Transformation Sharing Strategies for MLLR Speaker Adaptation

Arindam Mandal

A dissertation submitted in partial fulfillment of
the requirements for the degree of

Doctor of Philosophy

University of Washington

2007

Program Authorized to Offer Degree: Electrical Engineering

University of Washington
Graduate School

This is to certify that I have examined this copy of a doctoral dissertation by

Arindam Mandal

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Chair of the Supervisory Committee:

_____

Mari Ostendorf

Reading Committee:

_____

Mari Ostendorf

_____

Andreas Stolcke

_____

Jeffrey Bilmes

Date: _____

In presenting this dissertation in partial fulfillment of the requirements for the doctoral degree at the University of Washington, I agree that the Library shall make its copies freely available for inspection. I further agree that extensive copying of this dissertation is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for copying or reproduction of this dissertation may be referred to Proquest Information and Learning, 300 North Zeeb Road, Ann Arbor, MI 48106-1346, 1-800-521-0600, to whom the author has granted "the right to reproduce and sell (a) copies of the manuscript in microform and/or (b) printed copies of the manuscript made from microform."

Signature_____

Date_____

University of Washington

Abstract

Transformation Sharing Strategies for MLLR Speaker Adaptation

Arindam Mandal

Chair of the Supervisory Committee:
Professor Mari Ostendorf
Electrical Engineering

Maximum Likelihood Linear Regression (MLLR) estimates linear transformations of automatic speech recognition (ASR) parameters and has achieved significant performance improvements in speaker-independent ASR systems by adapting to target speakers. Evidence is presented in this dissertation that the performance improvements are not consistent across target speakers, and 15% show degradation in performance levels, i.e. increase in word error rates (WER). Robustness of MLLR adaptation is an important problem and solutions to it are crucial for ASR systems that must adapt to a wide-range of speakers. This dissertation presents new research directions that address this problem, exploring two aspects of MLLR transformation sharing using a regression class tree (RCT): the design of RCTs and the online complexity control of adaptation.

The standard approach for MLLR transformation sharing uses a single speaker-independent RCT. A new approach is proposed that uses multiple RCTs, each trained using speaker-cluster-specific data and represents types of speaker variability, determined by an algorithm that partitions a large corpus of speakers in the eigenspace of their MLLR transformations. ASR experiments show that choosing the appropriate RCT for target speakers leads to significant reduction in WER. For unsupervised adaptation, an algorithm is proposed that linearly combines MLLR transformations from cluster-specific RCTs using weights estimated by maximizing the likelihood of adaptation data and achieves small improvements in WER for several tasks in English and Mandarin. More significantly, distributional anal-

ysis shows that it reduces the number of speakers with performance loss from adaptation across ranges of adaptation data and WER.

The standard approach for complexity control in MLLR uses only the amount of adaptation data from a target speaker. Evidence is presented that this does not produce the optimal number of regression classes and significant improvements in WER are achieved using the oracle number of regression classes. A new solution for complexity control is proposed that predicts the number of regression classes in an RCT using speaker-level features with standard statistical classifiers and achieves moderate improvements in WER. Next, a more flexible approach is proposed that performs node-level pruning in an RCT, using node-level features and produces improved robustness of MLLR adaptation.

TABLE OF CONTENTS

Page

LIST OF ALGORITHMS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

ASR: Automatic Speech Recognition

WER: Word Error Rate

MLLR: Maximum Likelihood Linear Regression

RCT: Regression Class Tree

HMM: Hidden Markov Model

EM: Expectation Maximization

ML: Maximum Likelihood

MAP: Maximum A Posteriori Probability

ACKNOWLEDGMENTS

I am very grateful to my dissertation advisor Mari Ostendorf for giving me the opportunity to come to University of Washington for a PhD. As an advisor she has taught me how to perform research and impacted my approach to solving research issues and problems in general. Her mentoring is the most significant aspect of my education at the University of Washington. I am thankful to her for always carefully reading and providing me detailed feedback on drafts of my publications and this dissertation.

I also thank Andreas Stolcke for hosting me at SRI International's STAR Lab for a whole academic year and also carefully reading my publication drafts and giving me useful feedback. I am grateful for all the help I have received from other members of STAR Lab including Ramana Gadde, Jing Zheng, Wen Wang and Dimitra Vergyri. I also thank Jeff Bilmes for providing useful comments and suggestions during my defense.

I am especially grateful to my parents for supporting me in my decision to go back to graduate school for a PhD, for being patient with me and encouraging me to finish.

I feel fortunate that during my time at SSLI Lab, I was in the company of several wonderful and brilliant colleagues. Given the length of time I have spent in the lab I have had the chance to interact with my colleagues about several things including research, politics and ordering dinner everyday. I would like to thank Costas Boulis, Xin Lei, Xiao Li, Scott Otterson, Karim Filali, Dustin Hillard, Chris Bartels, Kevin Duh, Jeremy Kahn, Takahiro Shinozaki, Andrei Alexandrescu, Amarnag Subramanya, Jon Malkin and Sarah Schwarm. I also thank the systems administration staff of SSLI Lab for all their help. I am grateful to Karim Filali for helping with the final submission of this dissertation.

I thank Prithwish, Naved and Ekta for being good friends, for checking up on me from time to time in the past two years, for philosophical conversations and also for accompanying me on hiking trips around the country.

DEDICATION

To my parents

Chapter 1

INTRODUCTION

Humans possess sophisticated adaptation capabilities for recognizing and understanding speech in a range of unfamiliar conditions with a high rate of success. These conditions, which introduce variability in the acoustic signal of speech, include speakers with foreign accents, different dialects, physiological conditions that change voice quality, e.g., age, gender or illness, environmental conditions that affect the speech signal, e.g., telephone and recording channels, room acoustics, etc. The development of algorithms that attempt to produce similar adaptation behavior has played a key role in the steady progress of automatic speech recognition (ASR) systems from the research laboratory to commercial applications in real-world situations. At present, ASR systems can be encountered in a wide range of applications such as automated telephone information systems, command and control systems embedded in consumer electronics, in-car navigation systems, speech-to-speech translation systems, automated dictation systems, etc. Though the research community has devoted considerable effort in developing adaptation algorithms for such ASR systems, their performance in real-world situations is yet to match that achieved by humans, except for the simplest applications. The focus of this dissertation is to investigate the limitations of current adaptation paradigms for ASR and address them by developing new algorithms that leverage speaker variability and improve robustness of adaptation and ASR system performance for a range of real-world tasks.

1.1 Speaker Adaptation in ASR

Modern ASR systems have two main components: the acoustic model and the language model [51, 88]. The research findings presented in this dissertation are focused on the acoustic model, which are usually based on hidden Markov models (HMM) that use mixtures

of Gaussian distributions as the state output distributions [4, 5, 88]. For large vocabulary ASR systems, acoustic models are typically trained using data collected from a large corpus of speakers. This approach in training helps model the acoustic variation observed in a large speaker population. Such ASR systems are called speaker-independent (SI) since they can recognize unseen speakers without undergoing further training. Speaker-dependent (SD) ASR systems, which are trained using data of a single speaker and thus use a better acoustic model of that speaker, tend to produce 2-3 times better system performance compared to SI systems, when both are trained using comparable amounts of data [48]. The lower performance levels of SI ASR systems is due to the fact that they have to cope with modeling variability among speakers, typically with no training data from the target speaker. The goal of speaker adaptation in SI ASR systems is to shift the SI acoustic model "close" to the true SD model of a speaker unseen in training, and produce improved ASR system performance.

The adaptation "shift" or transformation of the SI acoustic model is usually estimated using (adaptation) training data from the unseen speaker. *Adaptation strategy* refers to the algorithms used to estimate the adaptation transformation and the subsequent transformation of the SI acoustic model. *Adaptation complexity* refers to the number of adaptation transformation parameters that need to be estimated. A transcription of the adaptation data is needed to estimate the transformations. If the true transcriptions are available then the adaptation procedure is called *supervised*, or if the transcriptions are "guessed" using an initial hypothesis of the SI acoustic model, then it is called *unsupervised*. The adaptation procedure is referred to as *static*, if the transformations are estimated using all adaptation data at once, or as *online* (or *incremental*) if the transformations are refined as more adaptation data becomes available. In addition, adaptation algorithms can also be classified as *model-space* if they directly transform the parameters of the acoustic model, or *feature-space* if they transform the acoustic feature vectors only [92].

1.2 Challenges in Speaker Adaptation

The primary goal of any speaker adaptation algorithm is to achieve ASR system performance levels that are significantly better than those of SI ASR systems and comparable to

that those of SD ASR systems. There are several challenges that any speaker adaptation algorithm must handle, in order to achieve the primary goal. The constraints include sparsity of adaptation data, variability among speakers, robustness of the adaptation approach, and computational resources available for performing adaptation.

1.2.1 Adaptation Data Sparsity

The number of samples of adaptation data available, from an individual speaker, is typically several orders of magnitude less than that used for training an SI acoustic model. In addition, the number of adaptation parameters to transform (or adapt) in the SI acoustic model of large vocabulary ASR systems, is several orders of magnitude higher than the number of samples of adaptation data. This implies that there will be insufficient or no data available for many parameters in the acoustic model. To deliver improved performance, adaptation algorithms require strategies to cope with training data insufficiency such that there is a graceful back-off in adaptation of acoustic model parameters with sufficient data to those with none, and a mechanism for sharing adaptation transformations across acoustic model parameters.

1.2.2 Speaker Variability

As previously mentioned, there is variability in a large speaker population with respect to linguistic, demographic and physiological attributes. For example, a speaker who grew up and lives in the southern dialect region of United States will have different pronunciation patterns compared to someone from Manhattan, New York. Variability in speech also arises from demographic factors such as gender, age, years of education, and cultural upbringing. Physiological differences in the human speech production systems, such as the vocal tract, which are determined in part by gender also introduces variability in speech. We conjecture that, in the presence of sparse adaptation data, adaptation strategies that account for such speaker variability would produce better speaker-adapted (SA) models compared to those produced by global adaptation strategies.

1.2.3 Adaptation Robustness

ASR systems that handle real-world situations encounter speech under a wide range of conditions that include amount of adaptation data available, speaker and channel variability, varying ASR performance level with respect to the SI ASR system. Adaptation algorithms should not only be able to improve overall system performance, but also be robust to deliver such improvements across the entire range of adaptation conditions presented to the system. This is especially important for usability of systems that handle large speaker populations, for example, automated call center support systems, where callers may additionally use several different channels (cellular, land-line, hands-free, etc.) to interact with the system.

1.2.4 Computation Complexity

Adaptation algorithms should have low overheads both in terms of the time and memory needed to estimate adaptation transformations. Real-world ASR applications such as ASR on portable computing devices, ASR-based dictation systems for desktop computers, telephone-based ASR applications that handle a large volume of users, all have resource constraints with respect to both available computing capacity and memory. For adaptation algorithms to have an impact on real-world applications, they must achieve a trade-off between computational efficiency and ASR performance gains.

1.3 Speaker Adaptation Approaches

Significant research efforts have been applied to the problem of adaptation for ASR systems and in a little over a decade, several important advances have been reported. Some of these approaches are now widely accepted as standards and an excellent survey can be found in [116]. A brief description of the major speaker adaptation approaches are presented in this section.

1.3.1 Maximum Likelihood Linear Regression (MLLR)

MLLR is the most widely-used adaptation technique in modern ASR systems [25, 59]. It has been successfully used in improving ASR system performance across a wide range of

domains and tasks. The key to its successful application lies in its use of a SI adaptation strategy and an SA back-off strategy to handle moderate to sparse amounts of adaptation data. In MLLR, using an initial transcription of the adaptation data and a maximum likelihood (ML) estimation approach, a transformation matrix is estimated for the means and variances of the Gaussian distributions of the SI acoustic model. The SI adaptation strategy is designed by organizing the SI Gaussian distributions into a tree, referred to as the *regression class tree*, using an appropriate similarity measure between the distributions. Given adaptation data of a speaker, the SA back-off strategy is implemented by descending down to those nodes in the tree that satisfy a pre-determined amount of data. These nodes are called *regression classes*, and an adaptation transformation is estimated for each such node and shared among all SI Gaussian distributions in that node, irrespective of whether they are observed in the adaptation data. Since the SA back-off strategy determines the number of regression classes to use, and by extension the number of adaptation parameters to estimate, it also serves the role of determining adaptation complexity.

1.3.2 Speaker Clustering

Speaker clustering approaches for adaptation primarily target cases of sparse adaptation data. This class of algorithms usually has two steps: an offline speaker clustering step and an online adaptation step. In the speaker clustering step, a large group of speakers are used to train component models representative of clusters that group similar speakers. The component models can be derived using an adaptive training approach [34] or in the eigenspace of SD acoustic models [53]. In the adaptation step, weights (one for each component, or cluster-specific, model), are estimated by maximizing the likelihood of an individual speaker's adaptation data. Since a single weight is estimated for cluster-specific model, these algorithms are naturally suited to cases of sparse adaptation data.

1.3.3 MAP Family

HMM-based ASR systems are typically trained using a ML approach such that the parameter values λ are chosen to maximize the likelihood of the training data $p(O|\lambda)$. In maximum

a posteriori parameter estimation (MAP), the parameters λ are set at the maximum of the distribution $p(\lambda|O)$ or equivalently $p(O|\lambda)p_o(\lambda)$, where $p_o(\lambda)$ is the prior distribution of the parameters [38]. MAP estimation requires the definition of a prior distribution. It is convenient if the prior density is from the same family as the posterior distribution (the conjugate prior) if it exits. For mixture Gaussian HMMs such a conjugate prior of finite dimension does not exist and an alternative approach that is presented in [38] is used.

A key advantage of the MAP approach is that as the amount of training data increases towards infinity the MAP estimate converges to the ML estimate. Its main drawback is that only parameters that are observed in the adaptation data can be adapted. To update poorly adapted, or unadapted parameters of an SI system, linear regression relationships that model correlations between the parameters are used as described in regression based model prediction (RMP) [1]. Under this approach, a set of speaker dependent model sets are computed and for each Gaussian mean element in the system other mean values are found that are well correlated with its speaker-dependent changes. RMP first updates models using standard MAP, and then uses parameters that have received a reasonable amount of adaptation to generate parameter estimates for each unadapted or poorly adapted target value.

1.3.4 Other Adaptation Approaches

There also exists a class of techniques referred to as speaker normalization, which removes variability amongst speakers such that acoustic feature vectors are better suited to use with a speaker-independent model. This is different from speaker adaptation, where the goal is to estimate a better SD model. Speaker normalization techniques do not alter model parameters in any way, but only alter the observation acoustic feature vectors such that some speaker-dependent variabilities are removed. The simplest of these is the widely used acoustic feature vector mean normalization technique, which subtracts the long term mean from individual speakers. Another technique that is popular is vocal tract length normalization (VTLN), which rescales the frequency axis with the aim of accounting for differences in vocal tract length between speakers [27, 56].

1.4 Problems with Existing Approaches

The major speaker adaptation approaches are able to cope with some or all of the challenges of adaptation with varying degrees of success. However, several shortcomings exist both with the design and performance of current approaches, which are enumerated below.

- *Robustness in adaptation* is a major shortcoming of current adaptation approaches in achieving ASR performance improvements across a wide range of adaptation conditions. In particular, our experiments show that MLLR adaptation degrades ASR performance levels for 15-30% of speakers, across a wide range of of the amount of adaptation data available, or ASR performance levels achieved with the SI acoustic model, and across several domains of ASR tasks.

- *Adaptation strategies* designed by mainstream adaptation approaches, e.g., MLLR and MAP, do not adequately model speaker variability. For example, in MLLR, the adaptation strategy (adaptation transformation sharing scheme) is specified for all target speakers using a single speaker-independent regression class tree. This rules out achieving any additional ASR performance improvements by varying the adaptation strategy that may be more suitable for individual speakers or groups of "similar" speakers.

- *Adaptation complexity* is usually determined by the amount of adaptation data available for individual speakers. Our experiments show that this approach, in the case of MLLR, is often not adequate in achieving the best possible ASR performance improvements from adaptation. Investigation of additional sources of information that can serve as good predictors of adaptation complexity can lead to higher ASR performance gains from adaptation.

1.5 Contributions of Dissertation

The focus of this dissertation is in particular on MLLR speaker adaptation when it is used in unsupervised, static and model-space mode. The instances of MLLR adaptation that are

used for transforming the feature vectors directly, are utilized by the ASR systems built on in this dissertation, without modification. The main contributions of the research presented in the dissertation are summarized below.

1.5.1 Speaker Variability in Adaptation Strategies

We introduce a strategy to model speaker variability in the eigenspace of MLLR transformations by partitioning a large corpus of speakers and learning cluster-specific regression class trees. Our experiments show that choosing the best possible regression class tree for individual speakers lead to significant improvements in ASR performance gains from MLLR adaptation, across different ASR tasks. In addition, when the best cluster-specific regression class tree is used with MLLR, there is reduction in the variance of ASR performance levels and significant improvement in performance for speakers with the worst ASR performance levels. By studying the structure of the cluster-specific regression class trees, we hypothesize that the speaker clusters are representative of dialect (and possibly sociolect) patterns in large speaker populations.

1.5.2 Robust Combination of Adaptation Transformations

To realize the potential ASR performance gains from using cluster-specific regression trees for individual speakers, we developed a robust strategy to estimate weights for linearly combining the transformations from each cluster and produce a composite adaptation transformation. The weight estimation strategy is a two-step back-off strategy for determining weights that maximize the likelihood of adaptation data with or without inequality constraints. The application of this approach across a range of ASR tasks show small to moderate improvements in ASR performance levels. In addition, the transformation combination approach has significantly less computational overheads compared to previously reported cluster-specific model interpolation algorithms.

1.5.3 Adaptation Complexity Control

The most widely-used strategy for adaptation complexity control of individual speakers in MLLR is to use a threshold on the amount of adaptation data available to determine the number of regression classes (and adaptation transformations) to use. Experiments show that this approach is not optimal and significant improvements in ASR performance can be achieved by choosing the best possible number of regression classes for individual speakers. To take advantage of such potential ASR performance gains, speaker-level information sources were used to train standard statistical learners to predict the optimal number of regression classes. Modest, but significant, improvements are achieved on applying this approach to several ASR tasks. Next, a new and more flexible approach is proposed that performs node-level pruning in regression class trees, again using standard statistical classifiers but with regression class-dependent features. A procedure is presented to incorporate this complexity control mechanism into MLLR adaptation in ASR experiments, which produced improved robustness in ASR system performance for English broadcast news tasks.

1.5.4 Improved Robustness in Adaptation

An important finding of this dissertation is the improvement in robustness of MLLR adaptation, across several ASR tasks, when using both of our proposed approaches: using composite adaptation transformations estimated by the linear combination of cluster-specific transformations and node-level pruning of regression class trees for adaptation complexity control. In particular, when using the composite adaptation transformations, our experiments show that there is a significant reduction in average ASR performance loss from adaptation. Both approaches show a reduction in the percentage of speakers who have degraded ASR performance from adaptation.

1.5.5 Adaptation Correlates

Several information sources were developed, both at the speaker-level and at the node-level of regression class trees, that can be viewed as adaptation correlates. The categories of these features include rate of speech, diversity of phones spoken, amount of speech,

confidence in initial hypothesis of words spoken, and several others. We have investigated these features with the aim to ascertain their suitability for predicting ASR performance gains (or losses) from adaptation for individual speakers. In addition, we have used a combination of these information sources in predicting complexity control in regression class trees for MLLR adaptation. Rate of speech, entropy of phone durations and amount of adaptation, computed for nodes in the regression class trees were useful in predicting regression class tree structures that eventually resulted in improved robustness of MLLR adaptation.

1.6 Organization of Dissertation

The rest of this dissertation is organized as follows: Chapter 2 briefly reviews the fundamentals of ASR systems and provides a detailed foundation of MLLR based and adaptation and its use of regression class trees. Chapter 3 provides a review of previous research done in incorporating speaker variability information into speaker adaptation and reports results of a pilot study. Chapter 4 describes in detail the architecture and components of multiple ASR systems used in this dissertation. Chapter 5 describes a speaker variability modeling approach to design multiple, speaker clustered, regression class trees and the outcomes of ASR experiments that user them. Chapter 6 and 7 describes proposed research directions for complexity control of complexity of MLLR adaptation using higher-level sources of information, but predicting the complexity at varying levels of granularity in regression class trees. Finally, Chapter 8 concludes this dissertation by listing its main findings and possible future directions of research.

Chapter 2

SPEAKER ADAPTATION USING MLLR

This chapter reviews the background literature on MLLR-based speaker adaptation, but first provides a short introduction to ASR. A considerable amount of research results have been published on speaker adaptation for ASR and the reader is referred to them, when appropriate, for further details.

2.1 ASR Review

The standard approach in automatic speech recognition is to find the most likely sequence of words $\hat{w}$ given the acoustic signal x from all possible word sequences W (Eqn. 2.1). Using Bayes' rule the problem can be broken into two components, as shown in Eqn. 2.2, where $p(x|w)$ is the *acoustic model* and $p(w)$ is the *language model*. As mentioned in Chapter 1, the research presented in this dissertation focuses only on the acoustic model. Other prominent approaches to ASR include applications of neural networks as described in [10, 61].

$$\hat{w} = \arg\max_{w \in W} \; p(w|x) \tag{2.1}$$

$$= \arg\max_{w \in W} \; p(x|w)p(w) \tag{2.2}$$

2.1.1 Parameter Estimation Framework for HMMs

A typical approach to designing an acoustic model is to use hidden Markov models (HMMs) to model sub-word units, e.g., tri-phones, with mixture Gaussian distributions modeling the state output distributions. Each individual HMM (for a tri-phone) is usually configured to have 3-states with only left-to-right transitions permitted [88]. The most common solution to training the models are based on maximum likelihood (ML) estimation. A closed form solution for ML estimation of the parameters of the HMMs does not exist. The solution is to use an iterative approach and maximize an auxiliary function, as described by the Baum-

Welch algorithm, which is an instance of the Expectation Maximization (EM) algorithm [6, 23, 87]. The auxiliary function for HMMs can be expressed as

$$\begin{aligned}\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) &= E_{P(\theta|o)}\Big[\log P(\mathbf{O}, \Theta|\hat{\mathcal{M}})\Big|\mathbf{O}, \mathcal{M}\Big] \\ &= \sum_{\theta \in \Theta} P(\theta|\mathbf{O}, \mathcal{M}) \log P(\mathbf{O}, \theta|\hat{\mathcal{M}}) \end{aligned} \tag{2.3}$$

where, $\mathcal{M}$ is the current model, $\hat{\mathcal{M}}$ is the model being estimated; $\mathbf{O}$ is the entire observation sequence and Θ represents the set of all possible HMM state sequences θ. It can be shown that finding the $\hat{\mathcal{M}}$, which maximizes the auxiliary function guarantees an increase in the likelihood of the training data $\mathbf{O}$, unless it is already at a maximum.

Discriminatively trained HMMs that directly minimize a WER criterion are in principle guaranteed to produce superior performance compared to ML-estimated HMMs. In modern ASR systems parameters of HMMs are often estimated using a discriminative training procedure that maximizes a mutual information criterion [41, 80], or a minimum phone error (MPE)-based criterion [84, 103]. In this dissertation, research results are presented on adaptation of both ML-estimated and discriminatively-estimated HMMs. MLLR adaptation is a maximum likelihood approach, but extensions based on discriminative training have been developed. Since the discriminative extensions are more expensive and have not led to significant gains over MLLR, they are not widely used and this work will be within the standard MLLR framework [43].

2.1.2 Decoding the State Sequence

Given an observation sequence $\mathbf{O}$, the most likely hidden sequence $\hat{\theta}$ is "decoded" by,

$$\hat{\theta} = \arg\max_{\theta \in \Theta} \; p(\theta|\mathbf{O}) \tag{2.4}$$

using the Viterbi algorithm as described in [87, 108]. A main advantage of this algorithm is that it can be easily extended to continuous speech and also allows disjoint models within the same model to be considered separately, which is equivalent to the case of having multiple pronunciations. For computational efficiency, a pruning algorithm is often used,

that considers only those paths that are above a certain threshold.

2.2 Maximum Likelihood Linear Regression

MLLR-based speaker adaptation belongs to the linear transformation family of adaptation algorithms [25,59,77,79]. Adaptation is performed by linearly transforming of the SI means and variances of Gaussian distributions of the acoustic model. The approach is reviewed here as presented in [25,59]. For example, the adapted Gaussian mean $\hat{\mu}_m$ can be represented as,

$$\hat{\mu}_m = \mathbf{W}_m \xi_m \tag{2.5}$$

where $\mathbf{W}_m$ is an $n \times (n+1)$ transformation matrix and ξ_m is the extended mean vector,

$$\xi_m = [1 \ \mu_m]^T = [1 \ \mu_{m_1} \ \mu_{m_n}]^T$$

2.2.1 Mean transformation

The linear transformation matrix for adaptation of Gaussian mean is estimated from a speaker's acoustic adaptation data using an ML approach and an initial transcription of the adaptation data. Again, the solution is iterative since the state sequence is hidden. The SI Gaussian distributions are grouped into R regression classes for the purpose of sharing adaptation transformation $\mathbf{W}_r$ among them. Considering only the terms that involve the mixture Gaussian distributions, the auxiliary function of Eqn. 2.3, can be written as

$$\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = K - \frac{1}{2} \sum_{r=1}^{R} \sum_{c=1}^{C_r} \sum_{m=1}^{M_c} \sum_{\tau=1}^{T} \gamma_m(\tau) \quad \big(\mathbf{o}(\tau) - \mathbf{W}_r \xi_m\big)^T \Sigma_m^{-1} \big(\mathbf{o}(\tau) - \mathbf{W}_r \xi_m\big) \tag{2.6}$$

where, K is the normalization constant; C_r is the number of mixture Gaussian distributions in each regression class r, and each mixture Gaussian distribution c has M_c component Gaussian distributions; $\mathbf{o}(\tau)$ is the observation vector at time τ and $\gamma_m(\tau)$, $\hat{\mu}_m$ and Σ_m^{-1} are the occupation probability at time τ, mean vector and inverse covariance of the of the mth Gaussian distribution.

Differentiating Eqn. 2.6, and equating it to 0, the following expression is obtained,

$$\sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\Sigma_m^{-1}\mathbf{o}(\tau)\xi_m^T = \sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\Sigma_m^{-1}\mathbf{W}_r\xi_m\xi_m^T$$
$$= \sum_{r=1}^{R}\mathbf{V}^{(r)}\mathbf{W}_r\mathbf{D}^{(r)} \tag{2.7}$$

where $\mathbf{V}^{(r)}$ is the state distribution inverse covariance matrix scaled by the state occupation probability,

$$\mathbf{V}^{(r)} = \sum_{t=1}^{T}\gamma_m(\tau)\Sigma_m^{-1} \tag{2.8}$$

and $\mathbf{D}^{(r)}$ is the outer product of the extended Gaussian mean vectors,

$$\mathbf{D}^{(r)} = \xi_m\xi_m^T \tag{2.9}$$

For the case when the HMM state Gaussian distributions are modeled by a diagonal covariance matrix, a closed form solution for $\mathbf{W}_r$ is obtained in the maximization step of the EM algorithm by solving a set of simultaneous equations, one for each row of $\mathbf{W}_r$ [59],

$$\mathbf{w}_i = \mathbf{G}^{(i)-1}z_i^T \tag{2.10}$$

where $\mathbf{w}_i$ and $\mathbf{z}_i$ are the i^{th} rows of $\mathbf{W}_r$ and $\mathbf{Z}$ respectively. $\mathbf{Z}$ is an $n \times (n+1)$ matrix whose elements are given by,

$$z_{ij} = \sum_{q=1}^{n+1} w_{iq}g_{jq}^{(i)} \tag{2.11}$$

and the elements of $\mathbf{G}^{(i)}$ is given by,

$$g_{jq}^{(i)} = \sum_{r=1}^{R} v_{ii}^{(r)}d_{jq}^{(r)} \tag{2.12}$$

The EM algorithm guarantees that the adapted Gaussian distribution obtained by applying the transformation matrix $\mathbf{W}_r$ will increase the likelihood of the adaptation data at each iteration. The row-by-row estimation procedure for $\mathbf{W}_r$ can be performed using Gaussian elimination or LU decomposition. MLLR adaptation of the Gaussian mean is very effective and is able to improve ASR system performance by 13%-17% across a range of tasks [37]. In most cases, one to three iterations of MLLR, is sufficient to achieve significant performance improvements.

