to sign a disclaimer form and fill in a small questionnaire, see Appendix A.6. Appendix A.7 contains a written version of the instructions for the subject. Usually these instructions were given orally. This way the recording administrator was able to ensure that the subject understood the task. On average each recording session lasted for about 45 minutes.

After each session the data were processed and stored on a computer hard drive as described in Appendix A.2.

5.3 Data annotation

The task of accurately transcribing non-native speech on a phone-by-phone basis is a challenging task in itself. Since there exists no such thing as a 'correct transcription', any transcription will always depend on the subjective judgments of the transcriber. Therefore, the initial instructions and training given to the transcriber can significantly influence the resulting transcriptions. Human assessment of whole sentences has been analysed in several literature sources. For instance, Flege et al., [38], observed a fairly high reliability of foreign accent judgments in a series of experiments, where native listeners had to identify the accent type for a given utterance. The dependency of the reliability of accent assessment on the type of judge has been investigated in more detail by Cucchiarini et al., [24]. Here, the grading of foreign-accented Dutch by three different groups of expert judges (1 group phoneticians, 2 groups of speech therapists for non-native speech) was analysed. Even though a high correlation of sentence level scores was observed among the three different types of judges, the correlation of automatic likelihood scores with human judges depended significantly on which expert judge group was used for comparison.

As discussed in the preceding paragraph, high correlations have been reported for the task of scoring whole sentences. However, little literature is available about pronunciation assessment on a sub-sentence level, e.g. the phone level of an utterance. Only, in [13] a similar project of annotating non-native speech, executed at SRI, was discussed. However, this type of annotation is important for the database described here. Precise transcriptions and assessments of the recorded non-native speech are necessary in order to be able to evaluate the output of algorithms which detect phone-level mispronunciations in non-native speech. The labeling by trained phoneticians for this database consisted of the following three components:

1. Each sentence was scored on a scale of 1 to 4. A score of 1 denoted native like speech, and a score of 4 denoted barely intelligible, highly accented speech.

2. Each word in a sentence was scored on the same scale of 1 to 4.

3. The initial transcriptions according to the BEEP Dictionary, representing Standard Southern English, were corrected and marked regarding any mispronunciations that occurred.

This annotation task was aided by an interface especially written for this purpose. For an example of a typical screen of the interface, see Figure 5.1, which shows the window structure as it looks when a phone is corrected. Appendix A.4 lists the instructions which were given to the phoneticians before they started to transcribe.

The correction of phone-level transcriptions represents both the most important and the most challenging component of the data annotation. The initial transcriptions were derived with the help of a pronunciation dictionary, which is described in the next section. Then, Section 5.3 describes the guidelines according to which the phoneticians corrected these initial transcriptions in order to create transcriptions which correspond to the actually produced phone sequence.

5.3.1 Pronunciation dictionary

For initial transcriptions, a pronunciation dictionary based on Standard Southern British English was used. This dictionary is predominantly based on the British English Example Pronunciation Dictionary (BEEP), see [41]. Only, those few words which did not occur in this dictionary such as names have been added using hand-derived pronunciations. The complete set of phone symbols used in this dictionary

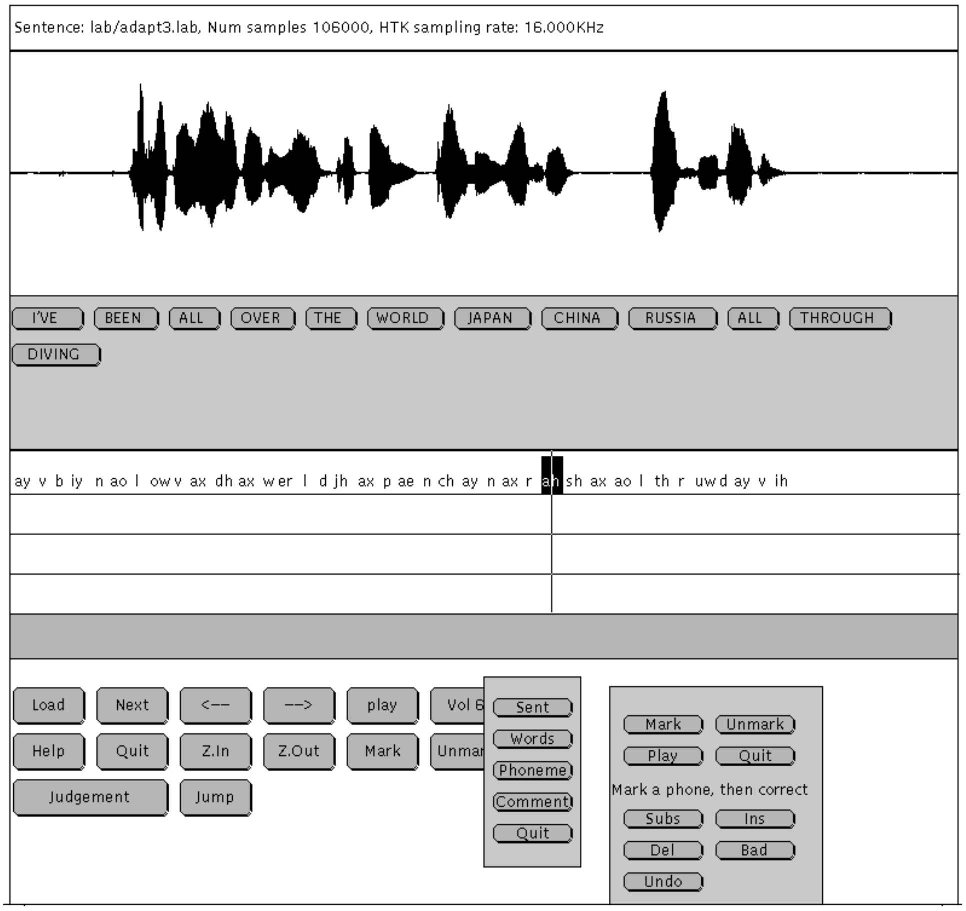

Figure 5.1: Example screen of the assessment interface when correcting the given transcription

is shown in Appendix A.3. Below an example section of the BEEP dictionary is given:

```
COLD            k ow l d sp
COME            k ah m sp
COMMERCIALLY    k ax m er sh l iy sp
COMPANY         k ah m p ax n iy sp
COMPLETELY      k ax m p l iy t l iy sp
CONTROLLING     k ax n t r ow l ih ng sp
COOKING         k uh k ih ng sp
```

In order to account for typical pronunciation variants, this pronunciation dictionary also contains multiple pronunciations, corresponding to the accepted variations in standard Southern British English.

5.3.2 Labeling mispronunciations

The two main problems of labeling mispronunciations are, firstly, the subjectivity of human transcribers and, secondly, the fact that non-native speakers produce sounds which cannot be described by the target language phone set alone. Often, these non-native sounds are a mixture of target and source language sounds or even source language sounds only.

The problem of subjectivity has been addressed by holding an initial meeting with the phoneticians. There, the labelling instructions were elaborated and discussed in order to ensure a labeling as homogeneous as far as possible. Additionally, each transcriber was asked to transcribe a common set of twenty calibration sentences. In Chapter 8, the transcriptions of these calibration sentences by all judges were evaluated in order to analyse how similar or dissimilar the transcribers tended to annotate the data.

The second problem of how to describe or label those phones which cannot be classified with the help of the target language phone set has been addressed by introducing new names for such sounds. Because at the start of the annotations it was not known what type of mispronunciation would be encountered, the transcribers were asked to introduce new names for new types of sounds and to use these new names consistently. They were also asked to give the database administrator a description of these new names. In the case that the sound of a non-native speaker was a mixture of the phone given in the transcription and of another target language phone, the transcription strategy was to substitute the other target phone. For instance, a phone might sound like a mixture of /v/ and /b/. If the correct transcription would have been /v/, but the speaker said something in-between /v/ and /b/, the original phone label would be substituted with /b/. In this way, the location of a mispronunciation has been marked, without using new phone symbols. Finally, the transcribing judge was also free to use phone labels of the phones as used in the source language but not the target language, for instance the palatal fricative /x/ instead of /h/ in "here" as it is used by Latin American Spanish speakers.

Altogether, the correction of the dictionary transcriptions consisted of the following options:

- **Insertion**: The assessed speaker pronounces a word with an additional phone.
- **Deletions**: The assessed speaker deletes a phones.
- **Substitutions**: The speaker uses an incorrect phone, i.e. a mispronunciation occurs.
- **Bad** : The speaker substitutes a sound which can not be identified.

In addition to this detailed labeling the phoneticians were told to add extra comments as to the overall quality of the sentence's pronunciation, for instance commenting on a speaker's characteristics.

Finally, for the purpose of pronunciation assessment, where the main information required is whether a phone was pronounced correctly or not, the transcriptions annotated by the phoneticians were additionally converted into binary strings. These strings only contain two symbols, one for correct and one for incorrect pronunciation.

5.4 Summary

This chapter describes the collection and annotation of a database of speech recorded from foreign language learners with different source languages. The database was designed to mirror the type of speech which is typically produced by beginners of English. Therefore, the prompting texts contained a simplified vocabulary and grammar. All collected speech was read aloud, i.e. it is not conversational speech. However, because the prompts were based on stories, these prompts contain sentences with all types of intonation patterns. This non-native speech data was annotated by trained phoneticians in two ways. Firstly, pronunciation scores were assigned for each sentence and word. Secondly, a phonetic transcription was created for each utterance, which corresponds to the phone sequence which has been uttered by the language student. These annotations were aided by a transcription tool which enabled the phoneticians to modify the given standard transcription of an utterance. The target was that the resulting transcriptions mirror the phone sequences produced by the subjects as closely as possible.

This database enables both the evaluation of the adaptation algorithms derived in Chapter 4 and the pronunciation scoring algorithm presented in Chapter 9.

Chapter 6

Evaluation of Non-native Speech Adaptation

6.1 Introduction

The non-native database described in Chapter 5 provides data for testing the non-native adaptation algorithms as derived in Chapter 4. This chapter presents experimental performance results for both the Linear Model Combination algorithm (LMC) and the Model Merging algorithm (MM). Each algorithm has been tested on two different types of foreign accented English: English spoken with a Latin-American Spanish accent and English spoken with a Japanese accent. The experiments have been designed to demonstrate the influence of several design parameters such as

- Choice of mapping between source and target language;
- Amount of adaptation data;
- Choice of initialisation parameters;

6.2 Experimental setup

The non-native database contains three speakers whose mother-tongue is Latin-American Spanish as well as two speakers whose mother-tongue is Japanese. The raw speech was encoded into feature vectors with 13 Mel-frequency cepstral coefficients, 13 delta and 13 acceleration coefficients. Additionally, all speech data was normalised with cepstral mean normalisation. The recognition system consists of 45 speaker-independent Gaussian mixture monophone models trained on Standard British English from the Wall Street Journal Corpus of British English

(WSJCAM0), [41], as target language models. Each model has five states including the non-emitting start and end states. The models of the source language have been trained on Latin-American Spanish and Japanese data respectively [1] . Each state was modelled by a Gaussian mixture with twelve components. The HTK Toolkit, [101] was used both for generating the model set and for the recognition experiments.

As language model, a word-pair grammar is used which is based on textbooks for learners of English as a second language, see [34, 16]. A word-pair grammar with perplexity of about 5 was chosen because it corresponds to the level of complexity of style and grammar which can be expected from typical exercise texts in a typical CALL system. Additionally, since heavily accented non-native speech is difficult to recognise accurately, tight constraints on the language model side are necessary, too. Moreover, the word insertion penalty has been adjusted for non-native speech, in order to balance insertion and deletion errors.

In all experiments, recognition performance is measured by word error rate (WER). Let N denote the total number of word labels, I the number of inserted words, S the number of substituted words and finally D the number of deleted words. Then, the word error rate is defined by

$$WER = \frac{I + D + S}{N} \tag{6.1}$$

Thus, when optimising the setup of the algorithms evaluated here, the aim is to minimise the word error rate.

The adaptation sentences used for all experiments in this chapter are taken from the 41 adaptation sentences of the three Latin-American Spanish speakers as well as the two Japanese speakers. The averaged word error rate in the experiments is based on the average of the WER of 80 test sentences of each of the five speakers.

6.3 MLLR adaptation

As previously mentioned in the overview of current adaptation technology, Section 4.2, the MLLR technique by Leggetter, [59], is an effective and widely used speaker adaptation algorithm. The recognition performance of MLLR will therefore be used as the benchmark against which to measure the performance of LMC and MM. In Figure 6.1 the recognition performance of MLLR is illustrated for different parameter setups. The recognition accuracy of MLLR heavily depends on which type of transformation matrix and on how many transformation matrices have to be estimated. The minimal number of HMM parameters has to be estimated in the case

[1]The author would like to thank Entropic Cambridge Research Lab for generously providing these data and models.

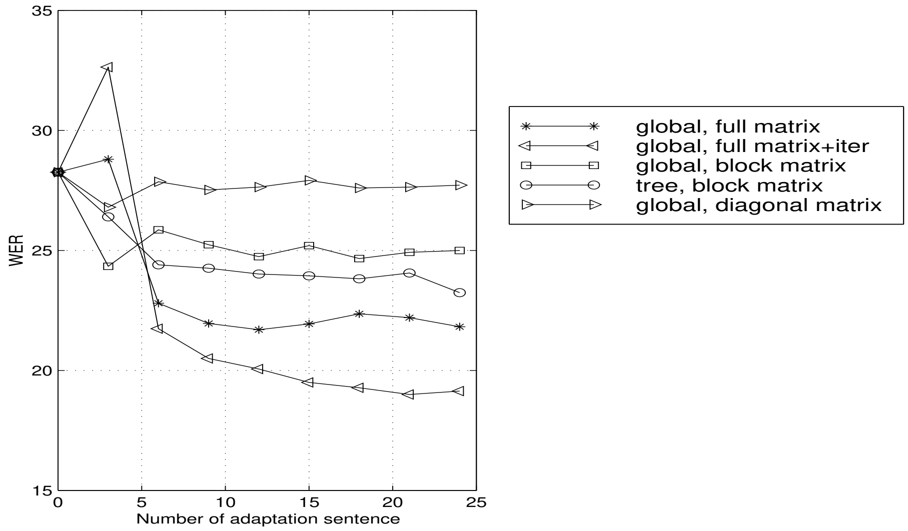

Figure 6.1: WERs for baseline setup and for different MLLR configurations. 'global'= global diagonal transformation matrix, 'full'=full transformation matrix, 'tree'=regression tree and 'block'=block-diagonal transformation matrix, 'iter'= 5 iterations.

of a single global, diagonal matrix. The best possible performance for this setup is found with three adaptation sentences. An increase in adaptation data does not yield any performance increase. If more data are available more parameters can be estimated reliably. In the case of more data, block-diagonal or full transformation matrices yield significantly lower word error rates. Moreover, with more adaptation data several transforms can be calculated for groups of HMMs instead of a single global transformation matrix only.

In this experiment, three different types of transformation matrices were used, a diagonal matrix, a block-diagonal matrix and a full matrix. These transformation matrices were tested with one global transform for all models and with a regression tree with groups of transforms. In Figure 6.1, it can be seen that an increase in adaptation data beyond the minimum of 5 sentences does not improve the performance for most setups except for the case of a full matrix re-estimated with 5 iterations. This effect arises from the large acoustic mismatch between the accented test data and the models trained on native speech, which causes inaccurate alignments. Multiple iterations however help to reduce the misalignments. A summary of the actual word error rates for both rapid adaptation, i.e. 6 or 9 adaptation sentences and full scale adaptation (24 sentences) is given in Table 6.1. The performance of the new algorithms will be measured against these results.

	Baseline	$\text{MLLR}_{diag,gl}$	$\text{MLLR}_{full,gl}$	$\text{MLLR}_{full,gl,iter}$
6 Sent.	28.3	27.9	22.8	21.7
9 Sent.	28.3	27.5	22.0	20.5
24 Sent.	28.3	27.7	21.8	19.2

Table 6.1: WER Summary for a) Baseline, b) MLLR with global, diagonal transform, c) with global, full transform and d) with global, full transform with iterations

6.4 Linear Model Combination

This section discusses the performance of LMC depending on different design parameters. In the case of CALL applications, mainly rapid adaptation with a few sentences is of interest because students cannot be expected to spend a long time recording adaptation sentences.

Therefore, the target of this work is to develop adaptation algorithms which both improve non-native recognition and enable rapid adaptation. For the latter reason the performance of LMC will be evaluated with regard to how fast recognition accuracy can be increased. All experiments with LMC estimate a single global transformation matrix which is applied to all models, because this is the setup where the minimal amount of parameters has to be estimated.

6.4.1 Effect of different mapping techniques

In Section 4.5 several mapping approaches between source and target language were discussed. The first experiment to evaluate LMC for rapid adaptation measures the WER of LMC using 6 adaptation sentences. Given this setup, the influence of choosing mappings on different levels such as state-level or model-level on the performance of LMC is shown in Table 6.2. Both the mixture-level mapping, LMC_m, and the state-level mapping, LMC_s, were calculated using the divergence distance measure, see equation 4.20, in order to map the mixture components between two mapped states. The first experiment with model-level mapping, LMC_A, uses the Euclidean distance measure to map between the mixture components within a state, whereas the second model mapping, LMC_B, uses the divergence measure.

As can be seen in this table, model-level mapping which is based on phonetic knowledge yields recognition improvement over the baseline. In the case of model mapping using the divergence distance measure, LMC_B, a relative improvement of 13.3% compared to the baseline was measured for the Latin-American Spanish data, whereas for Japanese accent the improvement is 13.2%. On the other hand, those mappings which are solely based on distance information, i.e. state and mixture mapping do not perform significantly better than the baseline. For both accent types, it can be seen that a distance measure which incorporates information about the variance of mixture components, i.e. the divergence measure, is more effective than a distance measure which only incorporates information about the means of mixture components, i.e. the Euclidean distance measure. Based on these results, all the following experiments will use model-level mapping with the divergence measure as the standard distance metric.

The observation that mappings based on phonetic knowledge yield better performance than acoustically based mappings indicates that knowledge about likely phone substitutions of a non-native speaker can improve the acoustic modeling of non-native speech. Finally, comparing the performance of LMC for speakers with a Latin-American Spanish accent with the performance for speakers with a Japanese accent, it can be seen that the algorithm works equally well for both types of accent.

Spkr (Accent)	Base	LMC_m	LMC_s	LMC_A	LMC_B
FL (Span)	20.3	19.1	19.0	17.9	14.8
PC (Span)	29.4	30.1	30.3	28.4	26.0
TS (Span)	26.7	26.0	26.0	24.7	25.1
MK (Jap)	19.3	21.9	19.7	18.8	18.7
SS (Jap)	45.8	44.5	44.4	43.8	39.1
Avg.	28.3	28.3	27.9	26.7	24.7

Table 6.2: WER for the baseline, for LMC with mixture-level mapping, LMC_m, and with state-level mapping, LMC_s. LMC_A denotes LMC with model-level mapping using Euclidean distance and LMC_B denotes model-level mapping using the divergence measure. All experiments use a global transform and 6 adaptation sentences.

Given that model-level mapping is most effective for LMC, the next experiment investigates how much of a difference in recognition performance can be caused by variations in the model-level mapping, i.e. how large is the effect of not choosing the "best" substitution model from the source language for a model from the target language. Since the substitution patterns of non-native speakers can still vary among individual speakers, there exist no "correct" mapping. In Table 6.3 the results for the automatically derived mappings for Spanish and Japanese accent are contrasted with the performance results for two slightly different mappings. In the mapping used in the experiment denoted as LMC_2, six vowels have been changed in places where the automatic mapping suggested several likely substitutions (see Tables 3.1

and 3.2). Likewise, in the mapping used in LMC_3, five consonants have been varied in a similar manner. In both cases the performance decreases only slightly. This leads to two conclusions: firstly, the automatic mapping technique, see Section 3.4.4, yields a suitable and reliable mapping. Secondly, changing the mapping by using alternative substitutions of source language models leads to a small performance decrease of only $2-4\%$ relative. This renders the automatic mapping calculation a reliable tool in order to create model mappings without much effort for any accent type.

Mapping	Base	LMC_1	LMC_2	LMC_3
WER	28.3	24.7	25.7	25.4

Table 6.3: Word error rate for baseline and different model mappings, LMC_1 (the original map), LMC_2 (6 changed vowels) and LMC_3 (5 changed consonants), all experiments use a global transform and 6 adaptation sentences

6.4.2 Amount of adaptation data

The improvements obtained by employing speaker adaptation also depends on the amount of adaptation data used for the re-estimation of the model parameters. In Figure 6.2, the word error rate of LMC is plotted together with the WER of two MLLR configurations (block matrix and full matrix with 5 iterations) as a function of the number of adaptation sentences. This comparison shows that LMC performs similar to MLLR using a block-diagonal transformation matrix. On the other hand, LMC performs significantly worse than MLLR with a full matrix and 5 iterations. Note also that LMC shows no performance increase when the amount of adaptation sentences is increased beyond the minimal three sentences. This might be due to the limited number of parameters that are estimated in LMC.

Another explanation for the fact that more data do not cause further performance increase is the acoustic mismatch between native and non-native speech. For instance, as described in Chapter 3, the speakers of the non-native database speak about 20% slower than native speakers. Moreover, these speakers speak with different intonation patterns and occasionally stop or hesitate. Under these circumstances, the alignment which is required for adaptation tends to be faulty. Thus, additional adaptation material might confuse instead of confirm the re-estimation process. All these observations about the difficulties of adaptation led to the development of the accent prediction methods described in Chapter 7.