2.2.2 Variance Transformation

The Gaussian covariance matrices can also be adapted using linear transformations as shown in Eqn. 2.13 or Eqn. 2.14 (proposed in [37]),

$$\hat{\Sigma}_m = L_m \mathbf{H}_m L_m^T \tag{2.13}$$

$$\hat{\Sigma}_m = \mathbf{H}_m \Sigma_m \mathbf{H}_m^T \tag{2.14}$$

where L_m is the Choleski factor of the original covariance matrix Σ_m, and $\mathbf{H}_m$ is the adaptation transformation matrix in both cases. An iterative estimation procedure for the variance transformation of Eqn. 2.14 that guarantees increase in likelihood of the adaptation data with variance-adapted acoustic model is described in [37]. The estimation of variance adaptation is carried out in two steps such that

$$P(\mathbf{O}|\mathcal{M}) \leq P(\mathbf{O}|\hat{\mathcal{M}}) \leq P(\mathbf{O}|\tilde{\mathcal{M}}) \tag{2.15}$$

where $\mathcal{M}$ is the SI model, $\hat{\mathcal{M}}$ is the model with the adapted Gaussian mean and $\tilde{\mathcal{M}}$ is the model with the adapted Gaussian mean and variance. The adapted covariance matrices are "full", which can lead to increased computational overhead. A diagonal variance transformation can be estimated by forcing the off-diagonal elements to be zero in the iterative procedure. ASR system performance gains obtained from variance adaptation are in the range of 2%-7%, which much less than those obtained from Gaussian mean adaptation only.

2.2.3 Transformation structures

In past work, various structures of the adaptation transformations, both mean and variance, have been explored. The transformations can be full matrix, or block diagonal when using feature vectors that have distinct subsets e.g., 1st and 2nd differential components [78] or diagonal transformation in the case of sparse adaptation data. In the general case, the mean and variance adaptation transformations are separately estimated leading to different transformations, which is referred to as *unconstrained* MLLR. If they share the same transformation, then it is referred to as *constrained* MLLR [25].

2.3 Regression Class Trees

MLLR-based speaker adaptation produces significant ASR system performance gains with relatively small amounts of adaptation data and low computational overhead. To achieve this, SI Gaussian distributions are clustered into regression classes and all distributions within a particular regression class share a single MLLR transformation that is estimated using the adaptation data for that class. This allows the adaptation of SI Gaussian distributions which are not observed in the adaptation data and also provides robustness in cases of small amounts of adaptation data and against errorful adaptation transcription in the case of unsupervised adaptation. By sharing adaptation transformations, fewer adaptation parameters need to be estimated, compared to the case of estimating a unique transformation for every SI Gaussian distribution in the acoustic model. The regression classes are usually organized into a tree structure which is referred to as a regression class tree. The generation of regression classes can be divided into two problems: offline design of the regression class tree structure and online complexity control of the tree for a target speaker.

The task of designing regression class trees has two components: clustering criteria to form regression classes and a clustering algorithm to organize these classes into a tree. The primary goal while designing regression classes is to pool together those acoustic units into the same cluster that need to be transformed similarly. However, it is difficult to know beforehand, for an unseen speaker, how the SI Gaussian distributions should be transformed due to adaptation. There are three current approaches to this problem: an acoustic-space

approach as in [15, 58], where it is assumed that Gaussian distributions that are close in an acoustic space can be transformed similarly; an approach that determines the regression class tree structure by maximizing the likelihood of adaptation training data [31, 32]; and a phonetic-space approach as in [100,106], where it is assumed that acoustic units that belong to the same phonetic class transform similarly. The first two approaches are data-driven and require the availability of a corpus of adaptation training data consisting of several speakers, to ensure that the clusters formed are speaker independent.

To measure similarity between SI Gaussian distributions in an acoustic-space, two forms of distance measures have been investigated: a symmetric divergence-based distance measure between two Gaussian distributions $\mathcal{N}(\mu_1, \Sigma_1)$ and $\mathcal{N}(\mu_2, \Sigma_2)$ as shown in Eqn. 2.16 [57]; and a likelihood change-based distance measure as shown in Eqn. 2.17 that relies on the availability of statistics collected while training SI acoustic models [81]. Eqn. 2.17 shows the change in likelihood of acoustic model training data when D Gaussian distributions are merged into one distribution c, assuming that the training data consisted of T observations and $\gamma_s(\tau)$ is the probability of occupying state s at time τ.

$$D_{sym} = \frac{1}{2} tr(\Sigma_1^{-1}\Sigma_2 + \Sigma_2^{-1}\Sigma_1 - 2I) + \frac{1}{2}(\mu_1 - \mu_2)'(\Sigma_1^{-1} + \Sigma_2^{-1})(\mu_1 - \mu_2) \quad (2.16)$$

$$\delta\mathcal{L} = \Big(\sum_{d \in D} \frac{1}{2}\log(|\Sigma_d|) \sum_{\tau=1}^{T} \gamma_d(\tau) \Big) - \Big(\frac{1}{2}\log(|\Sigma_c|) \sum_{\tau=1}^{T} \gamma_c(\tau) \Big) \quad (2.17)$$

For the acoustic-space based approaches, there are three main clustering algorithms that have been investigated: a hierarchical agglomerative clustering procedure [57] and a divisive clustering procedure [15, 31]. In [57], SI Gaussian distributions are first clustered into *base classes* using an agglomerative procedure and the divergence-based clustering criterion. A base class is the smallest collection of Gaussian distributions which can share the same transformation, and in the limit each Gaussian distribution will form its own base class. The base classes are again agglomeratively clustered to form a tree using the same criterion such that the root node contains all the base classes. The divisive clustering approach starts with all base classes in the root node and then proceeds in a recursive manner by splitting every node (regression class) into two (regression classes) based on a chosen clustering

criterion. The system in [15] uses a soft K-means algorithm [69] as the clustering criterion.

In [31,32], the clustering criterion is based on change in likelihood of adaptation training data for a particular assignment of base classes to regression classes. An initial set of base classes is obtained from acoustic model training as in [81], and they are agglomeratively clustered into two regression classes based on either of the two methods just described (acoustic-space). Each base class swaps its regression class till the assignment of base classes to regression classes is the one that maximizes likelihood of the adaptation data. A two-class regression class tree, generated by this process, achieved small improvements in performance, when compared to a two-class tree generated by the acoustic-space approach on a standard unlimited vocabulary task.[1]The phonetic-space approach uses knowledge of acoustic phonetics to form regression classes. In the system described in [100], phones were clustered into classes based on their membership of broad acoustic phonetic classes. Also, in [106], Venkataramani and Byrne investigated the use of pronunciation changes in forming regression classes by grouping phones based on their changes that were predicted by a statistical pronunciation model of [89].

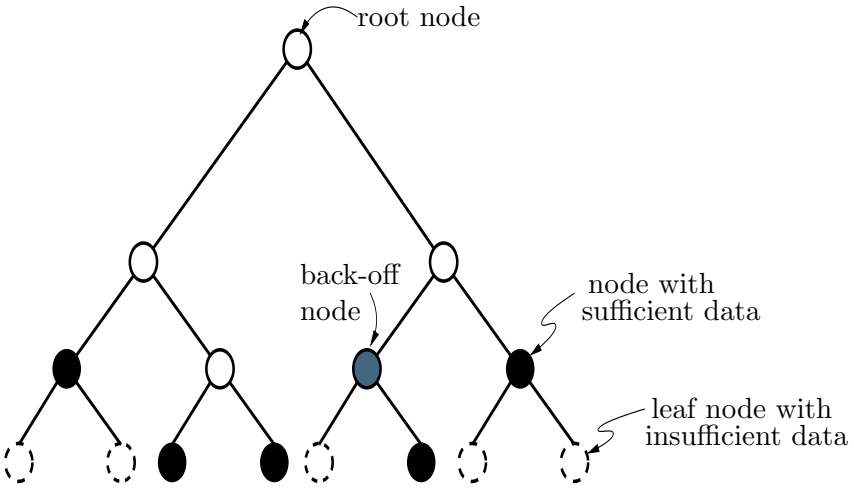

Figure 2.1: An example regression tree with 4 levels

The problem of online complexity control of the regression class tree is to decide a

[1]1994 ARPA Hub 1 unlimited vocabulary development and test set with approximately 15 sentences per speaker.

suitable depth of the tree to use for a target (unseen) speaker. Since each leaf node in the regression class tree represents a single regression class, this implicitly controls the number of MLLR transformations to be estimated. The most popular approach is to control tree depth based on the amount of adaptation data from a target speaker [15,57,58]. In this approach, the regression class tree is descended, starting from the root node, to the lowest node that has a sufficient amount of adaptation data to estimate a transformation for the Gaussian distributions assigned to it. An example of this approach is shown in the regression class tree of Figure 2.1. In [57,58,100,106] the approach to online complexity control was to use a fixed number of regression classes is that were predetermined based on the knowledge of the target task or development test results. Gales in [31] investigated two schemes for online complexity control: a cross-validation (CV) approach and an iterative MLLR approach both of which did not need any preset thresholds. Under the CV approach, the regression class tree is descended up to those nodes whose adaptation transformations, estimated using CV subsets of the adaptation data at that node, produced a higher likelihood gain on the adaptation data, compared to the transformation of the parent node. Under the iterative MLLR approach, as applied to a binary regression class tree, transformation estimation followed by subsequent recognition is performed for each level in the tree, and it is descended up to those nodes whose recognition hypothesis is different from the ones available in the parent nodes. Both these approaches, however, did not produce significant additional performance wins compared to the popular adaptation data threshold-based approach or just using a fixed number of regression classes.

Wang and Zhao describe a dynamic programming algorithm in [110], based on the minimum description length (MDL) principle to perform online complexity control of regression class trees. However this approach has a heavy computational overhead, since for every node of a full tree the estimates of different transformation matrix structures need to be obtained at first, and the procedure then performs an exhaustive search through the space of all tree-cuts of the full regression class tree.

2.4 Speaker Adaptive Training

Speaker-independent acoustic models tend to have higher variance than speaker-dependent models due to a wide-range of variations displayed by speakers in terms of vocal tract characteristics, dialect manifestations and other speaker-specific idiosyncrasies which are part of the speech signal. To model the inter-speaker and intra-speaker characteristics separately, a speaker-adaptive training (SAT) framework was proposed in [2,3], which extends the use of MLLR to the training speakers and produces a *compact* SI acoustic model that is a better starting point for adaptation. In the SAT paradigm, MLLR transformations are applied to both training and testing speakers, and SAT generally produces lower WER compared to non-SAT acoustic models. The general form of SAT has a significant computational overhead, since it uses unconstrained MLLR, which can be overcome by the use of constrained MLLR [33]. When a single (global) transform is used, the constrained MLLR transformation approach can be implemented as a feature-space transformation [33], which is not computationally expensive.

2.5 Related Research on MLLR

The background provided in this chapter serves as a basis for presenting the research results of this dissertation. Several promising variants of MLLR has been reported that are outside the scope of this work, but are briefly mentioned here. In cases of sparse adaptation data, the estimate of MLLR adaptation transformations are often biased and a better estimate can be obtained using prior densities for the transformations. In [17,19], the matrix-variate normal density served as the prior distribution for the adaptation transformation. A discriminative approach to estimating MLLR transformations was reported in [43]. MLLR has also been successfully applied for environmental adaptation [37], speaker recognition [28] and optical character recognition [62].

Chapter 3

MODELING SPEAKER VARIABILITY

MLLR-based speaker adaptation has been successful in improving ASR system performance for target speakers across a range of tasks. However, on closer inspection [65, 66], it has been observed that there are issues of robustness in the ASR system performance improvements obtained using MLLR. For example, about 10%-15% have degraded ASR performance when using adapted acoustic models. In addition, in cases of very sparse adaptation data, MLLR produces inferior ASR performance [53]. The most promising solutions to these problems have involved modeling speaker variability for large speaker populations and using such models to tune adaptation strategies for individual speakers. This chapter first provides a brief overview of sources of speaker variability, which is followed by a discussion of previous work in modeling speaker variability for ASR. Finally, a preliminary study is presented, that explores the relationship between several higher-level information sources of speaker variability and the performance improvements obtained using MLLR speaker adaptation.

3.1 Sources of Speaker Variability

In a large population, considerable diversity exists across the acoustic speech signal of individual speakers. This is referred to as speaker variability, in the context of ASR, and it arises due to several causes. The most important of these are enumerated below.

3.1.1 Pronunciation patterns

Differences in pronunciation patterns is a major source of speaker variability. These differences shown up when speakers use a different *dialect* e.g. speakers from different dialectal regions in United States; when individuals are *non-native* speakers, e.g., English spoken by a native Spanish speaker; due to *sociolectal* differences, which can include demographic

attributes such as age, years of education, social status, cultural upbringing, etc.

3.1.2 Transient

Intra-speaker variability can arise due to transient conditions that affect individuals, such as emotional state (e.g., anger, happiness, etc.) and sickness (e.g., common cold) that alter voice quality. In addition, the register, i.e., the formality of speaking situation and how familiar the speakers are with others, can impact the articulation quality and the extent to which a speaker uses dialectal variation.

3.1.3 Physiological

Physiological conditions that affect speech quality the most is the length and shape of the human vocal tract. The characteristics of the vocal tract is determined in part by gender, with males tending to have longer vocal tracts and lower pitch frequency in the speech signal. Other conditions such as damage to the larynx, for habitual smokers, can greatly affect the speech signal.

3.1.4 Environmental

Environmental conditions that influence the speech signal are widespread in everyday life, such as, speech from neighboring speakers, ambient noise from events, such as shutting doors, noise inside a car, etc. and conditions that affect the channel being used to transmit the speech signal, cellular vs. land-line telephone channels. Depending on loud the noise is, it can affect how a speaker talks (Lombard effect).

3.2 Speaker Clustering

In initial work that explored modeling speaker variability, the goal was to group a training speaker population into clusters and train a separate acoustic model for each. Then, a target speaker was assigned to the "closest" cluster-specific model before recognition. The idea was to capture speaker variability information in the cluster-specific models and use the appropriate one for a target speaker.

Imamura [50] proposed an approach in which initial speaker-specific acoustic models were clustered using a cross-entropy measure. Kosaka [52] presented an approach where speaker-specific models were organized into a tree such that any interior node in the tree stored an acoustic model trained using data of all speakers under that node, and a target speaker was assigned to that node which produced the highest likelihood on the adaptation data. In [82], a combination of speaker adaptive training and speaker clustering is reported, where MLLR transformations are applied to the N training speakers closest to a given test speaker, to train an SD model from the transformed data.

Major shortcomings of these approaches are that the speaker cluster-specific models may not be trained with sufficient amount of data to be representative of that cluster and the decision to assign a target speaker to a particular cluster may be errorful.

3.3 Model Combination Approaches

To address the problem of using a single speaker-cluster-specific model in classical speaker clustering approaches, the family of model combination approaches were developed. The common theme of all model combination approaches is to first train speaker-cluster-specific models (component models or adaptation transformations) from a large training speaker population. Then, for a target speaker, weights are estimated for each component model, using the adaptation data. Finally, the models are linearly combined to produce a composite (adapted) acoustic model for that particular speaker. The weights are the only parameters estimated, which are far less in number than the parameters in an adaptation transformation (mean or variance), which makes these approaches suitable for cases of sparse adaptation data. Some of the major model combination-based approaches are described below.

3.3.1 Cluster Adaptive Training

Gales proposed the cluster adaptive training (CAT) approach in [34]. It assumes that different speaker cluster models can have the same Gaussian covariances and mixture weights and only the Gaussian means vary across speakers. A set of canonical speaker cluster models can be trained in a framework similar to SAT and, given adaptation data from a target speaker, a set of weights for adapting the canonical model means are estimated using an

ML approach and a single regression class tree. CAT is able to achieve improvements in WER performance for large-vocabulary dictation tasks using small amounts of adaptation data.

3.3.2 Eigenspace approaches

Kuhn *et al.* proposed an adaptation technique in [53, 54] that also adapts the means of a set of canonical speaker-cluster-specific models. The canonical models, or eigenvoices, are formed by using principal components analysis (PCA) on a set of supervectors formed by all the mean vectors of a set of SD acoustic models. The vectors with the largest eigenvalues are chosen as the eigenvoices, or the basis set. During adaptation a maximum-likelihood eigen-decomposition (MLED) is used to estimate the weights (uses a single regression class tree) to be used for adapting the eigenvoices, which is similar to the weight estimation technique used in CAT. Eigenvoices have been useful in reducing WER for small-vocabulary tasks using little adaptation data.

In [14] and [63], an eigenspace representation of MLLR transformations (of a single acoustic model) was used to obtain basis MLLR transformations, and a test speaker's MLLR transformation is produced by interpolating the basis transformations, using weights estimated from the speaker's adaptation data and a single regression class tree.

3.3.3 Reference Speaker Weighting

Hazen proposed in [45] a scheme to decide on a set of reference speakers whose models were combined using weights estimated by a ML approach on a target speaker's adaptation data. The model of each reference speaker was represented using an estimate of the centroid of the Gaussians of each HMM state in the speaker-dependent models. The weights were constrained to sum to one and this approach was suitable for cases of sparse adaptation data.

3.3.4 Stochastic Transformations

The basic form of MLLR as in (2.5) assumes that a linear transformation adequately models the dependencies between training and testing speakers. This assumption may be too simplifying and an alternative approach, referred to as maximum likelihood stochastic transformation, was proposed by Diakoloukas and Digalakis in [104] where a probabilistic piecewise linear transformation is used instead. Using this approach the probability of state m generating observation $\mathbf{o}_t$ at time τ is expressed as,

$$p(\mathbf{o}(\tau)|j) = \sum_{i=1}^{N_\omega} \sum_{k=1}^{N_\alpha} p(\omega_i|m) p(\alpha_k|m, \omega_i) \cdot \mathcal{N}(\mathbf{o}(\tau); \mathbf{W}_{m,k}\xi_{m,i}, \Sigma_{m,i}) \tag{3.1}$$

where N_α is the number of component transformations used by each HMM state m, and $\sum_{k=1}^{N_\alpha} p(\alpha_k|m, \omega_i) = 1$. α_k denotes the event that the kth transformation is used, and the component transformation $\mathbf{W}_{j,k}$ are shared by all Gaussian distributions in state m and the output distribution of state m is modeled by a mixture Gaussian distribution. The probabilities $p(\alpha_k|m, \omega_i)$ that select the kth transformation at time t for the ith mixture Gaussian component of state m are specific to each mixture component. The parameters of the N_α transformation matrices are estimated by maximizing the likelihood of the adaptation data. The details of the re-estimation formulae are provided in [104]. Each of the N_α transformations can be thought of forming a set of basis transformations which are interpolated using the weights $p(\alpha_k|m, \omega_i)$. The basis set can represent a set of canonical speakers, speaker clusters, etc. The linear combination of transforms was successful in reducing WER for both speaker adaptation and dialect-adaptation experiments [8].

3.4 Improving MLLR Robustness

Several approaches have been explored with the specific aim of improving robustness of performance gains obtained using MLLR in applications where the amount of adaptation data is small. An MAP-like interpolation scheme between the SI Gaussian mean and the estimated mean was proposed in [42]. In [17] and [19], a prior distribution for the mean transformation parameters was used, which improves performance when adaptation data

available is small. A variant of the EM algorithm that uses a discounted likelihood criterion and does not quickly over-train was presented in [12]. These approaches, however, are mostly concerned with cases of sparse adaptation data, and as such are not less relevant to the results presented in this dissertation that are concerned with handling a wide range of available data, e.g., from several minutes to a only a few seconds.

3.5 Pilot Study

A pilot study was conducted with the aim of understanding the relationship, if any, between higher-level speaker variability information and ASR performance changes due to MLLR-based adaptation. The plan of the study was to first examine robustness issues in MLLR adaptation, and depending on the outcome, next use higher-level speaker-dependent features to predict (a) ASR performance change due to MLLR adaptation and (b) if an individual speaker would benefit or hurt due to adaptation. The eventual goal was to design adaptation strategies that took into account speaker variability information, based on conclusions (or evidence) that resulted from the pilot study.

3.5.1 ASR System & Task

A development version of the ASR system described in [100], focused on the task of recognizing conversational telephone speech (CTS) in North American English from 544 speakers[1] was used for this study. The system used unsupervised MLLR-based adaptation with a full transformation for Gaussian mean adaptation and diagonal transformation for variance adaptation and a regression class tree that was built using acoustic phonetic knowledge of association of phones into clusters. For online complexity control, the threshold on the amount of adaptation data was set to 200 frames of speech, which resulted in nearly all speakers in the test set to use 9 regression classes. The system description is kept brief here, in the interest of focusing on the pilot study itself. A more detailed description of all the ASR systems and tasks used for the research presented in this dissertation is available in Chapter 4.

[1] NIST English CTS 1998-2003 Evaluation test sets

3.5.2 Variable Success of MLLR

Figure 3.1 shows a histogram of the relative changes in WER due to MLLR adaptation for the speakers mentioned above. The horizontal axis represents relative Gain(+)/Loss(-) from MLLR adaptation. About 15% of the speakers show worse performance after adaptation, which demonstrates that MLLR is not successful in improving ASR system performance for all speakers.

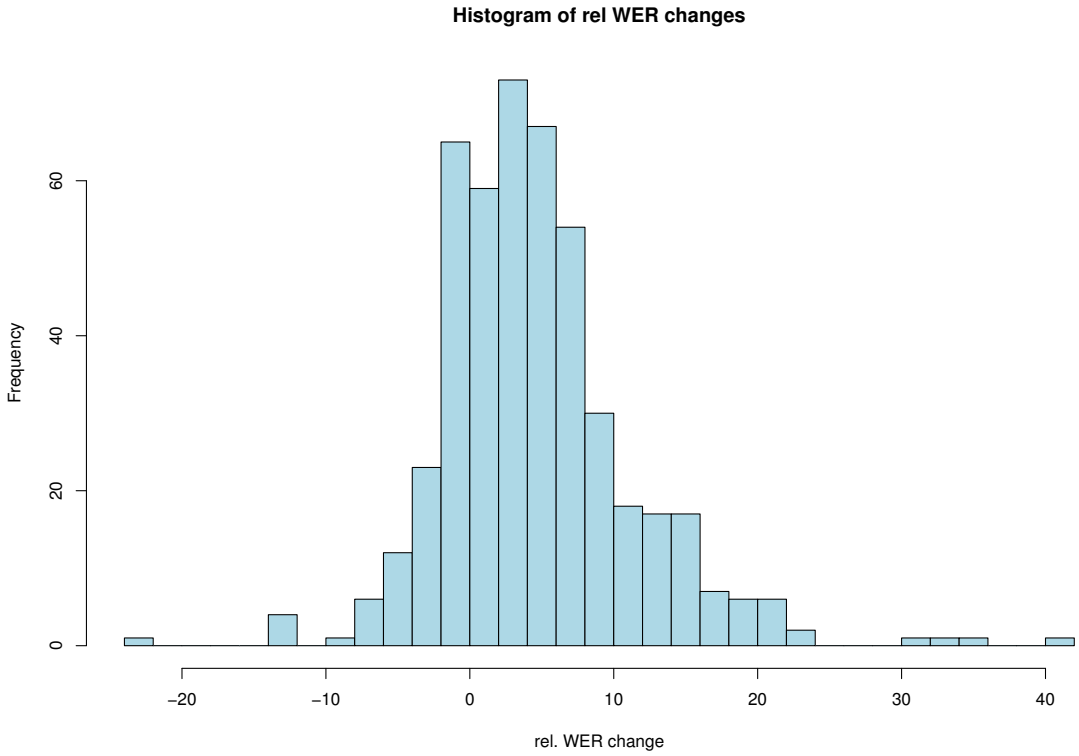

Figure 3.1: Relative WER (%) changes due to MLLR adaptation (English CTS)

3.5.3 Prediction Framework

With evidence of performance degradation due to MLLR adaptation at hand, a linear regression-based framework (Eqn. 3.2) [68] was used for predicting e, the ASR performance

change due to MLLR adaptation and a classification-based framework (Eqn. 3.3) was set up, using logistic regression [68], to predict, if a particular speaker would benefit from MLLR adaptation, with probability p, using speaker-level features, $x_1, x_2, \ldots x_K$ consisting of confusion network confidence measures [67] of the unsupervised adaptation hypothesis, statistics for rate of speech (ROS), fundamental frequency (F0) and energy of the acoustic signal, and vocal tract length (VTL) warping factor.

$$e = \alpha + \sum_{i=1}^{K} \beta_i x_i \qquad e \in \Re \tag{3.2}$$

$$\log\left(\frac{p}{1-p}\right) = \alpha + \sum_{i=1}^{K} \beta_i x_i \qquad p \in \{0, 1\} \tag{3.3}$$

The speakers were divided into two parts: a training set comprising 427 speakers that used for training the two predictors and a held-out test set comprising 72 speakers.

3.5.4 Prediction Results

Table 3.1 shows the root mean square error (RMSE) for predicting relative (rel.) change in WER due to MLLR adaptation using various subsets of features. As can be seen, the predictor is weak for all subsets of features, compared to the case of just predicting the training mean. For the case of predicting whether a particular speaker would benefit from adaptation[2], the classification error was 33%, which was not significantly better than chance, or predicting the most frequent case.

3.5.5 Conclusions

The pilot study provides clear evidence of problems with robustness of ASR performance improvements obtained due to MLLR adaptation. As mentioned in Section 1.2.3, this is an important issue for ASR systems that handle real-world conditions, that determines the overall usability of the system across a wide-range of conditions. The weak performance

[2]In ASR terminology, speakers who are difficult to recognize, or for example, have worse performance from adaptation, are referred to as "goats", while those having high performance levels are referred to as "sheep".

Table 3.1: RMSE of predicting Rel. Change in WER due to MLLR adaptation

Configuration	RMSE
Training Mean 1	8.72
Unadapted WER	8.65
Confidence	8.07
Confidence + F0 + ROS	8.06
All speaker-level features	8.06

of the higher-level speaker-dependent features for the two prediction tasks, in the pilot study, indicate that predicting "sheeps" vs. "goats" is not the correct approach to solve this problem, and the need to investigate automatically-derived speaker-level information sources, rather than knowledge-driven features for the purpose of incorporating speaker variability information in designing an adaptation strategy for target speakers. This issue is explored further, in Chapters 5 and 7, which describe results to address the robustness of MLLR adaptation.

Chapter 4

SYSTEM DESCRIPTION

The research presented in this dissertation is based on the Decipher^{TM} ASR platform developed at SRI International[1] with contributions from researchers at the University of Washington[2] and the International Computer Science Institute.[3] It is a general purpose large vocabulary ASR platform that can be customized for several different domains and achieved competitive performance levels on benchmark evaluations conducted recently by NIST.[4] This chapter provides details of the ASR system architecture for the following domains: (i) English conversational telephone speech (CTS) [100]; (ii) English broadcast news (BN) [105]; (iii) Mandarin BN and broadcast conversations (BC) [49]. The discussion focuses on the components relevant to speaker adaptation. Detailed information covering all aspects of the ASR platform, which have been developed over several years, are provided in [75, 76, 99, 100].

4.1 English CTS

The English CTS system is set up to be a multi-pass system as shown in Fig. 4.1. It uses two different front-end signal processing schemes: Mel-frequency cepstral coefficients (MFCC) [22] and perceptual linear prediction (PLP) [46] for deriving feature vectors. The MFCC features are concatenated additionally with Tandem/HATS features [74]. The feature vectors are processed using standard normalization schemes including mean, variance, vocal tract length normalization [112], and heteroschedastic linear discriminant analysis (HLDA)-based [40] normalization to produce feature vectors of 42-dimensional vectors.

---

[1]Speech Technology and Research Laboratory(`http://www.speech.sri.com`)

[2]Signal, Speech and Language Interpretation Laboratory(`http://ssli.ee.washington.edu`)

[3]`http://www.icsi.berkeley.edu/groups/speech`

[4]National Institute of Standards and Technology (`http://www.nist.gov/speech/`)

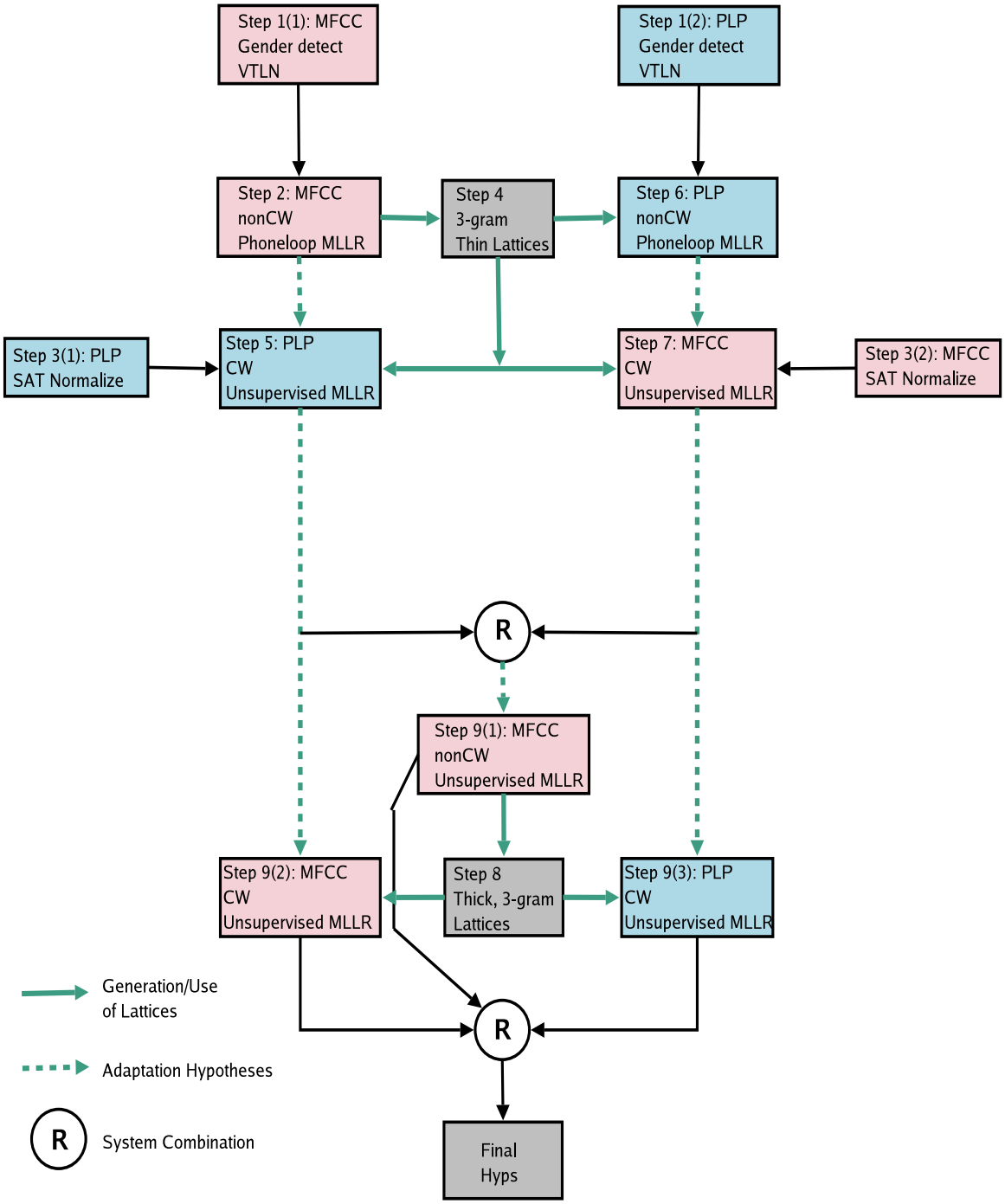

Figure 4.1: Architecture of 20xRT English CTS system

Gender-dependent acoustic models that characterize tri-phones as HMMs and use diagonal covariances for the state output distribution are trained for each front end. The acoustic models for first pass decoding (Step 2 & 6) use word-internal triphones, a bigram language model, and phone-loop MLLR to produce "thin" lattices. The lattices are re-scored using a 4-gram almost-parsing SuperARV [111] language model and processed using confusion networks [67] to produce higher-quality hypotheses to be used for adaptation transformation estimation in subsequent stages. The bigram lattices are also expanded using a trigram language model and used with cross-word acoustic models in the second and third-passes of the system (Steps 5, 7 and 9(1-3)). The HMM states of all acoustic models are clustered using a decision tree [81] and linguistic questions for sharing mixture Gaussian distributions. The word-internal acoustic models use 320,000 Gaussian distributions for each front end, and the cross-word ones use 384,000 distributions. In addition, the cross-word acoustic models uses normalization based on speaker adaptive training (SAT) using constrained MLLR (CMLLR) and are trained using the an alternating minimum phone error and maximum mutual information estimation (MPE-MMIE) criterion [85, 120]. The mean vectors and diagonal covariances of the Gaussian distributions of the crossword acoustic models, in the second and third passes, are adapted to test speakers using unsupervised MLLR and adaptation hypotheses from the previous passes. The adaptation hypotheses are exchanged between the two front ends, which is referred to as *cross-system adaptation*. Intermediate recognition hypotheses are produced by decoding lattices from the previous stage and the adapted acoustic models. The third-pass of the system uses "thicker" lattices compared to the second pass. The final recognition hypotheses are produced by confusion network-based system combination [29] of the three sub-systems of the third pass of the system that includes additional knowledge sources from a duration model [30] and a pause language model [107].

The acoustic model training comprised over 2000 hours of data from the Fisher [20] and Switchboard [39] corpora. The language model training data was drawn from these corpora and web sources [11]. This multi-pass system runs under 20 times real-time (13.8xRT) on 3.4 GhZ Intel Xeon processor that had 3 GB of memory.

The architecture of a faster-version of the multi-pass system is shown in Fig. 4.2. This

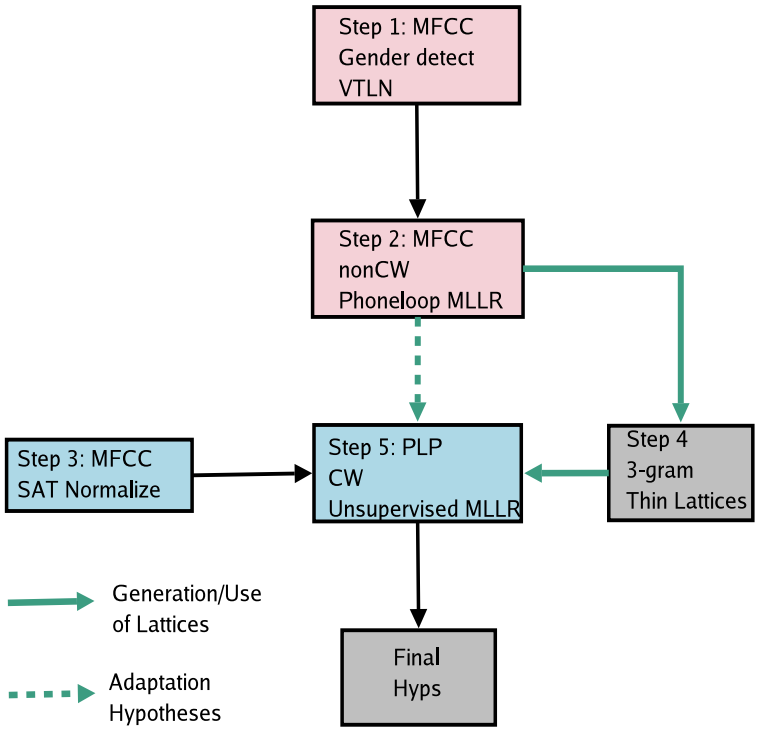

Figure 4.2: Architecture of "fast" 5xRT English CTS system

smaller system performs nearly all the functionality of the larger system, but runs under 5xRT. It also performs cross system adaptation by exchanging adaptation hypotheses between MFCC and PLP front ends.

4.2 English BN

The English BN version of the system shares several components of the CTS system and its architecture is shown in Fig. 4.3. It has some notable differences: it uses gender-independent acoustic models; it segments BN shows into speaker "groups" using an unsupervised clustering algorithm [90], so that these segments can be used as speaker labels for MLLR adaptation; and it does not employ cross-system adaptation, but instead uses PLP-based features for the two-passes of the system, which are 52-dimension vectors after standard normalization. The lattices generated in the first pass (Step 3) are expanded using

a 5-gram SuperARV language model to a 4-gram PFSG lattice, which were used with the adapted cross-word MPE-MMIE-based acoustic models in the second pass (Step 7). The adaptation hypothesis are generated after the first pass from the confusion network processing of 5-gram expanded and duration-model-rescored lattices. The number of Gaussian distributions trained for the two acoustic models were the same as in the CTS system.

The acoustic models were trained using over 3000 hours of BN data from several different corpora and the language models were trained from a training set of approximately one billion words. The system ran under 10xRT (9.38) on a 3.4 GHZ Intel Xeon processor with 2 GB memory.

4.3 Mandarin BN/BC systems

The Mandarin BN/BC system used an architecture similar to that of the English BN system, except it used MFCC-based features with additional pitch features and a tonal phone set in both recognition stages. The cross-word acoustic models, in the second-pass of the Mandarin BN/BC system, uses fMPE-based discriminatively trained feature vectors [84]. The acoustic model training data included 310 hours of Mandarin and 150 hours of Mandarin BC shows.

4.4 MLLR Adaptation

In the unsupervised MLLR step in all the above ASR systems, the adaptation hypotheses are first aligned using the forward-backward algorithm [6]. Next a full matrix transformation with an offset vector and a diagonal transformation vector are estimated for adapting the Gaussian means and covariances respectively. The regression class tree (RCT) used with unsupervised MLLR in the baseline version of the systems was manually designed using knowledge of acoustic phonetics. The RCT groups triphone states with the same center phone according to speaker manner classes (e.g., vowels, fricatives, stops, etc.) and organizes them into a tree with 9 leaf classes for English CTS/BN and 3 leaf classes for Mandarin BN/BC. The threshold used for online complexity control of MLLR adaptation was 200 frames.[5] This threshold was low enough that for nearly all speakers the number of MLLR

[5]The front-end signal processing for all the ASR systems set the frame rate to be 10ms with a 25ms window, which implies that 200 frames corresponds to 2 seconds of acoustic data.

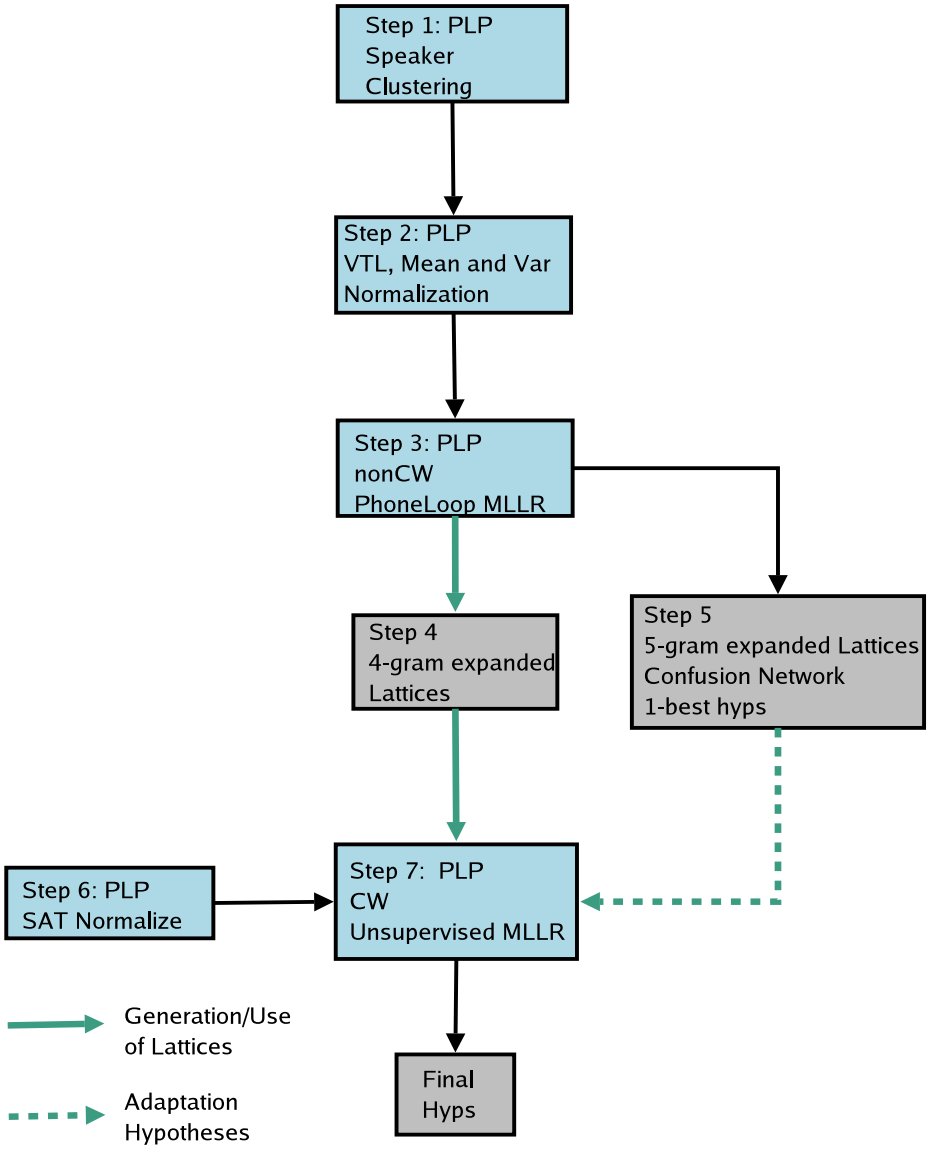

Figure 4.3: Architecture of 10xRT English BN system

transformations estimated was the same as the number of leaves in the tree.

4.5 Baseline Performance

Table 4.1 shows the baseline performance levels of the various ASR systems, after the last unsupervised MLLR step (Box labeled as "Final Hyps" in Figs. 4.1, 4.2 and 4.3), on recent NIST benchmark evaluations. These performance levels are competitive, when compared to other recently developed systems that participated in the same evaluations [15, 36, 70], and is used as the baseline performance levels for all the results presented in this dissertation.

Table 4.1: Baseline WER(%) of ASR systems for various domains

Domain(Year)	WER(%)
English CTS 2004	18.6
English CTS 2004 (fast)	21.5
English BN 2004	16.0
Mandarin BN	8.0
Mandarin BC	20.7

Chapter 5

SPEAKER-CLUSTERED REGRESSION CLASS TREES

5.1 Introduction

Speaker clustering has been investigated in previous work, as a means of characterizing speaker variability, with the aim of introducing flexibility in the adaptation strategy for target speakers. In Cluster Adaptive Training (CAT) [34], which is a speaker clustering approach to SAT, several cluster-specific acoustic models or MLLR transformations (and a single speaker-independent acoustic model) are trained. For a test speaker, a speaker-adapted model is derived by estimating weights, from adaptation data, to combine the component acoustic models (or component MLLR transformations) using a single regression class tree. Kuhn et al. [53] proposed eigenvoices or basis acoustic models, derived from an eigenspace representation of several cluster-specific acoustic models. The basis models are linearly combined using optimal weights, from adaptation data, to produce a speaker-adapted model that lies within the span of the eigenvoices. In [14] and [63], an eigenspace representation of MLLR transformations (of a single acoustic model) was used to obtain basis MLLR transformations, and a test speaker's MLLR transformation is produced by interpolating the basis transformations, using weights estimated from the speaker's adaptation data and a single regression class tree. In all these approaches, the component models or transformations are static, computed as part of the training process, and adaptation involves only estimating weights for combining transforms or models. Since the number of weights is typically small (one per cluster), these methods are good for cases of sparse adaptation data.

This chapter introduces a new strategy for using clustering to model speaker variability in speaker adaptation based on MLLR, that aims to take advantage of a range of amounts of adaptation data, and differs from the previous approaches in two respects. First, it leverages the benefits of speaker clustering in regression class tree structure design. Second,

it estimates component MLLR transformations dynamically for each test speaker using the cluster-specific regression class trees. The speaker clusters are created by clustering held-out training speakers (not used in acoustic model training) that are represented in the eigenspace of MLLR transformations of a single acoustic model. A regression class tree is then trained using the data available from the speakers in each cluster, motivated by the goal of capturing cluster-specific differences of dialect (or sociolect) or speaking style in the structure of the trees. This approach is also motivated by the hypothesis that sub-groups of speakers may achieve improved ASR system performance by using different regression class tree structures, i.e., that using the wrong structure may lead to lower performance levels. Evidence supporting this hypothesis is presented in Sec. 5.4 that shows significant ASR performance gains are achievable by choosing the optimal regression class tree structure for each target speaker.

In addition, an algorithm is presented that is used on target speakers to produce MLLR transformations by combining component transformations available from speaker-clustered regression class trees. The transformations are combined using optimal weights, that maximize the likelihood of adaptation data by extending the method described in [37] to include a backoff strategy, and the detailed transformations produce improved ASR performance across a range of tasks. The algorithm for estimating the optimal weights needs to store only the component MLLR transformations in memory and a single acoustic model. This results in reduced memory requirements for our approach, compared to eigenvoices and CAT, which interpolate component acoustic models and need memory for each one at recognition time.

5.2 Speaker Clustering for Regression Class Trees

5.2.1 Regression Class Trees

Two divisive clustering approaches were explored for building regression class trees (RCT): *constrained* and *unconstrained*. Both approaches start by estimating multivariate Gaussian distributions for each triphone state, and collect these to obtain phone-level sufficient statistics. Then, in the constrained approach, a decision tree is designed to cluster the Gaussian

distributions, choosing from linguistically motivated questions about the center phones to maximize likelihood of training data, similar to clustering triphone states of HMMs using decision trees [118]. In the unconstrained approach, a binary tree is built by splitting the distributions at each level into two clusters, using k-means clustering and a symmetric Kullback-Leibler (KL) distance measure between Gaussian distributions (Eqn. 2.16) as described in [57]. Both trees are grown to the point where all leaf nodes correspond to a single phone. The two types of RCT have the same set of leaf nodes but different branching structure leading to the leaves. For illustration purposes, pruned-versions of the two types of trees are shown in Fig. 5.1 and 5.2, in which the shaded nodes represent the leaves and the interior ones the questions learned for the splits. A variant of the unconstrained case was also explored, where the triphone state-level distributions were directly used in building the tree, which resulted in the leaves of the tree representing the individual state-level distributions. This tree is referred to as the *unconstrained state-level* regression class tree. The *constrained* and *unconstrained* RCTs have the same set of leaf nodes (one leaf for every phone), but all three RCTs have different branching structure leading to the leaves. Given limited adaptation data, it is often the case that the estimated transforms correspond to internal nodes (non-root, non-leaf node), which then leads to different adaptation results as a function of RCT structure.

5.2.2 Speaker Clustering in MLLR Eigenspace

MLLR transformations represent SD descriptions with reference to an SI model, and are thus a logical choice for modeling speaker variability. Eigenspace-based MLLR representations were found to be useful for gender classification [47] and as auxiliary features in mixtures-of-experts classifiers for speaker recognition [28]. In [14], such representations of MLLR transformations serve as basis transformations, which are linearly combined using weights, that maximize the likelihood of a target speaker's adaptation data, to obtain the final adaptation transformation. In contrast, the use of such representations of MLLR transformations in this dissertation is to obtain adaptation-relevant speaker clusters.

First, MLLR transformations are estimated for a large corpus of held-out training speak-

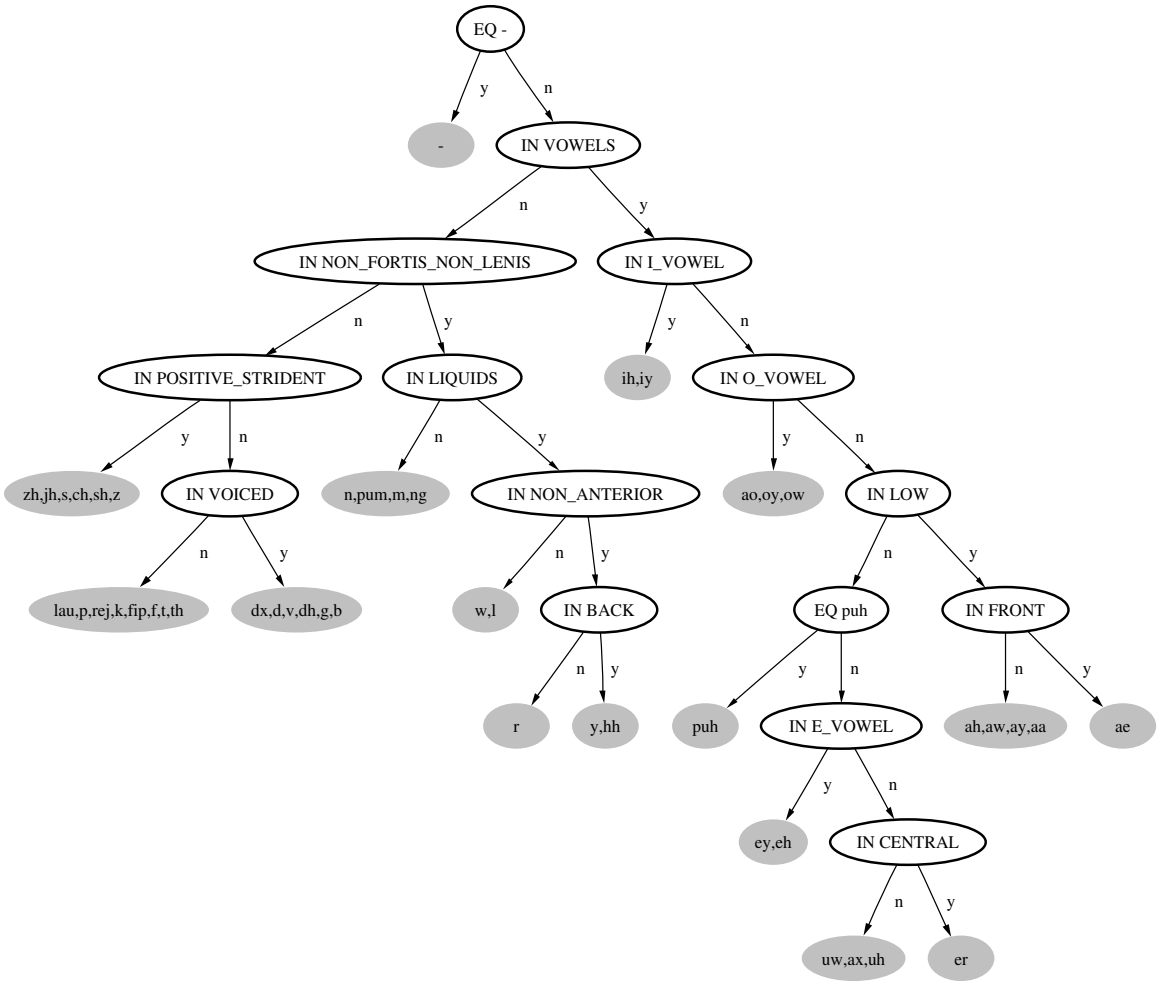

Figure 5.1: Constrained RCT for English CTS Male

ers, using a single constrained RCT that has R regression classes. Given a d-dimensional feature vector used in recognition, the MLLR transformations (mean transform, offset vector) are then vectorized to produce a $d(d+1)$-length vector, and normalize each dimension to have zero mean and unit variance. Next, principal component analysis (PCA) is performed on the vectorized MLLR transformations of all regression classes, except those corresponding to the nonspeech class. For purposes of numerical stability, PCA is performed using a singular value decomposition on the data matrix [68, 86]. The vectorized transforms are

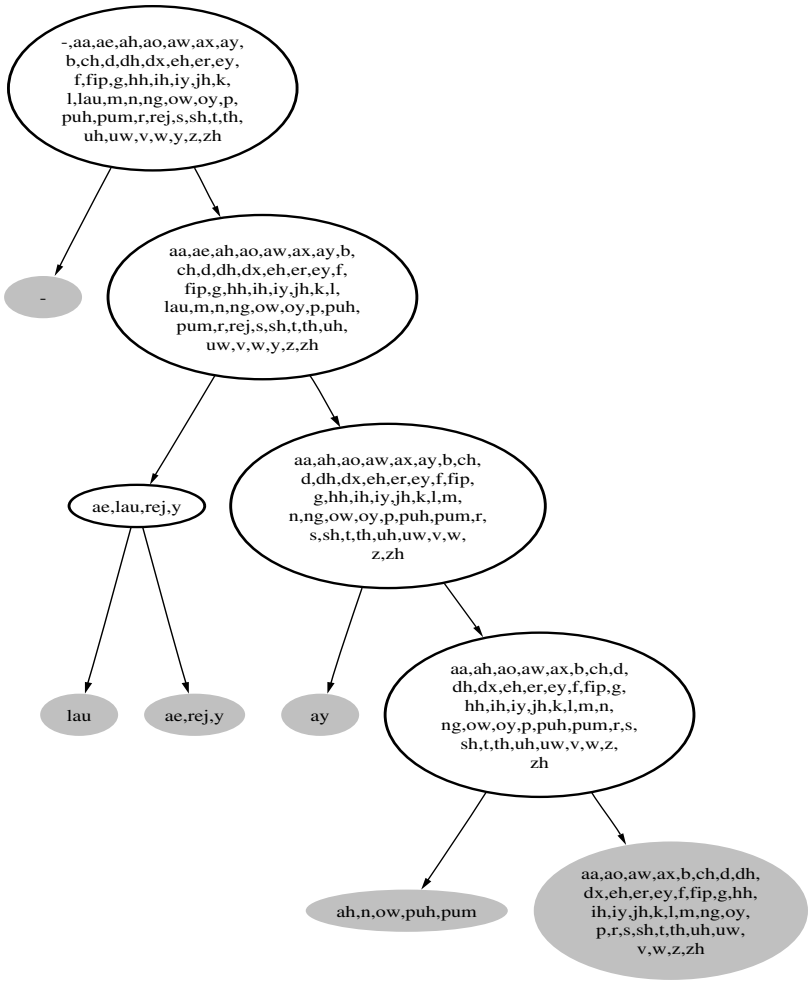

Figure 5.2: Unconstrained RCT for English CTS Male

then projected onto the first N principal components, and an RN-dimensional supervector is formed for each speaker by stacking together the PCA-reduced MLLR transforms for each of the R classes (excluding the nonspeech class). Finally, k-means clustering is used to partition the speakers into S clusters, using a Euclidean distance measure between the supervectors. The supervectors capture the speaker-dependent information present in MLLR transformations, and the clustering groups together speakers who share similar transform characteristics.

Given the speaker clusters, a separate RCT is trained, both constrained and unconstrained types, for each speaker cluster with the goal of capturing cluster-specific attributes in the structure of the RCT, or equivalently the appropriate phone groupings for transformation tying.

Since the training speakers in each cluster are more similar, in MLLR eigenspace, to speakers of their own cluster than to those of other clusters, it is hypothesized that the cluster-specific RCT will capture patterns representative of each speaker cluster in the structure of the RCT. This will produce diversity in RCT structures across clusters. It is also expected that choosing the appropriate MLLR RCT structure for every speaker should lead to improved ASR performance. In Section 5.4 evidence is presented that this strategy does indeed lead to different RCT structures and improved ASR performance results when using the oracle RCT.

5.3 Task and System Description

Three different ASR systems, based on SRI's Decipher^{TM}were used for all experiments presented in this chapter. The three ASR systems were: (i) English CTS "fast"; (ii) English BN and (iii) Mandarin BN/BC. All three systems perform unsupervised MLLR ("full" mean and diagonal variance transformation) once as shown in 4.3 using the two types of regression class trees described in Sec. 5.2.1 . The rest of the details of the these systems have already been described in Chapter 4.

Speaker clustering was performed separately for each domain in each language, and for each gender when gender-dependent acoustic models were used. Three different corpora were utilized for performing speaker clustering in each domain and language:

- conversations of speakers from the Fisher Phase 2 corpus [20] and recent NIST English CTS test sets (1998-2002) for use with the English CTS system, which together included 1186 male speakers and 567 female speakers (120 hours);

- BN in English (25 hours) for use with English BN system; and

- BN and BC shows in Mandarin (25 hours for each domain) for use with the Mandarin BN/BC system

The English BN and Mandarin BN/BC corpora were released by the Linguistic Data Consortium (LDC) in early 2006. The corpora used for speaker clustering was not part of the acoustic model training data for the different ASR systems. Only the NIST CTS test sets (1998-2002) and the NIST 2004 English BN test set were used for error analysis. Evaluation is performed on the NIST 2003 English CTS test set (12 hours), the NIST 2004 English BN test set (6 hours), the NIST 2006 Mandarin BN test set[1] (1 hour), and a 2005 Mandarin BC test set[2] (2.5 hours).

5.4 Oracle Cluster-Dependent Adaptation

5.4.1 Oracle Performance Improvements

The initial aim was to evaluate potential ASR performance gains from using the "best" RCT (in terms of word error rate) for individual speakers, choosing from among the ones produced by the speaker clustering algorithm.