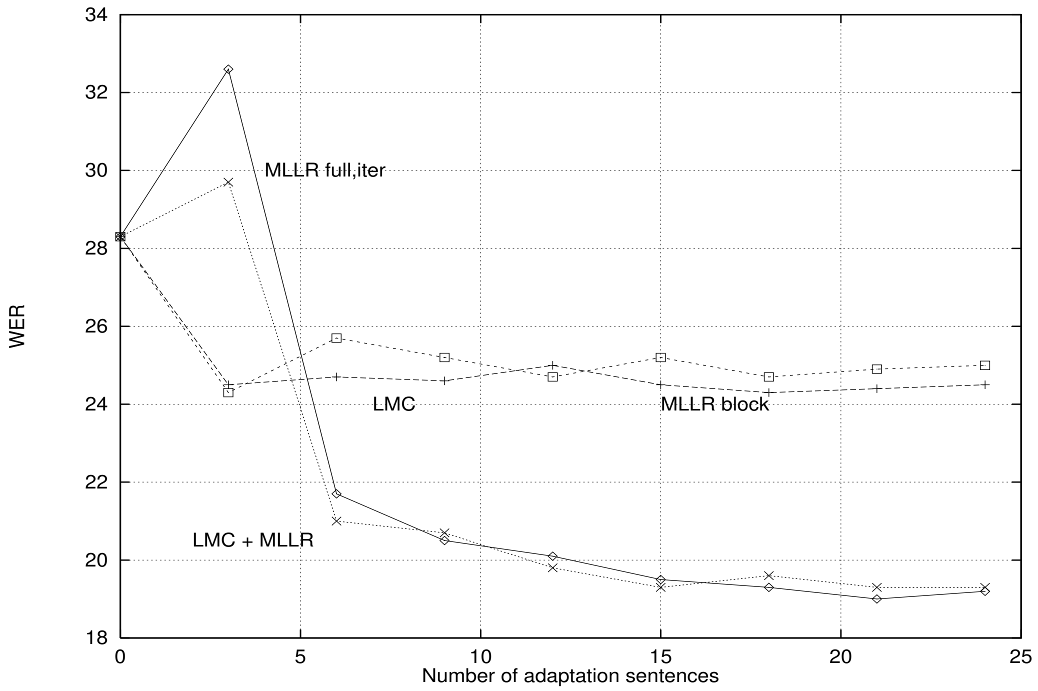

Figure 6.2: LMC: Word error rate dependency on the number of adaptation sentences. 0 sentences denotes the baseline WER

6.4.3 Combination of LMC and MLLR

In the experiments presented so far, it has been shown that LMC performs better than MLLR with a diagonal transformation matrix, but not better than MLLR with a full transformation matrix. On the other hand, the best performance of LMC is already found for the minimal amount of 6 adaptation sentences. For these reasons, this experiment uses LMC and MLLR adaptation in sequence. Thus, the improved model set estimated with LMC serves as the initial model set for MLLR providing a better initial alignment for estimating the MLLR transform. The results in Figure 6.2 show that providing a better initial model enables a small improvement of 3% relative in the case of rapid adaptation, i.e. 6 adaptation sentences. Otherwise, comparing MLLR and LMC+MLLR for varied amounts of adaptation data in Figure 6.2, shows basically equal performance. However, closer examination of the recognition results for each speaker revealed that there exist performance gains for those speakers which had lower initial error rates. In Figure 6.3 the results for MLLR versus LMC+MLLR are shown, both averaged over the four best speakers (fl,pc,ts,mk) and averaged over the three best speakers (fl,ts,mk). In the latter case the average improvement is 11% relative for rapid adaptation.

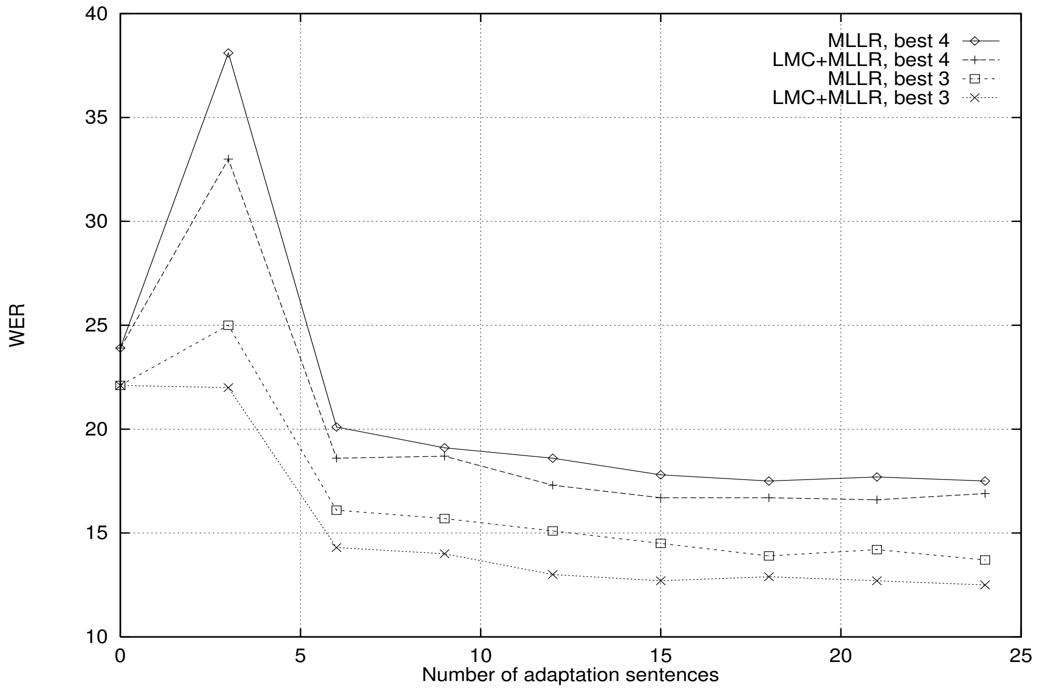

Figure 6.3: WER for MLLR and LMC+MLLR. 'best 4' = WER averaged only over best 4 speakers, 'best 3' = WER averaged over the best 3 speakers.

6.5 Evaluation of Model Merging

In this section the second adaptation scheme for non-native speech, MM, is evaluated. As described in Section 4.4, MM combines the Gaussian mixtures of each pair of target and source language states and merges them to a new Gaussian mixture with twice as many mixture components as before. This technique has the advantage over LMC that no mapping at the mixture component level is required.

The first experiment with MM evaluates the effect of reducing the amount of mixture components by applying a cut-off threshold for their weights. After an optimal threshold has been determined, this threshold has been used in the following experiments to test the influence of the choice of the mapping and of the initial combination weight α.

6.5.1 Decreasing the number of mixture components

This experiment applies a cut-off threshold to the weights of the mixture components in order to remove all mixture components whose weights fall below the cut-off threshold. In Table 6.4 the WER is presented for a range of such thresholds. The number in brackets indicates the percentage of mixture components kept after applying the cut-off threshold. Keeping on average 50% of the mixture components is equivalent to the number of mixture components in the original models. For low cut-off thresholds the performance varies only slightly even though the average number of mixture components is decreased to less than 60%, i.e. almost the original amount of model parameters. For the largest threshold, $T = 0.03$, the percentage is consistently below 40% for all test speakers, while the performance decreases by only 5% relative. The results of this experiment indicate that the improved performance of MM might be due to the incorporation of phonetic knowledge as opposed to an increase in the number of model parameters. In all following experiments a threshold of $T = 0.0005$ has been applied. For this threshold and 9 adaptation sentences, the recognition accuracy for MM decreases by 42% relative to the baseline. Moreover, MM is better than the optimal MLLR setup by 24% relative.

6.5.2 Mapping

As was observed for LMC, model mapping with the divergence measure also yields better performance for MM than state mapping using an acoustic-distance based metric, see Table 6.5. The same table also demonstrates that varying the mapping in the same manner as in Section 6.4.1 only has a small effect on the overall performance. The small performance decrease with small changes in the mapping makes MM stable against non-optimal mappings.

Spkr	MM	$MM_{T=0.0005}$	$MM_{T=0.001}$	$MM_{T=0.005}$	$MM_{T=0.01}$	$MM_{T=0.03}$
FL	11.1 (100)	10.4 (70)	10.4 (67)	10.6 (59)	9.8 (53)	9.2 (37)
PC	17.8 (100)	17.1 (74)	17.3 (70)	16.6 (62)	16.8 (56)	17.5 (39)
TS	17.1 (100)	16.7 (73)	16.8 (70)	17.0 (61)	17.5 (56)	18.0 (39)
MK	9.5 (100)	9.4 (60)	9.7 (52)	9.4 (48)	9.0 (47)	12.7 (21)
SS	28.7 (100)	28.5 (65)	28.4 (56)	27.2 (52)	28.6 (51)	31.0 (21)
Avg.	16.8 (100)	16.4 (68)	16.5 (63)	16.2 (56)	16.3 (53)	17.7 (31)

Table 6.4: Using a cut-off threshold to decrease the amount of mixture components. Bracketed numbers indicate average percentage of mixtures per model. 9 adaptation sentences and a full MLLR transformation matrix are used.

	Base	MM_{state}	MM_{model}	MM_{model_2}	MM_{model_3}
WER	28.3	25.9	16.4	17.8	16.9

Table 6.5: WER for MM (MM_{state} = state mapping, MM_{model} = model mapping. MM_{model_2} = model mapping with 6 changed vowels, MM_{model_3} = mapping with 5 changed consonants, 9 adaptation sentences.

6.5.3 Choice of the initialisation parameter α

For this experiment the value of the weighting factor α, which is used for the initial combination of the two model sets (see eq. (4.17)), is varied. The small variations in WER for different values of α show that the value of α does not significantly influence the performance as long as α lies within the interval $0.2 \leq \alpha \leq 0.8$. In order to illustrate the meaning of this parameter, assume an initial value of $\alpha = 0.5$. Then, this means that both mixtures are equally weighted. A larger α represents a higher weight of the source language mixture and vice versa.

Spkr	Base	$MM_{\alpha=0.2}$	$MM_{\alpha=0.5}$	$MM_{\alpha=0.8}$
Avg.	28.3	16.8	16.4	17.5

Table 6.6: Word error rate for model merging 6 adaptation sentences, varying the initial merging weight α.

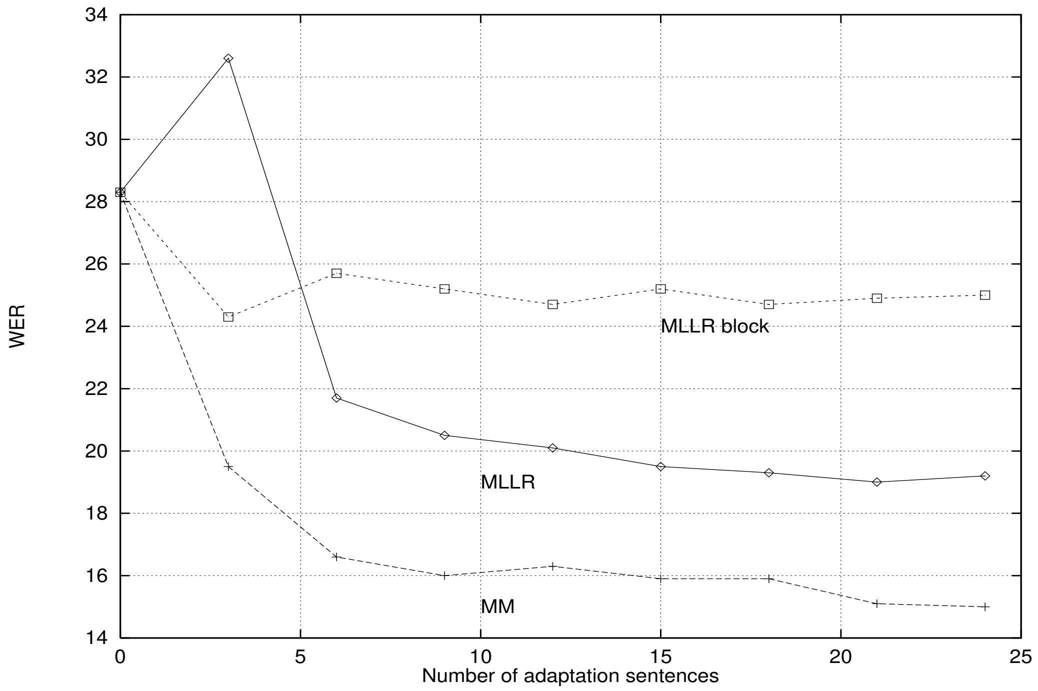

Figure 6.4: Model Merging: WER dependency on the number of adaptation sentences. 0 sentences denotes the baseline error rate.

6.5.4 Amount of adaptation data

Next, the effect of varying the amount of adaptation data is shown in Figure 6.4. Especially, for the case of rapid adaptation, i.e. 3 to 6 adaptation sentences, MM outperforms MLLR significantly. For example, for the minimum of 3 sentences, MM is better than MLLR with block matrix by relative 20%. For 24 sentences, the relative improvement reaches 22% in comparison with MLLR with a full global transformation matrix and 5 iterations.

6.6 Summary

In this chapter, two novel adaptation techniques for non-native data were evaluated. These algorithms, LMC and MM, are based on combining each Gaussian mixture of the target language model set with the corresponding Gaussian mixture of the source language model set. Both schemes have low computational requirements and have been proven to be effective for even small amounts of adaptation data. A summary of the results for rapid adaptation (i.e. 6 adaptation sentences) can be seen in Table 6.7. For the combination of LMC and MLLR as few as 6 adaptation sentences yields

Spkr	Base	MLLR	LMC+MLLR	MM
Avg.	28.3	21.7	21.0	16.6

Table 6.7: Summary of rapid non-native adaptation: WERs for baseline, MLLR (full matrix), LMC+MLLR and MM, always using 6 adaptation sentences.

a relative improvement of 26% over the baseline. In addition, relative to the MLLR algorithm which is used as a benchmark, performance is increased by 3%. The final set of experiments demonstrated that MM can perform better than the baseline and MLLR by relative 41% and 24% respectively.

Both LMC+MLLR and MM require a mapping between the source and target language model sets. Since this mapping is crucial for the performance of the algorithms presented here, the influence of several mapping approaches was discussed. Experimental results show that a mapping which incorporates phonetic knowledge about likely mispronunciations of a non-native speaker yields the largest performance increase. Furthermore, small changes in the mapping did not lead to significant performance losses. This indicates that these techniques are stable for applications.

Because LMC only requires adaptation of the diagonal elements of the combination matrix, few adaptation sentences suffice to achieve the best possible performance. Similarly, the WER for MM decreases rapidly with few adaptation sentences. However, in this case an increase in adaptation data still gives small additional improvements, mainly because adapting the means and re-estimating the weights yields a much larger number of parameters to be updated. The results presented in this chapter are based on a fairly small test database. In future work these experiments should be verified on a larger scale.

Chapter 7

Accent Prediction: Off-line Acoustic Modeling of Non-native Speech

7.1 Introduction

In contrast to the two non-native adaptation techniques, LMC and MM, as derived and evaluated in Chapters 4 and 6, this chapter introduces an approach which yields improvements in the recognition of non-native speech without requiring adaptation data. The improvement is found solely through exploiting the knowledge of the mother tongue of a language student. These **accent prediction** methods only require an additional model set for the source language and a mapping between the source and the target language in order to improve the recognition of foreign accented speech: they do not require adaptation data.

This approach is of practical importance because even though HMMs and training data for the major world languages are nowadays generally available, the transcribed non-native speech needed for adaptation will often be hard to obtain. Section 3.4 discussed methods for deriving the necessary mapping between two languages, either by using phonetic literature or by automatic calculation with the help of relatively little training data. These accent prediction techniques make it easier and cheaper to set up a CALL system.

In the following three sections three methods for accent prediction are presented and evaluated. These algorithms are called

1. **Parallel Bilingual Modeling**

2. **Linear Model Combination** with *a-priori* weights b_{jj}

3. **Model Merging** with *a-priori* α

All three methods combine acoustic models of the source and target language. They only differ in the way that the models are combined. This again influences the number of parameters which have to be estimated *a-priori*. The experiments will demonstrate the dependency of the performance on the choice of these *a priori* model parameters.

7.2 Accent Prediction using Parallel Bilingual Modeling (PBM)

The simplest way of combining models of the source and target language into new bilingual models is to combine each model of the target in parallel with it's mapped model of the source language. Choosing such models is equivalent to substituting models of the target language with models of the source language, if the source language models have a higher acoustic likelihood. Parallel Bilingual Modeling (PBM) combines each pair of acoustic models of the source and target language into a parallel model according to a given model mapping. An example of such a bilingual model is shown in Figure 7.1. This HMM combination requires the adjustment of the transition matrix. Let the transition matrix A of a three state HMM be

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & a_{22} & a_{23} & 0 & 0 \\ 0 & 0 & a_{33} & a_{34} & 0 \\ 0 & 0 & 0 & a_{44} & a_{45} \end{pmatrix} \tag{7.1}$$

and define another matrix B in the same way. Then the transition matrix C of the parallel-combined model combines the transition coefficients of the two matrices A and B :

$$C = \begin{pmatrix} 0 & a^*_{12} & 0 & 0 & b^*_{12} & 0 & 0 & 0 \\ 0 & a_{22} & a_{23} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{33} & a_{34} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{44} & 0 & 0 & 0 & a_{45} \\ 0 & 0 & 0 & 0 & b_{22} & b_{23} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & b_{33} & b_{34} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & b_{44} & b_{45} \end{pmatrix} \tag{7.2}$$

with the restriction that

$$b^*_{12} = 1 - a^*_{12} \tag{7.3}$$

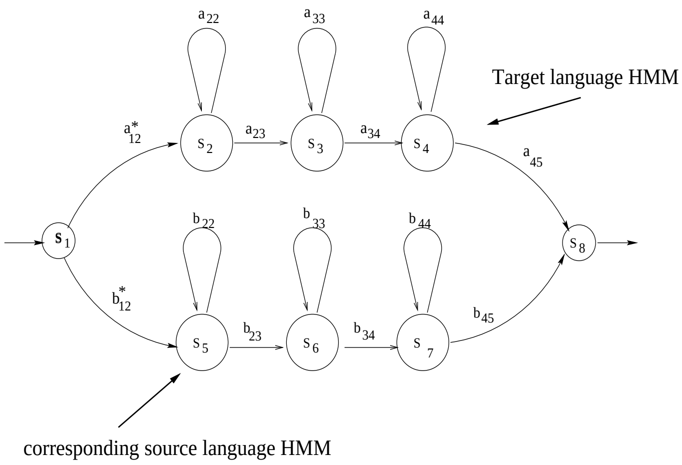

Figure 7.1: Bilingual HMM as a parallel combination of a source and a target language model

Thus, the only parameter to be estimated *a-priori* is a^*_{12}. This parameter represents the probability with which either the model of the target or source language is chosen.

Given these bilingual models, each phone in the transcription of a word will be extended in the lattice for recognition to two phones in parallel, the correct phone and the possible substitution. Table 7.1 gives the WER of each test speaker for recognition with acoustic models which were calculated using PBM. The *a-priori* estimate of the transition probability was $a^*_{12} = 0.5$. The Spanish-English and Japanese-English mappings used in these experiments were calculated with the automatically derived model-level mapping, see also Section 3.4.4. Table 7.1 shows that applying PBM improves the recognition performance for each speaker independent of the accent type. On average, a relative improvement of 30% over the baseline can be achieved.

The next experiment investigates the influence of the value the transition probability a^*_{12} on the recognition performance. Table 7.2 demonstrates that the recognition accuracy, averaged over all test speakers, does not change significantly for any value of a^*_{12} as long as it is within the range of $0.3 \leq a^*_{12} \leq 0.7$. Only for values at the edges of the interval $[0, 1]$ does the average WER decrease. If the value of a^*_{12} lies at the edges of this interval, then there exists a strong bias towards using the acoustic models of either the source or the target language only. Interestingly enough, if the probability of choosing an acoustic model of the target language is as low as 10% (i.e. $a^*_{12} = 0.1$), the overall recognition rate is higher than if this probability is 90%. This observation is another indicator that the use of acoustic

Spkr	Accent	Base	PBM
FL	Span	20.3	15.5
PC	Span	29.4	23.1
TS	Span	26.5	21.1
MK	Jap	19.1	16.5
SS	Jap	45.8	38.2
WER_{avg}	–	28.22	22.9

Table 7.1: WER results for accent prediction with PBM contrasted with the baseline results. ($a^*_{12} = 0.5$).

a^*_{12}	0.1	0.3	0.5	0.7	0.9
WER_{avg}	23.4	23.0	22.9	22.9	24.2

Table 7.2: Averaged WER Results of PBM for different values of a^*_{12} demonstrating the relative performance independence of PBM from the choice of a^*_{12}

models of the source helps to improve recognition accuracy.

7.3 Accent Prediction using Linear Model Combination (predicted LMC)

In this section another technique is proposed which predicts accented speech based on LMC adaptation as derived in Section 4.3. These LMC-predicted acoustic models are estimated by combining models of the source and target language in the same way as in the LMC adaptation algorithm. However, whereas for LMC adaptation the combination weights b_{jj} of the combination matrix $\mathbf{B_s}$ are estimated based on the adaptation data, for predicted LMC these weights are estimated *a-priori*. In other words, the predicted weights are chosen empirically.