Using the relevant constituents of the corpus in Section 5.3, speaker-clustered RCTs were trained for English CTS and BN. In the speaker-clustering algorithm, each speaker was represented by 8 vectorized MLLR transformations, since this produced stable clusters. Several different values of N, the number of principal components for projecting the vectorized MLLR transforms, were experimented with and $N = 8$ was chosen, since it produced the most diversity in structure among the cluster-specific RCTs. Diversity of tree structures is subjectively evaluated by visual examination of splits of phone clusters at the top levels in the cluster-specific RCTs, which have the most impact on tree structure. In the case of constrained RCTs, where splits correspond to linguistic questions the most diverse case had approximately 3 out of 4 questions being different at two levels below the root, across the different cluster-specific RCTs.

---

[1] As used during DARPA GALE Spring 2006 dry run tests.

[2] Test set prepared by Cambridge University and used for internal evaluations; not released publicly.

The k-means-based clustering algorithm was set up to produce 4 clusters for each gender (when using gender-dependent acoustic models) in each domain. This ensured that the speaker partitions had an adequate amount of data to train a cluster-specific RCT, and also maintained reasonable limits on computational costs of the MLLR transformation combination algorithm (described later). Next, to estimate the potential maximum ASR system performance gains from using a cluster-specific RCT the sequence of steps described in Algorithm 1 are followed.

Algorithm 1 Procedure to compute oracle cluster-dependent WER

1: Speaker Clusters $\mathcal{C}(\mathcal{S})$: Perform speaker clustering to produce k clusters, using the proposed approach and all speakers in $\mathcal{S}$, the training set of a given domain and language.

2: Hold out subset of speakers $\mathcal{S}_{\mathcal{H}}$ for evaluation purposes, and define the remaining training speakers as $\mathcal{S}_{\mathcal{T}} = \mathcal{S} - \mathcal{S}_{\mathcal{H}}$ from the speaker population $\mathcal{S}$. $\mathcal{C}(\mathcal{S}_{\mathcal{H}})$ is then the clustering of the speakers in $\mathcal{S}_{\mathcal{H}}$.

3: Train *constrained* or *unconstrained* RCTs $\mathcal{T}_1$, one for each speaker cluster in $\mathcal{C}(\mathcal{S})$, using the data of training speakers in that cluster, excluding the held-out speakers in $\mathcal{S}_{\mathcal{H}}$.

4: Speaker Clusters $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$: Produce new cluster assignments $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$ of speakers in $\mathcal{S}_{\mathcal{H}}$, by re-assigning each to that cluster index of $\mathcal{C}(\mathcal{S})$, whose RCT in $\mathcal{T}_1$ produces the lowest WER, after MLLR adaptation.

5: Compute the overall WER for each of these new clusters in $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$ using the WER of each speaker in it and for every cluster-specific RCT in $\mathcal{T}_1$.

The held-out subset of speakers $\mathcal{S}_{\mathcal{H}}$ of Step 2 in Algorithm 1 for English CTS experiments were drawn from recent NIST English CTS test sets (1998-2002) that were part of the training data for the speaker clustering algorithm, but not used in training cluster-specific RCTs, after ignoring around 1% of these speakers for whom there was no difference in ASR performance among the different cluster-specific RCTs. The results of Algorithm 1 (Step 5), using the unconstrained RCT, are shown in Table 5.1, where the rows represent test sets for each speaker cluster $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$ and the columns the cluster-specific RCT in $\mathcal{T}_1$ as defined in Algorithm 1. By definition, the overall WER of each new cluster in $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$, will be the lowest, using the RCT of its own cluster, compared to that achieved by using the RCT of

any other cluster. Not surprisingly, the best WER is seen for all cases when the cluster-specific test set matches its target RCT, i.e., the numbers along the diagonals of the Table 5.1. In addition, the upper bound of potential gains over the SI RCT is in the range of 0.6 to 0.8% (absolute) for the unconstrained RCT. On analyzing the performance numbers for each speaker, it was noticed that when the cluster-specific test set matches its target RCT, the error rate for the worst-performing speaker improves by 0.5 to 1.9% (absolute). Similar observations are made on analysis of the performance figures from the constrained trees, and are not presented here for brevity.

Table 5.1: Oracle WER(%) for English CTS using unconstrained RCT

	Clust 1	Clust 2	Clust 3	Clust 4	SI
Clust 1	**20.5**	21.2	21.2	21.3	**21.3**
Clust 2	22.0	**21.3**	22.1	22.1	**21.9**
Clust 3	24.1	24.4	**23.6**	24.0	**24.3**
Clust 4	21.6	21.8	21.6	**20.9**	**21.7**

These experiments were repeated with English BN. Cluster-specific RCTs were trained using only speakers in 25 hours of English BN data released by LDC in 2006 and tested on the held-out NIST 2004 English BN test set using the steps of Algorithm 1. The results are shown in Table 5.2. The trends in performance gains in this case, using oracle cluster RCT, are similar to that in the case of English CTS. The gains compared to the SI RCT vary from 0.5% to 0.8%, with the exception of one cluster, which shows no improvement. The quantitative evidence of ASR performance improvements presented here makes the case that applying the speaker-clustered RCT to target speakers would achieve improvements in overall ASR performance of MLLR adaptation.

The clustering of speakers in the held-out sub-set $\mathcal{S}_{\mathcal{H}}$ changed for many speakers between Step 1 and 4 of Algorithm 1. Since the assignment of speakers to clusters in $\mathcal{C}_2(\mathcal{S}_{\mathcal{T}})$ is based on minimum WER (rather than minimum squared error on transformations), a reclustering approach based on this criterion was explored, as detailed in Algorithm 2. A new assignment, $\mathcal{C}_2(\mathcal{S}_{\mathcal{T}})$, of speaker-clustering training speakers to that cluster whose tree produced the

Table 5.2: Oracle WER(%) for English BN using unconstrained RCT

	Clust 1	Clust 2	Clust 3	Clust 4	SI
Clust 1	**14.3**	14.8	14.8	14.8	**14.8**
Clust 2	17.1	**16.3**	17.2	17.1	**17.1**
Clust 3	14.6	14.7	**14.3**	14.8	**14.8**
Clust 4	16.4	16.7	16.3	**16.1**	**16.1**

Algorithm 2 Procedure to compute cluster-dependent WER with retrained RCT

1: Retain training set speakers $\mathcal{S}$, held-out speaker subset $\mathcal{S}_{\mathcal{H}}$, rest of the training speakers $\mathcal{S}_{\mathcal{T}}$, speaker cluster assignments $\mathcal{C}(\mathcal{S})$ and $\mathcal{C}(\mathcal{S}_{\mathcal{H}})$ and cluster-specific RCT $\mathcal{T}_1$ of Algorithm 1 for a given language and domain.

2: Speaker Clusters $\mathcal{C}_2(\mathcal{S}_{\mathcal{T}})$: Produce new cluster assignments $\mathcal{C}_2(\mathcal{S}_{\mathcal{T}})$ of speakers in $\mathcal{S}_{\mathcal{T}}$, by re-assigning each to that cluster index of $\mathcal{C}(\mathcal{S})$, whose RCT in $\mathcal{T}_1$ produces the lowest WER, after MLLR adaptation.

3: Train *constrained* or *unconstrained* RCTs $\mathcal{T}_2$, one for each speaker cluster in $\mathcal{C}_2(\mathcal{S}_{\mathcal{T}})$, using the data of training speakers in that cluster.

4: Compute the overall WER for each of the clusters in $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$ using the WER of each speaker in it and for every cluster-specific RCT in $\mathcal{T}_2$.

lowest WER was used, and an unconstrained RCT was retrained for each new cluster, and the error analysis procedure just described was performed. However, the results from this analysis (Step 5 of Algorithm 2), shown in Table 5.3, do not exhibit patterns similar to those in Table 5.1, which indicates the existence of a more complex relationship between speaker cluster membership and performance obtained from cluster-specific RCT. The difference in the two sets of results suggests that speaker variability for MLLR adaptation strategies can be better modeled by speaker clustering in the eigenspace of MLLR transformation, than by clustering speakers based on minimum WER.

5.5 Analysis of Regression Tree Structure

On manually examining the cluster-specific RCT, it was observed that a different sequence of questions was used by each constrained RCT, as expected. Since the structure of the

Table 5.3: Oracle WER(%) for English CTS using reclustering and retraining the unconstrained RCT. Lowest WERs in each row are highlighted.

	Clust 1	Clust 2	Clust 3	Clust 4	SI
Clust 1	21.2	21.1	**21.0**	21.1	21.3
Clust 2	**21.9**	**21.9**	22.0	**21.9**	21.9
Clust 3	**23.8**	**23.8**	23.9	24.1	24.3
Clust 4	**21.4**	**21.4**	**21.4**	**21.4**	21.7

RCT describes similarities among clusters of phones (based on phone-level statistics), it can be conjectured that each cluster-specific RCT reflects dialect or pronunciation patterns that are representative of its cluster. Further, on comparing the constrained RCT, for each speaker cluster, it was found that the branches of the trees that split the acoustic units describing vowels (Figure 5.3) exhibited more differences in the hierarchical structure than the branches involving consonants, which is consistent with linguistic studies on regional variation in American English [55]. For example, the relative proximity of "ao" and "aa" in the upper tree of Figure 5.3 compared to that in the lower tree is perhaps indicative of dialectal variation. The unconstrained RCT had structures that were considerably different from those of the constrained RCT and, across clusters, exhibited more diversity in structure details than the constrained ones, as illustrated in Figure 5.4.

5.6 Soft Regression Class Trees

5.6.1 Maximum Likelihood Weights for Transform Combination

The experiments in Section 5.4 serve as proof of concept that choosing the best RCT can lead to improved ASR performance, where the optimal RCT is determined for a test speaker by evaluating ASR performance for every cluster-specific tree. However, for actual ASR evaluations, an unsupervised method is needed to determine the optimal tree to use for a test speaker. The following approaches are compared: i) choosing a single RCT using a maximum likelihood (ML) criterion vs. ii) estimating weights for a linear combination of the MLLR transformations, where the weights are estimated to maximize the likelihood of

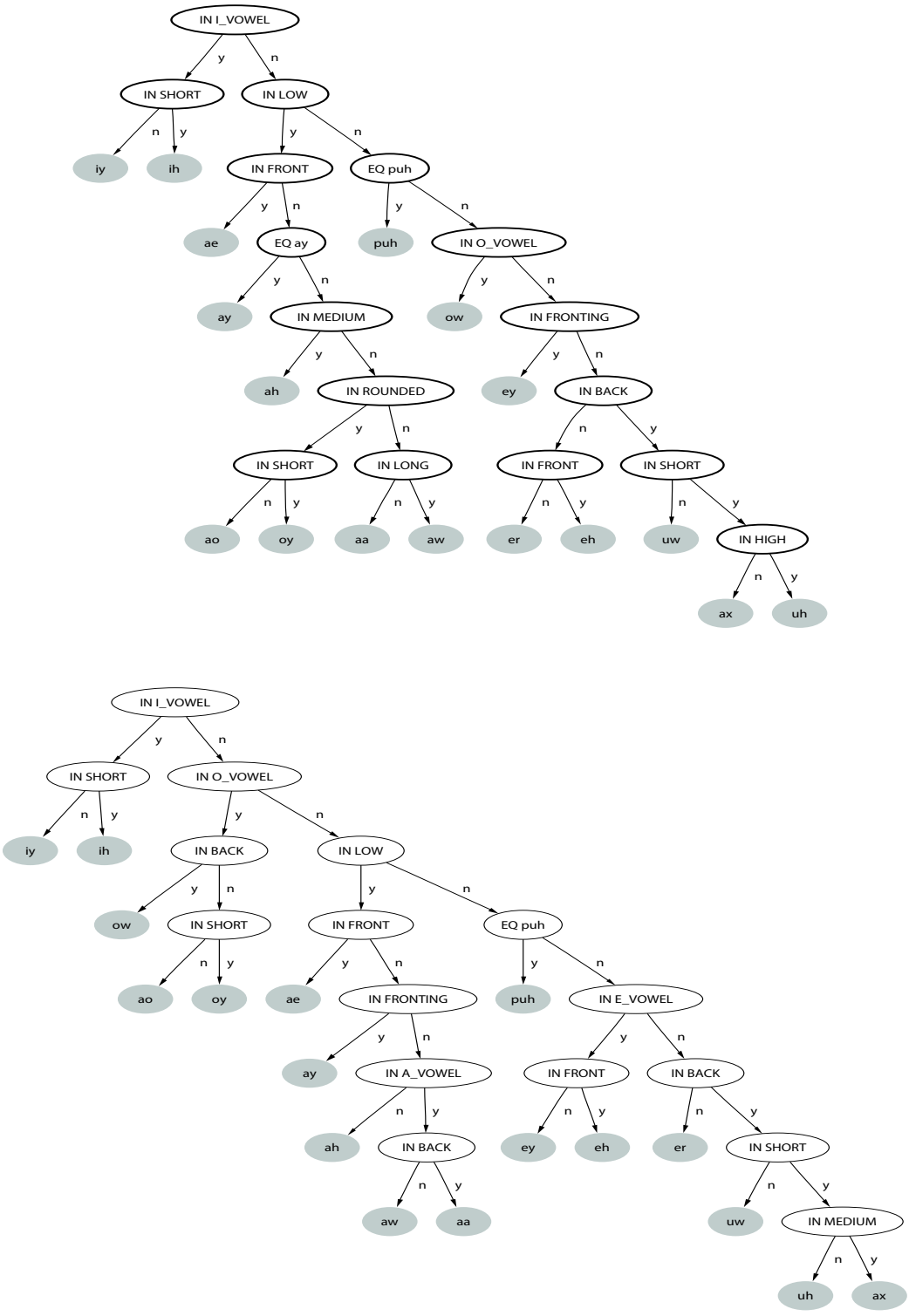

Figure 5.3: Vowel branches of the constrained RCT for two different English CTS Male clusters

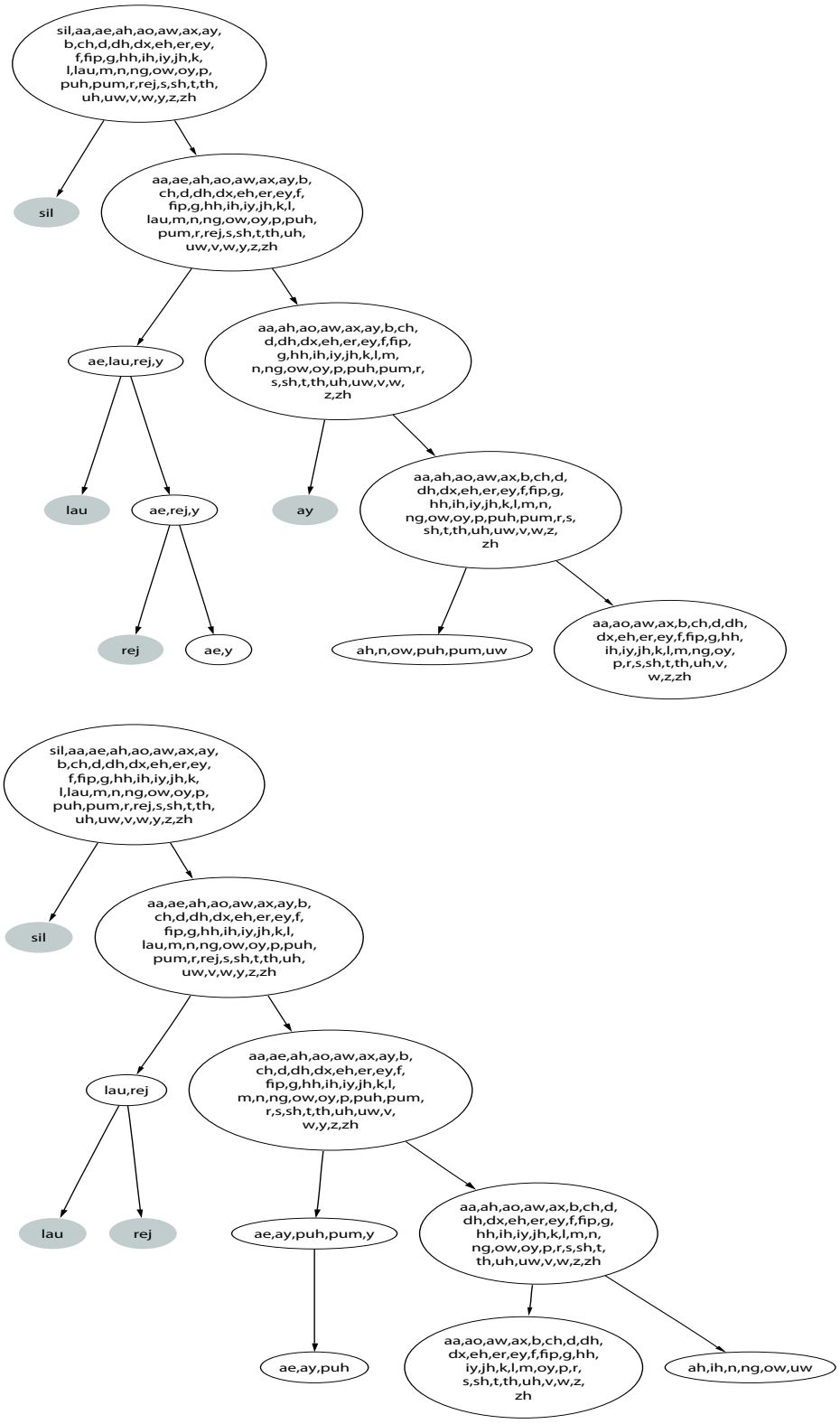

Figure 5.4: Top-levels of unconstrained RCT for two different clusters of English CTS (Female)

a speaker's adaptation data.

Combinations of multiple MLLR transformations using maximum likelihood (ML) weights have been proposed previously [8,31,91]. The component MLLR transformations have been estimated dynamically for test speakers using different nodes within a single RCT [31], or pre-computed for speaker clusters using various techniques [8], in both cases using a single RCT. The weights for combining the component MLLR transformations are estimated dynamically using an ML approach given a test speaker's adaptation data [8,31], or pre-computed using an ML approach on a corpus of training speakers [37]. The approach described here uses a test speaker's adaptation data to compute the component MLLR transformations, but relies on cluster-specific RCTs that are trained offline. The optimal weights to combine the component transformations are estimated to maximize the likelihood of the test data, as described below.

Define the transformed mean vector of the m-th Gaussian as

$$\hat{\mu}_m = \hat{\mathbf{M}}_m \hat{\alpha}^{(l)}$$

where

$$\hat{\mathbf{M}}_m = [\hat{\mu}_m^{(1)} \cdots \hat{\mu}_m^{(S)}], \quad \hat{\mu}_m^{(s)} = \hat{\mathbf{W}}^{(s,r)} \xi_m,$$

and $\hat{\mathbf{W}}^{(s,r)}$ is the transformation associated with the r-th regression class of the s-th speaker cluster on the extended mean vector ξ_m. The RCT used in this work clusters triphone states with the same center phone at leaf nodes. This implies that the l-th leaf node on each RCT corresponds to the same Gaussian distributions, and the weights for combining MLLR mean transformations can be tied at the leaf nodes across the S trees. The weights $\hat{\alpha}_s^{(l)}$ for the s-th RCT at l-th leaf node are represented by

$$\hat{\alpha}^{(l)} = [\hat{\alpha}_1^{(l)} \cdots \hat{\alpha}_S^{(l)}]^T$$

The auxiliary function of interest is:

$$\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = K - \frac{1}{2} \sum_{r=1}^{R} \sum_{c=1}^{C_r} \sum_{m=1}^{M_c} \sum_{\tau=1}^{T} \gamma_m(\tau) \Big(\mathbf{o}(\tau) - \hat{\mu}_m \Big)^T \Sigma_m^{-1} \Big(\mathbf{o}(\tau) - \hat{\mu}_m \Big) \ , \tag{5.1}$$

where K is a normalization constant, R is the number of regression classes containing C_r mixture Gaussian distributions, each of which has M_c component Gaussian distributions; $\mathbf{o}(\tau)$ is the observation vector at time τ; and $\gamma_m(\tau)$, $\hat{\mu}_m$ and Σ_m^{-1} are the occupation probability at time τ, mean vector, and inverse covariance of the mth Gaussian distribution, respectively. Using a procedure similar to that of [31], the weights $\hat{\alpha}_s^{(l)}$ can be estimated by first differentiating Eqn. 5.1 with respect to $\hat{\alpha}_s^{(l)}$, which results in Eqn. 5.2

$$\Big[\sum_{r=1}^{R} \sum_{c=1}^{C_r} \sum_{m=1}^{M_c} \sum_{\tau=1}^{T} \gamma_m(\tau) \hat{\mu}_m^{(s)T} \Sigma_m^{-1} \hat{\mathbf{M}}_m \Big] \hat{\alpha}^{(l)} = \sum_{r=1}^{R} \sum_{c=1}^{C_r} \sum_{m=1}^{M_c} \sum_{\tau=1}^{T} \gamma_m(\tau) \hat{\mu}_m^{(s)T} \Sigma_m^{-1} \mathbf{o}(\tau) \ , \tag{5.2}$$

which can be further represented as a set of simultaneous equations as shown in Eqn. 5.3

$$\mathbf{Z}^{(l)} \hat{\alpha}^{(l)} = \mathbf{V}^{(l)} \ . \tag{5.3}$$

where $\mathbf{Z}$ is a $S \times S$ matrix and $\mathbf{V}$ is a $S \times 1$ vector. Each row of $\mathbf{Z}$ and the corresponding element of $\mathbf{V}$ represents the left and right side, respectively, of Eqn. 5.2 that is obtained by differentiating Eqn. 5.1 with respect to each weight $\hat{\alpha}_s^{(l)}$.

The maximum likelihood solutions for the weights do not require constraints that the weights be positive or sum to one, i.e., as would be the case if they were interpolated as a mixture. However, to handle instances when numerical instability (e.g., inversion of matrix $\mathbf{Z}^{(l)}$ fails)[3]leads to bad estimates of weights, Lagrange multipliers, $\lambda^{(l)}$ and $\beta_s^{(l)}$, are

[3]If the structure of the cluster-specific RCTs are similar, then the rows of $\mathbf{Z}$ will be nearly identical (numerically) as computed using Eqn. 5.2. In such cases the inversion of $\mathbf{Z}$ may fail.

introduced into the objective function of Eqn. 5.1 to obtain Eqn. 5.4:

$$\mathcal{Q}(\mathcal{M},\hat{\mathcal{M}}) = K - \frac{1}{2}\sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\Big(\mathbf{o}(\tau)-\hat{\mu}_m\Big)^T\Sigma_m^{-1}\Big(\mathbf{o}(\tau)-\hat{\mu}_m\Big) + \sum_{l=1}^{L}\lambda^{(l)}(\sum_{s=1}^{S}\hat{\alpha}_s^{(l)}-1) + \beta_s^{(l)}(\hat{\alpha}_s^{(l)}\geq 0)\ . \tag{5.4}$$

To solve for the transform mixing weights $\hat{\alpha}_s^{(l)}$ a simple method is employed that explores only the possibilities that arise by setting one inequality constraint active at a time[4]to cover 2^S possible cases when there are S constraints. When a particular inequality constraint becomes active, its corresponding weight is set to zero and the remaining non-zero weights are solved under the constraint that they sum to one. Under this assumption, Eqn. 5.4 can be reduced to Eqn. 5.5, with the appropriate weights set to zero.

$$\mathcal{Q}(\mathcal{M},\hat{\mathcal{M}}) = K - \frac{1}{2}\sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\Big(\mathbf{o}(\tau)-\hat{\mu}_m\Big)^T\Sigma_m^{-1}\Big(\mathbf{o}(\tau)-\hat{\mu}_m\Big) + \sum_{l=1}^{L}\lambda^{(l)}(\sum_{s=1}^{S}\hat{\alpha}_s^{(l)}-1)\ . \tag{5.5}$$

Differentiating with respect to $\hat{\alpha}_s^{(l)}$,

$$\frac{\delta\mathcal{Q}(\mathcal{M},\hat{\mathcal{M}})}{\delta\hat{\alpha_s^{(l)}}} = \sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\Big(\hat{\mu}_m^{(s)T}\Sigma_m^{-1}\hat{\mathbf{M}}_m\hat{\alpha}^{(l)} - \hat{\mu}_m^{(s)T}\Sigma_m^{-1}\mathbf{o}(\tau)\Big) + \lambda_l\ . \tag{5.6}$$

Setting the right hand side of Eqn. 5.6 to zero,

$$\Big[\sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\hat{\mu}_m^{(s)T}\Sigma_m^{-1}\hat{\mathbf{M}}_m\Big]\hat{\alpha}^{(l)} = \sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\hat{\mu}_m^{(s)T}\Sigma_m^{-1}\mathbf{o}(\tau) - \lambda_l\ . \tag{5.7}$$

[4]Setting an inequality constraint active implies making it an equality

Eqn. 5.7 can be re-written as,

$$\mathbf{z}_i{}^{(l)}\hat{\alpha}^{(l)} = v_i^{(l)} - \lambda_l \ , \tag{5.8}$$

where $\mathbf{z}_i$ is the i^{th} row of the $S \times S$ matrix $\mathbf{Z}$ and v_i is the i^{th} dimension of vector $\mathbf{v}$. Solving for l^{th} Lagrange multiplier we get,

$$\lambda_l = \frac{\sum_{i=1}^{S} \left(\mathbf{z}_i{}^{-1} \cdot \mathbf{v} - \lambda_l \sum_{j=1}^{S} z_{ij}\right) - 1}{\sum_{i=1}^{S}\sum_{j=1}^{S} z_{ij}} \ . \tag{5.9}$$

The weight $\hat{\alpha}^{(l)}$ can be estimated from,

$$\hat{\alpha}^{(l)} = \mathbf{Z}^{(l)-1}\left(\mathbf{v}^{(l)} - \lambda_l\right) \tag{5.10}$$

This weight estimation is process is continued, till a set of non-zero weights are found that also satisfy the inequality constraint, i.e. they are less than one.

5.6.2 Backoff Strategy

For each test speaker, a two-step ML weight estimation procedure is applied. First, the mean and diagonal variance MLLR transformations are estimated for every cluster-specific RCT from HMM state occupancy statistics collected using the speaker's adaptation data and the unadapted SI acoustic model. Next, we determine the cluster-specific RCT that produces the highest gain in likelihood on the adaptation data using the acoustic model adapted by its corresponding MLLR mean and diagonal variance transformations. Then, using this adapted acoustic model, we reestimate the HMM state occupancy statistics, which are subsequently used for estimating the mean transformation smoothing weights, without any inequality constraints (first step) (as described in Section 5.6.1) and its corresponding diagonal variance transformation, and determine its likelihood gain on the adaptation data. If the gain is less than the best gain from the cluster-specific RCTs, we re-estimate the smoothing weights with inequality constraints (second step), and its corresponding diagonal variance transformation. Depending on the set of smoothing weights chosen, either from

the first or the second step, its corresponding combined mean transformations and diagonal variance transformations are used to adapt the SI acoustic model.

5.6.3 Computation Requirements

Compared to standard MLLR adaptation with a single transform, the computational cost of adaptation using multiple cluster-dependent RCT structures increases linearly with the number of clusters. With S clusters, there are $S + 1$ transformations estimated (including one for the SI RCT). For example, if there is only one iteration of MLLR estimation and the backoff strategy (Sec. 5.6.2) is not used, then the state occupancy statistics from the SI model are used for estimating transformations for each cluster-specific RCT, and the total computational cost of estimating the MLLR transformations is $S + 1$ times that of using only the SI RCT. As mentioned in Sec. 5.6.1, the ML weights are only estimated for those nodes in the cluster-specific RCTs that represent equivalence classes, in terms of the SI Gaussian distributions they cluster, such as the set of leaf nodes of each tree. The added cost of ML weight estimation for combining the transformations is negligible, since there are far fewer weights to estimate than transformation parameters. For example, for the *unconstrained* and *constrained* RCTs, which have as many leaves as phones in the SI acoustic model, one weight needs to be estimated for each leaf node of each cluster-specific RCT. For an ASR system with 45 phones and 4 cluster-specific RCTs the total number of weights that need to be estimated is 180. On the other hand a typical MLLR full mean transformation, for a 39-dimensional feature vector needs 1560 parameters to be estimated.

The memory requirements are twice that of standard MLLR, since two acoustic models are stored in memory: the SI acoustic model to estimate state occupancy counts, and an adapted acoustic model, obtained by applying cluster-specific MLLR transformations to the SI acoustic model and used to compute overall likelihood gain from the cluster-specific RCTs. Compared to approaches such as eigenvoices, that combine multiple models and need to store each component acoustic model in memory, our approach also has much smaller memory requirements. Since weight estimation requires computing likelihoods with each component model, and this dominates the computational costs in transform estimation, the

overall cost of the two approaches is similar.