In order to see what are typical estimates of these weights in the case of LMC adaptation, Figure 7.2 shows example estimates of b_{jj} for two non-native and one native speaker. All weights have positive values. This indicates that the linear combination approach can provide the correct direction in which to move in the acoustic space. For the first non-native speaker significantly higher combination weights were estimated than the second one, this means that the second speaker is likely to be more fluent than the first one. This figure also shows that acoustic models adapted with LMC adaptation are likely to have combination weights b_{jj} in

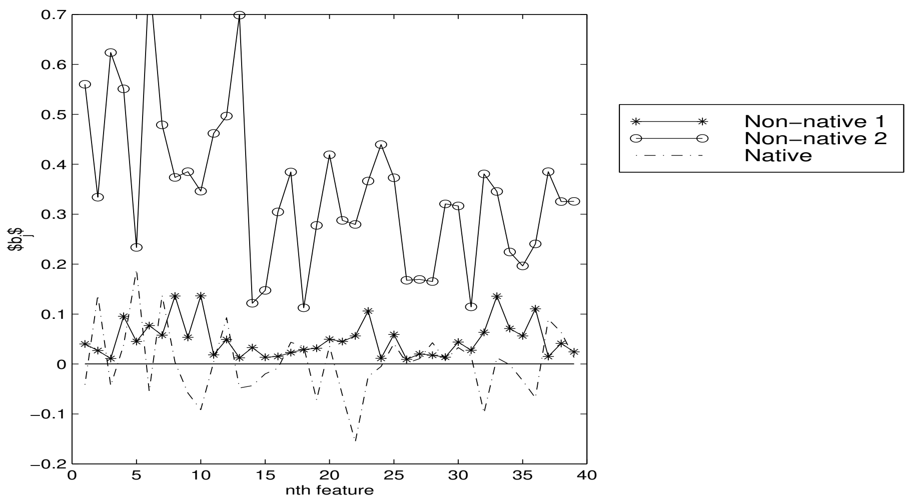

Figure 7.2: Example of model combination weights for two non-native speakers and one native speaker

the range of $0.0 \leq b_j \leq 0.7$. Thus, any *a-priori* estimates of these weights should be within this interval, too.

The choice of *a-priori* combination weights is important for the success of this approach. However, since there exist no closed form solution for finding optimal weights, the only possibility is to choose weights by trial and error. The following two experiments measure the recognition performance of predicted LMC for two types of guesses of the b_{jj}.

7.3.1 Constant *a-priori* values for the combination weights b_{jj}

In this experiment the same constant value is used for all combination weights b_{jj}. In Figure 7.3, this constant weight has been varied in the interval of $[0, 1]$. $b_{apriori_j} = 0$ means that only models of the target language are used. The WERs of all speakers follow a similar pattern. First, the error rate decreases with increasing weight b_{jj} until a minimum has been reached. Any further increase of b_{jj} beyond its minimum causes an increase in the recognition accuracy. For all tested speakers this minimum is found for weights within the following interval

$$b_{apriori_j} \in \; [0.20, 0.70] \quad \forall\, j \in 0, \ldots N. \tag{7.4}$$

The WER minima for each speaker have been listed in Table 7.3 together with the optimal constant weights for which the minimum is achieved and with the relative recognition improvement. The relative improvements are quite significant, they

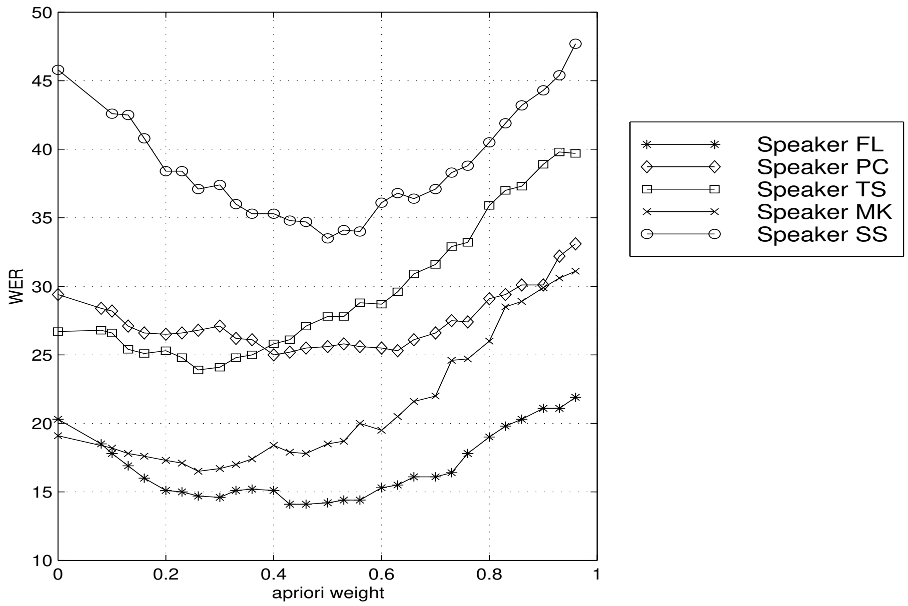

Figure 7.3: WER dependency on the choice of *a-priori* weights. Weight 0.0 denotes the baseline error rate

vary in the range of $26-50\%$. However, the drawback of this LMC based prediction algorithm is that the performance heavily depends on the guess of the combination weight $b_{j_{apriori}}$ and on the speaker, because in this case the newly combined mixture components will be more similar to the mixture components of the source language.

Even if the LMC-based prediction algorithm is not very suitable for improving the recognition accuracy of any speaker of given accent group, it might be possible to use such weights as a measure of fluency. That value of $b_{j_{apriori}}$ which minimises the WER for a given set of test sentences can be interpreted as a measure of the degree of foreign accent. A higher weight would indicate a less fluent speaker. For example, listening to random speech samples indicates that the Japanese speaker 'SS' is less fluent than speaker 'MK'. This agrees with the observation that the minimal WER for 'SS' is found around $b_{j_{apriori}} = 0.6$, whereas the minimum for 'MK' is found at $b_{j_{apriori}} = 0.4$. On the other hand, the WER changes fairly slowly with changing weight. Therefore, a suitable a-priori weight could be chosen for a group of subjects with similar degrees of fluency.

7.3.2 Influence of the choice of the source language

This experiment checks whether it is indeed the inclusion of phonetic knowledge that helps to improve recognition. The experiment is based on using a model set of a

Spkr	FL	PC	TS	MK	SS
$b_{j,best}$	0.43	0.40	0.26	0.26	0.50
$\%WER$	31	15	10	13	27

Table 7.3: Relative WER improvements for optimal predicted combination coefficients $b_{j,best}$

Spkr	Base	$\text{LMC}_{0.5}$	$\text{LMC}_{span,0.5}$
MK	19.1	18.5	23.3
SS	45.8	33.5	45.6
Avg.	32.5	26.0	34.5

Table 7.4: Word error rate for accent prediction using models combined for a different accent.

different language other than the source language to calculate a combined model set. Thus, the Spanish and the English models have been combined with the *a-priori* weight $b_{japriori} = 0.5$. These Spanish-English models are then used to recognise Japanese accented speech. The results are shown in Table 7.4. Using LMC adapted models based on Spanish as the source language yields worse recognition than the baseline, whereas the performance of LMC with Japanese source models increases by 20%. This is a clear indication that knowledge about the source language is the reason for the improvements achieved with LMC based accent prediction.

7.4 Accent prediction using Model Merging (predicted MM)

The final accent prediction method presented in this chapter is based on MM adaptation. The models are combined in the same way as before, but the adaptation and re-estimation steps of the MM technique are skipped, since in this case no adaptation material is available. However, the results in Figure 6.4 demonstrated that even few adaptation sentences yield significant recognition improvement. So, it can be assumed that even if no data are available in order to reestimate means and weights, the use of acoustic models based on predicted Model Merging might still yield some improvements.

The only variable to be estimated in this case is the weighting factor α. This factor has been varied between $\alpha = 0.0$, i.e. using only the target models and $\alpha = 1.0$, i.e. using solely the source models. In Figure 7.4 the WER is shown as

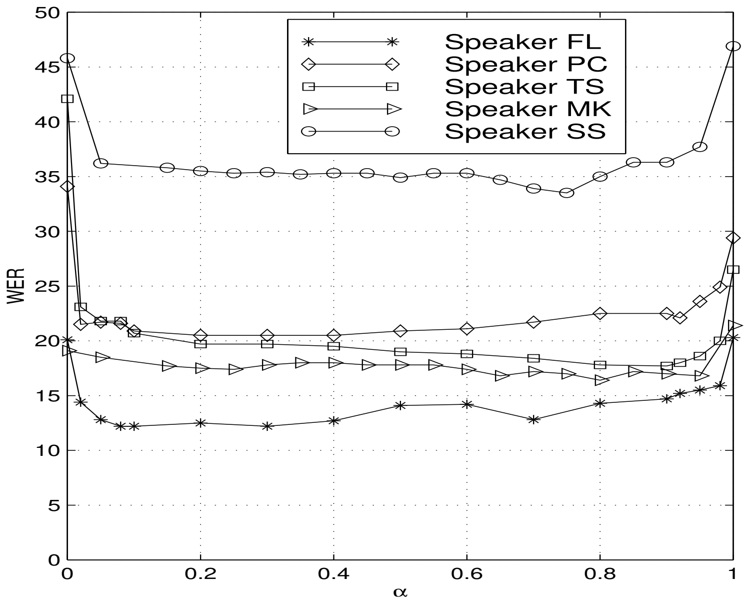

Figure 7.4: WER for predicted MM with dependency on merging weight α. $\alpha = 1.0$ denotes target models only.

Spkr	FL	PC	TS	MK	SS	Avg
Base	20.3	29.4	26.5	19.3	45.8	28.3
MM	12.1	19.6	18.9	17.5	34.9	20.6

Table 7.5: Word error rate for baseline and A-priori model merging ($\alpha = 0.5$)

a function of the merging coefficient α. For any $0.1 \leq \alpha \leq 0.9$ the performance of predicted MM is roughly constant for all speakers. This renders the algorithm almost independent of the value of α. Thus, predicted MM can be implemented without requiring any additional fine tuning. Because of this characteristic, this algorithm is well suited for any kind of recognition system, where non-native speech of one type of accent has to be recognised.

It can be seen from Table 7.5 that predicted MM reduce the word error rate from $WER = 28.3\%$ for the baseline of speaker-independent native models to $WER = 20.6$ for predicted MM with $\alpha = 0.5$. This error rate reduction represents a relative improvement of 27.2% in recognition accuracy.

Spkr	Base	MM	MM_{span}
MK	19.1	17.5	19.1
SS	45.8	34.9	39.0
Avg.	32.5	26.2	29.1

Table 7.6: WER for accent prediction using models merged for a different accent.

7.4.1 Influence of the choice of the source language

The importance of using the correct source language models when merging models has been tested in the same manner as for LMC, see Section 7.3.2. Again, a model set of a different language other than the source language has been used to calculate a merged model set. Here, the Spanish and English models have been merged to recognise Japanese accented speech. The results are shown in Table 7.6. As with LMC, the performance of MM with Japanese source models increases by 17% compared to the baseline as opposed to the performance with Spanish source models where a lower, but still significant improvement of 10% is achieved.

Probably, it is the additional information provided by the second model set which enables recognition improvement for even the wrong model set. Because the additional modelling "material" is adapted to the speaker's characteristics by updating the weights and adapting the means, higher accuracy is possible. Closer examination of the results in Table 7.6 also shows that the improvement for the wrong model set is only obtained for speaker "ss" who has a much lower overall performance than speaker "mk".

7.5 Summary

This chapter presented three different techniques which are capable of improving the recognition accuracy of foreign accented speech without requiring adaptation material. The improvement is gained solely by exploiting the knowledge of possible substitution errors. This knowledge is incorporated in a mapping between a given pair of source and target languages. These algorithms also require an additional model set of the source language.

In Table 7.7 the recognition performance of predicted PBM, LMC and MM is compared. The largest performance increase is obtained for predicted MM, whereas predicted PBM and LMC perform roughly the same. The relative improvement obtainable with MM is as high as 27.2%. In other words, with this algorithm the recognition performance can be increased by a factor of one third if the source

Spkr	Base	PBM	$LMC_{0.3/0.46}$	MM
FL	20.3	15.5	14.1	12.1
PC	29.4	23.1	25.5	19.6
TS	26.5	21.1	24.1	18.9
MK	19.3	16.5	16.7	17.5
SS	45.8	38.2	34.7	34.9
Avg.	28.3	22.9	23.0	20.6

Table 7.7: Results summary for accent prediction using bilingual models: WER for baseline, predicted PBM with $a^*_{12} = 0.5$, LMC with optimal $b_{jj,best}$ per speaker, and MM with $\alpha = 0.5$

language of a speaker is known but no adaptation data are available.

Even though both predicted PBM and LMC yield similar improvements of up to 19% relative, they differ significantly in their dependency on good initial estimates. The recognition accuracy of predicted PBM varies only a little with varying the transition probability a^*_{12} between the source or target language model. On the other hand, the performance of predicted LMC heavily depends on the weights $b_{j,apriori}$. Because of this characteristic, it would be difficult to implement predicted LMC in a CALL system. However, the value of the optimal a-priori weights could be used to measure the fluency of a language student.

As with predicted PBM, predicted MM is quite independent of the choice of the merging parameter. Given its significant performance gains and its easy implementation, predicted MM appears to be the most suitable algorithm to implement in a CALL system in order to improve overall recognition accuracy without the need for non-native adaptation data. However, indirectly a small amount of data is required in order to build a mapping between the source and target langauge. Note also that a small bias might has been introduced in these experiments due to the fact that the test data were also part of those data that were used to derive the mapping.

Chapter 8

Measurement of Pronunciation Assessment

8.1 Introduction

It is relatively easy to calculate scores, which assess pronunciation, but it is far more difficult to derive methods to validate any scoring or assessment of accented speech; for a discussion of this topic see also [72]. As outlined in Section 2.3, previous work on automatic pronunciation scoring mainly concentrates on calculating word or sentence level scores, [40, 68, 23]. In order to validate such scores, the non-native data used by these research groups were annotated with similar scores on word or sentence level by human judges — either language teachers, trained phoneticians or native speakers without extra training. In all these cases it was possible to relate and validate the scores with a standard correlation measure. For example, in Neumeyer et al., [68], comparison of the scoring behaviour of the human judges yielded a correlation of 0.76 on sentence level. In another study, [54], where individual phone segments have been scored, this inter-judge correlation decreased to 0.55. This example indicates how difficult it is even for human judges to consistently judge pronunciation on a detailed level. It also demonstrates the subjectivity of the rating task.

However, in this work the target is to compare strings of phones which have been marked as to whether they have been mispronounced or not. This involves comparing strings of binary decisions rather than scores. Furthermore, these strings are of differing lengths due to insertions and deletions. Thus, standard correlation coefficients cannot be used directly to determine whether or not two different judgment agree. Therefore, this chapter presents a set of four performance measures to compare any set of two judgments. These two judgment can be two human judgments or one human and a computer-based judgment. In the latter case, these measures can be used to assess the effectiveness of the *GOP* scoring for detecting pronunciation errors.

8.2 The transcription of pronunciation errors

The non-native database used for assessment consists of target transcriptions based on a pronunciation dictionary and of transcriptions which have been annotated by human judges to contain the phone sequence which had actually been spoken. The utterance transcriptions marked with corrections will be referred to as *corrected transcriptions* and the transcriptions derived directly from the pronunciation dictionary will be referred to as the *dictionary-based transcriptions*. Finally, transcriptions in which each phone correction has been replaced by a single rejection symbol are referred to as *rejection-marked transcriptions*.

Two corrected transcriptions of the same utterance are difficult to align with each other due to insertions and deletions of phones. Therefore, all performance measures which are presented in this chapter compare transcriptions on a frame by frame basis. With this approach, measuring the similarity of two differently corrected transcriptions of the same utterance becomes equivalent to comparing the rejection/acceptance marking of corresponding speech frames.

Based on the rejection-marked transcriptions, the frame level markings are calculated as follows:

1. The phone level segmentation for each sentence is calculated by forced alignment of the acoustic waveform with the corrected transcriptions.

2. All frames corresponding to substituted, inserted or deleted phones are marked with “1”, all other ones with “0”. This yields a vector $\mathbf{x}$ of length N with $x(i) \in \{0, 1\}$. These vectors will be called transcription vectors.

3. The transitions between “0” and “1” in the transcription vectors are abrupt whereas in practice the precise location of the boundaries between correctly and incorrectly pronounced speech segments is uncertain. Moreover, segmentation based on forced alignments can be erroneous due to the poor acoustic modeling of non-native speech. For these two reasons, the vectors representing corrected transcriptions are smoothed by a Hamming window

$$x'(n) = \sum_{k=-N/2}^{N/2} x(k)w(n-k) \tag{8.1}$$

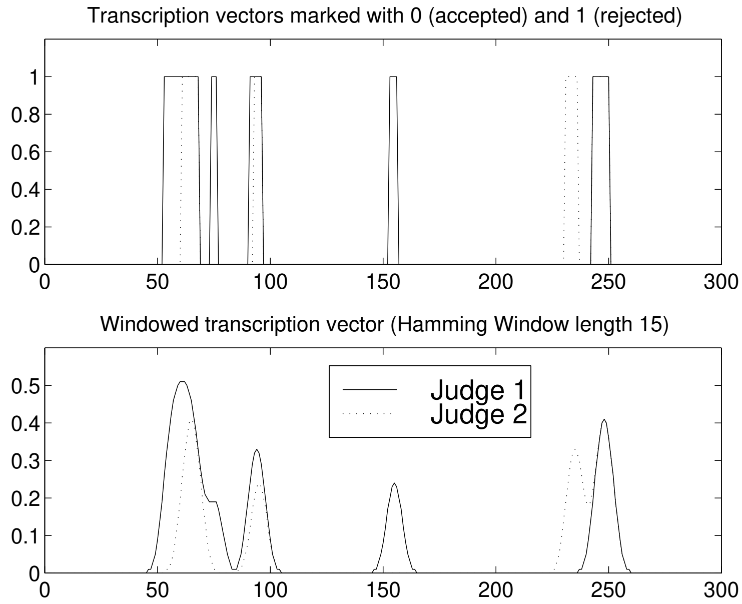

Figure 8.1: Smoothing effect of the windowing. Overlapping regions denote areas where both judges decided to reject the pronunciation of a phone

Using a speech frame period of 10msec, the length of a vowel tends to extend over 6-20 frames, whereas consonants can be much shorter. Also, if rejected frames in one transcription are immediately followed by rejected frames in the other transcription, the rejections can be considered to have been caused by the same pronunciation error. Based on these considerations, a window length of $N = 15$ was selected for all experiments. The effect of the smoothing window is illustrated in Figure 8.1.

8.3 Performance measures

This section defines the four performance measures used to compare transcriptions corrected by two judges or one judge and the automatic GOP scoring system presented in Chapter 9. These measures are based on similarity measurements between two reference transcriptions. Since the production of reference transcriptions must be done by human judges and is highly subjective, the same performance measures are also used to cross-validate the judges. Note that the performance measures are only concerned with the detection of pronunciation errors. They do not take account of the type of error which has occurred.

To cover all aspects of performance, four different dimensions are considered

- *Strictness* - how strict was the judge in marking pronunciation errors?

- *Agreement* - what is the overall agreement between the reference transcription and the automatically derived transcription? This measure takes account of all phones whether mispronounced or not.

- *Cross-correlation* - what is the overall agreement between the errors marked in the reference and the automatically detected errors? This measure only takes account of phones for which an error has been marked in one or both transcriptions.

- *Overall phone correlation* - how well do the overall rejection statistics for each phone agree between the two references?

8.3.1 Strictness

Firstly, human correction of the pronunciation of non-native speakers depends on subjective personal judgment. There will always be a large number of phones whose pronunciation is on the borderline between correct and incorrect; a stricter judge might declare more borderline cases as incorrect than another judge who is more benign. In the case of computer-based scoring, the choice of a rejection threshold determines how strict the scoring system will be. This *strictness of labeling, S*, can be defined as the overall fraction of phones which are rejected, i.e. relative strictness

$$S = \frac{Count\ of\ Rejected\ Phones}{Total\ Count\ of\ Phones} \tag{8.2}$$

As an example, the database used for assessment, see Chapter 5 contains a set of calibration sentences which were labeled by six different judges. Figure 8.2 shows the strictness of the judges for these calibration sentences where the mean and standard deviation are $\mu_S = 0.18$ and $\sigma_S = 0.05$, respectively.

A simple way to compare the strictness of two judges $J1$ and $J2$ is to use the difference between strictness levels for the two references, i.e.

$$\delta_S = \mid S_{J1} - S_{J2} \mid \tag{8.3}$$

8.3.2 Agreement

The overall *Agreement* (A) between two rejection marked transcriptions is defined in terms of the city-block distance between the corresponding transcription vectors, i.e.

$$A_{J1J2} = 1 - \frac{1}{N} \parallel \mathbf{x}_{J1} - \mathbf{x}_{J2} \parallel_C \tag{8.4}$$

where $\parallel \mathbf{x} \parallel_C = \sum_{i=0}^{N-1} \mid x(i) \mid$.

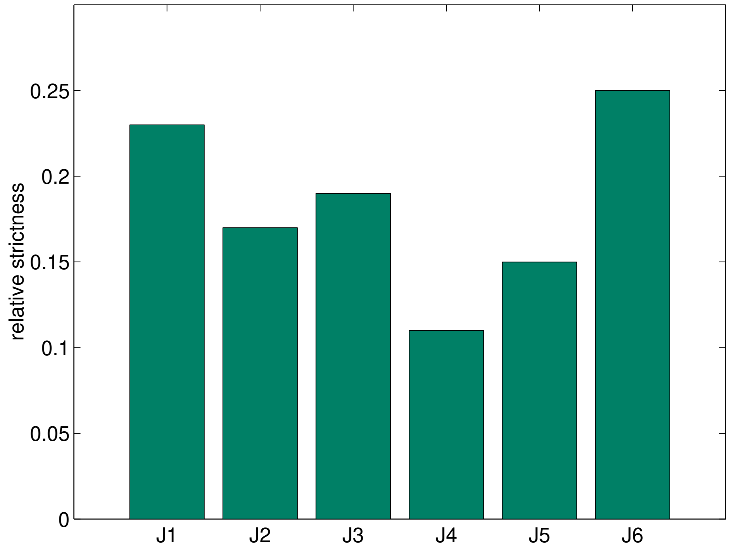

Figure 8.2: Relative strictness for all human judges measured on the calibration sentences

8.3.3 Cross-Correlation

Agreement measures overall similarity of two transcriptions by comparing all frames of an utterance. In contrast, the *Cross-Correlation* (CC) measure takes into account only those frames where either of these frames has a rejection marking.

$$CC_{J1,J2} = \frac{\mathbf{x}_{J1}^T \mathbf{x}_{J2}}{\| \mathbf{x}_{J1} \|_E \| \mathbf{x}_{J2} \|_E} \tag{8.5}$$

where $\| \mathbf{x} \|_E = \sqrt{\sum_{i=0}^{N-1} x(i)^2}$ is the standard Euclidean norm.

In other words, Cross-Correlation measures similarity between all segments which contain rejections in either of the two transcriptions. Because similarity of the rejection patterns with a human judge is the main design objective of the GOP scoring system, this measure has the highest importance.

8.3.4 Phone Correlation

Finally, *Phone Correlation* (PC) measures the overall similarity of the phone rejection statistics of all M phones in the given models set. Given a vector $\mathbf{c}$ of length M whose elements contain the count of rejections for each phone in a complete model

A	CC	PC	δ_S
0.91	0.47	0.78	0.06

Table 8.1: Averaged A, CC, PC and δ_S results based on correlating all possible pairs of judges. These values are the baseline against which automatic scoring performance will be measured.

set, phone correlation is defined as

$$PC_{J1,J2} = \frac{\sum_{m=0}^{M}(c_{J1}(m) - \mu_{c_{J1}})(c_{J2}(m) - \mu_{c_{J2}})}{\sum_{m=0}^{M}\sqrt{(c_{J1}(m) - \mu_{c_{J1}})^2(c_{J1}(m) - \mu_{c_{J1}})^2}} \tag{8.6}$$

where μ_c denotes the mean rejection counts.

8.4 Inter-judge labeling comparison

With the above derived set of performance measures it is now possible to compare the ratings of the human judges. Similarly as in [68], we measure inter-judge correlation based on the corrected transcriptions for the 20 calibration sentences.

In order to properly interpret the results of assessing a computer-based pronunciation system using manually-derived transcriptions as the reference, it is necessary to measure the inter-judge labeling consistency and to obtain an understanding of how the judges label the data. Their labeling is characterised by the phones they consider important for good pronunciation and thus tend to correct, by the consistency of the rejection patterns across different judges and finally by their strictness. In this section, the four performance measures described above are used in conjunction with the 20 calibration sentences to determine these characteristics.

Figure 8.3 shows averaged results of all the measures for each judge. These results have been calculated by averaging A, CC, PC and δ_S between the respective judge and all other ones. All results vary within an acceptable range, that is $0.85 < A < 0.95$, $0.40 < CC < 0.65$, $0.70 < PC < 0.85$ and $0.03 < \delta_S < 0.14$. Therefore, the labeling by different human judges can be considered as being reasonably consistent. However, Judge 5 is a slight outlier in that he has a lower average cross-correlation with the other judges. The total mean values over all pairs of judges of all four measures are shown in Table 8.1. These mean values will be used as benchmark values against which the performance of the automatic scoring as presented in Chapter 9 is measured.

Table 8.2 shows the similarity between the human judges and the baseline GOP scoring method (see Chapter 9) for each non-native speaker in that judge's group. It

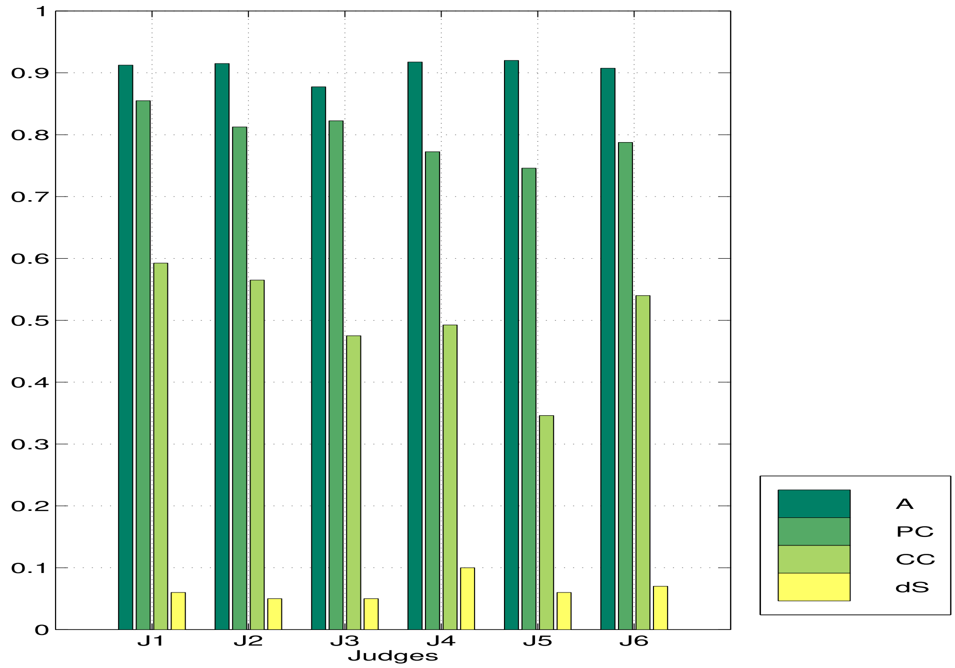

Figure 8.3: A, CC, PC and δ_S for each judge based on averaging the measures between the respective judge and all the other judges.

Judge	Speaker	Strictness	CC	PC
J1	Cal.	0.25	0.51	0.77
	ss	0.25	0.56	0.73
	ts	0.21	0.49	0.84
J2	Cal.	0.19	0.53	0.81
	yp	0.16	0.49	0.62
J3	Cal.	0.21	0.50	0.68
	mk	0.13	0.38	0.57
J4	Cal.	0.13	0.37	0.62
	sk	0.07	0.12	0.61
	as	0.11	0.37	0.50
J5	Cal.	0.16	0.22	0.71
	ay	0.19	0.50	0.61
	fl	0.16	0.43	0.56
	pc	0.19	0.50	0.62
	ky	0.23	0.48	0.34

Table 8.2: Similarity results between judges and the baseline GOP scoring grouped according to the judge who labeled the respective speaker sets. The speaker name *Cal.* denotes the calibration sentences.

can be seen that the intra-judge results are quite consistent. However, Judge 4 had a high acceptance level of non-native pronunciation, and thus corrected a significantly smaller portion of the data.

Figure 8.4 shows the CC and PC measures for each speaker grouped according to their native languages. Also shown on this figure are the genders of each speaker. From this figure and the data shown in Figure 8.3, it appears that the labeling of the human judges does not depend significantly on the mother tongue or the gender of the subjects, but mostly depends on the variability of human judges.

Finally, the rejection patterns for all judges are shown in Figure 8.5, which depicts the rejection counts for all phones for the judges. The strong correlation between the rejection pattern of all judges is clearly evident.

8.5 Summary

In this chapter four performance measures were developed in order to assess the quality of the phone level pronunciation scoring, both by human judges and by an

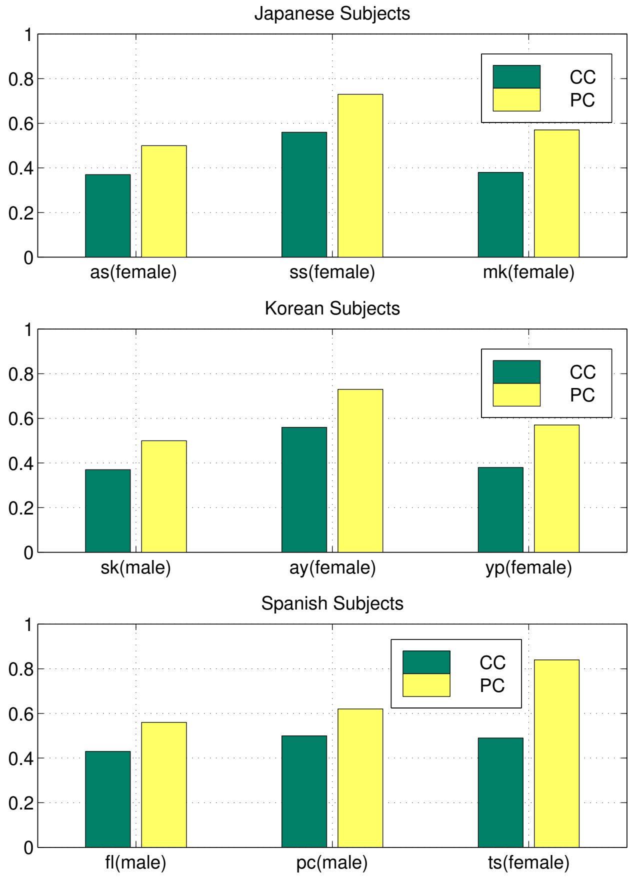

Figure 8.4: CC and PC results grouped according to each student's mother-tongue.

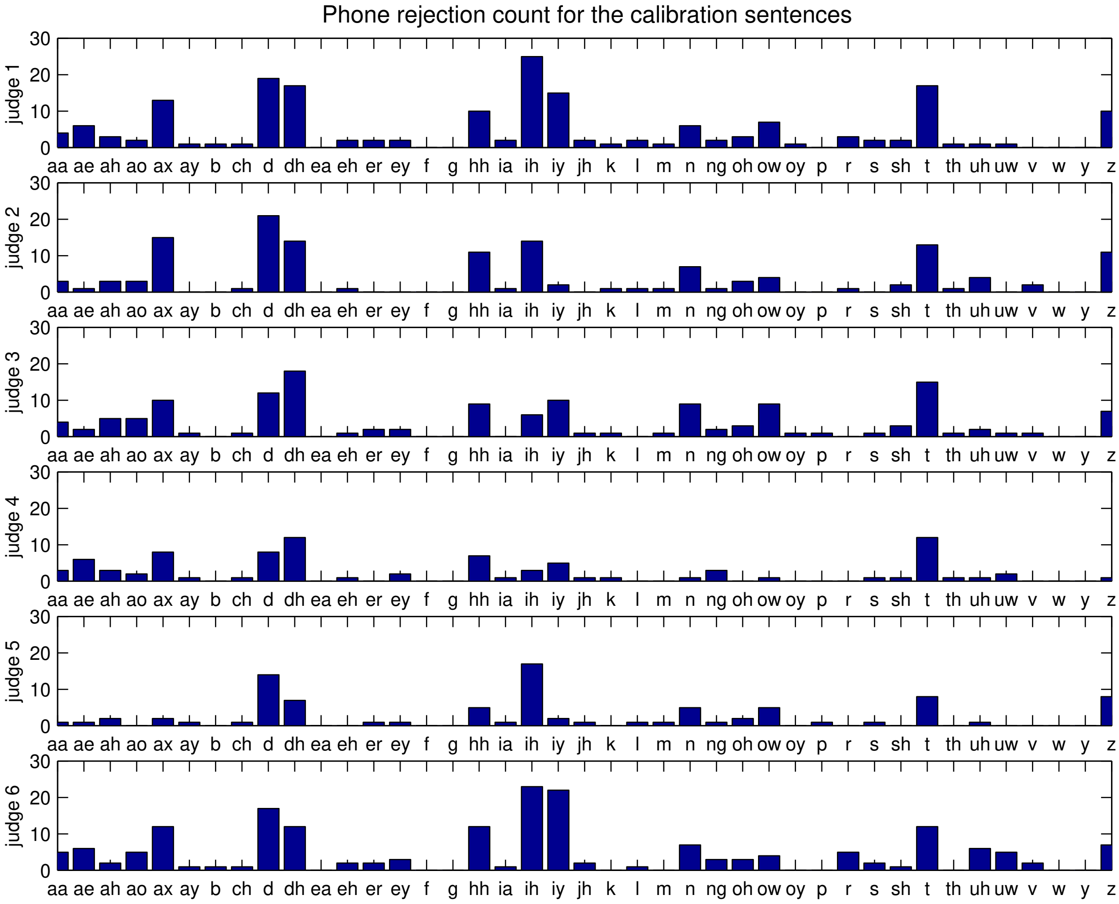

Figure 8.5: Rejection counts of all phones for all judges based on the calibration sentences to show the correlation between the rejection pattern of different judges.

automatic scoring system as described in the following chapter.

The analysis of human judgment characteristics showed that although there is significant variability in the labeling of each judge, there is nevertheless sufficient common ground to form a basis for assessing the performance of the various automatically derived pronunciation scoring methods. The averaged values for human judges of these four performance measures provide the baseline against which the performance of the automatic scoring technique has to be measured.

Chapter 9

Pronunciation Assessment: The GOP Scoring Algorithm

9.1 Introduction

In this chapter a family of newly developed pronunciation scoring techniques will be presented and experimentally evaluated. This family of techniques consists of a baseline algorithm which calculates an individual score for each phoneme in an utterance of a language student. Then, several modifications are developed in order to refine the baseline technique. These refinements also build upon the non-native modeling techniques of the previous chapters. This chapter is organised as follows: Section 9.2 describes the current state-of-the-art of confidence scoring in automatic speech recognition applications, since the algorithms here proposed are based on confidence scoring. Next, Section 9.3 gives the theoretical foundation of the new pronunciation scoring algorithms. Then, Sections 9.4 and 9.5 discuss the performance of these algorithms. First, the algorithms are tested with speech where pronunciation errors have been artificially introduced. Following this, they are applied to data from the non-native database discussed in Chapter 5.

9.2 State-of-the-art of confidence scoring

The task of assessing the pronunciation quality of individual phones can be compared to the task of calculating the confidence that a phones has been correctly recognised. Based on this comparison, the target of the work in this chapter is to develop a scoring algorithm which obtains high confidence scores for correctly pronounced phones and low scores for non-standard pronunciation. This section presents an overview of the state-of-the-art in confidence scoring which will serve as the theoretical foundation of the new scoring techniques. Existing confidence scoring algorithms can be grouped into two main classes: either the algorithms are based on *a-posteriori* likelihoods or on binary classifiers. Additionally, there exist more confidence scoring techniques. For example, [18] developed a technique to combine acoustic and language model scores.

9.2.1 A-posteriori based confidence scores

A large number of confidence calculation techniques uses *a-posteriori* recognition probability scores. However, depending on the setup of a recogniser such a-posteriori scores can often only be approximated yielding slightly different actual implementations of confidence measures.

The algorithms proposed in [56, 76, 22] differ only in the approximations of the calculation of the *a-posteriori* based confidence scores for keyword spotting. In all these approaches the confidence score of having spotted a keyword is given by an approximation of the *a-posteriori* probability of the keyword. Additionally, in [22], these confidence measures are combined with other confidence related features, such as the number of strong recognition hypotheses available, in order to build a post-classifier. This classifier has been used in a second recognition pass to decide whether a keyword has been spotted correctly or not. With this two-pass approach an increase in the decision accuracy from roughly 1/2 to 2/3 has been reported for deciding whether a word was correct or not. Likewise, Caminero et al., [15], and De la Torre et al., [26], proposed to combine *a-posteriori* probability scores with garbage model likelihood scores after a first recognition pass using confidence tests based on linguistic information. Other approaches to confidence scoring, too, have been based on combining a range of different knowledge sources into feature vectors which are put through a classifier yielding a tag for each hypothesised word, see for example [82]. In general, this idea can be extended to combine *a-posteriori* scores with other knowledges sources. For example, knowledge of a speaker's native language will be employed to refine the baseline algorithm, see Section 9.3.4.

9.2.2 Classification based confidence scores

Confidence scoring can also be interpreted as a binary classification task where the amount of false acceptances has to be minimised while maximising the amount of correct acceptances and correct rejections. Along this line of thought lie the confidence scoring approaches described in [75, 102]. Here, the acoustic confidence score for a phone is presented as the ratio of the likelihood that a phone was correctly recognised given its recognition likelihood score versus the likelihood that a phone was incorrectly recognised given its recognition likelihood score. Similarly, in [20, 79, 61], a neural net classifier using a range of features such as likelihood ratios and durations was built. The results of a first recognition pass are fed into the classifier, whose output then determines whether a keyword has been spotted or not.

However, in all cases where detectors or classifier are built, correct transcriptions are required in order to collect the likelihood score statistics on which the classifier is based. Therefore, it is difficult to apply these measures to the task of pronunciation assessment because it is not possible to find correct transcriptions of non-native speech. It is only possible to obtain approximations of these transcriptions, see also Chapter 8. Thus, the algorithm developed in the next section is based on a *a-posteriori* confidence score.

9.3 "Goodness of Pronunciation(GOP)" Scoring

In the case of pronunciation scoring, the underlying assumption is that the transcription of an utterance spoken by the language student is given. For example, a typical task is to ask the student to read out a given text prompt or to choose an answer from a small number of choices. In this section, a family of "Goodness of Pronunciation(GOP)" scoring methods is presented which calculate an individual score for each phone of an utterance for which the dictionary-based transcription is known.

These GOP algorithms have to be seen within the framework of the research regarding pronunciation assessment in CALL done by other research groups, as described in detail in Section 2.3. In particular, during the period of time when the research of this thesis was executed, other research groups have developed related pronunciation scoring techniques. Although few of them attempt to score pronunciation on the phone level, Neumeyer et al., [68], and Cucchiarini et al., [23] have shown that *a-posteriori* phone probabilities can be applied to pronunciation scores on the word and sentence level or even selected phones, see also [54].

9.3.1 Basic GOP algorithm

The aim of the GOP measure is to provide a score for each phone in an utterance. In computing this score it is assumed that the orthographic transcription is known and that a set of hidden Markov models is available to determine the likelihood, $p(O|q_j)$, of the acoustic segment O corresponding to a phone q_j. Under these assumptions, the quality of pronunciation for any phone q_i is defined to be the duration normalised log of the posterior probability $P(q_i|O)$ that the speaker uttered phone q_i given the corresponding acoustic segment O. That is

$$GOP_1(q_i) \equiv |\log(P(q_i|O))| / NF(O) \tag{9.1}$$

$$= |\log\left(\frac{p(O|q_i)P(q_i)}{\sum_{j=1}^{J} p(O|q_j)P(q_j)}\right)| / NF(O) \tag{9.2}$$

where J is the total number of phone models and $NF(O)$ is the number of frames in the acoustic segment O which starts in frame f_s and ends in f_e.

Assuming all phones are equally likely ($P(q_j) = P(q_i)$ for any j and i) and that the sum in the denominator can be approximated by its maximum, the basic GOP measure becomes

$$GOP_1(q_i) = |\log\left(\frac{p(O|q_i)}{\max_{j=1}^{J} p(O|q_j)}\right)| / NF(O) \tag{9.3}$$

$$= \left|\frac{log(p(O|q_i))}{NF(O)} - \frac{\max_{j=1}^{J} log(p(O|q_j))}{NF(O)}\right| \tag{9.4}$$

The acoustic segment boundaries and the corresponding likelihoods are determined from Viterbi alignments. Firstly, the numerator of equation 9.3 is computed using a forced alignment in which the sequence of phone models is fixed by the known transcription. Secondly, the denominator is determined using an unconstrained phone loop. This is the same arrangement as commonly used in word spotting, see also [56]. One difficulty in equation 9.3 is that if a mispronunciation has occurred, the alignments of the phone loop will differ from the alignment in the forced alignment. Hence, the denominator score is determined by simply summing the log likelihood per frame over the duration of the segment O. In practice, this will often mean that more than one phone in the unconstrained phone sequence has contributed to the computation of $\max_{j=1}^{J} p(O|q_j)$. If N phones contribute to this likelihood, the actual computation will take the following form

$$\frac{log(p(O|q_j))}{NF(O)} = \sum_{i=1}^{N} \frac{log(p(O|q_{ji})}{f_{ie} - f_{is}} \tag{9.5}$$

where f_{is} and f_{ie} denote start and end frame number for the ith phone occurring during the current interval from f_s to f_e. An illustration of these alignments is given in Figure 9.1.

With this GOP score, a system to score pronunciation on the phone-level is easily implemented. A block diagram of the resulting GOP scoring mechanism can be seen in Figure 9.2. The front-end feature extraction converts the speech waveform to a sequence of frames with mel-frequency cepstral coefficients (MFCC) and these are used in two recognition passes: the forced alignment pass and the phone recognition

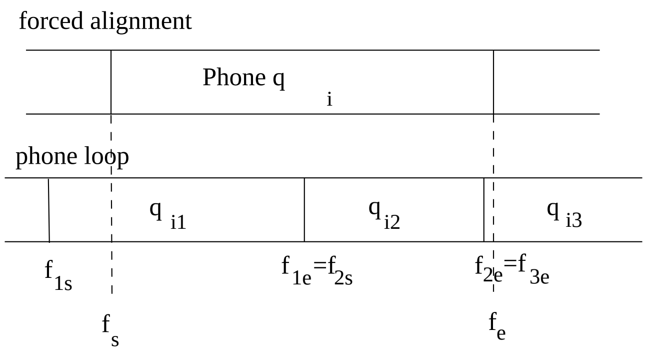

Figure 9.1: Contribution of several phones in the phone loop to the GOP score of phone q_i

pass where each phone can follow the previous one with equal probability. Based on these results, the individual GOP scores are calculated for each phone as defined in the previous equations. Finally, a threshold is applied to each GOP score to reject badly pronounced phones. The choice of the threshold depends on the level of strictness required. The selection of suitable thresholds is discussed further in Section 9.5.