5.7 Recognition Experiments with Soft Regression Trees

5.7.1 Baseline Regression Class Tree Performance

We first present the baseline ASR system performance after MLLR adaptation (as described in Sec. 5.3) when using the different types of RCTs trained for this work. As mentioned before we examined four different types of RCTs: *constrained*, *unconstrained*, *unconstrained state-level* and a *manually* designed tree (using knowledge of acoustic phonetics as described in Sec. 5.3). Tables 5.4 and 5.5 show the ASR system performance (WER) improvements obtained using each of these trees for the 2004 English BN and 2003 English CTS test sets respectively. In row 1 of Tables 5.4 and 5.5, "Unconstr." refers to the unconstrained type RCT, "Unconstr. State" refers to the unconstrained state-level type RCT, "Constr." refers to the constrained type RCT and "Manual" refers to the manually designed RCT. In the same tables, row 2 refers to the WER of the adaptation hypothesis ("Adapt Hyps") used for MLLR adaptation and row 3 refers to the WER after MLLR adaptation ("MLLR").

Table 5.4: WER(%) after MLLR adaptation using 4 different RCT building schemes on 2004 English BN test set

	Unconstr.	Unconstr. State	Constr.	Manual
Adapt Hyps	17.9	17.9	17.9	17.9
MLLR	15.9	15.9	15.9	16.0

Table 5.5: WER(%) after MLLR adaptation using 4 different RCT building schemes on 2003 English CTS test set

	Unconstr.	Unconstr. State	Constr.	Manual
Adapt Hyps	23.1	23.1	23.1	23.1
MLLR	21.5	21.4	21.4	21.4

The results presented in Tables 5.4 and 5.5 show that the performance improvements

obtained from each of the three types of automatically-built RCTs: *unconstrained*, *unconstrained state-level* and *constrained,* are similar. Among these three cases, the *unconstrained state-level* type RCTs have higher degrees of freedom (or number of leaf nodes) since each of its leaf nodes represent a triphone HMM state, compared to the *unconstrained* and *constrained* type RCTs, which have one leaf per phone[5]. It is hypothesized that the higher degrees of freedom will make the resulting RCT less stable than the phone-based trees and thus more sensitive to over-training and errorful adaptation hypothesis. In a pilot experiment with the 2004 English BN test set, we saw evidence of over-training with the *unconstrained state-level* type RCT that resulted in larger improvements in overall likelihood of the adaptation data, but lower ASR system performance improvements due to MLLR adaptation. Due to this, we will only focus on the *unconstrained* and *constrained* type RCTs for subsequent ASR experiments. We chose to explore both of these types because they lead to very different tree structures, which might impact the effectiveness of the multi-tree combination.

5.7.2 Average ASR Performance Results

To evaluate the performance of the two-step ML procedure for combining MLLR transformations from speaker-clustered RCT, its performance was tested on a range of standard NIST test sets. As mentioned in Section 5.4, for each domain, 4 cluster-specific RCTs were trained for each of three types: constrained, unconstrained, and unconstrained state-level types. The transformations were combined using the two-step ML procedure. In Tables 5.7, 5.8 and 5.9, the columns denoted by "Unconstr.", "Constr." and "Unconstr. state-level" refer to the unconstrained, constrained and unconstrained state-level RCT, respectively. The row labels refer to experiment configurations which are explain in Table 5.6.

Table 5.7 shows the results of experiments that were run with the NIST 2003 English CTS test set and the CTS-domain specific RCT. The results show small improvements of 0.1% to 0.2% (absolute) for both kinds of RCTs, using the two-step ML procedure, compared to using only one SI RCT. Only the improvement of 0.2% (absolute) for the configuration "Soft

[5]For the ASR system used in this work, there were 3129 HMM states (for the English BN system) while the number of phones was only 45.

Table 5.6: Various configurations for ASR experiments.

Experiment Name	Experiment Configuration
Adapt Hyps	WER of adaptation hypothesis
SI	speaker-independent, automatically-derived RCT
Soft SC (root)	multiple speaker-clustered RCT with MLLR transformation combination weights tied at the root of the cluster-specific trees (global weights)
"Soft SC (leaves)" and "Soft SC"	multiple speaker-cluster RCT with weights tied at the leaves of the cluster-specific trees
ML SC	cluster-specific RCT that achieves the highest likelihood gain on adaptation data
Soft SC + No adapt	cluster-specific RCTs with the an extra weight for the case of no adaptation
Oracle SC	cluster-specific RCT that achieves the lowest WER

SC (leaves)" using the unconstrained RCT is significantly different from the "SI" case ($p < 0.01$)[6]. The difference in performance associated with tying the weights, at the root vs. at the leaves of the cluster-specific RCT is not significant. However, given that speakers usually have enough data to estimate weights at the leaves, and this method gave slightly better results, we picked the leaf-based weight-tying configuration for all subsequent experiments in

[6]Unless otherwise noted, significance tests on recognition results use a matched pair sentence segment test.

other domains. The "ML SC" configuration did not achieve better performance than "Soft SC (leaves)", while the performance of "Oracle SC" confirms our observation in Section 5.4 that the overall WER can be reduced significantly by choosing the optimal RCT.

Table 5.7: WER(%) using speaker-clustered RCT for the 2003 English CTS test set

	Unconstr.	Constr.
Adapt Hyps	23.1	23.1
SI	21.5	21.4
Soft SC (root)	21.4	21.4
Soft SC (leaves)	21.3	21.3
ML SC	21.3	21.3
Soft SC + No adapt	21.4	**21.2**
Oracle SC	20.8	20.9

Similar experiments were run with the NIST 2004 English BN test set using the speaker-clustered RCT trained on the BN training data. The results are shown in Table 5.8, where we can see that the "Soft SC (leaves)" configuration is able to achieve small improvements over the baseline (manually designed) RCT and the SI RCT in the range of 0.1%-0.2% (absolute) for both kinds of RCT, though these are not statistically significant.

Table 5.8: WER(%) using speaker-clustered RCT for the 2004 English BN test set

	Unconstr.	Constr.
Adapt Hyps	17.9	17.9
SI	16.0	15.9
Soft SC (root)	15.9	15.9
Soft SC (leaves)	**15.9**	**15.8**
ML SC	15.9	15.9
Soft SC + No adapt	15.9	15.8
Oracle SC	**15.2**	**15.3**

The performance of the speaker-clustered RCTs was tested on the NIST 2006 Mandarin BN and 2005 BC (development) test sets using the Mandarin BN/BC ASR system. The results are shown in Table 5.9, where the constrained SI RCT achieved improvements of 0.3%

(absolute) for both test sets, compared to the baseline (manually designed[7]) RCT, which are statistically significant at the level of $p < 0.002$. The use of unconstrained speaker-clustered RCT ("Soft SC") results in improvements of 0.2% (absolute) for the NIST 2006 Mandarin BN test set (significant, $p < 0.012$) and 0.6% (absolute) for the 2005 Mandarin BC (dev) test set (significant, $p < 0.001$). The use of constrained speaker-clustered RCT did not result in significant improvements[8]. As in the case of the speaker-clustered RCTs for the English domains, the cluster-specific constrained RCTs for Mandarin exhibited differences mainly in the vowel branches, and the structure of the cluster-specific unconstrained RCTs shows more diversity compared to the constrained RCTs.

Table 5.9: WER(%) using speaker-clustered RCT for the 2006 Mandarin BN and 2005 BC (dev) test sets

	2006 Mandarin BN		2005 Mandarin BC	
	Unconstr.	Constr.	Unconstr.	Constr.
Adapt Hyps	9.6	9.6	22.6	22.6
SI	7.5	**7.7**	20.3	20.4
Soft SC	**7.3**	7.8	**19.7**	**20.4**
Soft SC + No adapt	**7.2**	7.8	19.7	20.4

Finally, we experimented with adding an identity MLLR mean transformation when estimating the ML weights. The identity transformation represents the case of "no adaptation" and is referred to as "Soft SC + No adapt" in Tables 5.7, 5.8 and 5.9. The motivation for these experiments is our observation in previous work [65,66] that 10-15% speakers have lower ASR system performance from using MLLR adaptation in the case of English CTS. This experiment is able to achieve small improvements (0.1% absolute) with the constrained type RCT for 2003 English CTS test set and the unconstrained type RCT for 2006 Mandarin BN test set, compared to the "Soft SC" case.

[7]The manually designed RCT had three leaf classes: vowels, consonants and nonspeech organized into a tree.

[8]The linguistic question set that is used for building the constrained RCT is not as richly developed for Mandarin as is the case for English, which provides an explanation for the better ASR performance when using the unconstrained RCT in the Mandarin case.

5.7.3 Performance Analysis of Two-Step ML Weight Estimation

As mentioned earlier, this work extends the weight estimation framework of [37] by introducing a two-step ML weight estimation procedure with or without inequality constraints, mainly for the purposes of numerical stability. We compared the performance of the two-step procedure to the one-step procedure that used unconstrained ML weights for combination of cluster-specific MLLR mean transformations. The results shown in Table 5.10 indicate that the two-step procedure indeed produces better ASR performance, compared to the one-step procedure in all cases. The one-step procedure is not stable, and performance is worse on average than when using a single SI RCT.

Table 5.10: Comparison of performance [WER(%)] using two-step or one-step ML weight estimation (2004 English BN and 2003 English CTS test sets)

	2004 English BN		2003 English CTS	
	Unconstr.	Constr.	Unconstr.	Constr.
SI	16.0	15.9	21.5	21.4
One step	16.6	16.5	22.1	22.0
Two step	15.9	15.8	21.3	21.3

5.7.4 WER Distribution Analysis

Since the overall gains from using speaker-clustered RCT are small, an error analysis was performed on the various experiments on the NIST 2003 English CTS and 2004 English BN test sets to investigate whether there are marked differences in the *distribution* of the WERs by speaker. Such an analysis is motivated by a desire for adaptation methods that give robust improvements across speakers, e.g., reducing the number of cases where MLLR adaptation hurts ASR system performance.

In Figures 5.5 and 5.6, the speakers in the NIST 2004 English BN (234 speakers) and 2003 English CTS (144 speakers) test sets were ordered by the duration of adaptation data.[9] In Figure 5.7 speaker-level analysis of WER change from adaptation, relative to the

[9]Total length of waveforms after segmentation.

unadapted case, is shown with the same ordering of speakers as in Figure 5.5, using both the SI RCT (left) and speaker-clustered RCT (right) for speakers in the NIST 2004 English BN test set. The points where relative WER change is greater than zero correspond to cases when MLLR adaptation hurts ASR performance. The plots in Figure 5.7 indicate that, contrary to expectations, not all speakers benefit from MLLR adaptation. Further the amount of adaptation data is not a good predictor of performance gains (or losses) from adaptation for a specific speaker, though there is a trend of increased variance of performance change from adaptation as the amount of adaptation data decreases. It can also be seen in Figure 5.7 that fewer speakers have performance losses (relative WER change > 0) when the speaker-clustered RCT is used in adaptation. Similar trends are observed for speakers in the NIST 2003 English CTS test set, though most speakers there have more than 100 seconds of speech, rendering the increased variance trend less clear.

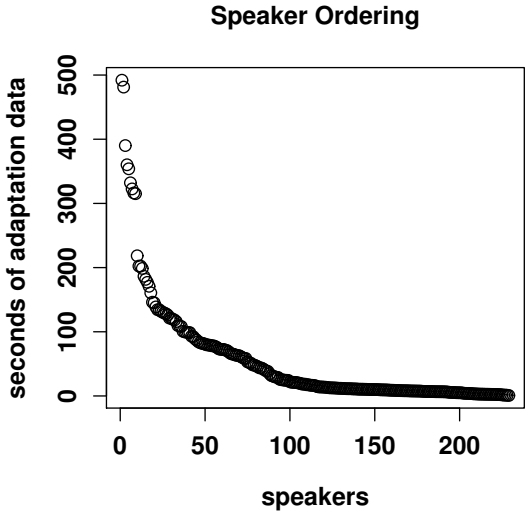

Figure 5.5: Speakers ordered by amount of adaptation data in seconds (2004 English BN test set)

Based on the error reductions with the oracle cluster-specific RCT in Section 5.4 and the trends seen in Figure 5.7 for English BN, it can be concluded that using multiple speaker-

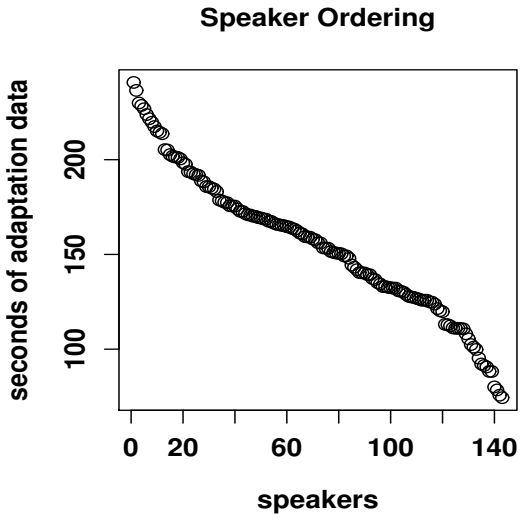

Figure 5.6: Speakers ordered by amount of adaptation data in seconds (2003 English CTS test set)

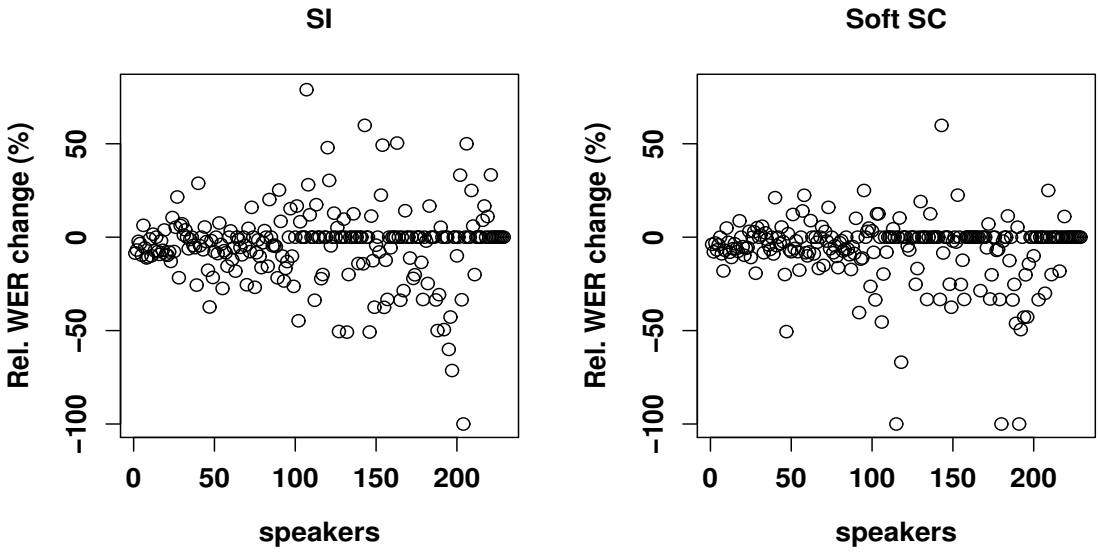

Figure 5.7: Relative change in WER for all speakers in the NIST 2004 English BN test set, ordered by decreasing amount of adaptation data.

clustered RCT leads to more robust adaptation strategies, compared to a single SI RCT.

Quantitative evidence for this conclusion is presented in Tables 5.11 and 5.12, which show the difference between the percentage of speakers that benefit from adaptation and the percentage of speakers that are hurt by it (the "net benefit" of adaptation), using different adaptation strategies for both the NIST 2004 English BN and the 2003 English CTS test sets. The different configurations are (rows 3 through 6): SI RCT (SI), the cluster-specific RCT that achieves the highest gain of likelihood on adaptation data (ML SC), multiple speaker-clustered RCT (Soft SC), and the RCT that achieved the lowest WER (Oracle SC). The net benefits are reported for different speaker subsets (columns 2 through 7): all speakers ("All"), speakers who achieve more than 5% relative WER reduction (or increase) from adaptation ("Rel. 5%"), and speakers whose WER reduction (or increase) from adaptation was significant at the level of $p < 0.15$. Denoting the WER for a speaker obtained using the SI RCT by p_{SI} and that obtained using any other configuration by p_X, the difference in WER is significant at the level of $p < 0.15$ if

$$p_X \notin [p_{SI} + \epsilon, p_{SI} - \epsilon]$$

where $\epsilon = 1.0364\sqrt{\frac{p_{SI}(1-p_{SI})}{n}}$ and n is the number of words spoken by the speaker. Note this is a simple, weaker significance test than the matched pairs test used with earlier results. A significance threshold of $p < 0.15$ was chosen since few speakers satisfy higher significance thresholds because the number of words for an individual speaker is small. Still, this is a stricter criterion for WER change than the simple relative difference of WERs.

It can be seen in Table 5.11, for both the NIST 2004 English BN and 2003 English CTS test sets, that the net percentage of speakers benefiting (or benefiting significantly at the level of $p < 0.15$) in the "Oracle SC" case is substantially higher than in the SI case, confirming the previous observation that the optimal RCT structure varies across speakers. Table 5.11 also shows that using multiple speaker-clustered RCTs, a greater net percentage of speakers benefit from adaptation, compared to both "SI" and the "ML SC" cases. In the case of English BN, the percentage of speakers significantly net benefiting from "Soft SC" is twice that in the "SI" case, while for English CTS the same difference is almost 30%

higher in the "Soft SC" case, compared to the "SI" case.

The distributional information for the speakers with significant performance differences for different adaptation methods is shown graphically in Figures 5.8 and 5.9 with the ordering of speakers the same as in Figures 5.5 and 5.6, respectively. Again, it is observed that fewer speakers are hurt by adaptation with "Soft SC". In Figure 5.8, which gives results for English BN, there is one speaker who shows a large relative performance loss (59%). On examining the performance patterns for this speaker, it is observed that while this is a "difficult" speaker for adaptation, it is not for ASR on the whole. The speaker's unadapted WER is 12.2%, compared to 19.5% using any of the RCT configurations that were experimented with. This speaker is particularly disfluent, but the unusually poor performance is most likely due to the fact that he is grouped in a cluster with another speaker that has a lot of background noise (keyboard clicks, etc.) that can negatively affect the adaptation transformations. This finding motivates some of the features explored in Chapter 7 for complexity control of MLLR adaptation.

The same analysis was performed for only those speakers who had less than 120 seconds of adaptation data in both the NIST 2004 English BN (190 speakers) and 2003 English CTS (24 speakers) test sets. The results, shown in Table 5.12, indicate that for speakers with less adaptation data, using speaker-clustered RCT in adaptation is again a better choice than both "SI" and "ML SC" cases. The impact is particularly notable for English CTS where the net percentage of speakers significantly benefiting from "Soft SC" is twice that in the "SI" case. This indicates that using speaker-clustered RCT with MLLR leads to ASR performance gains that are robust to cases with varying amounts of adaptation data, as is also evident from Fig. 5.8 and 5.9.

The information shown in Figure 5.7 is presented again in Figure 5.11, with the only difference being that the speakers are now ordered by their unadapted WER (ordering shown in Figure 5.10). The graphs in Figure 5.11 indicate that the unadapted WER is only a weak predictor of performance gains from MLLR adaptation, with correlation coefficients being 0.04 and 0.03 for "SI" and "Soft SC" cases, respectively. In Figure 5.12, only those speakers are shown whose relative WER change from adaptation is significant (at the level of $p < 0.15$) with the same ordering of speakers as in Figure 5.10. On computing the average

Table 5.11: Net benefit (%) analysis of all speakers in English CTS and BN

	NIST 2004 English BN			NIST 2003 English CTS		
	All	Rel. 5%	$p < 0.15$	All	Rel. 5%	$p < 0.15$
SI	24.1	18.2	5.0	67.2	38.5	24.5
ML SC	25.9	20.5	6.4	60.1	43.4	29.4
Soft SC	32.2	27.8	10.0	59.4	41.3	31.4
Oracle SC	62.2	54.1	18.6	87.4	67.1	44.8

Table 5.12: Net benefit (%) analysis of speakers with less than 120 seconds of speech in English BN and CTS

	NIST 2004 English BN			NIST 2003 English CTS		
	All	Rel. 5%	$p < 0.15$	All	Rel. 5%	$p < 0.15$
SI	22.1	15.3	4.2	41.7	25.0	12.5
ML SC	20.0	16.8	5.3	58.3	37.5	20.8
Soft SC	29.0	24.8	8.9	50.0	50.0	25.0
Oracle SC	58.4	50.5	16.3	75.0	50.0	33.3

relative performance loss (WER increase), for the speakers who have a loss in performance, it was observed that the case of using speaker-clustered RCT had a lower average performance loss (11% vs. 18 %), compared to the case of using the SI RCT.

5.7.5 Performance Analysis of ML weights

To understand the behavior of the ML weight estimation procedure we compared its performance when combining MLLR mean transformations estimated from the same RCT and when combining MLLR mean transformations from cluster-specific RCTs. We conducted ASR experiments with the 2004 English BN, 2006 Mandarin BN and 2005 Mandarin BC test sets and the SI unconstrained type RCT for each. The results of these ASR experiments are shown in Table 5.13. We first applied the data threshold on the amount of adaptation data to determine the regression classes in the SI RCT and the set of initial MLLR transformations to use. The results of using the SI RCT is shown in row 3 ("SI") of Table 5.13. Then, we explored two possibilities to smooth the initial MLLR mean transformations with

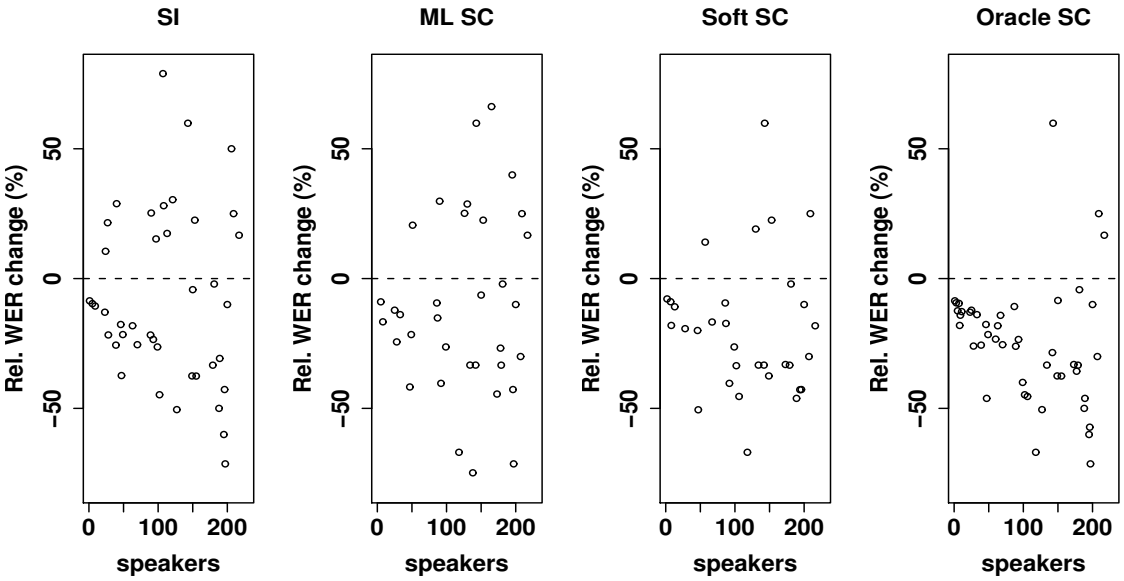

Figure 5.8: Significant ($p < 0.15$) performance changes of speakers from adaptation with various tree configurations (NIST 2003 English BN test set)

ML estimated weights: with mean transformations from one level up in the SI RCT, and with mean transformations up to two levels up in the SI RCT, which are shown in row 4 ("SI + One level") and row 5 ("SI + Two levels") respectively in Table 5.13 and row 6 ("Soft SC") refers to the results of using the cluster-specific RCT. For all three test sets, the "SI + One level" and the "SI + Two levels" cases do not show better performance over the "Soft SC" case and for the 2004 English BN test set the performance of "SI + One level" and ""Soft SC" are the same. More importantly, for the 2006 Mandarin BN and 2005 Mandarin BC test sets, the "SI + One level" and "SI + Two level" cases show lower performance improvements from MLLR adaptation, compared to the "SI" and "Soft SC" cases. Since larger relative performance improvements is achieved by the "Soft SC" case compared to the "SI" case for these the two Mandarin test sets it provides evidence that the performance improvements obtained by combining MLLR mean transformations from multiple cluster-specific RCTs using ML weights is due to the differences in RCT structures

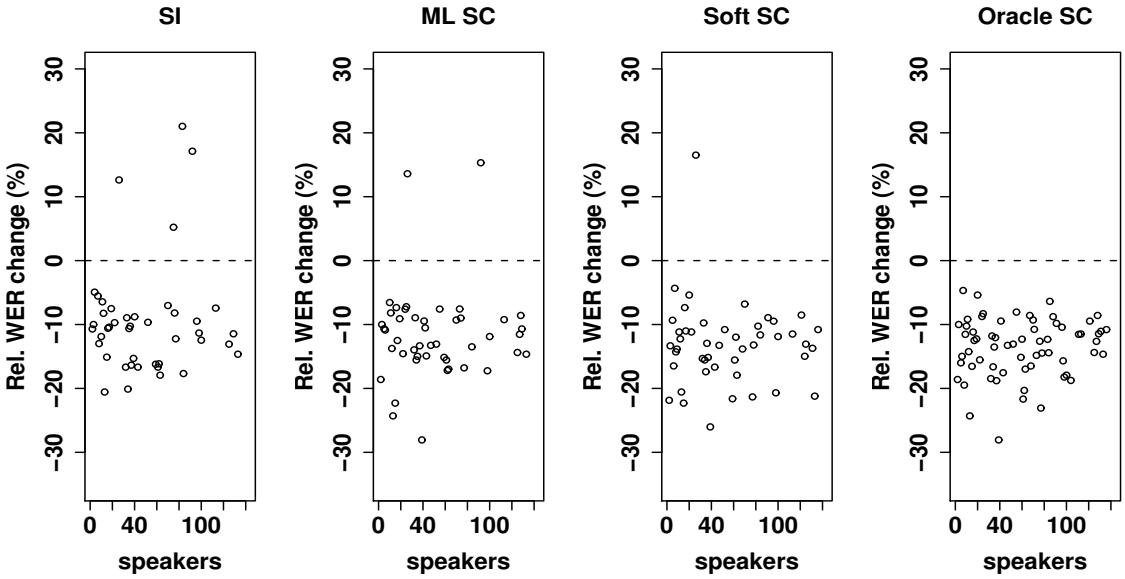

Figure 5.9: Significant ($p < 0.15$) performance changes of speakers from adaptation with various tree configurations (NIST 2003 English CTS test set)

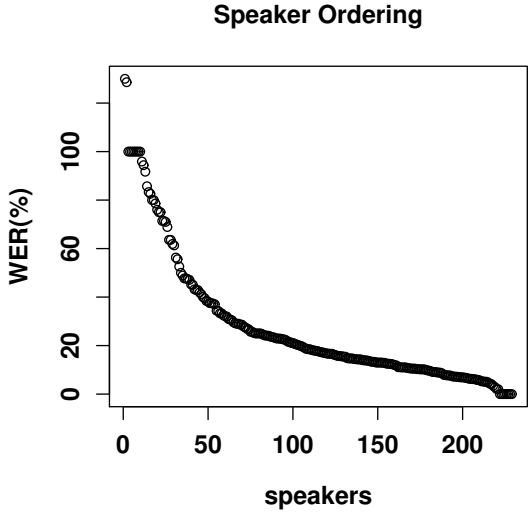

Figure 5.10: Speakers ordered by decreasing unadapted WER (2004 English BN test set).