9.3.2 Speaker adaptation

The reliability of the GOP scoring procedure described above depends on the quality of the acoustic models used. Since the aim of the GOP measure is to assess pronunciation quality with respect to native speaker performance, it is reasonable to use native speakers to train the acoustic models. However, non-native speech is characterised by different formant structures compared to those of a native speaker for the same phones, see also Section 3.3.1 and [9]. This can lead to phone recognition errors. Hence, some degree of speaker adaptation may be justified. To test this hypothesis, the GOP measure can be computed using models whose Gaussian means have been adapted using Maximum Likelihood Linear Regression (MLLR), [58]. In order to achieve speaker normalisation without adapting to specific phone error patterns, this adaptation is limited to a single global transform with a full transformation matrix and several iterations.

The adaptation or prediction methods as developed in the previous chapters could be applied, too. However, all the techniques of foreign-accent adaptation as presented in Chapters 4 and 7 already incorporate knowledge about systematic mispronunciations. Therefore, using such adaptation would have a counter-effect on

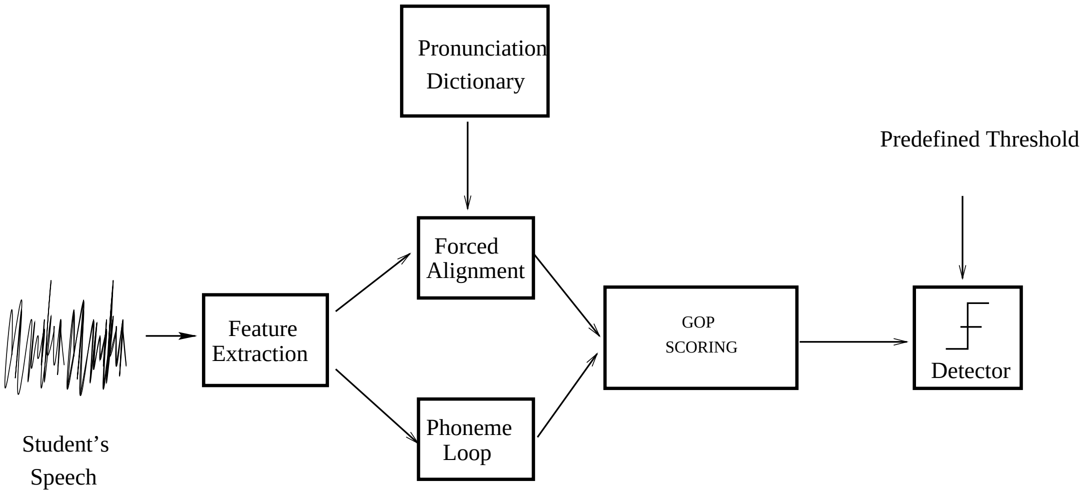

Figure 9.2: Block-diagram of a pronunciation scoring system: phones whose scores are above the predefined threshold are assumed to be badly pronounced and are therefore rejected.

the desired capability of the GOP scoring to detect mispronunciations.

9.3.3 Phone dependent thresholds

So far a single acceptance/rejection threshold for all phones has been assumed. However, in practice, the acoustic fit of phone-based HMMs differs from phone to phone. For example, fricatives tend to have larger variability in their realisations than vowels suggesting that a higher threshold should be used in the latter case.

A simple phone-specific threshold can be computed from native GOP statistics. For example, the threshold for a phone q_i can be defined in terms of the mean μ_{q_i} and variance σ_{q_i} of all the GOP scores for phone q_i in the training data of native speech

$$T_{q_{i1}} = \mu_{q_i} + \alpha\sigma_{q_i} + \beta \tag{9.6}$$

where $0.8 < \alpha < 1.3$ and $-1.0 < \beta < -2.0$ are empirically determined scaling constants, yielding thresholds on a similar scale as the global threshold, but adapted to individual phones. The assumption here is that averaging native GOP scores will reduce the effect of errors in the phone recogniser.

A reasonable target for an automatic pronunciation system is to perform as well as a human judge. One way to approximate human performance is to learn from human labeling behaviour. Let $c_n(q_i)$ be the total number of times that phone q_i uttered by speaker n was marked as mispronounced by one of the human judges in the training database, see also Section 8.3.4. Then, a second phone dependent threshold can be defined by averaging the normalised rejection counts over all speakers

$$T_{q_{i2}} = log \frac{1}{N} \sum_{n=1}^{N} \left(c_n(q_i) / \sum_{m=1}^{M} c_n(m) \right) \tag{9.7}$$

where M is the total number of distinct phones and N is the total number of speakers in the training set. Due to the normalisation the average counts are in the range of 0 to 1 so that the resulting logarithmic values yield thresholds on a similar scale to those defined in equation 9.6. In Figure 8.5 the rejection counts for each speaker can be seen.

9.3.4 Explicit error modeling

Pronunciation errors can be grouped into two main error classes. The first class contains individual mispronunciations which occur if a student is not familiar with the pronunciation of a specific word. The second class consists of substitutions of native sounds for sounds of the target language which do not exist in the native language. This latter error type will be referred to as systematic mispronunciations. Because the GOP method described so far does not employ models for the phones of a student's native language, incorrect acoustic modeling of the non-native speech will occur especially in the case of systematic mispronunciations. Therefore, the detection of these errors might be improved if knowledge of the native tongue of the learner can be included in the GOP scoring. This idea is similar to the underlying principle used in all the algorithms for the acoustic modelling of non-native speech as presented in the previous chapters.

For this purpose a recognition network has been implemented incorporating both correct pronunciation and common pronunciation errors in the form of error sub-lattices for each phone, using the phone model sets of both the target and the source language. Concatenating these sub-lattices according to the target transcriptions yields the desired error network for any utterance. For example, Figure 9.3 shows the resulting network for the word "but". The list of possible errors of a Spanish speaker learning English has been taken from [53], some examples of which are listed in Table 9.1. Alternatively, the substitution patterns derived in Section 3.4 can be used as well. The recognition output of such an error network will be a sequence of phones corresponding to the target transcription $q_i = q_{i_t}$ in the case that the target pronunciation was more likely or to an error phone $q_i = q_{i_e}$ otherwise.

A straightforward detector of systematic mispronunciations based on an error network could consist of rejecting all phone segments where an error phone has

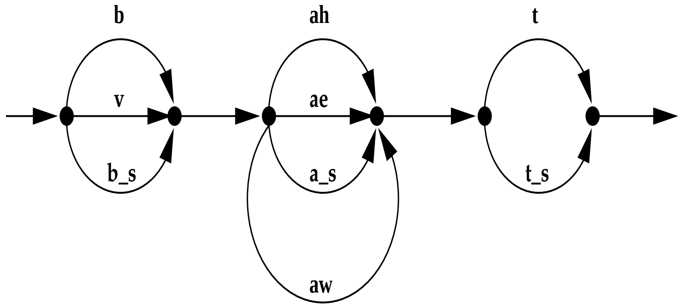

Figure 9.3: Example error-network for the word 'but', created through concatenating the sub-lattice of possible errors for each phone, the topmost phones correspond to the target transcription. (Phone names with subscript 's' denote Spanish models.)

British	Expected Errors	British	Expected Errors
/b/	/b/, /v/, /b_s/	/k/	del, /k_s/
/d/	/dh/, /d_s/	/l/	/l_s/
/th/	/f/, /s/, /f_s/, /s_s/	/ah/	/a_s/, /aw/, /ae/
/s/	/hh/, del	/uh/	/uw/, /u_s/
/ch/	/ch_s/	/ae/	/eh/, /e_s/
/jh/	/ch/, /ch_s/	/oh/	/o_s/

Table 9.1: Expected errors of a Spanish speaker for some British-English phones. (Phone names with subscript 's' denote Spanish models).

been recognised. However, such an approach would ignore the information about the likelihood of the occurrence of such an error. Hence, the posterior likelihood of each error phone $P(q_{i_e}|O)$ is computed by normalising the likelihood of the likely phone in the output of the error network with the recognition results of a phone loop network including acoustic models of both the target language and the source language applying equation 9.3.

Knowledge of $P(q_{i_e}|O)$ allows the posterior probability of the target phones q_{i_t} to be calculated in all phone segments containing systematic mispronunciations:

$$\begin{aligned} P(q_{i_t}|O) &= 1 - \sum_{j \neq q_{i_t}} P(q_j|O) \\ &\approx 1 - \max_{q \neq q_{i_t}} P(q_j|O) \\ &= 1 - P(q_{i_e}|O) \end{aligned} \tag{9.8}$$

Again the assumption has been made that the above sum can be approximated by its maximum. Thus, scores for systematic mispronunciations $GOP_e(q_i)$ are defined as

$$GOP_e(q_i) = \begin{cases} \mid log(1 - P(q_{i_e}|O)) \mid & \text{if } q_i = q_{i_e} \\ 0.0 & \text{otherwise} \end{cases} \tag{9.9}$$

Combining the basic GOP_1 with GOP_e yields a second GOP metric which includes additional penalties for scores of phone segments where systematic errors were recognised.

$$GOP_2(q_i) = GOP_1(q_i) + K\ GOP_e(q_i) \tag{9.10}$$

where K is a scaling constant.

9.4 GOP Experiments with artificial speech

Before evaluating this new GOP scoring algorithm with the non-native database, see also Chapter 5, some preliminary tests have been executed with artificial data. This artificial data was taken from the Resource Management (RM) database, which consists of continuous speech spoken by a variety of North American speakers. The dictionary used was based on a dictionary from Carnegie Mellon University using a set of 48 phones for the phonetic transcriptions, including two silence models. Based on these RM data, the artificial data was created by manipulating the pronunciation dictionary so that the pronunciations are changed to contain different phones. For instance, all occurrences of the sound $/aa/$ have been changed to $/iy/$ and so forth. Thus, speech data with known locations of pronunciation errors has been created.

Because these error locations are known, it is now easier to measure the reliability of GOP-based error detection with these artificial data than with non-native speech, where the judgment from the phoneticians are coloured by subjectivity.

In order to enable a more detailed analysis of the GOP scoring reliability for the artificial data, the following four decision types are defined:

1. Correct Acceptance (CA): A phone was pronounced correctly and is detected to be correct;

2. False Acceptance (FA): A phone was pronounced incorrectly and is detected to be correct;

3. Correct Rejection (CR): A phone was pronounced incorrectly and is detected to be incorrect;

4. False Rejection (FR): A phone was pronounced correctly and is detected to be incorrect.

For a given threshold, statistics of all these four decision types can be collected. Let the accuracy of making the correct decision of accepting or rejection the pronunciation of a phone be defined as scoring accuracy, $SA = CA + CR$. Then SA can be plotted as a function of FA for a range of thresholds. Such plots allow a system to be designed for optimal performance, which is defined as optimal scoring accuracy for a given acceptance level of false acceptances. In Figure 9.4 the performance results of an experiment with speaker-independent Gaussian mixture monophones, measuring scoring accuracy versus the false acceptance rate, can be seen. For comparison, the performance using speaker-independent tied-state triphones, which model more closely context and coarticulation is given as well. The poorer performance of triphone models is perhaps due to the fact that there are far fewer monophones (49) compared to the 112849 logical (6900 physical) triphones resulting in the monophones being more discriminative. Both model sets were trained on 73 speakers of the RM database; each speaker set consisted of 40 sentences. For the Gaussian mixture monophones a scoring accuracy of 90% at a false acceptance rate of 8% can be achieved when choosing a suitable threshold. These results show that — at least for this setup with artificially generated pronunciation errors — the GOP scoring method is a viable assessment tool.

In addition to the above experiment with monophones and triphones, different choices of spectral feature vectors have been tested as well, see also [94]. In the above described experiment, a feature vector consisting of 13 Mel-frequency cepstral coefficients together with 13 delta, 13 acceleration and 3 energy coefficients was used. In Table 9.2, results for Mel-frequency cepstral coefficients (MFCC), Mel-frequency

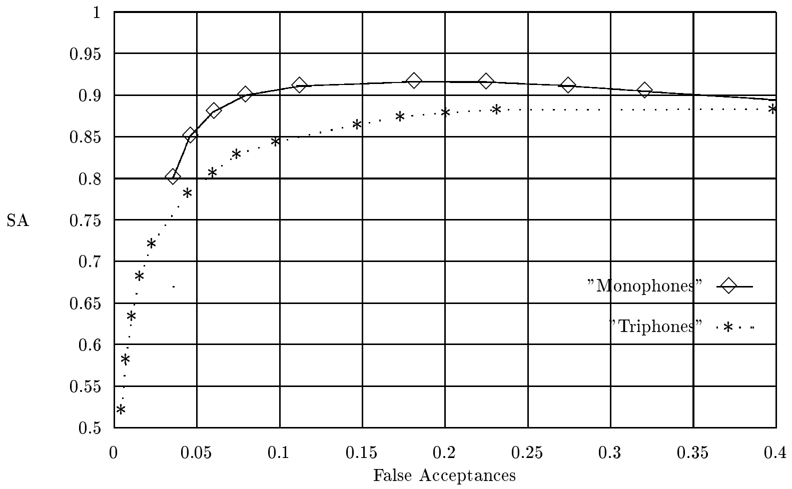

Figure 9.4: Scoring accuracy versus false acceptance for monophones and triphones

Model	SA in % for fa= 8%
MFCC	80
MFCC-E-D-A	88
MFCC-E-D-A-Z	90

Table 9.2: Scoring Accuracy for different feature vector selections

cepstral coefficients together with energy, delta and acceleration coefficients (MFCC-E-D-A), and the same vector with mean cepstral normalisation (MFCC-E-D-A-Z) can be seen. The feature vector which includes delta and acceleration coefficients performs best of all different feature vector types.

Based on these experimental results, Gaussian mixture monophones will be used in all following experiments with the non-native data together with feature vectors consisting of 13 Mel-frequency cepstral coefficients (with cepstral mean normalisation) plus 13 delta, 13 acceleration and 3 energy coefficients.

9.5 GOP experiments with non-native speech

This section presents performance results for both the basic GOP scoring method and its refinements as described in Section 9.3. The recognition setup consists of Gaussian mixture monophone HMMs trained on the British English corpus WSJ-CAM0 [41]. These hidden Markov models were built using the HTK Toolkit, [101].

For automatic GOP scoring, the values of agreement A, cross-correlation CC and phone correlation PC vary according to the level of strictness applied, which

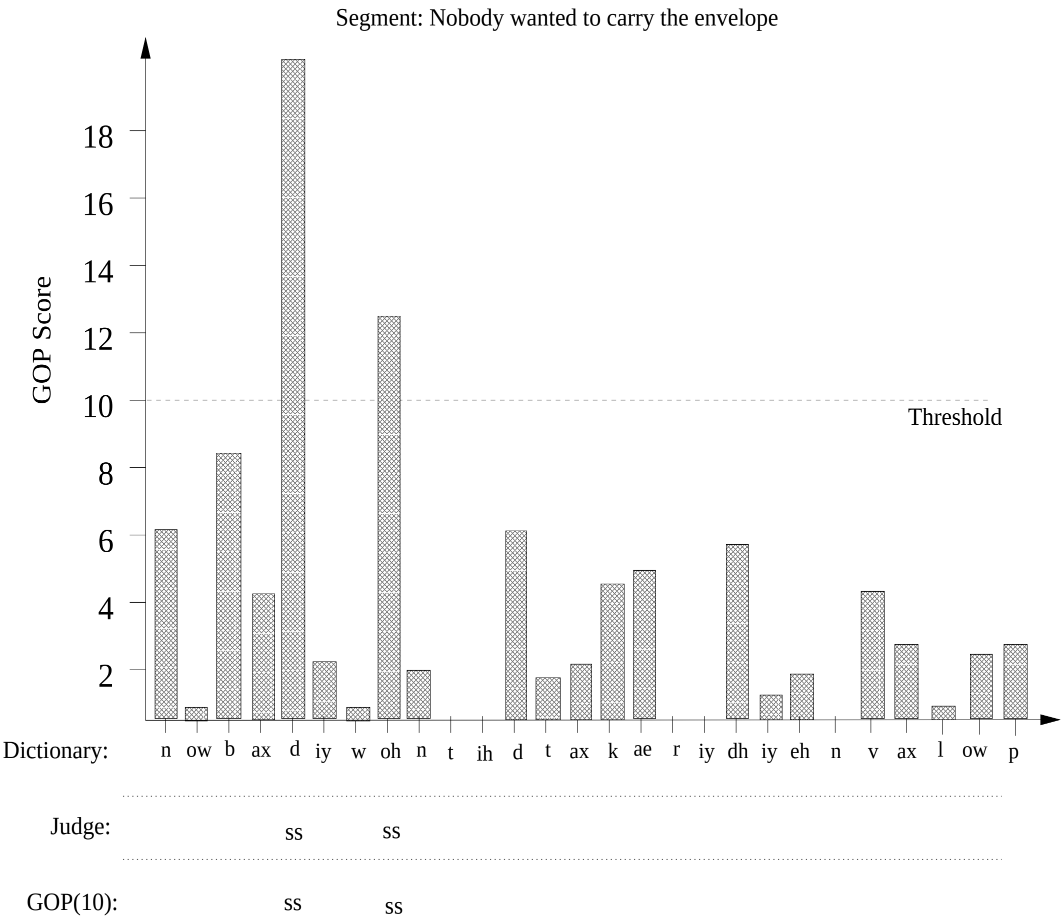

Figure 9.5: GOP Scoring results for first example sentence, 'ss' denotes the location of a rejection, the automatically rejected phones correspond to GOP scores above the threshold.

again depends on the choice of a threshold.

Three examples of GOP scoring are depicted in Figures 9.5, 9.6 and 9.7. Varying the threshold determines the number of rejections. In the first example sentence, Figure 9.5, complete agreement between the human judge and the GOP scoring can be found for a threshold of 10.0. The subject, a female speaker with Latin-American accent, pronounced "nobody" more like "nobory" and the vowel in "want" like $/ah/$.

The next example in Figure 9.6 demonstrates that swallowing or deleting of sounds can be detected as well. In this example the student, a female Japanese speaker, says "visibily" instead of "visibility". In order to demonstrate the influence of varying the strictness of the GOP scoring by varying the threshold, the acceptance/rejection pattern for two thresholds, $T = 4$ and $T = 5.5$ are shown. Variations in strictness mainly affect sounds which are not that clearly pronounced, but which

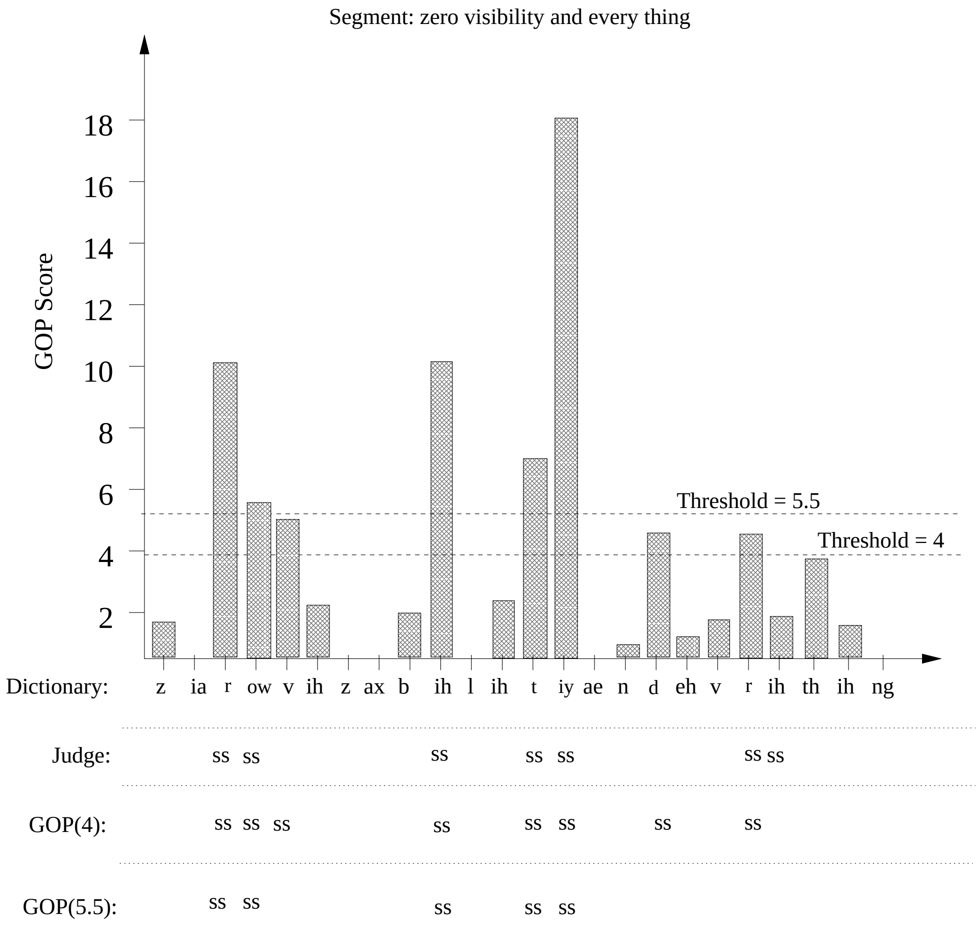

Figure 9.6: GOP scoring example sentence,

are not exactly wrong either. In the stricter case, i.e. $T = 4$, the computer score indicates a mediocre pronunciation of the $/v/$ in visibility, whereas the judge accepted it as correct. The same applies to the $/d/$ in "and". The judgments for "every" are an example of the difficulty in precisely locating mispronunciations. Both judgments detect a mistake. However, the judge does not accept both the $/r/$ and the $/ih/$ whereas the automatic scoring only rejects the $/r/$.

The previous example demonstrated that obvious mispronunciations can be reliably detected by the automatic scoring. In this last example, Figure 9.7, the male Latin-American speaker does not know the pronunciation of the word 'pint' and therefore pronounces the vowel in this word as $/ih/$ instead of $/ay/$. Additionally, this example again shows that due to bad alignments in the recognition passes the GOP scoring can be inprecise in the localisation of a mispronunciation. The subject mispronounces the $/v/$ in 'very', however, the automatic scoring attaches a bad

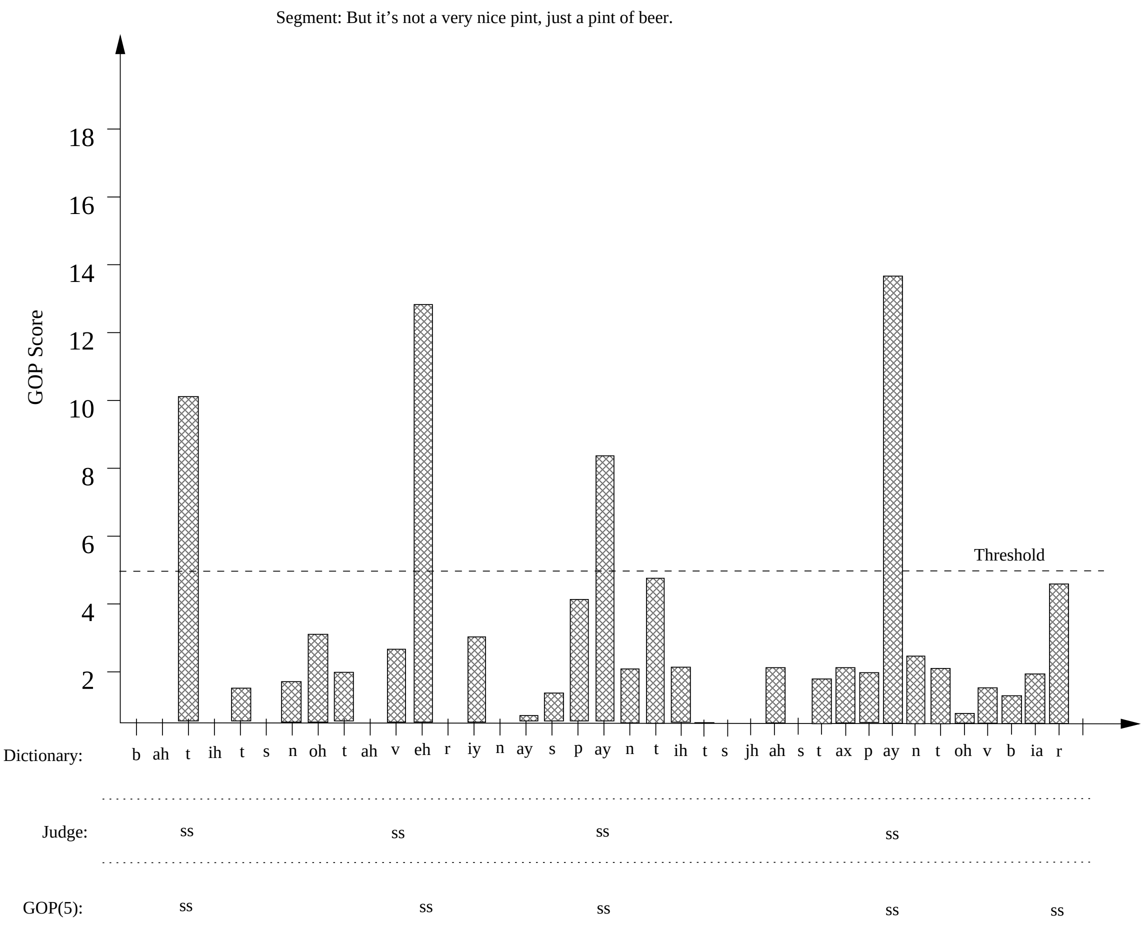

Figure 9.7: *GOP* Scoring results for third example sentence. Student pronounce 'pint' as 'peent'.

score to the following sound $/eh/$.

These examples demonstrate how the automatic scoring operates. In the following sections the optimal setting of variable parameters will be discussed in more detail. The performance measures, which have been derived in Chapter 8, will be applied to compare the automatic scoring with the scoring by human raters. These measures will be used for the baseline method as well as for its modifications. All scoring methods share one main parameter which influences the pronunciation assessment. This is the threshold. Additionally, some of the refinements also have scaling constants which have to be optimised, too.

9.5.1 Varying the threshold

The value of the rejection threshold can influence the scoring performance considerably. Because the experiments in this section attempt to achieve an automatic scoring that comes as close as possible to the scoring made by human judges, the target here is to find a rejection threshold which yields a rejection/acceptance pattern similar to the judges. Therefore, the range of rejection thresholds, as studied in this experiment, is restricted to be within one standard deviation of the judges strictness, see also Section 8.3.1 for the definition of δ_S,

$$|\delta_S| \leq \sigma_S \qquad with\ \sigma_S = 0.05. \tag{9.11}$$

In Figure 9.8 the values of the three performance measures, agreement (A), cross-correlation (CC) and phone correlation (PC) as derived in Chapter 8, are given for a single speaker as a function of the rejection threshold. In this figure the vertical lines denote the acceptable range of threshold settings. It can be seen, that the performance values do not vary greatly within this range. Thus, the performance of an automatic assessment system will not degrade significantly if a non-optimal threshold is chosen. This finding renders the GOP scoring suitable for commercial applications, because any threshold is always going to be less suitable for some students than for other students. Also, in a commercial system it would be a good idea to automatically adjust the threshold to the proficiency level of a student. A beginner might be less frustrated if he or she is judged more benignly than a more fluent student.

9.5.2 Results for the basic GOP algorithm

The previous section discussed the choice of the rejection threshold which influences the degree of strictness of the automatic scoring system. This section will now analyse the performance results for the baseline GOP algorithm in more detail.

In Table 9.3 the scoring of the baseline GOP for the calibration sentences has been compared with the labeling pattern of those six judges, who labeled the calibration data, too. For the example threshold of $T = 4.5$ high similarity between a judge and the automatic scoring can be achieved, see, for instance, Judge 2 versus $GOP(4.5)$. On the other hand, there also exists some disparity, see for instance Judge 4 or 5. This result confirms the observation in Chapter 8 that the rejection pattern among human judges can vary considerably, especially with regard to the cross-correlation measure. In Table 9.3 it can also be seen that CC has a significantly higher standard deviation than the other measures.

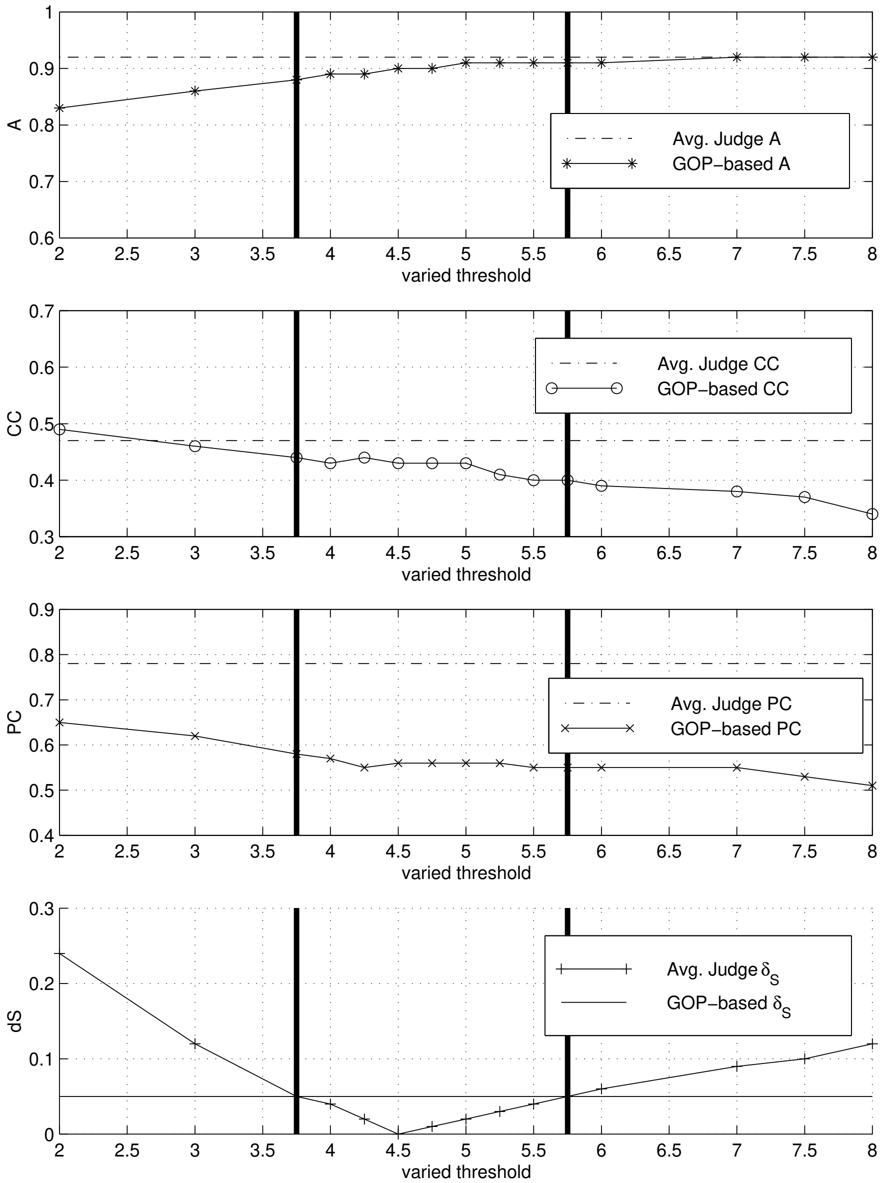

Figure 9.8: Dependency of A, CC, PC and δ_S on threshold variation, based on data for 'fl', a male Spanish speaker. The range inside the bold lines is the range of valid δ_S.

Judges	A	CC	PC	δ_S
J1 vs GOP(4.5)	0.86	0.51	0.77	0.01
J2 vs GOP(4.5)	0.87	0.53	0.80	0.06
J3 vs GOP(4.5)	0.86	0.50	0.68	0.05
J4 vs GOP(4.5)	0.86	0.36	0.67	0.13
J5 vs GOP(4.5)	0.86	0.23	0.75	0.10
J6 vs GOP(4.5)	0.86	0.50	0.67	0.01
Mean (σ)	0.86 (0.01)	0.44 (0.11)	0.72 (0.05)	0.06 (0.04)

Table 9.3: Performance of baseline GOP versus each judge. The results are based on the assessment by the judges and the GOP scoring of the calibration sentences (σ = standard deviation)

Next, Table 9.4 shows the optimal values of A, CC and PC which are achievable for each speaker when using the basic GOP scoring technique within the allowed threshold range. As can be seen, the optimal threshold is speaker dependent. However, apart from speakers 'sk' and 'as', a threshold of $T = 4.5$ is close to optimal for all speakers. Because 'sk' and 'as' were the two speakers whose transcriptions were annotated by the very strict judge (Judge 4), see Section 8.3.1, these two speakers are not included in the averaged results presented in the remainder of this chapter.

The performance results for the automatic GOP scoring metrics as discussed in Section 9.3 are summarised in Figure 9.9. The first bar on the left marked "Baseline" shows the performance of the basic GOP_1 metric with a fixed overall threshold as discussed in Section 9.3.1. The final bar on the left shows the human-human performance on the calibration sentences used as the benchmark values. As can be seen, the scores for A and CC are similar between human and automatic scoring whereas for PC, the automatic scoring is worse by about 20%. The second bar marked "MLLR" shows the effect of applying global speaker adaptation. An improvement of 5% has been obtained for PC at the cost of a small decrease in CC. The third and fourth bars show the effects of using individual thresholds for each phone based on averaging native scores T_{p_1} and on averaging the judges' scores T_{p_2}. As can be seen, thresholds derived from the statistics of the judges' scoring yield the best performance. This is probably so because these thresholds are directly related to desired rejection statistics, that is the individual thresholds are based onrejction patterns which were obtained from data which included the test data.

In the next experiment, MM adaptation is contrasted with MLLR adaptation. In Table 9.5 the averaged results are given for the three Spanish accented as well as

ID	Thres	A	CC	PC	δ_S
fl	5	0.91	0.43	0.56	0.02
pc	4.5	0.87	0.50	0.62	0.04
yp	4.0	0.90	0.49	0.62	0.02
ts	4	0.87	0.49	0.84	0.03
ky	5	0.84	0.48	0.34	0.04
sk	7	0.90	0.12	0.61	0.06
ss	4.5	0.85	0.56	0.73	0.05
as	7	0.90	0.37	0.50	0.07
mk	4.5	0.90	0.38	0.57	0.07
ay	4.5	0.90	0.50	0.61	0.05
j1	4.5	0.86	0.51	0.77	0.01
j2	5	0.87	0.53	0.81	0.04
j3	4.5	0.86	0.50	0.68	0.05
j4	7.5	0.91	0.43	0.62	0.00
j5	5.5	0.88	0.22	0.71	0.05
j6	4	0.85	0.52	0.65	0.03
GOP Mean		0.88	0.47	0.60	0.05
Human Mean		0.91	0.47	0.78	0.05

Table 9.4: Thresholds yielding optimal performance for all non-native speakers of the database (using basic GOP scoring).

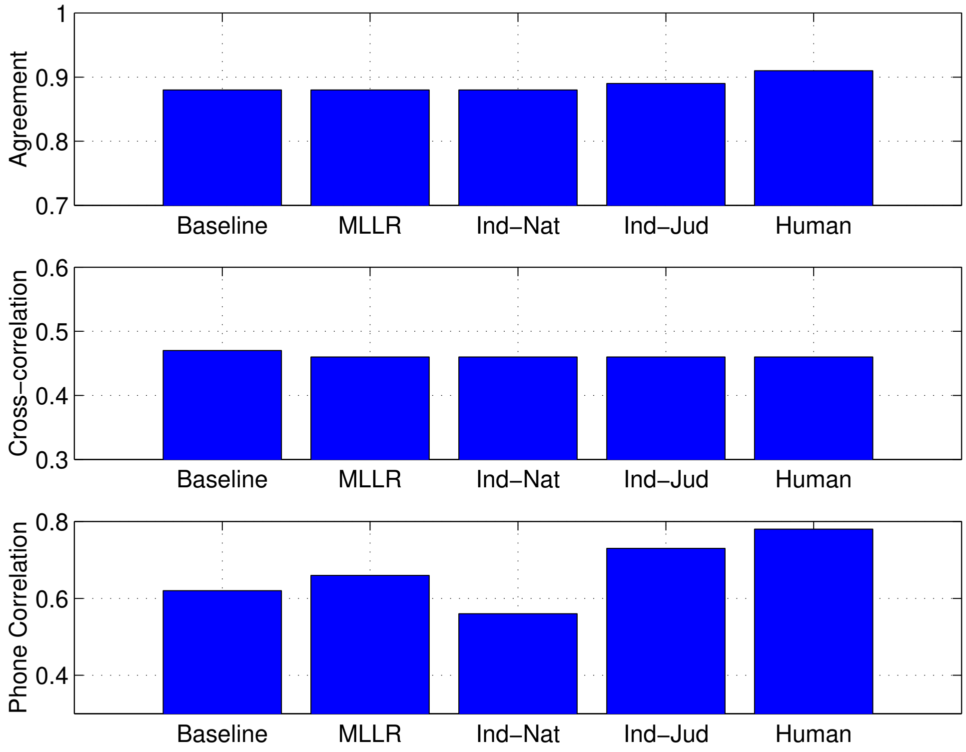

Figure 9.9: Comparison of the A, CC and PC performance measures using (a) the basic GOP scoring (Baseline), (b) basic *GOP* with adaptation (MLLR), (c) individual thresholds based on average native GOP scores (Ind-Nat), (d) individual thresholds based on human judge statistics (Ind-Jud), and (e) Human-human average performance (Human).

Speaker ID	A	CC	PC
Baseline	0.89	0.47	0.67
MLLR	0.89	0.46	0.71
MM adapt.	0.88	0.46	0.66
MLLR + Ind-Jud	0.88	0.49	0.77
MM + Ind-Jud	0.89	0.46	0.72
Human Mean	0.91	0.47	0.78

Table 9.5: Performance results of the individual speakers when using MM predicted models

for the two Japanese accented speakers. In order to enable a comparison, the results for the baseline and for GOP scoring with incorporated judge statistics (Ind-Jud) are given, too. MM adaptation adapts individually to each model as opposed to the global transformation applied by MLLR. Therefore, MM adaptation adapts to likely errors whereas MLLR only adapts to the global vocal characteristics. This effect causes the worse performance of MM compared with MLLR.

Finally, Table 9.6 summarises the effects of incorporating error modeling into the GOP algorithm. The British English hidden Markov models were augmented by a set of similar Spanish HMMs trained on a database of Latin-American Spanish. The data from the three Spanish speakers in the database is analysed using the extended GOP_2 metric with the scale factor K adjusted to give optimal performance. Instead of averaging over all eight speakers, the results shown in this table are only averaged over the three Spanish speakers in the database. The averaged baseline performance of these speakers is shown in the first line of Table 9.6. Comparison of these results with those for the metric GOP_2 demonstrate that a slight improvement in CC and PC can be obtain through including extra information for the detection of systematic mispronunciations. Finally, the results of combining all proposed refinements of the baseline algorithm, i.e. global MLLR adaptation, judge-based individual thresholds and error modeling, are as high as the human benchmark values. This leads to the conclusion that GOP scoring can approach the reliability of human scoring. These results correspond to the findeings by other researchers, see also, [24,38].

This section contains one more experiment. A recognition pass using the error networks can be regarded as an automatic detector of systematic mispronunciations. Table 9.7 compares human rejections with all automatically detected systematic mispronunciations, i.e. all rejections at phone segments where a native phone had been more likely that the target phone. The relatively high values for CC and PC indicate

Experimental Setup	A	CC	PC
Baseline	0.89	0.46	0.71
Ind-Judge	0.89	0.48	0.76
Error modeling	0.88	0.48	0.72
Ind-Judge + Error Modeling	0.90	0.49	0.78
Human Mean	0.91	0.47	0.78

Table 9.6: Scoring performance with and without Error Modeling averaged over the three Spanish accented speakers (all experiments include MLLR adaptation).

Speaker ID	A	CC	PC	S_{ID}	δ_S
fl	0.90	0.34	0.42	0.18	0.01
pc	0.88	0.39	0.57	0.23	0.04
ts	0.89	0.27	0.40	0.19	0.02

Table 9.7: Performance results of the individual speakers when using an error network to detect systematic mispronunciations only.

that a large proportion of pronunciation errors are due to systematic mispronunciations and that a significant proportion of these can also be detected by the use of error networks. Additionally, the results of this experiment can be used to collect statistics about how often each phone was systematically mispronounced. These results also provide information about which phones of the source language are typically substituted, i.e. these results provide statistics of typical mispronunciations.

This metric provides information about which type of mispronunciations occurred and whether the pronunciation of a given phone sounds more Spanish than English. This information might be used in future work to provide additional feedback about error types in addition to detecting error locations within an utterance or a word.