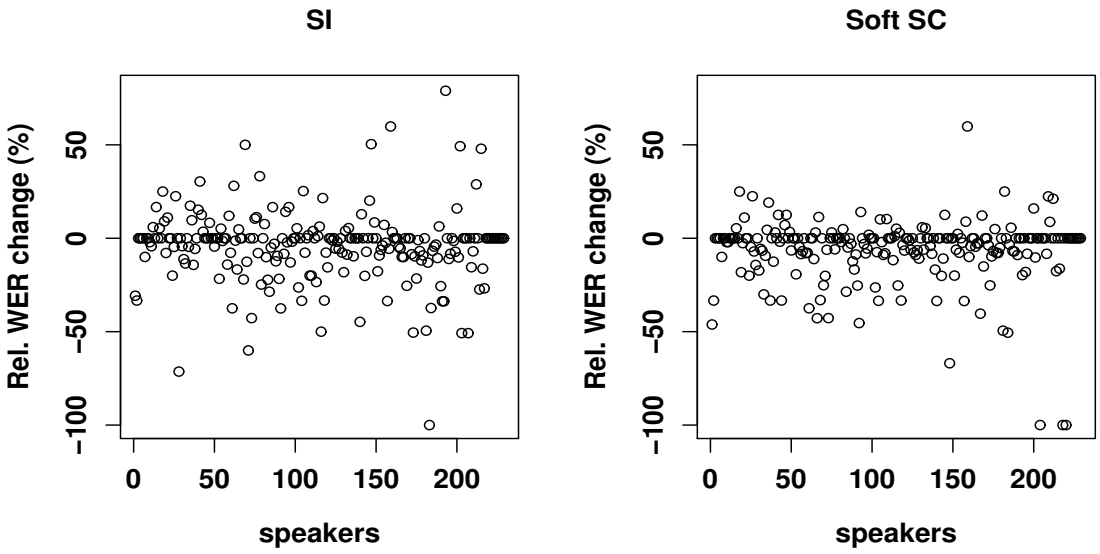

Figure 5.11: Effect of unadapted WER on adaptation success (2004 English BN test set); speakers are ordered by decreasing unadapted WER.

and not only due to the ML weights themselves. In the case of the 2005 Mandarin BC test set, the structures of the cluster-specific RCTs are perhaps able to capture variations of dialect or register[10] in conversations.

Table 5.13: WER(%) using ML weights to smooth MLLR mean transformations with those from higher nodes in the SI unconstrained RCT

	WER(%)		
	2004 English BN	2006 Mandarin BN	2005 Mandarin BC
SI	15.9	7.5	20.3
SI + One level	15.9	7.8	20.5
SI + Two levels	16.5	8.1	21.1
Soft SC	15.9	7.3	19.7

[10]Register is the formality of speaking situation and how familiar the speakers are with others.

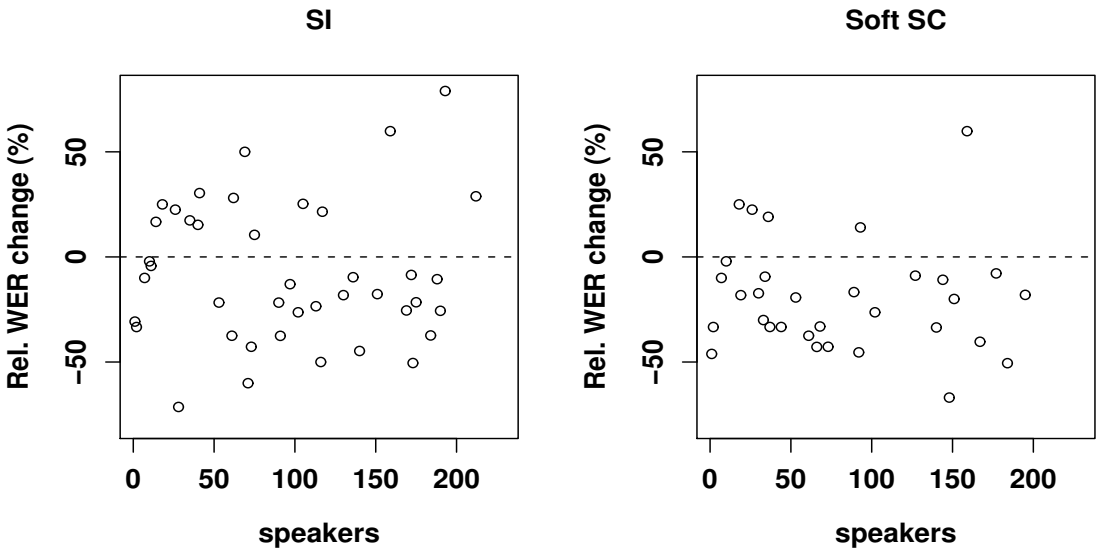

Figure 5.12: Speakers with significant ($p < 0.15$) performance change from adaptation (NIST 2004 English BN test set); speakers are ordered by decreasing unadapted WER.

5.8 Discussion

In MLLR-based speaker adaptation, the conventional approach to designing speaker-specific adaptation strategies is to use a global RCT for all speakers to decide on the regression classes (MLLR transformations) to use. Evidence is presented here that this approach sometimes leads to WER increases, and more robust performance across a population of speakers is possible by modeling speaker variability in designing speaker-specific adaptation strategies. A speaker clustering algorithm was introduced that models speaker variability by partitioning a large corpus of speakers in the eigenspace of their MLLR transformations, and captures the speaker variability information in the diversity of the structures of RCT trained for each speaker cluster. By choosing the optimal cluster-specific RCT to use for each individual test speaker, it is possible to achieve significantly lower overall WER, compared to the case where a global RCT is used, and there is also a smaller variance in error rates across speakers. On examining the different RCT structures produced, in the case of the

constrained RCT, it is noticed that more diversity was exhibited by the vowel branches than the consonant branches, which is conjectured to be indicative of dialectal variations in the training speaker population.

To take advantage of the speaker-clustered RCT in evaluating ASR systems, a procedure is developed that linearly combines MLLR transformations for a given speaker, estimated for each cluster-specific RCT, with weights that are estimated by maximizing the likelihood of the adaptation data in the framework of a two-step ML procedure that estimates weights with and without inequality constraints. The two-step ML procedure produces small improvements, compared to using only one SI RCT, for both English BN and CTS tasks, and larger improvements for Mandarin BN and BC test sets. Further, it is observed that the use of speaker-clustered RCT leads to ASR performance gains that are robust to the amount of adaptation data and the unadapted WER. As the amount of adaptation data decreases, regression classes are chosen higher up in the RCT (based on a given data count threshold), but the tying across phone classes differs depending on the RCT structure. This results in diverse MLLR transformations being linearly combined by the two-step ML procedure, and explains the robustness of WER gains from adaptation across a range of conditions. It is also observed that the speaker-clustered RCTs benefited a majority of the speakers who were hurt by MLLR adaptation with a single SI RCT, and reduced the average performance loss for those speakers who were hurt by MLLR.

In future work, it may be useful to relax the constraint that the speaker clustering data be disjoint from the acoustic model training set. While the strategy adopted in this paper avoids bias, it might turn out to be unnecessary in practice. By using a much larger speaker population in clustering, it is hoped to more diverse structures will be learned in the ensemble of RCTs, further improving robustness of the proposed method.

Chapter 6

TREE-LEVEL RCT COMPLEXITY CONTROL

The research presented in Chapter 5 focused on learning RCT structures for particular speaker clusters. Another important problem in the context of MLLR speaker adaptation is the online complexity control to determine the complexity of adaptation, or number of adaptation transformations, to estimate for target speakers. The standard solution for this problem is to descend down the regression class tree to those nodes that satisfy a predetermined threshold on the amount of adaptation data available in that node and estimate a transformation for each such node. In this chapter, evidence is presented that data-driven pruning does not yield the best regression class tree size (number of regression classes), and that significant gains in WER, compared to the standard approach, are achievable by choosing the oracle number of regression classes, which may include no adaptation. Previous work on online complexity control have investigated solutions that overcome the limitations of only considering amount of adaptation data - indirectly in [110] where a minimum description length (MDL)-based solution is proposed and directly in a cross-validation approach in [31] - but are computationally expensive. It is hypothesized that higher ASR performance improvements from MLLR adaptation can be obtained by a less costly predictor of regression class tree complexity based on features of the adaptation hypothesis. An automatic solution is proposed that predicts the best regression class tree size, if any, for a speaker by using standard statistical learning paradigms and speaker-level information sources that include acoustic-based and recognizer-based features. While the pilot study in Chapter 3 showed that speaker-level features are only weak predictors of potential ASR system performance gains from MLLR adaptation, the experiments were performed with a fixed size RCT. The work presented in this chapter investigates the possibility that these features might provide additional information over amount of adaptation data alone for predicting RCT size, i.e., the online complexity of adaptation.

6.1 Task and ASR System

The research results presented in this chapter is based on experiments on English CTS. The NIST benchmark test set corpus for English CTS was split into two parts: a training set, comprising the Switchboard and Fisher speakers in benchmark test sets of 1998, 2000, 2001, 2002, and 2003 (464 speakers total) and a test set, comprising the test set of 2004 (72 speakers total). The acoustic data in all these test sets was not used for acoustic model training purposes.

The ASR system used in this work was the full version of the English CTS system described in Sec. 4.1 and shown in Fig. 4.1. The system uses unsupervised MLLR in five decoding steps and exchanges adaptation hypotheses between the two different front ends – MFCC and PLP – to perform cross-system adaptation. For the purposes of description, the five instances of application of MLLR adaptation are split in two groups: the "early" stage comprising Steps 5 & 7 and the "later" stage comprising Steps 9(1,2,3). The final performance level of the system, is determined from the system combination outputs, after Step 9(1,2,3) and, in terms of WER it was 18.6% on the NIST RT 2004 English CTS test set.

6.2 Baseline Regression Class Tree

The ASR system used a baseline regression class tree that was manually constructed based on prior knowledge of acoustic phonetics. The tree, which has 9 leaf classes, is shown in Fig. 6.1, which employs a hierarchical back-off strategy, when less than 2 seconds of data is available for a regression class. However, a sufficient amount of data was available for the nine phonetic classes for most speakers, and the back-off mechanism was rarely needed. It is trivial to construct a tree with six clusters by pruning back some nodes in the tree.

6.3 Best Tree Sizes

Five different regression class trees, each with different number of leaves, were constructed. The tree sizes were 3, 6, 7, 9, or 11 leaves, which corresponded to the number of number of regression classes for each, since the data threshold was set low enough at 2 seconds. A final

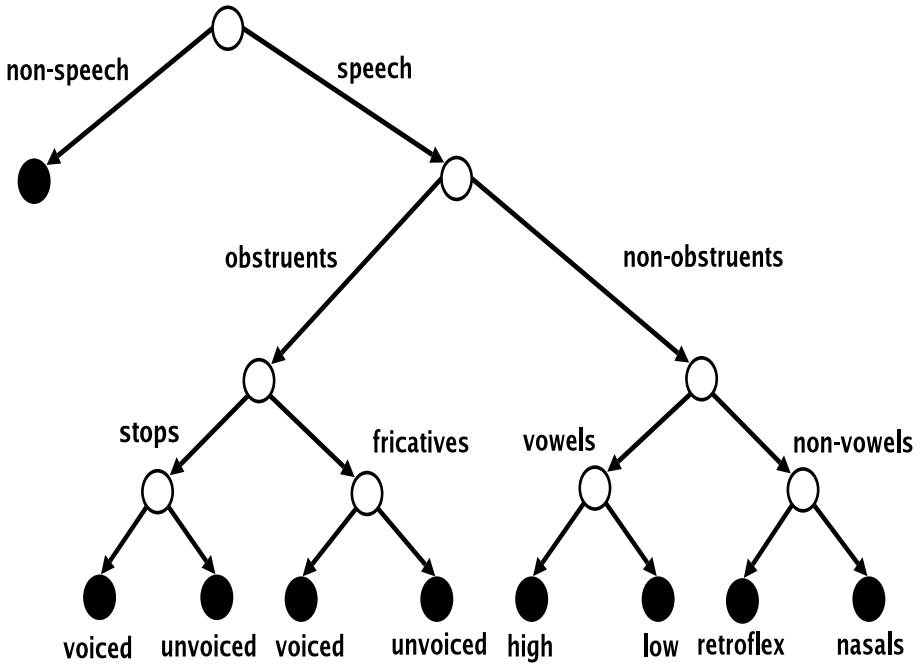

Figure 6.1: Regression class tree for phone clustering.

possibility is the use of unadapted speaker-independent models. The default case is to use the tree with nine regression classes in all stages in the ASR system, when MLLR adaptation is performed. Choosing a different size tree can be thought of as using a more sophisticated back-off to determine the transforms in addition to data-driven pruning. To determine the best tree size for each speaker, recognition experiments were performed for each available regression class tree size for each of the steps in the system that used unsupervised MLLR. Focusing on two NIST English test sets, Table 6.1 shows that significant gains in over ASR system performance can be achieved if the best regression class tree size can be determined for each target speaker.

Table 6.1: WER(%) with oracle regression class tree sizes.

Test Set	Default	Oracle
eval2004	18.6	17.4
eval2003	18.9	17.9

Analysis of the speakers in the training portion of the corpus provides clear evidence that the best regression class tree size for speakers varies, as shown in the histograms in Figure 6.2. The two graphs show the distribution of oracle regression class tree sizes in the early (left) and later (right) stages. Not surprisingly, the distribution weight for higher sizes increases in the last stage because of higher-quality of the adaptation hypotheses. Also evident from these histograms, and consistent with observations in the previous chapter, is the fact that several speakers have the lowest WER when they are recognized using unadapted models (0 classes).

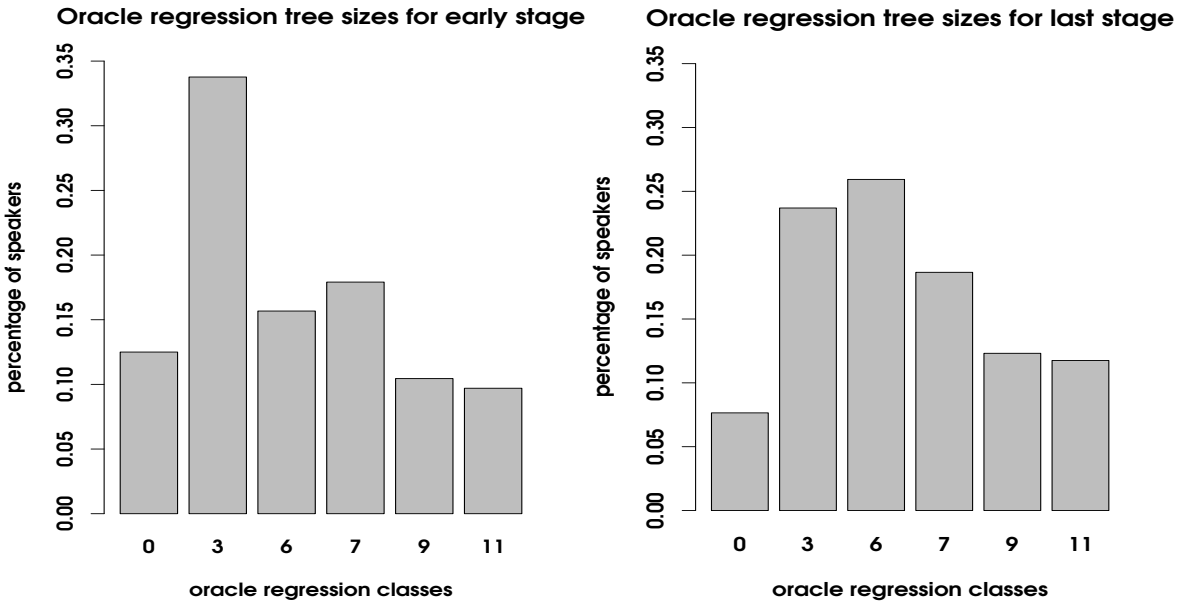

Figure 6.2: Distribution of oracle tree sizes.

To further understand the variability among speakers, they were clustered into groups in which speakers have similar relative gains (or losses) from different regression class trees. Each speaker was associated with a vector of relative WERs, normalizing the rate for each possible regression class tree with that obtained by the default regression class tree. They were then clustered using k-means with the number of clusters fixed at 5. Figure 6.3 shows the mean vector of relative WER change for the speakers in a given cluster, from the case

using unadapted acoustic models through every available regression class tree. For speakers in clusters 1 and 4, the larger trees (9, 11) are best; for cluster 2, the mid-size trees (3, 6) are best; for cluster 3, there is no advantage to any size; and for cluster 5, no adaptation is the best strategy.

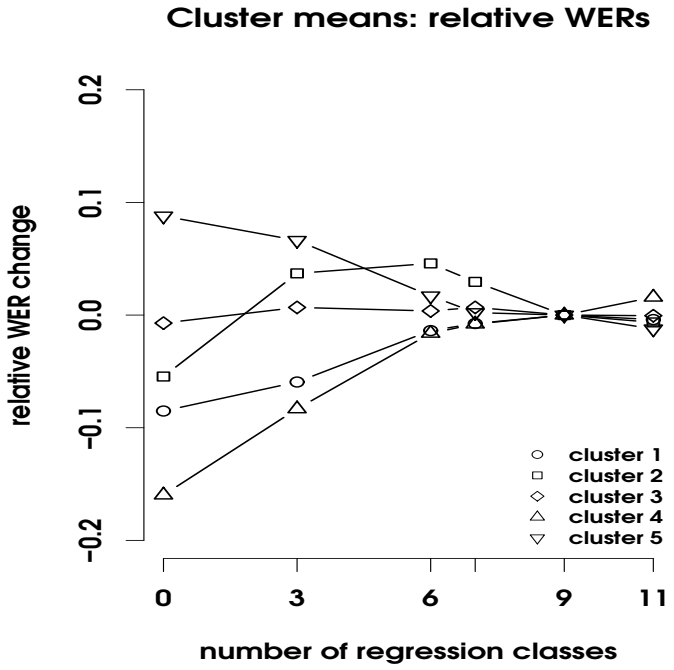

Figure 6.3: Mean relative change in WER (compared to default) for speaker clusters over different tree sizes.

6.4 Prediction of Tree Size

With evidence of potential performance gains from tuning the regression classes, the next step was to automatically predict the regression class tree size for each speaker, based on information observed in the adaptation data. The prediction problem could be formulated in three possible ways: *classification* (tree size is one of the possible six cases), *regression* (tree size is predicted as a real number), and *classification based on cluster membership* as shown in Figure 6.3. A significant number of speakers had multiple local maxima (i.e.,

WER change could not be ordered by tree size); hence, the regression approach was not suitable. An experiment was performed using perfect cluster classification, but it produced a gain of only 0.2% in WER compared to around 1%, which can be achieved by using perfect regression class tree size. Based on this evidence, it was decided to classify each speaker into one of six possible regression class tree sizes shown in Figure 6.2.

Several speaker-level features computed from the adaptation data were investigated:

- acoustic scores per-cluster in RCTs,
- seconds of speech per-cluster,
- average word-based confidence scores of adaptation hypothesis (from system combination),
- normalized energy measure (per frame),
- vocal tract length normalization factor for MFCC and PLP front ends, and
- rate of speech (in phones per second).

For each of the per-cluster features, there were 11 scores, one for each node in the 11-leaf tree. The seconds of speech per cluster feature was used since it is the standard back-off criterion in the case of insufficient adaptation data. Word-level confidence features have been found to be useful in improving performance of MLLR-based adaptation in [102]. This provided the motivation for using confidence scores in this work. However, the word-level confidence scores were averaged over a speaker to compute a speaker-level confidence measure.

After the raw features are extracted from adaptation data, they were processed as follows. Each training sample represented a speaker and was labeled with the best regression class tree size for that speaker. To compensate for the small number of training samples available, only 464 speakers, the second best regression class tree size was added to the set of training labels, if the WER was not significantly worse than the best regression class tree. Thus, each

training sample could have as many as two training labels. In addition, to counter the lack of training data, bagging with replacement was used for selecting training samples. Since the number of samples from each class was the same in the bagged training set, the final classifier posteriors were renormalized with the priors seen in the examples in the training set. Next, dimensionality reduction was performed on the training feature vectors using PCA followed by LDA and used the resulting features to train standard statistical learners.

A 4-fold cross-validation training paradigm (1998+2000, 2001, 2002, 2003) was used, designing classifiers on three sets and tuning parameters on the fourth validation set. For each of the classifiers, the number of samples to use from each class for bagging and the number of PCA components were decided by the accuracy on the validation set. The best configuration to use was chosen by varying the bag size per class from 25 to 75 in steps of 25 and the number of PCA components from 10 to 35. The LDA transformation always produced a 5-dimensional feature vector, since the number of possible classes was six. The ensemble of classifiers from the cross-validation partitions were combined to form a stacked learner. The posteriors from each of the classifiers in the stacked learner were then averaged to obtain the final class posteriors.

Several statistical learning paradigms were explored including decision trees, support vector machines, k-nearest neighbor and multinomial neural networks. Decision trees were found to perform best, although the performance of the other methods was not significantly different. The overall classification error rate obtained was in the range of 55%-64% for each of the held out sets. The relative reduction in classification error rate compared to chance error was in the range of 4%-20%.

Table 6.2 shows the percentage of times a particular class of features was used by the decision trees in training classifiers when the allowed subsets of features were recognizer-independent (seconds of speech, VTL, energy, and rate of speech measures), recognizer-dependent (acoustic scores and confidence measures), or all features. Acoustic scores and seconds of speech per cluster are used more frequently than other features. This observation could also be explained by the fact that the number of dimensions representing these two classes of features was much higher than the others, most of which were represented by one dimension.

Table 6.2: Features usage in the decision trees that were trained on different feature subsets.

	% of questions		
Features	Rec. indep	Rec. dep	All
Acoustic Scores	-	85.7	45.2
Seconds of speech	78.2	-	37.2
Confidence	-	14.3	8.6
VTL	10.4	-	5.4
Energy	5.7	-	2.2
Rate of speech	5.7	-	1.4

6.5 Recognition Experiments

Various combinations were experimented with in predicting the regression class tree size for individual speakers in the NIST RT 2004 test set using the decision tree described in Section 6.4. In Table 6.3, column 2 is the case where a single prediction of regression class tree sizes is used throughout the system (auto1); column 3 is the case where two different sets of predicted regression class tree sizes are used: one each for the early and later stages (auto2); and column 4 is the case where predicted regression class tree sizes are used only in the later stages (auto3). The results of recognition experiments are shown in rows 10-12 in Table 6.3 for different feature subsets, which correspond to columns 2-4 in Table 6.2. Compared to the baseline (18.6% WER), gains are seen for all cases where predicted regression class tree sizes are used. However, compared to the case when oracle regression class tree sizes are used (17.4% WER), there is substantial room for further improvement. The strategy that uses different predicted regression class tree sizes for the early and later stages (auto2) produces the best improvement in WER, an absolute 0.4% compared to the baseline, using recognizer-dependent features, which is statistically significant at the level $p = 0.002$ according to the matched pair sentence segment test. However, differences relative to using a single tree prediction are not significant. Contrary to trends in the oracle case, the average predicted regression class tree size was 7 for the early stage and 3 for the later stage.

Table 6.3: Results of using predicted regression class tree sizes with features from steps X and Y (PX+Y)

	auto1	auto2	auto3
step 5	P2+6	P2+6	9
step 7	P2+6	P2+6	9
step 5+7	P2+6	P2+6	9
step 9(1)	P2+6	P5+7	P5+7
step 9(2)	P2+6	P5+7	P5+7
step 9(3)	P2+6	P5+7	P5+7
	WER(%) for Eval2004		
Default	18.6	18.6	18.6
Rec. indep	18.3	18.3	18.4
Rec. dep	18.3	**18.2**	18.5
All	18.3	18.3	18.4
Oracle	17.4	17.4	17.4

6.6 Discussion

This work shows that significant improvement in WER can be achieved by selecting the correct size of regression class trees for individual speakers, including the possibility of no adaptation. Initial efforts at developing an automatic procedure to predict the regression class tree sizes have yielded modest improvements in WER. Analysis of the features used for prediction shows that acoustic scores of adaptation data along with amount of adaptation data available are the most useful features. A limitation of this work is that the RCTs are not automatically derived, as in more recent versions of the system, and that a fixed progression of tree cuts is specified for the different sizes. In the next chapter, we look at the more flexible methods for determining tree size, using node-level pruning with automatically derived trees.

Chapter 7

NODE-LEVEL COMPLEXITY CONTROL IN RCT

7.1 Introduction

In Chapter 6, it is shown that the standard minimum data count approach is not an optimal solution for online complexity control of RCTs, and better strategies can be designed by using higher-level speaker-dependent features in addition to the amount of data in a node to predict tree size - the effective number of regression classes to use for a target speaker. The solution proposed in Chapter 6, determined RCT complexity for the tree as a whole by using a predictor that chose from among a small set of possible tree configurations (e.g. 0-11 leaves) based on speaker and adaptation hypothesis characteristics. The work presented in this chapter, extends this idea to predict node-level pruning (or extension), allowing a greater number of tree configurations with a very simple classifier. In addition, new features are investigated for the predictor, in part motivated by problems that arise in adapting segments of broadcast news, where the hypothesized "speaker" is based on automatic clustering of speech segments. Using these features with standard statistical classifiers, a prediction is made for a new regression class to use, corresponding to pruning or growing a branch of the RCT. In addition, a scheme is described that applies the proposed complexity control scheme simultaneously to all regression classes that leads to improvements in robustness of ASR system performance for English broadcast news on recent NIST evaluation test sets.

7.2 Task and System Description

The ASR system used in the this work was the English BN system of Sec. 4.2. It uses unsupervised MLLR ("full" mean and diagonal variance transformation) once as shown in 4.3. The rest of the details of these systems have been described in Chapter 4.

The corpus used in this work is comprised of five NIST English BN development and evaluation data sets released between 2003 and 2004: dev2003, dev2004-ldc, dev2004-tdt4,

eval2003 and eval2004, which totaled approximately 12 hours of North American broadcast news shows.

7.3 Classifier for Complexity Control

The goal of this work was to improve on the standard adaptation data threshold-based approach for complexity control by using a standard statistical classifier and higher-level features to predict the optimal regression class to use. We first start with the constrained RCT described in Sec. 5.2.1 and an initial set of regression classes that are determined by the standard data-threshold approach. Then we apply a classifier to each initial regression class, using higher-level features specific to that class, to choose a neighboring regression class that is (ideally) more appropriate for MLLR adaptation transformation estimation. The tasks involved in designing the classifier were: deciding on the training labels, deciding on the classifier objective function, and choosing the higher-level features to use, which we describe in the next two sections.

7.3.1 Training Labels for Classifier

The most appropriate training label for the classifier would be the node label of the regression class that leads to the greatest improvement in ASR system performance after MLLR adaptation. However, even after starting with an initial set of regression classes, the number of possible new sets of regression classes that need to be explored is very large. To simplify the number of possibilities to explore, each initial regression class is examined separately, keeping others fixed, and considering only its neighboring regression classes in the branch of a tree as alternatives. This scheme is illustrated in Fig. 7.1. Four possibilities are explored for each initial regression class: *move up*, *move down*, *stay at current position* or *no adaptation*. Next, the MLLR adaptation transformations are estimated for each case and used to adapt the SI Gaussian distributions in the initial regression class under consideration, and also adapt the rest of the SI Gaussian distributions with the MLLR transformation estimated with their respective initial regression class. Then, recognition is performed using the adapted acoustic model, and the WER is computed. The possibility that leads to the best improvement in WER, is chosen as the training label for the classifier. Thus, there

were four possible training labels: -1 (move down), 0 (stay unchanged), 1 (move up) or X (no adaptation).