9.6 GOP demonstration software

In order to demonstrated the ability of the GOP algorithm to localise pronunciation errors, a Java-based demonstration program has been designed. This software has functions to record and playback a student's speech, just as a traditional pronunciation teaching system. However, the new component of this system is that a GOP score is calculated for each phone in an utterance. Because a rejection threshold is

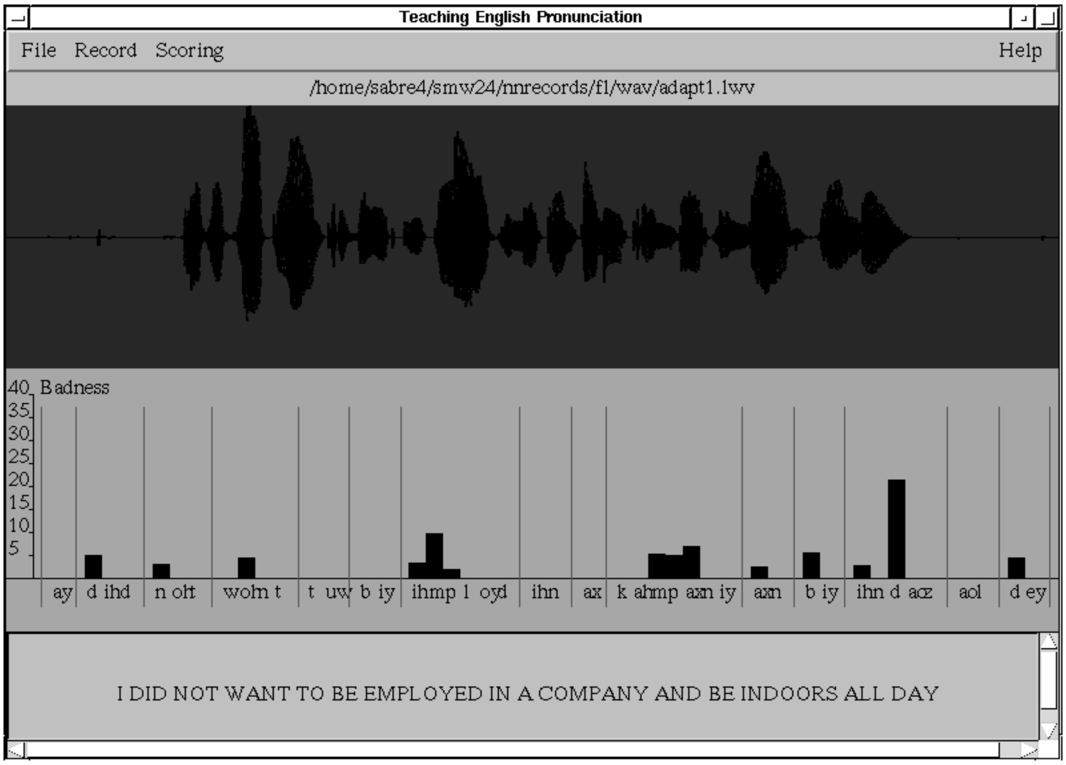

Figure 9.10: Typical window of the demonstration software. Below the waveform of the utterance the scores are given for each phone. The higher a bar, the worse the pronunciation.

already built in, all phones which have been rejected as incorrect are shown with a bar. After being provided with this localisation information, the student can listen to his own pronunciation repeatedly in order to obtain an understanding of his or her mistake(s).

The main window displays the utterance which the student is asked to read out, the actual waveform and the scores for each phone in the utterance together with the phonetic transcriptions. An example of the analysis of an utterance is shown in Figure 9.10. Initial user testing of the pronunciation of a few random native and non-native speakers showed that generally mispronunciations can be detected.

9.7 Summary

This chapter was dedicated to the development and verifications of techniques to detect and localise pronunciation errors on the phone level within an utterance. For this purpose, a group of algorithms was presented and evaluated on two sets of data.

Firstly, artificial data were used in order to determine which type of feature vector and which type of HMM are most suitable for the task at hand. Next, the GOP algorithm together with all modifications was tested using the heavily accented non-native speech of the database described in Chapter 5. The experimental results have shown that the automatic scoring techniques can produce pronunciation assessments approaching the reliability of human judges. A similar result has been found by Franco et al, [38]. Furthermore, incorporating knowledge about the source language in the form of alternative pronunciations helps with the task of locating pronunciation errors.

Chapter 10

Conclusions

This thesis began with a discussion of the current state of the art in the usage of speech recognition in CALL. In this thesis, a set of new algorithms have been derived and evaluated. These algorithms address two major challenges in CALL. The first is to improve acoustic modeling of heavily accented non-native speech either with the help of adaptation or off-line modeling techniques. The second is to automatically score and assess the pronunciation of individual phones in a student's speech.

This chapter will discuss and review the results of this research. Sections 10.1 and 10.2 reflect on the findings regarding acoustic modeling and pronunciation scoring, and Section 10.3 discusses the design criteria for a well-rounded pronunciation teaching system based on the insights gained from the research of this thesis. Finally, Section 10.4 concludes this thesis outlining possible directions of future research.

10.1 Acoustic modeling of non-native speech

The large decrease in recognition accuracy when recognising non-native speech, especially if the speech was produced by novice speakers, motivated the development of a family of new algorithms to improve recognition of non-native speech. The main advantage of this new bilingual approach to acoustic modeling lies in the fact that such adaptation and off-line model prediction do not require large amounts of non-native speech of a particular type of accent. The only requirements for these algorithms are a mapping between the source and the target language and a set of native, speaker-independent models of the source language.

These new algorithms fall into two groups: adaptation and prediction, but all of them share the idea that non-native speech can be modeled through a combination of models from both the target and the source language. This idea is based on the theory that non-native speech consists of sounds which are a mixture of the source and the target language of a non-native speaker. A technique was developed which automatically creates a complete list of substitution patterns in order to obtain a mapping. Once a mapping between a language pair has been calculated, it is independent of any specific task or model set.

The effectiveness of this bilingual modelling approach was proven in a comprehensive set of experiments. If adaptation material is available, Model Merging adaptation can decrease the baseline WER by 42% relative. Comparison with the standard adaptation technique MLLR demonstrated that Model Merging can also outperform MLLR by about 20% relative. On the other hand, the second adaptation algorithm, LMC, has proven to be less effective. Both accent adaptation techniques were tested on two different types of accent, Latin-American Spanish and Japanese, in order to demonstrate that these new algorithms yield performance improvements independent of the accent type.

The experimental evaluation of these new adaptation algorithms led to two observations. Firstly, even little adaptation data can yield significant performance improvements. Secondly, an increase in adaptation data does not create large additional improvements. Therefore, off-line model prediction methods which combine the models of the source and target language by *a-priori* estimating the combination parameters can be almost as effective as re-estimating these parameters using adaptation data. This approach enables us to improve the acoustic modeling of non-native speech without requiring adaptation sentences. Indirectly, however, non-native data are required, in order to build the mapping.

The first of the three off-line modeling algorithms, parallel bilingual modeling (PBM) creates a new non-native model by combining each target language model in parallel with its most likely substituted model of the source language. It is interesting to note that the performance gain of this technique is fairly independent of the value of the transition probability which determines how likely it is that the source or target model is chosen. The second method is based on the LMC adaptation technique, the only difference being that the elements of the diagonal combination matrix are estimated *a-priori* and not with the help of adaptation data. Unlike PBM, the performance of this technique is highly dependent on the fluency of each individual speaker. However, *a-priori* LMC can be used for measuring overall fluency of non-native speakers.

The last of the prediction techniques is predicted Model Merging. Similar to the case of adaptation, predicted MM proved to be most successful. Firstly, this method achieved a relative improvement of 27% over the baseline. Secondly, experiments with different values for the weighting between the two models showed that the overall performance gain is independent of this *a-priori* combination parameter, as long it is within a given interval. This fact combined with the low computational cost of Model Merging renders it a very powerful technique for applications using non-native speech recognition.

10.2 Pronunciation scoring on phone level

The overview of the state-of-the-art in pronunciation teaching in Chapter 2 noted the fact that even though several sentence and word level pronunciation scoring techniques have been developed, there are very few techniques to automatically assess pronunciation on a phone level. Therefore, this work focused on the development of an algorithm, called *Goodness of Pronunciation(GOP)*, which analyses a whole utterance and then returns scores for each phone in this utterance. In conjunction with an acceptance/rejection threshold, this method is well suited to localising the pronunciation errors within an utterance.

Before being able to assess this automatic scoring technique, it was necessary to develop measures which can compare phone-by-phone judgments of an utterance. Four performance measures were developed which incorporate different aspects of the characteristics of pronunciation assessment. These measures were used to analyse how human judges rated non-native speech on a phone-level. The averaged judgment similarity among several human judges served as the baseline against which to measure the GOP performance.

Several enhancements to the basic GOP algorithm have been introduced. These include individual thresholds for each phone as well as the use of error networks. These networks compare the recognition likelihood of the phones which should be recognised with the recognition likelihood of phones which are typical mispronunciations. The combination of the baseline algorithm and all the enhancements has been shown to localise pronunciation errors as reliably as human judges.

10.3 Design criteria for an overall pronunciation teaching system

The techniques of pronunciation scoring and non-native speech recognition as developed in this thesis are designed to be embedded within an overall computer-assisted language teaching system.

A CALL system should cover all four dimensions of language, that is reading, writing, listening and speaking. Regarding the speaking skill, this work provides the pronunciation scoring method to identify regions of bad pronunciations. Once an pronunciation error has been identified, error feedback could be provided through multiple sources of textual, audio, and/or visual materials. For example, in [62] it has been shown that learners tend to utilise the information of a visual signal about specific segments and timing characteristics of speech, but it was also shown that learners primarily rely on the acoustic signal. Examples of visualisation are explained in [7, 6, 27], where tools have been proposed for the visualisations of intonation and prosody.

In all situations where oral interaction is desired, it will be necessary to implement some form of non-native adaptation in order to obtain a level of recognition performance which is capable of sustaining an interactive dialogue. Such recognition tasks will need to be very constrained in terms of the language model used — i.e. such a task should use a limited vocabulary and simplified grammar. Additionally, the implementation of keyword spotting techniques could be helpful to test whether crucial words within a dialogue were spoken. Acoustic modeling of non-native speech could be improved by the model combination techniques developed in this thesis.

Another important aspect of a CALL system is to provide means of monitoring the progress of a user. In [92] a system is described which maintains an evolving model of what the user knows in order to direct the course of further interaction. This approach is based on applying dialogue modeling techniques in combination with speech recognition. A similar approach can be applied to a system using the GOP scoring technique, where the scores from each session are recorded and evaluated in order to determine which study material to present next to the user according to his progress.

At the time when the research described in this thesis was undertaken, a group of university laboratories and publishing houses started work on a project called "**I**nteractive **S**poken **L**anguage **E**ducation" , [11]. This project aims at introducing speech recognition technology into new CALL products for adult learners of English. One of the main goals of this project is to provide appropriate levels of specific feedback in order to guide the student to improve her pronunciation. Such feedback requires localisation and diagnosis of pronunciation errors on a phone level. This project encounters exactly those challenges for which the algorithms presented in this thesis have been derived. Therefore, this project provides a good example how this research could be incorporated within a comprehensive language learning system that addresses all components of oral instruction as discussed in the section above.

The ISLE systems consists of a language tutoring system with an ISLE monitor which mediates the communication between the tutoring system and the four modules concerned with pronunciation training. These four modules are recognition, error localisation, pronunciation diagnosis and word-stress diagnosis. The work of this thesis could be incorporated into the first three of these modules. For the recognition module, the Model Merging adaptation algorithm could be used. The GOP technique can be used for error localisation. The error networks of Section 9.3.4 could help with error diagnosis, and in fact, such networks are already being incorporated into the ISLE system.

10.4 Future work

The solutions to two major challenges of using automatic speech recognition in computer-assisted language learning systems which have been developed in this thesis, can only be regarded as the beginning of further research. This section will briefly outline possible directions of future work.

Further development of accent modeling techniques

Further improvements in acoustic modeling of non-native speech could be in the direction of modeling speech disfluencies as well as durational and intonational differences between native speech and heavily accented speech by beginners.

Also, it would be worth investigating how the model combination and model prediction techniques perform when applied to triphone models. Another idea would be to determine whether these model combination approaches for foreign-accented speech can be applied to other tasks, such as accent identification or dialogue modeling. For instance, current accent identification systems usually train several model sets with data from different accents, see also [46]. Instead of training several accent-specific HMM model sets, model sets based on accent prediction could be used. By doing so, the need for accent-specific recordings would be eliminated.

Another area of research concerns experiments testing whether accented speech by speakers of a source language can be better recognised by a model set which combines the target language with a closely related source language. For example, it would be interesting to see, if Danish accented speech would have a higher recognition rate when using a Swedish-English combined model set instead of an English set alone. Such grouping of languages might be a means to reduce the number of mappings and source model sets required in order to cover a large range of accents.

Further development of automatic pronunciation teaching

Like the acoustic modelling of non-native speech, the GOP scoring techniques could be refined too. In Section 9.2 different types of confidence scoring were described. Variations of these algorithms could be applied to phone pronunciation scoring as well. Moreover, the work presented here has predominantly been concerned with the localisation of pronunciation errors. Future work could investigate extensions of these techniques to error correction methods. For example, instead of just providing the answer that the /b/ in 'but' has been pronounced incorrectly, it would be desirable to have an algorithm which reliably returns the phone which has been produced instead. For this purpose, the error network developed for enhancing the GOP algorithm can be useful. In those instances where a source phone was more likely than a target phone, this information can be used for error feedback.

Finally, the acoustic modeling techniques could also be used for the development of various pronunciation fluency measures. For example, the optimal *a-priori* weight of the LMC algorithm can be interpreted as an indicator of fluency. Another possibility would be to use the percentage of mixture components of the target mixture which is retained in the re-estimated merged model set of the Model Merging algorithm as an indicator of fluency. This percentage can be found by summing all mixture component weights of those mixture components which belongs to the original target language Gaussian mixture. The lower this percentage, the lower the fluency of the subject will be.

The results of this thesis as well as the results from other groups showed significant progress during the past three years. Viable solutions have been proposed for almost every aspect of automatic pronunciation teaching, but the remaining challenge is to combine all these techniques into well-rounded CALL systems based on both technical and pedagogical findings.

Appendix A

Recording specifications for the Non-native Database

A.1 Technical specifications of recording equipment

A.1.1 The head mounted close-talking microphone

Name:	Sennheiser 414-6
Frequency Response:	50Hz-12kHz
Mode of Operation:	Pressure gradient transducer for close talking
Directional Characteristic:	super-cardioid
Rejection at 120 and 1000 Hz:	20dB-2dB
Impedance at 1000 Hz:	200Ω
Sensitivity:	$1\mu V/50mG \approx 1\mu V/5\mu T$

A.1.2 The Head-mounted Microphone Pre-amplifier: Symetric SX202 Dual Mic Preamp

Frequency Response:	20 Hz - 20 kHz, +0dB, -1dB
SNR:	$95dB(-50dBV, 150\Omega)$
Max. Gain/ Min. Gain:	60/20 dB

The signal from the Sennheiser is fed into the Symetric and then into the analogue input. Average measured close-talking microphone SNR in speech after digitisation, see [41], was roughly 35-45 dB.

A.1.3 Silicon Graphics IRIS Indigo's stereo line-level analogue input

Nominal Input Impedance:	$5k\Omega$
Input Signal: Max. Amplitude:	10 Vpp
Minimum Level:	1 Vpp (for full-scale input)

A.1.4 Silicon Graphics IRIS Indigo's A/D converter

Resolution:	Stereo 16-bit
Modulation:	delta-sigma
Sampling Rate Used:	16 kHz

Over sampling: 64 kHz

Official SNR at 48 kHz: ¿ 80dB (20 Hz - 20 kHz)

Recordings were made directly onto the IRIS Indigo's hard disk.

A.2 Data structure

For each subject a directory was created with the following structure:

```
                  id
      /         |         \       \
   wav        mfc        lab     cor
```

Generally the initials of each speaker were used to identify the data from one person, here denoted by "id". The directory "wav" contains all waveform files, which are stored in the HTK waveform format, see [101]. Each recorded sentence is stored in a separate waveform. The naming structure is such that the first 40 adaptation sentence are called "adapt1.lwv" to adapt40.lwv, the remaining sentences "s1.lwv" to s80.lwv". The file names are the same for each speaker, because the data for each speaker is stored in separate directories. The same naming structure is used for the encoded files in directory "mfc" using the suffix ".mfc", for the corrected transcriptions in directory "lab", using suffix ".lab". Finally the comment files are stored in the directory "cor" using the suffix ".cor".

The encoded data in directory "mfc" are encoded as the Mel-frequency cepstral coefficients with the following script being responsible for the encoding:

```
HCopy -A -C ~/nncorpus/sabre/gop/wav2mfc.cfg -S idwav2mfc.scp
```

with the following parameters set in the config file `wav2mfc.cfg`

```
SOURCEKIND     = WAVEFORM
SOURCEFORMAT   = HTK
SOURCERATE     = 625

ZMEANSOURCE    = FALSE
TARGETKIND     = MFCC_E_D_A_Z
TARGETFORMAT   = HTK
TARGETRATE     = 100000

SAVECOMPRESSED = TRUE

WINDOWSIZE     = 250000.0
NUMCHANS       = 24
```

The “lab” directory contains the files with the transcriptions as they were corrected by phoneticians. The “cor” directory contains word and sentence level score and also additional comments. Furthermore, each “id”- directory contains the following files:

- id.scp : scriptfile of all waveform files
- id.mlf : word level mlf file
- idmono.mlf : phoneme level mlf file
- id.ptx : prompting file for this speaker
- idlabs.mlf : phonetic transcriptions according to BEEP dictionary
- id.txt : ascii file with basic information about speaker

An example for a “id.txt” file is given below:

```
 SJK:
- from Korea
- 34 years
- male
- English for 1 year
- intermediate to advanced
```

The word level mlf file was create from id.ptx with script: ptx2mlf.pl. From the word level mlf the phoneme level mlf was created by:

```
HLEd -l '*' -i idmono.mlf -d new.dct ~/dep/lib/labs/mono.hled id.mlf
```

A.3 Examples of how the BEEP Phone set is used

BEEP	Examples	BEEP	Examples
aa	After, bArs, chAnce, lAUgh	l	weLL, whiLe
ae	mAn, sAck, sAt, tAxi	m	aMong, caMe, criMe
ah	tOUch, Under, amOng, anOther	n	corNer, crowN, daNger
ao	bOred, cOUrse, dOOr, fOUr	ng	cuttiNG, fiNGer
aw	fOUnd, hOUse, nOW, OUr	oh	gOt, jOb, lOng
ax	ovEr, pErfect, pOlice, quiEt	ow	mOst, nO, Only
ay	quIet, sAY, sIde, wAY	oy	pOInted, rOYal, vOIce
b	aBout, Bad, Back, elBow	p	accePt, comPany, couPle
ch	inCH, muCH, quesTIOn, temperaTure	r	diveR, eveRy, faR
d	tiDe, toDay, tolD, Day	s	faSt, faCe, handSome
dh	furTHer, oTHer, THat	sh	inspecTion, oCean, selfiSH
ea	thERE, awARE, cAREful	t	fronT, sTared, shouTing
eh	clEver, dEAdly, Elbow	th	teeTH, THanks
er	EArn, fIrst, fURther, gIRl	ua	you're
ey	gAve, grEAt, lAter, holidAY	uh	carefUl, cOUld, gOOd
f	leFt, oFF, saFer	uw	grEW, intO, jEWel
g	toGether, younGer, aGain	v	I'Ve, leaVe, moVe
hh	beHind, Had, Hard	w	Once, QUarter, Waited
ia	EAr, fEAr	y	Year, alIen, bEautiful
ih	fIsh, fIt, gettIng	z	buSiness, carS, cloSe
iy	friendlY, hE, lEAve	zh	expoSure
jh	Just, stranGE, oriGinal		
k	taKe, thanK, tooK		

Table A.1: Examples of the BEEP Phone Set

A.4 Usage of the assessment software

Before starting to work, change to the directory of the subject you are going to label. Then start the program with : `../Assess &`. Once the program is started you have to load the subject data by clicking on **Load** and then typing the subject ID in the line where it says: **Load filelist:**.

Then, the first waveform can be seen together with the word and phone level transcriptions. The **Help** button contains some help regarding how to do the pronunciation judgments. Generally, for each sentence you click the judgment button, score the sentence and then the word level. Following this, you should modify the phonetic transcription.

In order to score the whole sentence, listen to the sentence with **play**, then click the **Sent** button in the **Judgment** panel and choose that score which you think best describes the current sentence.

To score each word of a sentence, click the **Word** button in the **Judgment** panel. Next, click the word you want to judge, use **play** to listen to it and use **score** for scoring.

The main task is correcting the given phonetic transcription based on Standard Southern British English so that they reflect what actually has been spoken by the language student. The following options exist to modify the transcriptions:

1. **Subs**: Substitutes a given phoneme with the correct one. First click on the phone you want to change, then on **Subs** in the **Phoneme** menu. Then you get asked to type in the correct phone.

2. **Bad**: Sometimes a phoneme might sound wrong but there is no phoneme which would describe the sound correctly. Then you can mark it as **bad**.

3. **Del**: For a deletion, click on the respective phoneme and then on **Del**.

4. **Ins**: For an insertion click on the phoneme, after which you want to insert something, then click **Ins** and type in the phone you want to insert. It can happen that the student repeated a word where he or she was unsure about the pronunciation, you would have to insert all these phonemes and add a comment that there was a repetition in the sentence.

The last judgment option is **Comment** which allows you to note down anything special about the sentence, i.e. a repetition or some odd ways of pronouncing a sounds which doesn't really fit in the given phoneme set.

The remaining buttons in the main window, i.e. **Z.in**, **Z.out**, **Mark**, **Unmark**,

Play, −¿, ¡− allow you to mark, zoom in and zoom out different parts of the waveform, depending on whichever part you are transcription at the moment.

A.5 Some additional comments:

- I will assign each one who is doing the labelling a set of student data to label. Each Data set is marked by an two-letter id, the initials of the foreign language student. The data are organised in one directory for each subject, named with the respective ID. Each of these directories contains subdirectories called *wav* (this one contains the waveforms), *cor* (this one contains the sentence scores, the word scores and the comments for each sentence) and *lab*, which contains the corrected transcriptions, again one file per sentence. If you want to see the modifications you have done so far, you can have a look at these files, the program itself does not allow you to see any of your changes.
- You can use any other phone names to describe foreign sounds for the corrections but I will need a list describing them.
- The scoring scale on the word and sentence level is such that “good” represents almost native like speech, i.e. only slight accent, whereas “Very Poor” denotes close to unintelligible speech.
- In order to evaluate these judgments, there will be 20 calibration sentences which each of you would have to judge.
- You can use the comment section to note down if there were any stops or repetitions in the sentence or for added or missed words of the desired transcription.
- The word boundaries are based on alignments with British English models. Since that performs really bad for non-native speech, it happens that you don’t hear the word you actually want to listen to in the Word Menu. You can get around this by using **Mark** in the appropriate part of the waveform.
- You have to be in the speaker’s directory whose data you are to analyse when starting up the program, which you could do by typing something like : `../Assess &` .
- If you start a session, but want to finish before doing all sentences, just quit. When you restart use **Next** or **Jump** to get to the sentence where you left off. When using **Jump** it asks you for the sentence name where you what to

start. These names you can see in the top line of the program window. For sentence “adapt1.lab” you would have to type “adapt1”, etc.

- The window does not refresh automatically, i.e. if you minimise it or have another window on top, you get only white space back. If you go to the next sentence it refreshes.

A.6 Questionnaire about subjects

Reader's Questionnaire

We would be grateful if you could complete the questions below. None of the information will be distributed except information about your sex, age and mother tongue. Thank you for volunteering !

ID (for experimenter's use only): ....

Please print

Age ....

Sex ....

Do you have any speech or hearing impairment? ....

What is your mother tongue? ....

For how long have you been studying English? ....

....

Where did you live between 4-16? ....

Do you speak other languages and what level of fluency? ....

....

A.7 Recording instructions

Dear Speaker,

We thank you for taking part in our project. Your recorded speech will be used for non-native speech recognition research. Everything will be explained to you, but if you do get lost, read this or contact the administrator.

You are going to talk into the microphone attached to the headphone that you wear on your head.

When you start, a box will appear on the screen that will show the text to be read. These texts have been extracted from simplified stories for students of English as a second language.

First read the sentence on the screen in silence. Then to start recording that sentence, you move the cursor to the RECORD square and press the left mouse button. The square will now say STOP and time will start running. Now repeat the sentence aloud into the microphone. If you press the left mouse button again with the cursor on the STOP, the recording will stop. Don't click STOP before you finish the sentence! You can now click PLAY on the screen to hear your recording of the sentence. If everything is all right, go on to the next sentence by clicking CONTINUE. However, if you said a different word than in the text on the screen, then click RECORD and the recording starts again. After a session is over you can click on the QUIT button that will come up after the last sentence and find the administrator.

Your recording will consist of three parts. First you can practise on three sentences, which will not be stored. Then there will be a story about a diver for about 40 sentences and after that another story for about 80 sentences (at the top of the recording box you will see a number saying how many of the total number of sentences you have recorded so far).

The basic things that you have to think about are to try to speak as naturally and clearly as possible and to speak at normal loudness. Don't worry if you mispronounce a word, we expect and want this to happen.

If you have any questions now or during the recordings, please do not hesitate to ask the administrator what to do. Thank you and Good Luck!

Silke Witt (Test Administrator)

Bibliography

[1] G. Abbott, editor. *The teaching of English as an international language*, chapter 2:Pronunciation — perception and production, pages 37–56. Collins, 1981.

[2] D. Abercrombie. Teaching pronunciation. In A. Brown, editor, *Teaching English Pronunciation*, pages 87–95. Routledge, London, 1991.

[3] V. Abrash, A. Sankar, Franco H., and M. Cohen. Acoustic adaptation using non-linear transformations of HMM parameters. In *Proc. ICASSP*, pages 729–732, 1995.

[4] S.M. Ahadi and P.C. Woodland. Combined Bayesian and predictive techniques for rapid speaker adaptation of continuous density hidden Markov models. *Computer Speech and Language*, 1997.

[5] R. Akahane-Yamada, E. McDermott, T. Adachi, H. Kawahara, and J. S. Pruitt. Computer-based second language production training by using spectrographic representation and HMM-based speech recognition scores. In *Proceedings ICSLP*, Sydney, Australia, 1998.

[6] A. Alvarez, P. Martinez, P. Gomez, and J.L. Dominguez. A signal processing technique for speech visualisation. In *STiLL Proceedings*, pages 33–36. ESCA Workshop, 1998.

[7] A. Alvarez, R. Martinez, V. Nieto, V. Rodellar, and P. Gomez. Continuous formant tracking applied to visual representations of the speech and speech recognition. In *Proceedings Eurospeech*, Rhodes, Greece, 1997.

[8] S. Anderson and D. Kewley-Port. Evaluation of speech recognizers for speech training applications. *IEEE Trans. Speech and Audio Processing*, 3(4):229–241, July 1995.

[9] L. Arslan and J.H.L. Hansen. Frequency characteristics of foreign accented speech. In *ICASSP '97*, Munich, Germany, April 1997.

[10] L.M. Arslan. *Foreign Accent classification in American English.* PhD thesis, Dept. of Elec. and Comp. Engineering, Duke University, USA, 1996.

[11] E. Atwell. ISLE interactive spoken language education: Pronunciation training: requirements and solutions. Technical Report Project:LE4-8353, European Union, 1999.

[12] J. Bellegarda, P.V. de Souza, A. Nadas, D. Nahamoo, M.A. Picheny, and L.R. Bahl. The metamorphic algorithm: a speaker mapping approach to data augmentation. *IEEE Trans. on Speech and Audio Processing*, 2(3):413–420, July 1994.

[13] H. Bratt, L. Neumeyer, E. Shriberg, and H. Franco. Collection and detailed transcription of a speech database for development of language learning technology. In *Proceedings ICSLP*, Sydney, Australia, 1998.

[14] W. Byrne, E. Knodt, S. Khudanpur, and J. Bernstein. Is automatic speech recognition ready for non-native speech? a data-collection effort and initial experiments in modeling conversational Hispanic English. In *Proceedings STiLL*, pages 37–40, Marholmen, Sweden, 1998.

[15] J. Caminero, L. Hernandez, C. de la Torre, and C. Martin. Improving utterance verification using hierarchical confidence measures in continuous natural numbers recognition. In *Proc. ICASSP*, pages 891–894, 1997.

[16] R. Chandler. *Farewell, my lovely.* Penguin Readers Series. Penguin Book Ltd, 1991.

[17] C.A. Chapelle. Multimedia CALL: Lessons to be learned from research on instructed SLA. *Language Learning and Technology*, 2(1):22–34, 1998.

[18] L.L. Chase. *Error-responsive Feedback Mechanisms for Speech recognisers.* PhD thesis, Carnegie Mellon University, Pittsburgh,USA, 1997.

[19] H.C. Choi and R.W. King. On the use of spectral transformation for speaker adaptation in HMM based isolated-word speech recognition. *Speech Communication*, (17):131–143, 1995.

[20] D. Colton, M. Fanty, and R. Cole. Utterance verification improves closed-set recognition and out-of-vocabulary rejection. In *Proc. Eurospeech*, Madrid, Spain, 1995.

[21] S. Cox. Predictive speaker adaptation in speech recognition. *Computer Speech and Language*, 9:1–17, 1995.

[22] S. Cox and R. Rose. Confidence measures for the switchboard database. In *Proc. ICASSP*, 1996.

[23] C. Cucchiarini, F. De Wet, H. Strik, and L. Boves. Assessment of dutch pronunciation by means of automatic speech recognition technology. In *Proceedings ICSLP*, Sydney, Australia, 1998.

[24] C. Cucchiarini, H. Strik, and Lou Boves. Automatic pronunciation grading for dutch. In *Proceedings STiLL*, pages 95–99. ESCA Workshop, 1998.

[25] J. Dalby, D. Kewley-Port, and R. Sillings. Language-specific pronunciation training using the HearSay system. In *Proceedings STiLL*, pages 25–28, Marholmen, Sweden, 1998. ESCA Workshop.

[26] C. de la Torre, L. Hernandez-Gomez, F.J. Caminero-Gil, and C. Martin. Online garbage modeling for word and utterance verification in natural numbers recognition. In *Proc. ICASSP*, pages 845–848, 1996.

[27] R. Delmonte. Prosodic modeling for automatic language tutors. In *Proceedings STiLL*, pages 57–60, Marholmen, Sweden, 1998. ESCA Workshop.

[28] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from data via the em algorithm. *J. Roy. Stat. Soc.*, 39:1–38, 1977.

[29] V. Digalakis and L. Neumeyer. Fast speaker adaptation using constrained estimation of Gaussian mixtures. In *Proc. ARPA SLST workshop*, 1994.

[30] F. Ehsani. NTT-data Japanese-English ATC ASR system description. Technical report, Entropic, Inc., 1996.

[31] F. Ehsani, J. Bernstein, A. Najmi, and O. Todic. Subarashi:Japanese interactive spoken language education. In *Proceedings EUROSPEECH '97*, Rhodes, Greece, 1997.

[32] M. Eskenazi. Using automatic speech processing for foreign language pronunciation tutoring:some issues and a prototype. *Language Learning and Technology*, 2(2):62–76, Januar 1999.

[33] M.-W. Feng. Speaker adaptation based on spectral normalization and dynamic HMM parameter adaptation. In *Proc. ICASSP*, pages 704–707, 1995.

[34] A. Fine. *Madame Doubtfire*. Penguin Readers Series. Penguin Book Ltd, 1995.

[35] J.E. Flege. *Sound Patterns in Second Language Acquisition*, chapter 2: Effects of Equivalence Classification on the Production of Foreign Language Speech Sounds, pages 9–39. Foris Publications, 1987.

[36] J.E. Flege. Perception and production: the relevance of phonetic input to L2 phonological learning. In Ferguson, editor, *Crosscurrents in Second Language Acquisition*. Benjamins, Amsterdam, 1991.

[37] J.E. Flege. Production and perception of a novel, second-language phonetic contrast. *J. Acoust. Soc. Am.*, 93(3):1589–1608, March 1993.

[38] J.E. Flege and K.L. Fletcher. Talker and listener effects on degree of perceived foreign accent. *J. Acoust. Soc. Am.*, 91(1):370–389, January 1992.

[39] H. Franco and L. Neumeyer. Calibration of machine scores for pronunciation grading. In *Proceedings ICSLP*, Sydney, Australia, 1998.

[40] H. Franco, L. Neumeyer, Y. Kim, and Ronen O. Automatic pronunciation scoring for foreign instruction. In *ICASSP '97*, München, Germany, April 1997.

[41] J. Fransen, D. Pye, A.J. Robinson, P.C. Woodland, and S.J. Young. WSJCAM0 corpus and recording description. Technical Report CUED/F-INFENG/TR 192, Cambridge University Engineering Department, Cambridge, U.K., 1994.

[42] S. Furui. Cepstral analysis technique for automatic speaker verification. *IEEE Trans. on Speech and Audio Processing*, 29(2):254–272, 1982.

[43] J. L. Gauvain and C. H. Lee. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *IEEE Trans. on Speech and Audio Processing*, 2(2):291–298, April 1994.

[44] V. Gupta and P. Mermelstein. Effects of speaker accent on the performance of a speaker-independent, isolated-word recognizer. *Journal Acoust. Soc. Am.*, 71(6):1581–1587, June 1982.

[45] H. Hamada, S. Miki, and R. Nakatsu. Automatic evaluation of English pronunciation based on speech recognition techniques. *IEICE Trans. Inf. and Sys.*, E76-D(3):352–359, March 1993.

[46] J.H.L. Hansen and L.M. Arslan. Foreign accent classification using source generator based prosodic features. In *ICASSP*, pages 836–839, 1995.

[47] S. Hiller, E. Rooney, J. Laver, and M. Jack. SPELL: An automated system for computer-aided pronunciation teaching. *Speech Communication*, 13:463–473, 1993.

[48] J. Ishii and M. Tonomura. Speaker normalisation and adaptation based on linear transformations. In *Proc. ICASSP*, pages 1055–1058, 1997.

[49] C.-H. Jo, T. Kawahara, S. Doshita, and M. Dantsuji. Automatic pronunciation error detection and guidance for foreign language learning. In *Proceedings ICSLP*, Sydney, Australia, 1998.

[50] G. Kawai and K. Hirose. A call system using speech recognition to train the pronunciation of Japanese long vowels, the mora nasal and mora obstruent. In *Proceedings EUROSPEECH '97*, Rhodes, Greece, 1997.

[51] G. Kawai and K. Hirose. A CALL system using speech recognition to teach the pronunciation of Japanese tokushuhaku. In *Proceedings STiLL*, pages 73–76. ESCA Workshop, May 1998.

[52] G. Kawai and K. Hirose. A method for measuring the intelligibility and non-nativeness of phone quality in foreign language pronunciation training. In *Proceedings ICSLP*, Sydney, 1998. 241-244.

[53] J. Kenworthy. *Teaching English Pronunciation.* Longman, 1987.

[54] Y. Kim, H. Franco, and L. Neumeyer. Automatic pronunciation scoring of specific phone segments for language instruction. In *Proceedings EUROSPEECH '97*, Rhodes, Greece, 1997.

[55] J. Kingston, C. Bartels, J. Benki, D. Moore, J. Rise, R. Thorburn, and N. Macmillan. Learning non-native vowel-categories. In *ICSLP*, Philadelphia, 1996.

[56] K.M. Knill and S.J. Young. Speaker Dependent Keyword Spotting for Accessing Stored Speech. Technical Report CUED/F-INFENG/TR 193, Cambridge University Engineering Department, Cambridge, U.K., Oct 1994.

[57] L. Lee and R.C. Rose. Speaker normalisation using efficient frequency warping procedures. In *Proc. ICASSP*, pages 353–356, 1996.

[58] C. J. Leggetter and P. C. Woodland. Speaker adaptation of HMMs using linear regression. Technical Report CUED/F-INFENG/TR. 181, Cambridge University Engineering Department, Cambridge, U.K., June 1994.

[59] C.J. Leggetter. *Improved Acoustic Modelling for HMMs using Linear Transformations.* Phd thesis, Cambridge University, 1995.

[60] C.J. Leggetter and P.C. Woodland. Flexible speaker adaptation using maximum likelihood linear regression. In *Proc. ARPA SLST Workshop*, 1994.

[61] E. Lleida and R.C. Rose. Efficient decoding and training procedures for utterance verification in continuous speech recognition. In *Proc. ICASSP*, 1996.

[62] D.J. Markham and Y. Nagano-Madsen. Input modality effects in foreign accent. In *Proc. ICSLP*, pages 1473–1476, 1996.

[63] J. McDonough, W. Byrne, and X. Luo. Speaker normalisation with all-pass transforms. In *Proc. ICSLP*, 1998.

[64] H. Murakawa. PROTS (Pronunciation Training System). In *Proceedings ICSLP*, pages 641–643, 1990.

[65] H. Murakawa. Teaching pronunciation to Japanese university students: the voiceless fricative /s/ sound. In *Proceedings ICSLP*, pages 1181–1184, 1990.

[66] A. Nakamura. Restructuring Gaussian mixture density functions in speaker-independent acoustic models. In *Procs of ICASSP*, pages 649–652, 1998.

[67] S. Nakamura and K. Shikano. A comparative study of spectral mapping for speaker adaptation. In *Proc. ICASSP*, pages 157–160, 1990.

[68] L. Neumeyer, H. Franco, M. Weintraub, and P. Price. Automatic Text-independent Pronunciation Scoring of Foreign Language Student Speech. In *ICSLP '96*, Philadelphia, PA, USA, Oct 1996.

[69] L. Neumeyer, A. Sankar, and V. Digalakis. A comparative study of speaker adaptation techniques. In *Proc. EUROSPEECH*, pages 1127–1130, Madrid, September 1995.

[70] A.-M. Öster. Spoken L2 teaching with contrastive visual and auditory feedback. In *Proceedings ICSLP*, Sydney,Australia, 1998.

[71] M. Padmanabhan, L. R. Bahl, D. Nahamoo, and M.A. Picheny. Speaker clustering and transformation for speaker adaptation in large-vocabulary speech recognition systems. In *Proc. ICASSP*, pages 701–704, 1996.

[72] P. Price. How can speech technology replicate and complement skills of good language teachers in ways that help people to learn language? In *Proceedings STiLL*, pages 81–86. ESCA Workshop, 1998.

[73] D. Pye and P.C. Woodland. Experiments in speaker normalisation and adaptation for large vocabulary speech recognition. In *Proc. ICASSP*, 1997.

[74] L. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition.* Prentice Hall, 1993.

[75] Z. Rivlin. A confidence measure for acoustic likelihood scores. In *EUROSPEECH*, volume 1, pages 523–526, Madrid, Spain, 1995.

[76] Z. Rivlin, M. Cohen, V. Abrash, and T. Chung. A phone-dependent confidence measure for utterance rejection. In *Proc. ICASSP*, pages 515–518, 1995.

[77] C.L. Rogers and J.M. Dalby. Prediction of foreign-accented speech intelligibility form segmental contrast measures. *J. Acoust. Soc. Am.*, Vol. 100:no. 4, pt. 2, 1996.

[78] C.L. Rogers, J.M. Dalby, and G. DeVane. Intelligibility training for foreign-accented speech: A preliminary study. *J. Acoust. Soc. Am.*, Vol. 96:no. 5, pt. 2, 1994.

[79] R. C. Rose, B. H. Juang, and C. H. Lee. A training procedure for verifying string hypothesis in continuous speech recognition. In *Proc. ICASSP*, 1995.

[80] A. Sankar and C.H. Lee. Robust speech recognition based on stochastic matching. In *Proc. ICASSP*, 1995.

[81] M. Saraclar, H. Nock, and S. Khudanpur. Pronunciation modeling by sharing gaussian densities across phonetic models. In *Proc. EUROSPEECH*, Budapest, Hungary, Sept. 1999.

[82] T. Schaaf and T. Kemp. Confidence measures for spontaneous speech recognition. In *Proc. ICASSP*, pages 875–878, 1997.

[83] R. Schwartz, Y.-L. Chow, and F. Kubala. Rapid speaker adaptation using a probabilistic spectral mapping. In *Proc. ICASSP*, pages 633–636, 1987.

[84] B. Sevenster, G. de Krom, and G. Bloothooft. Evaluation and training of second-language learners' pronunciation using phoneme-based HMMs. In *Proceedings STiLL*, pages 91–94, Marholmen, Sweden, 1998. ESCA Workshop.

[85] A.R.M. Simoes. Assessing the contribution of instructional technology in the teaching of pronunciation. In *Proceedings ICSLP*, 1996.

[86] P. Strevens. A rationale for teaching pronunciation:the rival virtues of innocence and sophistication. In A. Brown, editor, *Teaching English Pronunciation*, pages 96–103. Routledge, London, 1991.

[87] K. Tajima, R. Port, and J. Dalby. Effects of temporal correction on intelligibility of foreign-accented English. *Journal of Phonetics*, 25, 1997.

[88] E.E. Tarone. Some influences on the syllable structure of interlanguage phonology. *International review of applied linguistics in Language Teaching*, 18:139–152, 1980.

[89] E. Thelen, X. Aubert, and P. Beyerlein. Speaker adaptation in the Philips system for large vocabulary continuous speech recognition. In *Proc. ICASSP*, pages 1035–1038, 1996.

[90] R.L. Trask. *A Dictionary of Phonetics and Phonology.* Routledge, 1996.

[91] U. Uebler, M. Schuessler, and H. Niemann. Bilingual and dialectical adaptation and retraining. In *Proc. Int. Conf. on Speech Language Proc.*, 1998.

[92] R.C. Waters. The audio interactive tutor. *Computer Assisted Language Learning*, 8:326–354, 1995.

[93] S.M. Witt and S.J. Young. computer-assisted pronunciation teaching based on automatic speech recognition. In *Language Teaching and Language Technology*, Groningen, The Netherlands, April 1997.

[94] S.M. Witt and S.J. Young. Computer-assisted pronunciation teaching based on automatic speech recognition. In *Proceedings of Language Teaching and Language Technology*, Groningen, Netherlands, 1997. Swet and Zeitlinger.

[95] S.M. Witt and S.J. Young. Language Learning based on Non-native Speech Recognition. In *Proceedings EUROSPEECH '97*, pages 633–636, Rhodes, Greece, 1997.

[96] S.M. Witt and S.J. Young. Bilingual model combination for non-native speech recognition. In *Proc. Institute of Acoustics Conference on Speech and Hearing*, 1998.

[97] S.M. Witt and S.J. Young. Estimation of models for non-native speech in computer-assisted language learning based on linear model combination. In *Proceedings ICSLP*, Sydney, Australia, 1998.

[98] S.M. Witt and S.J. Young. Performance measures for phone-level pronunciation teaching in call. In *STiLL:Speech Technology in Language Learning*, pages 99–102, Marholmen, Sweden, May 1998. ESCA.

[99] S.M. Witt and S.J. Young. Off-line acoustic modelling of non-native accents. In *Proc. EUROSPEECH*, Budapest, Hungary, 1999.

[100] M. Yoram and K. Hirose. Language training system utilizing speech modification. In *Proceedings ICSLP*, pages 1449–1452, 1996.

[101] S. J. Young, J. Odell, D. Ollason, and P. C. Woodland. *The HTK Book.* Entropic Cambridge Research Laboratory, 1996.

[102] S.R. Young. Recognition Confidence Measures: Detection of Misrecognitions and Out-of-vocabularyWords. Technical Report CMU-CS-94-157, Carnegie Mellon University, Pittsburgh, PA 15213, May 1994.

[103] G. Zavaliagkos, R. Schwartz, and J. Makhoul. Adaptation algorithms for bbn's phonetically tied mixture system. In *Proc. ARPA SLST workshop*, 1994.

[104] Y. Zhao. An acoustic-phonetic-based speaker adaptation technique for improving speaker-independent continuous speech recognition. *IEEE Trans. on Speech and Audio Processing*, 2(3):380–394, July 1994.