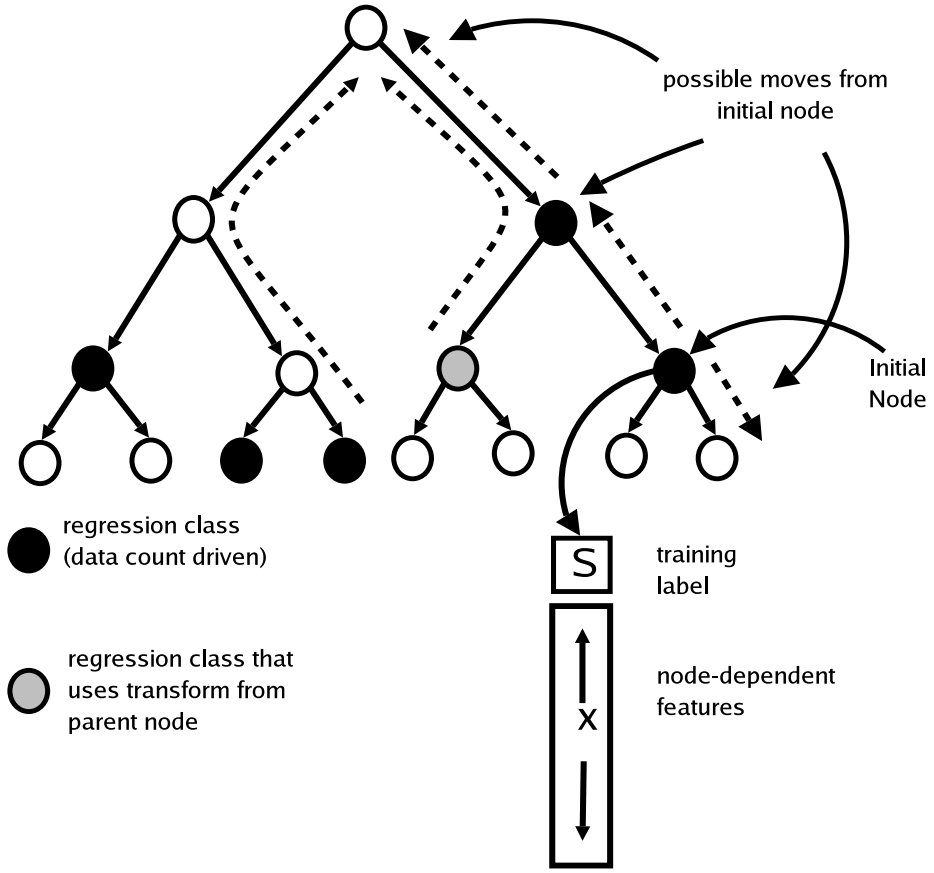

Figure 7.1: Determining training labels for classifier

In the process of generating the training labels for the classifier, it is observed that, for many regression classes, there is more than one training label category that produced similar ASR system performance levels. This ambiguity in training is represented by grouping together instances of training cases which produce similar ASR system performance (at the level of $p = 0.3$) to form a new training category.[1] The approach taken to compute if two different WER, obtained with two different regression classes, were significantly different

[1]While $p = 0.3$ seems high, many instances are still grouped because the small sample size in each case means differences must be quite large to be highly significant.

is the same as described in Sec. 5.7.4. For example, all regression classes for which *move up* and *stay unchanged* produce the similar WER, are assigned to a single category. As there were 4 initial training label categories, all 2^4 cases of the different training categories producing equivalent WER are considered. Then three different training label assignment schemes are designed:

- *Smallest Tree*: In this case, a new training label category is chosen, from the initial categories, that is biased towards predicting a smaller tree or lesser degree of adaptation. For example, the group of regression classes for which move up and stay unchanged produce equivalent WER, we choose to use move up as the new training label since it leads to smoother MLLR transformations. (conservative approach)

- *Largest Tree*: In this case, a new training label category is chosen, from the initial categories that is biased towards a larger tree or higher degree of adaptation. (aggressive approach)

- *Regression*: In this case we assign a new numerical target, equally-spaced between -1 to 1, to each of the 24 new possibilities and perform regression, instead of classification. This case effectively involves predicting a larger number of possibilities.

The above the three schemes are shown in Table 7.1, where each row lists a one of the possible 2^4 cases of the four possible training labels having equivalent WER[2]; columns 1 through 4 represent the four possible training labels; ; column 5 the percentage of training samples for each row; and columns 6 through 8 ; represent the three different labeling schemes.

7.3.2 Features

Several different features were considered that were dependent on a given regression class for use with the classifier. The features were computed at the phone-level from the adaptation

[2]For example 0 1 1 0 represents the case that *move up* and *stay unchanged* have similar WER (at the level of $p = 0.3$)

Table 7.1: Training Label assignments and distribution of training samples pooled from five English BN test sets

Training Labels							
X	1	0	-1	% samples	Regression	Smallest Tree	Largest Tree
1	0	0	0	9.7	-1.00	X	X
1	1	0	0	3.9	-0.85	1	1
1	0	1	0	3.4	-0.71	0	0
1	1	1	0	8.3	-0.57	1	0
0	1	0	0	2.4	-0.43	1	1
1	1	0	1	0.7	-0.29	1	-1
0	1	1	0	4.8	-0.14	1	0
1	0	0	1	0.9	0.00	-1	-1
1	1	1	1	41.9	0.14	1	-1
1	0	1	1	2.3	0.29	0	-1
0	0	1	0	7.0	0.43	0	0
0	1	1	1	6.5	0.71	1	-1
0	0	1	1	3.7	0.85	0	-1
0	0	0	1	3.8	1.00	-1	-1

hypothesis and aggregated at the level of the regression classes. The broad categories of these features explored are listed in Table 7.2, where column 1 lists the feature category and column 2 provides the assumptions this work makes about how the degree of adaptation (smoother or more detailed transformations) is correlated with increasing (or decreasing) levels of each feature. For example, it can be argued, that as the signal-to-noise ratio becomes higher, detailed transformations may not be effective and smoother ones may be a better option. For rate-of-speech, likelihood of adaptation data, frequency of different phones spoken and amount of adaptation data, it can be argued that increasing levels of any of these features may create conditions where more detailed transformations are more suitable. In addition, the clustering score from the unsupervised speaker clustering algorithm that determines the pseudo speaker-labels was also considered as a feature, with the idea that this might indicate clusters with multiple speakers, where less adaptation might be better. Evidence is provided next, that shows the above reasoning holds for some of the features. For each of these feature categories the mean, standard deviation and

entropy were computed from the phone-level statistics within each regression class.

Table 7.2: Feature categories used for predicting adaptation complexity.

Feature Categories	Assumed correlation with degree of adaptation
Amount of adaptation data	+
Rate of speech (ROS)	+
Signal-to-noise ratio (SNR)	−
Likelihood of adaptation data with SI acoustic model	+
Frequency of different phones	+
Clustering score of unsupervised speaker clusters	−

The training labels were generated and the classifier features (zero-normalized) for all the acoustic data in the five different test sets. Next, each regression class was assigned to its training label category and histograms were generated for four different classifier feature categories shown along the columns in Fig. 7.2: amount of adaptation data (column 1), ROS (column 2) and entropy of phone frequency (column 3) for each regression class. Each row in Fig. 7.2 corresponds to one training label category, and the plots are ordered in descending order of "degree" of adaptation: move down (row 1), stay unchanged (row 2), move up (row 3) and no adapt (row 4). The categories are referred to in terms of degree of adaptation since moving up the tree involves using smoother MLLR transformations (more sharing) and moving down the tree obtains more detailed transformations. It can be seen in column 3, that classes with lower entropy of the frequency of the different phones

spoken are better handled by using less adaptation. There is also a weak trend of classes showing lower ROS tending to need a decreasing degree of adaptation. Normalized ROS close to -1 correspond to slower speakers and correspondingly normalized ROS close to 1 indicate faster speakers. Column 1 in Fig. 7.2 shows that amount of adaptation data is an indicator of degree of adaptation, since classes with a large amount of data benefit from more detailed adaptation (row 1), but the trend is weak. The empirical evidence in these examples supports the hypothesis that other features, besides amount of adaptation data, may be useful for predicting complexity control of regression class trees.

7.3.3 Classifier Performance

The classification tasks used standard support vector machine (SVM)-based classifiers and the regression used support vector regression (SVR) [95], provided by the implementation in [13]. Three of the five test sets were used for evaluation purposes: dev2004, eval2003 and eval2004. For each test set, we used the other four test sets as the training set. The support vector machines used a radial basis function kernel and its parameters were determined by five-fold cross validation for each training set. The classification error rates of the SVM classifiers and the corresponding chance[3]error rate , for the *smallest tree* and *largest tree* classifier designs, and the root mean square error (RMSE) of the SVR and its corresponding RMSE for predicting the mean target, are shown in columns 2 and 3 in Tables 7.3, 7.4 and 7.5 respectively. Since the relative WER change for each possible training label for every regression class was available, the estimated relative change in WER was computed for the predicted case, the chance (for classification) or mean (for regression) case, and the oracle (best WER improvement) case. The estimate for a test set and particular condition is found by weighting the relative error associated with a class by the number of words in the class and averaging over all classes. These figures are shown in columns 3, 4 & 5 in Tables 7.3, 7.4 and 7.5. The performance of the SVM classifier is better than predicting chance in all cases as is the estimated relative WER change, and the labels based on the smallest tree (conservative tree size) lead to the biggest gains in estimated WER. The WER improvement

[3]We use the term "chance" to refer to predicting the most frequent class.

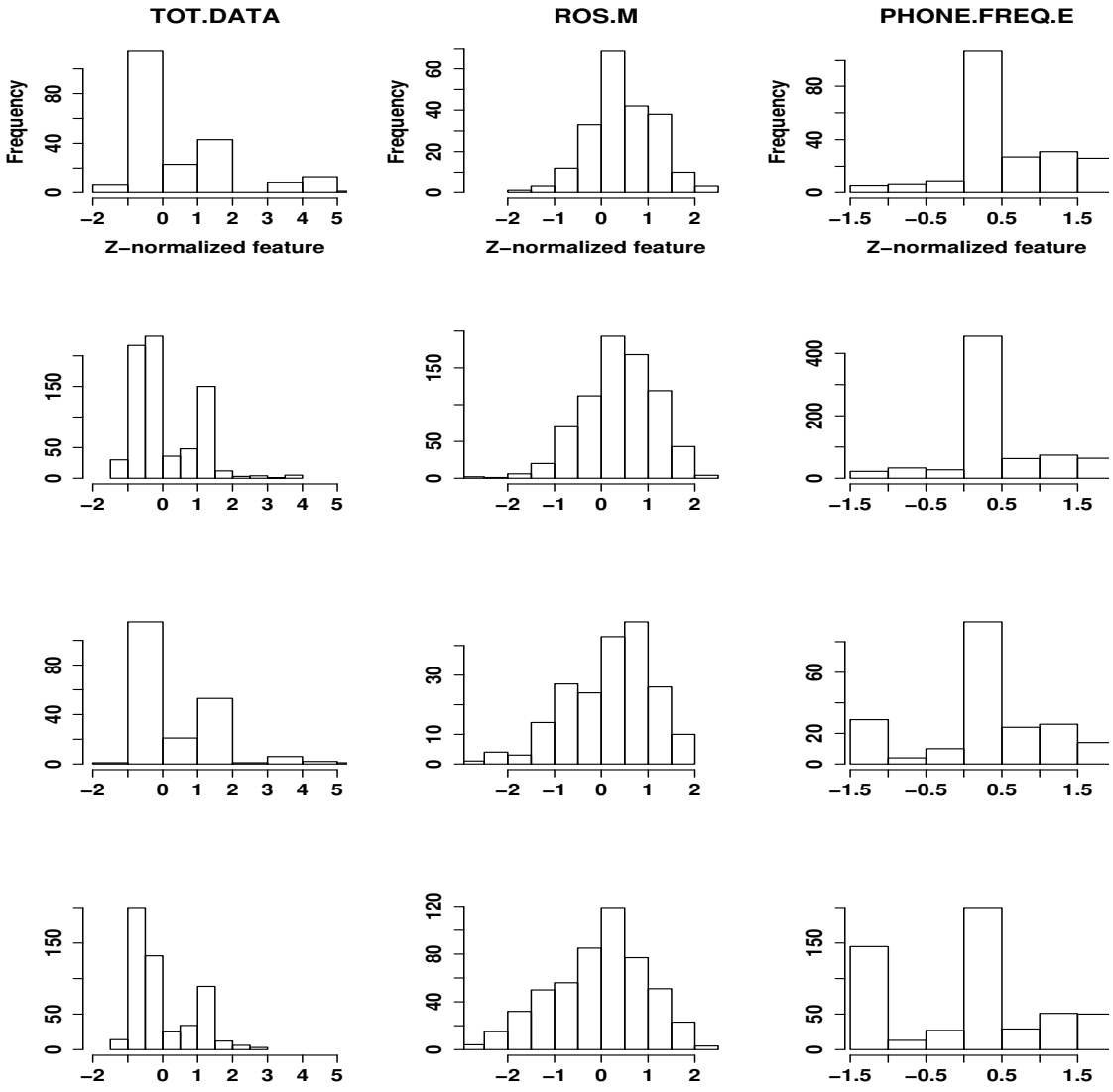

Figure 7.2: Four feature classes with rows in descending order of "degree" of adaptation: move down (row 1), stay unchanged (row 2), move up (row 3) and no adapt (row 4).

in the oracle case is higher in most cases, compared to the classifier, though still not large. In the case of using the SVR, the estimated WER gains are somewhat larger, but its RMSE is higher than just predicting the mean target in the training data.

Table 7.3: Classifier performance (smallest tree)

Test set	Classifier Performance		Relative WER change compared to baseline (Estimated)		
	SVM	Chance	SVM	Chance	Oracle
dev2004	0.50	0.59	0.9	0.0	4.6
eval2003	0.40	0.47	0.4	0.0	2.6
eval2004	0.43	0.53	0.5	0.0	2.0

Table 7.4: Classifier performance (largest tree)

Test set	Classifier Performance		Relative WER change compared to baseline (Estimated)		
	SVM	Chance	SVM	Chance	Oracle
dev2004	0.17	0.29	0.1	0.0	1.1
eval2003	0.21	0.25	0.3	0.1	2.3
eval2004	0.31	0.34	-1.0	-1.1	2.0

Table 7.5: SVR performance (regression)

Test set	RMSE		Relative WER change compared to baseline (Estimated)		
	SVR	Mean	SVR	Mean	Oracle
dev2004	0.90	0.53	2.5	1.5	4.6
eval2003	0.92	0.50	0.4	0.1	2.6
eval2004	0.72	0.54	0.0	-1.0	2.0

7.4 Online Complexity Control

The evidence presented above suggests the potential for improved ASR system performance by applying the proposed complexity control prediction scheme. To use it in an actual recognition experiment, a procedure was designed, where the classifier (or predictor) is applied simultaneously to all the initial data-threshold regression classes. While applying

this procedure, it was noticed that for the cases when the classifier predicts "move down (-1)", sometimes we get "poor" MLLR transformation estimates from the lower nodes in the tree. To overcome this issue, we decided to smooth the mean MLLR transformations only, of the lower nodes with those from the initial regression class. The weights for smoothing were estimated by maximizing the likelihood of the adaptation data using an approach similar to that described in Sec. 5.6.1 and proposed by Gales in [37].

To apply the online complexity control scheme described in the previous section, an initial set of regression classes is first determined (based on applying a threshold to the adaptation data), and apply the complexity control classifier of Sec. 7.3 to obtain predictions of regression classes to use. Next, the mean and diagonal covariance MLLR transformations are estimated for the predicted classes and the initial regression classes and stored in memory. If there are any regression classes predicted from the lower nodes in the tree (closer to the leaves), the MLLR transformations of the predicted classes are applied to the SI acoustic model and the HMM state occupation statistics are re-estimated. Using the new statistics, smoothing weights are estimated for the mean MLLR transformations in the lower node and its corresponding mean transformation from the initial regression class, to produce the final set of MLLR transformations.

7.5 Recognition Experiments

This overall complexity control scheme is applied to the English BN ASR system of Fig. 4.3, and tested on the three test sets of Table 7.6 for the two classifier design schemes described earlier. The column titled "Baseline" refers to the adaptation strategy with the standard adaptation data-threshold-based complexity control. The classifier for the *largest tree* scheme produces small (but insignificant) improvements for two test sets, while the classifier for the *smallest tree* scheme and the SVR used for the *regression* framework is not able improve over the baseline, in contrast to the estimated WER analysis that suggested the smaller tree would be better. A reason for this may be that the estimate of one node at a time effectively assumes a larger tree, since pruning is only applied to one node. In order to relate this approach to the previous work on predicting whole RCT sizes (Chapter 6, a majority voting procedure was applied to the predictions of the classifier in the largest tree

case and the most frequent prediction was applied to all the nodes in the RCT. The case of predicting whole RCT sizes is equivalent to pruning back or growing further all initial regression classes. This results of the voting procedure in shown in column 6 in Table 7.6, which is able to improve over the baseline cases for two test sets.

Table 7.6: WER (%) for various predicted complexity control strategies applied simultaneously to all regression classes

Test set	Baseline	Smallest Tree	Largest Tree	Regression	Vote
dev2004	19.1	19.1	**19.0**	19.1	**19.0**
eval2003	10.9	10.9	**10.8**	10.9	10.9
eval2004	15.9	15.9	15.9	15.9	**15.8**

To further understand the impact of the proposed complexity control approach, in particular on the robustness of ASR system performance, the per-speaker level WER were examined. The "net (%)" of speakers who benefit using the proposed approach, compared to the baseline adaptation case, are shown in Table 7.7 for the two classifier design schemes and the voting scheme. The trends in the table clearly show that a higher percentage of speakers benefit from using the proposed complexity control scheme, which implies an improved robustness of ASR performance. Again, in contrast to the estimated WER gains in the previous section, the best results here are obtained with the more aggressive (largest) tree size prediction strategy both for the largest tree classifier and the voting scheme applied to the predictions of this classifier. A possible explanation for this can be the fact that the *largest tree* scheme makes greater use of the transformation smoothing weights resulting in stable performance. The positive results for the voting scheme suggest that voting may be somewhat more robust than making independent decisions at the leaf nodes. The same analysis in the case of the *regression* scheme, using SVR, did not show similar trends.

7.6 Another View of Complexity Control

Since online complexity control of MLLR-based adaptation determines the degree of adaptation, it is analogous to determining the amount of "shift" of the SI acoustic model needed

Table 7.7: Net (%) of speakers who benefit from classifier-based complexity control compared to standard case.

Test set	Smallest Tree (%)	Largest Tree (%)	Voting (%)
dev2004	7	5	12
eval2003	13	30	37
eval2004	14	12	22

to move it "closer" to the true SD acoustic model. It can be argued that target speakers whose true SD acoustic model is "close" to the SI acoustic model would require a lesser degree of adaptation compared to speakers whose true SD acoustic models are very different from the SI acoustic model. In related work in [60], it was reported that for supervised adaptation of statistical classifiers, when the distributions of training and target data were similar (low KL distance between distributions), fewer adaptation data samples were needed to achieve the same confidence in classifier error as that in the case of dissimilar distributions (higher KL distance) and greater number of adaptation samples. The problem of deciding the online complexity control or the degree of adaptation in MLLR can also be addressed by investigating measures of dissimilarity between training and target speaker distributions.

In this dissertation, preliminary investigations were conducted to study the dissimilarity of distributions of rate of speech measures in training and target speaker populations and examine the relationship of such dissimilarities and the four "degrees" of adaptation of the rows in Fig 7.2. In Fig. 7.3, histograms of zero-normalized rate of speech (measured in phones per second) at the level of nodes in the RCT are shown for both training and target speaker populations for English BN for different "degrees" of adaptation. Training speaker population refers to the speakers used for training the SI acoustic model and comprised 900 hours of English BN acoustic data. The target speaker population is the NIST 2004 English BN test set comprising 6 hours of acoustic data. In addition, the rate of speech measures for the training and target speaker populations are normalized using the same normalization factors used in column 2 of Fig. 7.2 in order for them to be comparable. KL distance measures were computed between each pair of histograms in each column of Fig. 7.3 with

normalized and smoothed histograms to compare them for similarity. The mismatch in the distributions is greatest in column 1 (KL distance of 0.64), which corresponds to the case of higher degree of adaptation (larger trees and faster speech). The mismatch between distributions is least for column 4 (KL distance 0.31), which correspond to lesser degree of adaptation (smaller trees and slower speech). However, the KL distance measures for the distributions of columns 2 and 3 are 0.42 and 0.60 respectively. The KL distance measures do not increase from left to right, as one would expect if less mismatch implies that fewer samples are required in adaptation (allowing for bigger trees). However, it may be that ROS is not the right space to examine mismatch and/or complications from using unsupervised (vs. supervised adaptation).

Also noticeable in Fig. 7.3, is the overall trend that phones in the training speaker population have higher rate of speech measures compared to the target speaker population. A likely reason for this phenomenon is that the rate of speech measures for the training speaker population is computed from forced alignments that are generated from supervised transcriptions and may be constrained to include extra phones. The rate of speech measures for target speakers are based on recognition hypothesis which do not have such constraints and may effectively have a lower rate of speech as measured in phones per second.

Next, in Fig. 7.4 the distribution of the relative ASR performance improvements of MLLR adaptation using the SI RCT, compared to using the SI acoustic (unadapted) model, is shown for speakers in the English BN 2004 test set. The horizontal axis in Fig. 7.4, represents speakers who are ordered by decreasing rate of speech (measured in phones/sec) from left to right. The remaining presentation details of Fig. 7.4 is similar to that of Fig. 5.7. Complexity control for MLLR adaptation for the experiment shown in Fig. 7.4 is performed using the standard approach of using only amount of adaptation data. It can be seen that speakers with performance loss due to MLLR adaptation extends across the entire range of rate of speech (and by extension the entire range of amount of adaptation data) and not just regions of mismatch in distributions, particularly the fast speaking rates as shown in Fig. 7.3.

The conclusions of [60] with regard to the amount of adaptation data required should be qualified by the fact that it is derived from experimental evidence in the case of supervised

adaptation, while the work in this dissertation is only on unsupervised adaptation where adaptation hypotheses are errorful. In this scenario, it may be useful to consider confidence of word hypotheses in assessing train/test differences. In addition, it may be necessary to consider distribution dissimilarities in terms of multiple factors jointly.

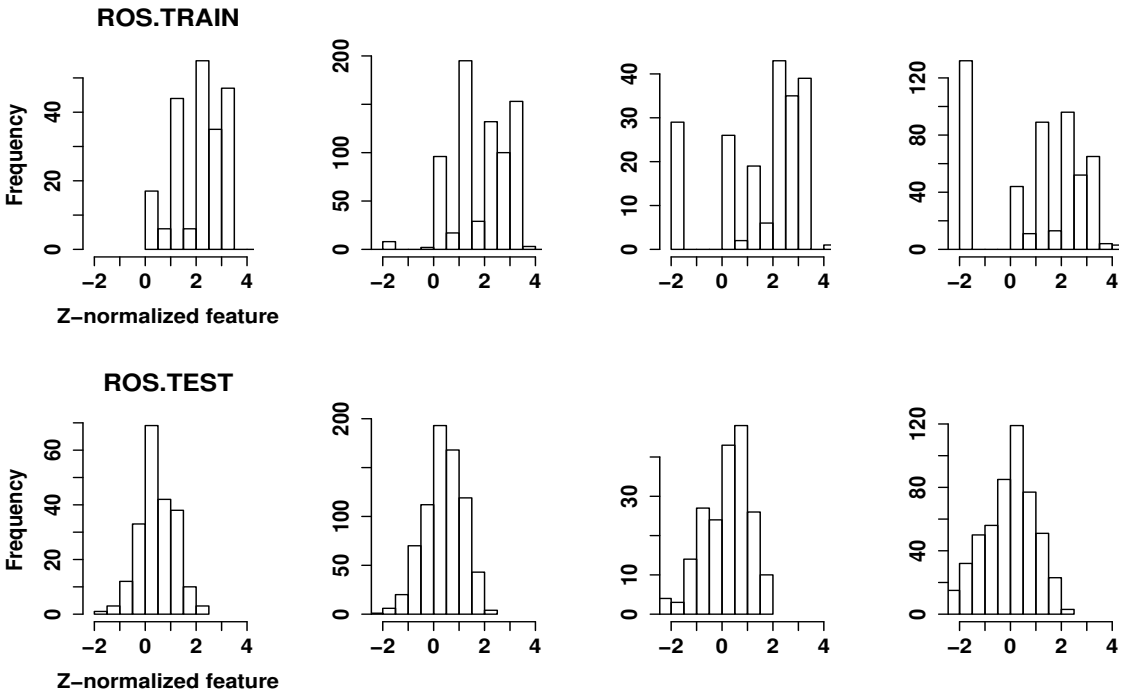

Figure 7.3: Histograms of rate of speech measures at the regression tree node level for training and test speaker populations for four "degrees of adaptation": move down (col 1), stay unchanged (col 2), move up (col 3) and no adapt (col 4).

7.7 Discussion

In summary, a new approach for online complexity control of RCTs is proposed for MLLR adaptation that attempts to improve upon the standard approach to this problem. Two classifiers were designed to predict the optimal regression class to use based on several higher-level information sources that are derived from the initial regression class, obtained by the standard approach. It is shown that rate-of-speech and phone distribution entropy

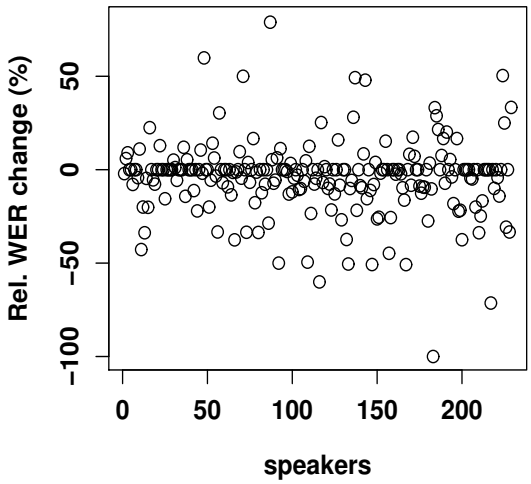

Figure 7.4: Relative change in WER for all speakers in the NIST 2004 English BN test set, ordered by decreasing rate of speech from left to right.

features can potentially be useful in predicting complexity of MLLR adaptation, in addition to the amount of adaptation data. These classifiers performed better than chance, though by modest margins and produced improvements in an estimate of change in WER due to MLLR adaptation. In addition, we also proposed a procedure for applying the complexity control scheme simultaneously to all initial estimates of regression classes that led to only modest improvements in ASR system performance. However, the robustness of ASR system performance is clearly improved, as shown by the large net percentage of speakers who benefit from using our complexity control approach, compared to the baseline adaptation case.

The overall gains observed here are not as large as those reported in Chapter 6 with a more constrained tree structure prediction, but the tasks were different (conversational speech vs. broadcast news) and the use of speaker clustering to define adaptation segments makes the broadcast news domain somewhat more challenging for robust adaptation. The voting scheme is most similar in that it uses a constrained set of subtrees, but these are

relative to the data-driven cut through the original tree rather than the hand-determined set in Chapter 6.

Chapter 8

CONCLUSIONS AND FUTURE WORK

This chapter presents a summary of the main contributions of this dissertation and proposed future directions of research related to this work.

8.1 Main Findings and Contributions

The pilot study presented in Chapter 3, provides evidence that MLLR adaptation leads to worse ASR system performance for 15% of speakers (Fig. 3.1) in the case of English CTS and about 30% of speakers in the case of English BN. This is an important problem that concerns the robustness of performance improvements obtained using MLLR adaptation. The research presented in this dissertation have proposed new solutions to improve the robustness of MLLR adaptation by exploring two aspects of MLLR transformation sharing using regression class trees: the design of regression class trees and the online complexity control of adaptation.

Based on evidence in previous work [14, 34, 53], that incorporating speaker variability information during adaptation leads to improved ASR system performance, the approach in Chapter 5 was to move beyond the use of a single SI RCT, training multiple RCTs, each of which represented distinct types of speaker variability within a large training speaker population. A new speaker clustering algorithm was introduced that modeled speaker variability by partitioning a large corpus of speakers in the eigenspace of their MLLR transformations, and captured speaker variability information in the diversity of the structures of RCTs trained for each speaker cluster. By using the optimal cluster-specific RCT for each individual target speaker, it was possible to achieve significant overall improvement in ASR system performance, compared to the case where a single SI RCT is used. Visual examination of the cluster-specific RCT structures revealed that in the case of the constrained RCT, the vowel branches exhibited more diversity than the consonant branches, which

was consistent with findings of independent studies on dialect markers in North American English.

To incorporate the multiple, cluster-specific RCTs in MLLR adaptation and subsequent recognition experiments, a procedure was developed that linearly combined MLLR mean transformations for a given speaker. The linear combination of the transformations was done using weights, one for each cluster-specific RCT, that were estimated using an ML-based approach on the adaptation data in the framework of a two-step ML procedure that estimated the weights with and without inequality constraints. The two-step ML procedure produced small, but insignificant overall improvements, compared to using only one SI RCT, for both English BN and CTS tasks, and larger improvements (1.9-2.9% relative, which is significant) for Mandarin BN and BC test sets. More importantly, it was observed that the use of speaker-clustered RCT led to ASR performance gains that were robust to a wide range of amounts of adaptation data and WERs on the adaptation hypothesis. Approximately 5% more speakers, in the case of English CTS, and 7% more speakers, in the case of English BN, benefited significantly from using the speaker-clustered RCT, compared to the baseline case of a single SI RCT. As the amount of adaptation data decreases, regression classes are chosen higher up in the RCT (based on a given data count threshold), but the tying across phone classes differs depending on the RCT structure. This results in diverse MLLR transformations being linearly combined, and explains the robustness of WER gains from adaptation across a range of conditions. It is also observed that the speaker-clustered RCTs benefited a majority of the speakers who were hurt by MLLR adaptation with a single SI RCT, and reduced the average performance loss for those speakers who were hurt by MLLR.

In Chapter 6, evidence (Fig. 6.2) was presented that using the standard approach for complexity control for target speaker does not yield the optimal number of regression classes to use. Based on this evidence, the work in Chapter 6 and 7 was focused on exploring solutions that involved using higher-level information sources, aggregated at different levels of granularity (speaker or RCT node) and including the standard amount of adaptation data, to predict better complexity levels for MLLR adaptation. Chapter 6 proposed an approach to predict adaptation complexity, for target speakers, from a pre-determined set of fixed number of regression class sharing schemes, including performing no adaptation, using

standard statistical classifiers and speaker-level features. This solution resulted in modest (2.1% relative), but significant, improvement in ASR system performance for English CTS. Among the speaker-level features, likelihood of different phone classes in the adaptation hypotheses, with the unadapted SI acoustic model, and the standard criterion of amount of adaptation data were found to be most useful. A limitation of this work was that the final number of regression classes to use was predicted as a "whole", which reduces the flexibility in the possible of different RCT structures that can be predicted for a target speaker.

In Chapter 7, we overcame this limitation by proposing a new, more flexible strategy that can predict complexity of adaptation by performing node-level pruning and showed there are other higher-level features that can be useful for this purpose. The node-level pruning was performed by a support vector machine-based classifier, which predicted the best regression class to use from among the neighboring classes in the tree to the initial minimum adaptation data count regression classes in an SI RCT, by exploring a somewhat larger number of cuts through an SI RCT. A new set of features was designed at the node-level for use with the classifier, among which rate of speech and entropy of amount of adaptation data per-phone in a node were found to be useful (along with the amount of adaptation data) and produced modest gains in the classifier accuracy over predicting the most frequent case. A procedure was proposed that allowed the incorporation of the node-level classifier into the framework of MLLR adaptation such that it could be used in recognition experiments. This procedure used weights, similar to those in Chapter 5, to smooth the MLLR mean transformations from lower nodes in an RCT with those of the initial regression classes. While this approach achieves only insignificant improvements in overall ASR system performance, it is able to improve robustness of ASR performance as evidenced by the higher "net" percentage of speakers (7-30%) who benefit from it, compared to using the standard approach to complexity control.

Overall the research directions presented in this dissertation are general purpose in the context of MLLR adaptation, and as such can impact any application that uses it besides ASR, such as optical character recognition and speaker recognition. The future directions of this research, presented in the next section can be equivalently extended for such applications.

8.2 Future Directions

There are several future directions that can be pursued based on the research presented in this dissertation. This section discusses these by focusing on both algorithm-specific variations and impact over a broader scope.

The effectiveness of the linear combination of cluster-specific MLLR mean transformations for a target speaker, as described in Chapter 5, depends on the combined transformation being different from each component transformation, which in turn depends on the structure of cluster-specific RCTs being very different. Variations of the speaker clustering algorithm of Chapter 5 can be pursued with the aim of training cluster-specific RCTs that show variation in structure. It is possible to obtain a new set of MLLR transformations using the speaker clustering obtained by assigning training speakers to the cluster whose RCT produced the lower WER after MLLR adaptation, and then performing the k-means based clustering in the eigenspace of these new transformations. To further validate the oracle-cluster dependent ASR performance improvements (shown in Table 5.1 and 5.2) obtained from the speaker clustering algorithm, the training speakers can be randomly clustered and repeating the oracle cluster error analysis. We can relax the constraint that the speakers used for clustering not be included in acoustic model training can be relaxed to utilize a larger population for speaker clustering, which may lead to more diverse RCTs.

The two-step ML weight estimation procedure in Chapter 5, involving the estimation of weights with inequality constraints, is a simple one that may not always be successful in finding weights that satisfy all the constraints. Perhaps a solution is to explore more advanced methods in constrained optimization.

It was mentioned that the cluster-specific RCTs learned in Chapter 5, are conjectured to be representative of dialectal (or sociolectal) information in large speaker populations. An immediate extension of this work is to perform more detailed analysis, in collaboration with linguists, to determine the validity of this hypothesis. A related direction of this of this work is to design an automatic system for dialect clustering which will benefit ASR systems for languages such as Arabic. It may also be of interest to investigate whether this approach can be used for detecting changes in register, for example between news reporting

and talk show conversations, which may help ASR systems to regulate adaptation strategies for each domain. The eventual goal of this direction of research is to automatically learn new subspaces that are representative of speaker variability as it relates to adaptation, whether it is a function of dialects or other types of variability that can lead to more effective modeling in the final adaptation strategy.

The main direction of future research for the new strategies for complexity control presented in Chapter 6 and 7 is to search for features that have greater predictive power for this task. Pronunciation probabilities for words spoken by individual speakers and reliable confidence estimates for the words in the adaptation hypotheses can be explored. The current set of features explored show evidence that they add more information, in predicting complexity of adaptation, compared to the case of amount of adaptation data alone. However, they are somewhat weak in their predictive power as seen from the accuracy levels of the classifiers. While these features are easy to interpret, since they are derived using knowledge of the properties of the speech signal, they are also somewhat noisy. A better approach might be to use the posterior of statistical models of these information sources (e.g., rate of speech) as features. Another promising direction to pursue is to investigate features that model the differences in distributions of higher-level information sources between training and target speaker populations, as motivated by the discussion in Sec. 7.6. In yet another interesting research direction, the problem of online complexity control can be solved as an extension of the eigenspace MLLR work reported in [14]. In this approach, the final MLLR transformation to use would be a weighted combination of eigen MLLR transformations as in [14], but using a different predicted number of eigen MLLR transformations (of Chapter 5) to use for each target speaker. The eventual extension of this work is to develop a set of adaptation correlates, which can be used a basis for two other tasks: "difficulty" of adaptation for individual speakers, potential gains (or losses) from adaptation. Both of these tasks are important in being able to design adaptation strategies that provide robustness across a wide range of conditions, in addition to improving system performance.

A final extension of the work presented in the dissertation is to incorporate the complexity control strategy of Chapter 7 into the design of adaptation strategies using multiple, speaker-clustered RCTs of Chapter 5. Since both approaches are successful in impacting

the robustness of MLLR adaptation, such a combined approach may lead to more stable ASR system performance.

Bibliography

[1] S. Ahadi and P. Woodland, "Combined Bayesian and predictive techniques for rapid speaker adaptation of continuous density hidden Markov models," *Computer Speech & Language*, vol. 11, pp. 187–206, 1997.

[2] T. Anastasakos, J. McDonough, and J. Makhoul, "Speaker adaptive training: A maximum likelihood approach to speaker normalization," in *Proc. of ICASSP*, vol. 2, 1997, pp. 1043–1046.

[3] T. Anastasakos, J. McDonough, R. Schwartz, and J. Makhoul, "A compact model for speaker adaptive training," in *Proc. of ICSLP*, vol. 2, 1996, pp. 1137–1140.

[4] L. R. Bahl, F. Jelinek, and R. L. Mercer, "A maximum likelihood approach to continuous speech recognition," *IEEE Transaactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-5, pp. 179–190, 1983.

[5] J. Baker, "The Dragon system-An overview," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. ASSP-23, pp. 24–29, 1975.

[6] L. Baum, T. Petrie, G. Soules, and N. Weiss, "A maximization technique in the statistical analysis of probabilistic functions of finite state Markov chain," *Annals of Mathematical Statistics*, vol. 41, no. 164, pp. 164–171, 1970.

[7] E. Bocchieri, V. Digalakis, A. Corduneanu, and C. Boulis, "Correlation modeling of MLLR transform biases for rapid HMM adaptation to new speakers," in *Proc. of ICASSP*, vol. 2, 1999, pp. 773–776.

[8] C. Boulis, V. Diakoloukas, and V. Digalakis, "Maximum likelihood stochastic transformations adaptation for medium and small data sets," *Computer Speech & Language*, vol. 15, no. 3, pp. 257–287, 2001.

[9] C. Boulis and V. Digalakis, "Fast speaker adaptation of large vocabulary continuous speech recognizer using a basis transform approach," in *Proc. of ICASSP*, vol. 2, 2000, pp. 989–992.

[10] H. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*. Kluwer Press, 1993.

[11] I. Bulyko, M. Ostendorf, and A. Stolcke, "Getting more mileage from web text sources for conversational speech language modeling using class-dependent mixtures," in *Proc. HLT/NAACL*, 2003, pp. 7–9.

[12] W. Byrne and A. Gunawardana, "Discounted likelihood linear regression for rapid adaptation," in *Proc. of Eurospeech*, vol. I, 1999, pp. 203–206.

[13] C.-C. Chang and C.-J. Lin, *LIBSVM: a library for support vector machines*, 2001, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[14] K. T. Chen, W. W. Liau, H. M. Wang, and L. S. Lee, "Fast speaker adaptation using eigenspace-based maximum likelihood linear regression," in *Proc. of ICSLP*, vol. III, 2000, pp. 742–745.

[15] S. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau, and G. Zweig, "Advances in speech transcription at IBM under the DARPA EARS program," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1596–1608, 2006.

[16] R. Chengalvarayan, "Speaker adaptation using discriminative linear regression on time-varying mean parameters in trended HMM," *IEEE Signal Processing Letters*, vol. 5, pp. 63–65, 1998.

[17] C. Chesta, O. Siohand, and C. H. Lee, "Maximum a posteriori linear regression for hidden Markov model adaptation," in *Proc. of Eurospeech*, vol. I, 1999, pp. 211–214.

[18] J. Chien, "Online hierarchical transformation of hidden Markov models for speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 6, pp. 656–667, 1999.

[19] W. Chou, "A posteriori linear regression with elliptically symmetric matrix priors," in *Proc. of Eurospeech*, vol. I, 1999, pp. 1–4.

[20] C. Cieri, D. Miller, and K. Walker, "The Fisher corpus: A resource for the next generations of speech-to-text," in *Fourth International Conference on Language Resources and Evaluation*, 2004.

[21] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, pp. 357–366, 1980.

[22] ——, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[23] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.

[24] V. Digalakis, S. Berkowitz, E. Bocchieri, C. Boulis, W. Byrne, H. Collier, A. Corduneanu, A. Kannan, S. Khudanpur, and A. Sankar, "Rapid speech recognizer adaptation to new speakers," in *Proc. of ICASSP*, vol. 2, 1999, pp. 2102–2105.

[25] V. Digalakis, D. Rtischev, and L. Neumeyer, "Speaker adaptation using constrained estimation of Gaussian mixtures," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 5, pp. 357–366, 1995.

[26] S. Doh and R. Stern, "Inter-class MLLR for speaker adaptation," in *Proc. of ICASSP*, vol. 3, 2000, pp. 1755–1758.

[27] E. Eide and H. Gish, "A parametric approach to vocal tract length normalization," in *Proc. of ICASSP*, vol. 1, 1996, pp. 346–348.

[28] L. Ferrer, K. Sönmez, and S. Kajarekar, "Class-based score combination for speaker recognition," in *Proc. of Eurospeech*, 2005, pp. 2173–2176.

[29] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER)," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997, pp. 347–352.

[30] V. R. Gadde, "Modeling word durations," in *Proc. of ICSLP*, vol. 1, 2000, pp. 601–604.

[31] M. Gales, "The generation and use of regression class trees for MLLR adaptation," Cambridge University, Tech. Rep. CUED/F-INFENG/TR263, 1996.

[32] ——, "Transformation smoothing for speaker and environmental adaptation," in *Proc. of Eurospeech*, vol. 4, 1997, pp. 2067–2070.

[33] ——, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech & Language*, vol. 12, pp. 75–98, 1998.

[34] ——, "Cluster adaptive training of hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 4, pp. 417–428, 2000.

[35] ——, "Multiple-cluster adaptive training schemes," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE International Conference on*, vol. 1, 2001, pp. 361–364vol.1.

[36] M. Gales, K. Do, P. Woodland, C. Ho, D. Mrva, R. Sinha, and S. E. Tranter, "Progress in the CU-HTK broadcast news transcription system," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1513–1525, 2006.

[37] M. Gales and P. Woodland, "Mean and variance compensation within the MLLR framework," *Computer Speech & Language*, vol. 10, pp. 249–264, 1996.

[38] J. Gauvian and C. Lee, "Maximum a-posteriori estimation for multivariate Gaussian mixture observations of Markov chains," *IEEE Transactions on Speech and Audio Processing*, vol. 2, pp. 291–298, 1994.

[39] J. Godfrey, E. Holliman, and J. McDaniel, "Switchboard: Telephone corpus for research and development," in *Proc. of ICASSP*, vol. 1, 1992, pp. 517–520.

[40] N. Goel, "Investigation of silicon auditory models and generalization of linear discrminant models for improved speech recognition," Ph.D. dissertation, Johns Hopkins University, 1997.

[41] P. Gopalakrishnan, D. Kanvesky, A. Nadas, and D. N. and, "An inequality for some rational functions with applications to some statistical estimation problems," *IEEE Transactions on Information Theory*, vol. 37, pp. 107–113, 1991.

[42] S. Goronzy and R. Kompe, "A MAP-like weighting scheme for MLLR speaker adaptation," in *Proc. of Eurospeech*, vol. I, 1999, pp. 5–8.

[43] A. Gunawardana and W. Byrne, "Discriminative speaker adaptation with conditional maximum likelihood linear regression," in *Proc. of Eurospeech*, vol. 2, 2001, pp. 1203–1206.

[44] R. Haeb-Umbach, "Automatic generation of phonetic regression class trees for MLLR adaptation," *Speech and Audio Processing, IEEE Transactions on*, vol. 9, no. 3, pp. 299–302, 2001.

[45] T. Hazen, "A comparison of novel techniques for rapid speaker adaptation," *Speech Communication*, vol. 31, pp. 15–33, 2000.

[46] H. Hermansky, "Perceptual linear prediction (PLP) analysis of speech," *Speech Communication*, vol. 87, pp. 1738–1752, 1990.

[47] C. Huang, T. Chen, S. Li, E. Chang, and J. Zhou, "Analysis of speaker variability," in *Proc. of Eurospeech*, vol. 2, 2001, pp. 1377–1380.

[48] X. Huang and K. Lee, "On speaker-independent, speaker-dependent and speaker-adaptive speech recognition," in *Proc. of ICASSP*, vol. 2, 1991, pp. 877–880.

[49] M.-Y. Hwang, X. Lei, W. Wang, and T. Shinozaki, "Investigation on Mandarin broadcast news speech recognition," in *Proc. of ICSLP*, 2006, pp. 1233–1236.

[50] A. Imamura, "Speaker adaptive HMM-based speech recognition with a stochastic speaker classifier," in *Proc. of ICASSP*, vol. 2, 1991, pp. 841–844.

[51] F. Jelinek, *Statistical Methods for Speech Recognition*. MIT Press, 1997.

[52] T. Kosaka and S. Sagayama, "Tree structured speaker clustering for fast speaker adaptation," in *Proc. of ICASSP*, vol. 1, 1994, pp. 245–248.

[53] R. Kuhn, J. Junqua, P. Nguyen, and N. Niedzielski, "Rapid speaker adaptation in eigenvoice space," *IEEE Transactions on Speech and Audio Processing*, vol. 8, no. 6, pp. 695–707, 2000.

[54] R. Kuhn, P. Nguyen, J.-C. Junqua, L. Goldwasser, N. Niedzielski, S. Fincke, K. Field, and M. Contolini, "Eigen voices for speaker adaptation," in *Proc. of ICSLP*, vol. 5, 1998, pp. 303–306.

[55] W. Labov, "The organization of dialect diversity in North America," in *Fourth International Conference on Spoken Language Processing*, 1996.

[56] L. Lee and R. Rose, "A frequency warping approach to speaker normalization," *IEEE Transactions on Speech and Audio Processing*, vol. 6, pp. 49–60, 1998.

[57] C. Leggetter, "Improved acoustic modelling for HMMs using linear transformations," Ph.D. dissertation, University of Cambridge, 1995.

[58] C. Leggetter and P. Woodland, "Flexible speaker adaptation using maximum likelihood linear regression," in *Proc. ARPA Spoken Language Technology Workshop*, 1995, pp. 104–109.

[59] ——, "Maximum likelihood linear regression for speaker adaptation of HMMs," *Computer Speech & Language*, vol. 9, pp. 171–185, 1995.

[60] X. Li and J. Bilmes, "A Bayesian divergence prior for classifier adaptation," in *AI STATS*, 2007.

[61] R. Lippmann, "Review of neural networks for speech recognition," *Neural Computation*, vol. 1, pp. 1–38, 1989.

[62] Z. Lu, I. Bazzi, A. Kornai, and J. Makhoul, "A robust, language-independent OCR system," in *Proc. 27th Advances in Computer-Assisted Recognition Workshop, SPIE*, 1999.

[63] B. Mak and R. Hsiao, "Improving eigenspace-based MLLR adaptation by kernel PCA," in *Proc. of ICSLP*, vol. I, 2004, pp. 13–16.

[64] A. Mandal, L. Davis, C. Espy-Wilson, and M. Matthies, "The use of temporal vs. spectral cues to recognize speech," in *Journal of the Acoustical Society of America*, vol. 6, no. 4, Pt. 2, 1999, p. 2182, presented at the 138th Meeting of the Acoustical Society of America, Columbus, OH, November 1-5, 1999.

[65] A. Mandal, M. Ostendorf, and A. Stolcke, "Leveraging speaker-dependent variation of adaptation," in *Proc. of Eurospeech*, 2005, pp. 1793–1796.

[66] ——, "Speaker clustered regression-class trees for MLLR adaptation," in *Proc. of ICSLP*, 2006, pp. 1133–1136.

[67] L. Mangu, E. Brill, and A. Stolcke, "Finding consensus in speech recognition: Word error minimization and other applications of confusion networks," *Computer Speech & Language*, vol. 14, pp. 373–400, 2000.

[68] K. Mardia, J. Kent, and J. Bibby, *Multivariate Analysis*. Academic Press, 1979.

[69] T. Martinetz, S. Berkovich, and K. Schulten, "Neural-gas network for vector quantization and its application to time-series prediction," *IEEE Transactions on Neural Networks*, vol. 4, pp. 558–569, 1993.

[70] S. Matsoukas, J.-L. Gauvain, G. Adda, T. Colhurst, K. Chia-Lin, O. Kimball, L. Lamel, F. Lefevre, J. Z. Ma, J. Makhoul, L. Nguyen, R. Prasad, R. Schwartz, H. Schwenk, and X. Bing, "Advances in transcription of broadcast news and conversational telephone speech within the combined EARS BBN/LIMSI system," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1541–1556, 2006.

[71] S. Matsoukas, R. Schwartz, H. Jin, and L. Nguyen, "Practical implementations of speaker-adaptive training," in *Proc. DARPA Speech Recognition Workshop*, 1997.

[72] J. McDonough, T. Anastasakos, G. Zavaliagkos, and H. Gish, "Speaker-adapted training on the switchboard corpus," in *Proc. of ICASSP*, vol. 2, 1997, pp. 1059–1063.

[73] J. McDonough, V. Venkataramani, and W. Byrne, "On the incremental addition of regression classes for speaker adaptation," in *Proc. of ICASSP*, vol. 3, 2000, pp. 7141–1743.

[74] N. Morgan, B. Chen, Q. Zhu, and A. Stolcke, "TRAPping conversational speech: Extending TRAP/Tandem approaches to conversational speech recognition," in *Proc. of ICASSP*, vol. 1, 2004, pp. 536–539.

[75] H. Murveit, J. Butzberger, V. Digilakis, and M. Weintraub, "Large-vocabulary dictation using SRI's DECIPHER(TM) speech recognition system: Progressive-search techniques," in *Proc. of ICASSP*, vol. II, 1993, pp. 319–322.

[76] H. Murveit, P. Monaco, V. Digilakis, and J. Butzberger, "Techniques to achieve an accurate real-time large-vocabulary speech recognition system," in *Proc. ARPA Spoken Language Technology Workshop*, 1994.

[77] J. Neto, L. Almeida, M. M. Hochberg, C. Martins, L. Nunes, S. J. Renals, and A. J. Robinson, "Unsupervised speaker-adaptation for hybrid HMM-MLP continuous speech recognition systems," in *Proc. of Eurospeech*, vol. 1, 1995, pp. 187–190.

[78] L. Neumeyer, A. Sankar, and V. Digilakis, "A comparative study of speaker adaptation techniques," in *Proc. of Eurospeech*, vol. 2, 1995, pp. 1127–1130.

[79] L. Neumeyer and M. Weintraub, "Probabilistic optimum filtering for robust speech recognition," in *Proc. of ICASSP*, vol. 1, 1994, pp. 417–420.

[80] Y. Normandin and S. D. Morgan, "An improved MMIE algorithm for speaker-independent small vocabulary continuous speech recognition," in *Proc. of ICASSP*, vol. 1, 1991, pp. 537–540.

[81] J. Odell, "The use of context in large vocabulary speech recognition," Ph.D. dissertation, University of Cambridge, 1995.

[82] M. Padmanabhan, L. Bahl, D. Nahamoo, and M. Picheny, "Speaker clustering and transformation for speaker adaptation in speech recognition systems," *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 1, pp. 71–77, 1998.

[83] M. Padmanabhan, G. Saon, and G. Zweig, "Lattice-based unsupervised MLLR for speaker adaptation," in *Proc. ISCA ITRW ASR2000*, 2000, pp. 128–131.

[84] D. Povey, B. Kingsbury, L. mangu, G. Saon, H. Soltau, and G. Zweig, "fMPE: Discriminatively trained features for speech recognition," in *Proc. of ICASSP*, vol. I, 2005, pp. 961–964.

[85] D. Povey and P. Woodland, "Minimum phone error and I-smoothing for improved discriminative training," in *Proc. of ICASSP*, vol. 1, 2002, pp. 105–108.

[86] R Development Core Team, *R: A language and environment for statistical computing*, R Foundation for Statistical Computing, 2005, ISBN 3-900051-07-0; http://www.R-project.org.

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

[88] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[89] M. Riley, W. Byrne, M. Fincke, S. Khudanpur, A. Ljolje, J. McDonough, H. Nock, M. Saraclar, C. Wooters, and G. Zavaliagkos, "Stochastic pronunciation modelling from handlabelled phonetic corpora," in *Proc. of ESCA Workshop on Modeling Pronunciation Variation for Automatic Speech Recognition*, 1998, pp. 109–116.

[90] A. Sankar, F. Beaufays, and V. Digilakis, "Training data clustering for improved speech recognition," in *Proc. of Eurospeech*, vol. 1, 1995, pp. 502–505.

[91] A. Sankar, R. Gadde, and F. Weng, "SRI's 1998 broadcast news system - towards faster, smaller, and better speech recognition," in *DARPA Broadcast News Workshop*, 1999, pp. 281–286.

[92] A. Sankar and C. H. Lee, "Robust speech recognition based on stochastic matching," in *Proc. of ICASSP*, vol. 1, 1995, pp. 121–124.

[93] A. Sankar and C. Lee, "A maximum likelihood approach to stochastic matching for robust speech recognition," *IEEE Transactions on Speech and Audio Processing*, vol. 4, pp. 190–202, 1996.

[94] A. Sankar, L. Neumeyer, and M. Weintraub, "An experimental study of acoustic adaptation algorithms," in *Proc. of ICASSP*, vol. 2, 1996, pp. 713–716.

[95] B. Schlkopf, C. J. C. Burges, and A. J. Smola, *Advances in Kernel Methods: Support Vector Learning*. MIT Press, 1999.

[96] K. Shinoda and C. Lee, "Structural MAP speaker adaptation using hierarchical priors," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 1997, pp. 381–388.

[97] K. Shinoda and T. Watanabe, "Speaker adaptation with autonomous model complexity control by MDL principle," in *Proc. of ICASSP*, vol. 2, 1995, pp. 717–720.

[98] A. Stolcke, X. Anguera, K. Boakye, O. Cetin, F. Grezl, A. Janin, A. Mandal, B. Peskin, C. Wooters, and J. Zheng, "Further progress in meeting recognition: The ICSI-SRI spring 2005 meeting recogntion system," in *Proc. NIST MLMI Meeting Recognition Workshop*, 2005.

[99] A. Stolcke, H. Bratt, J. Butzberger, H. Franco, V. R. R. Gadde, M. Plauche, C. Richey, E. Shriberg, K. Sonmez, F. Weng, and J. Zheng, "The SRI March 2000 Hub-5 conversational speech transcription system," in *Proc. NIST Speech Transcription Workshop*, 2000.

[100] A. Stolcke, B. Chen, H. Franco, R. Gadde, M. Graciarena, M. Y. Hwang, K. Kirchoff, X. Lei, A. Mandal, N. Morgan, T. Ng, M. Ostendorf, K. Sonmez, A. Venkataraman, D. Vergyri, W. Wang, J. Zheng, and Q. Zhu, "Recent innovations in speech-to-text transcription at SRI-ICSI-UW," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 14, no. 5, pp. 1729–1744, 2006.

[101] A. Stolcke, L. Ferrer, S. Kajarekar, E. Shriberg, and A. Venkataraman, "MLLR transforms as features in speaker recognition," in *Proc. of Eurospeech*, 2005, pp. 2425–2428.

[102] L. Ubel and P. Woodland, "Improvements in linear transform based speaker adaptation," in *Proc. of ICASSP*, vol. 1, 2001, pp. 49–52.

[103] V. Valtchev, J. Odell, P. C. Woodland, and S. J. Young, "MMIE training of large vocabulary recognition systems," *Speech Communication*, vol. 22, pp. 303–314, 1997.

[104] V.Diakoloukas and V.Digalakis, "Maximum likelihood stochastic transformation adaptation of hidden Markov models," *IEEE Transactions on Speech and Audio Processing*, vol. 7, pp. 177–187, 1999.

[105] A. Venkataraman, A. Stolcke, W. Wang, D. Vergyri, V. Gadde, and J. Zheng, "SRI's 2004 broadcast news speech to text system," in *EARS RT04 Workshop*, 2004.

[106] V. Venkataramani and W. Byrne, "MLLR adaptation techniques for pronunciation modeling," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 2001, pp. 421–424.

[107] D. Vergyri, A. Stolcke, V. R. R. Gadde, L. Ferrer, and E. Shriberg, "Prosodic knowledge sources for automatic speech recognition," in *Proc. of ICASSP*, vol. 1, 2003, pp. 208–211.

[108] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *Information Theory, IEEE Transactions on*, vol. 13, no. 2, pp. 260–269, 1967.

[109] L. Wang and P. Woodland, "Discriminative adaptive training using the MPE criterion," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 2003, pp. 279–284.

[110] S. Wang and Y. Zhao, "Online Bayesian tree-structured transformations of HMMs with optimal model selection for speaker adaptation," *IEEE Transactions on Speech and Audio Processing*, vol. 9, pp. 663–677, 2001.

[111] W. Wang and M. Harper, "The SuperARV language model: Investigating the effectiveness of tightly integrated multiple knowledge sources," in *Proc. of the Conference on Empirical Methods in Natural Language Processing*, 2002, pp. 238–247.

[112] S. Wegmann, D. McAllaster, J. Orloff, and B. Peskin, "Speaker normalization on conversational telephone speech," in *Proc. of ICASSP*, vol. 1, 1996, pp. 339–341.

[113] F. Weng, A. Stolcke, and A. Sankar, "Efficient lattice representation and generation," in *Proc. of ICSLP*, vol. 6, 1998, pp. 2531–2534.

[114] R. Westwood, "Speaker adaptation using eigenvoices," Master's thesis, University of Cambridge., 1999.

[115] K. M. Wong and B. Mak, "Rapid speaker adaptation using MLLR and subspace regression classes," in *Proc. of Eurospeech*, vol. 2, 2001, pp. 1253–1256.

[116] P. Woodland, "Speaker adaptation: Techniques and challenges," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, vol. I, 1999, pp. 85–90.

[117] P. Woodland, M. Gales, and D. Pye, "Improving enviromental robustness in large vocabulary speech recognition," in *Proc. of ICASSP*, vol. 1, 1996, pp. 65–68.

[118] S. Young, J. Odell, and P. Woodland, "Tree based state tying for high accuracy modelling," in *Proc. ARPA Spoken Language Technology Workshop*, 1994, pp. 405–410.

[119] X. H. Y. Zhao, "Fast model selection based speaker adaptation for nonnative speech," *IEEE Transactions on Speech and Audio Processing*, vol. 11, pp. 298–307, 2003.

[120] J. Zheng, J. Butzberger, H. Franco, and A. Stolcke, "Improved maximum mutual information estimation training of continuous density HMMs," in *Proc. of Eurospeech*, vol. 2, 2001, pp. 679–682.

VITA

Arindam Mandal was born in Kolkata (formerly Calcutta), India and grew up there. He received the Bachelor of Engineering degree in Electrical and Elecronics Engineering in 1997 from Birla Institute of Technology in Ranchi, India and a Master of Science degree in Electrical Engineering from Boston University in Boston, MA in 2000. In 2007, he received the PhD degree in Electrical Engineering from the University of Washington in Seattle, WA.