Improving Robustness of MLLR Adaptation with Speaker-Clustered Regression Class Trees

Arindam Mandal[a,1] Mari Ostendorf[a] Andreas Stolcke[b]

[a] *University of Washington, Department of Electrical Engineering, Seattle, WA, USA*

[b] *Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, USA, and*

International Computer Science Institute, Berkeley, CA, USA

Abstract

We introduce a strategy for modeling speaker variability in speaker adaptation based on maximum likelihood linear regression (MLLR). The approach uses a speaker clustering procedure that models speaker variability by partitioning a large corpus of speakers in the eigenspace of their MLLR transformations and learning cluster-specific regression class tree structures. We present experiments showing that choosing the appropriate regression class tree structure for speakers leads to a significant reduction in overall word error rates in automatic speech recognition systems. To realize these gains in unsupervised adaptation, we describe an algorithm that produces a linear combination of MLLR transformations from cluster-specific trees using weights estimated by maximizing the likelihood of a speaker's adaptation data. This algorithm produces small improvements in overall recognition performance across a range of tasks for both English and Mandarin. More significantly, distributional analysis shows that it reduces the number of speakers with performance loss due to adaptation across a range of adaptation data sizes and word error rates.

Key words:
speech recognition, speaker adaptation, speaker clustering, regression class trees

1 Introduction

Over the past few years, speaker adaptation based on maximum likelihood linear regression (MLLR) (Leggetter and Woodland, 1995; Digalakis et al., 1995) has proven to be an effective technique that yields significant performance wins in speaker-independent (SI) automatic speech recognition (ASR) systems. It has also been successfully applied in the case of environmental adaptation (Woodland et al., 1996). The basic approach in MLLR is to estimate an affine transformation, by maximizing the likelihood of a speaker's adaptation data, to shift the parameters of an SI acoustic model closer to that of a speaker-dependent (SD) model. The transformations can be estimated for both means and variances of Gaussian distributions (Gales and Woodland, 1996) of SI acoustic models. The affine transformations can also be estimated and applied directly to the acoustic feature vectors as reported in Gales (1998). In speaker adaptive training (SAT) (Anastasakos et al., 1997), MLLR is applied

Email addresses: marindam@ee.washington.edu (Arindam Mandal), mo@ee.washington.edu (Mari Ostendorf), stolcke@speech.sri.com (Andreas Stolcke).

[1] Author was a visiting International Fellow at the Speech Technology and Research Laboratory, SRI International, Menlo Park, CA, for the duration of this work. At present the author is a staff member at SRI International.

to both training and testing speakers to model inter-speaker variability and produce a canonical model that primarily captures intra-speaker variability. Furthermore, the MLLR transforms themselves may be used as features for speaker modeling purposes, e.g., in speaker recognition (Stolcke et al., 2005).

With MLLR, it is possible to adapt all parameters of an acoustic model (mean and covariance of SI Gaussian distributions), irrespective of whether they are observed in a limited amount of adaptation data, through the use of regression classes that define groups of models that share a transformation (Gales, 1996, 1997). Regression class trees organize SI Gaussian distributions into hierarchical classes in a tree either by using expert knowledge or by applying a data-driven clustering procedure (agglomerative or divisive) and an appropriate similarity measure for comparing the distributions. In general, regression class trees can be built such that the lowest node in the tree preserves phone-level, triphone state-level or Gaussian distribution-level granularity using appropriate statistics. The statistics are typically collected for a particular domain, e.g., conversational telephone speech (CTS), broadcast news (BN), etc. and used for building the SI regression class tree for that domain. This results in different tree structures, and hence different possible regression classes, for each domain, which impacts the estimation of MLLR transformations. Given adaptation data from a test speaker, the tree is descended from the top to those nodes that satisfy a predetermined minimum count of data frames, and a transformation is estimated for each such node (regression class) to be shared by all its members, allowing for adaptation of both observed and unobserved units.

Previous work on speaker adaptation has also investigated the use of speaker clustering with the aim of introducing flexibility in the adaptation strategy for individual speakers. In Padmanabhan et al. (1998), 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 a speaker adapted (SA) model from the transformed data. Imamura (1991) proposed an approach in which initial speaker-specific acoustic models were clustered using a cross-entropy measure. Kosaka and Sagayama (1994) 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. Shortcomings of these approaches are that the speaker cluster-specific models may not be trained with a 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.

To address the problem of using a single speaker-cluster-specific model in classical speaker clustering approaches, the family of model combination approaches was developed. Bocchieri et al. (1999) proposed an approach to use a cascade of bias transformation vectors, in addition to using a full mean transformation. The cascade of bias vectors provided robust transformation estimates and improved ASR system performance in cases of sparse adaptation data. 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. Cluster adaptive training (CAT) (Gales, 2000) is a speaker clustering approach to SAT, where several cluster-specific acoustic models or MLLR transformations (and a single speaker-independent acoustic model) are trained. For a test speaker, an SD 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. (2000) 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 an SD model that lies within the span of the eigenvoices. In Chen et al. (2000) and Mak and Hsiao (2004), 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 work aims to take advantage of larger amounts of adaptation data, as well as small amounts, and differs from previous approaches in two respects. First, we leverage the benefits of speaker clustering, by designing multiple regression class tree structures (i.e., transformation tying schemes) for different clusters of speakers. Second, we estimate 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. We show evidence that significant ASR performance gains are achievable by choosing the optimal regression class tree structure for each speaker.

We also describe additional algorithms, used on test speakers, that 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 Gales and Woodland (1996) to include a robust weight estimation strategy, and the detailed transformations produce improved robustness in 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.

The rest of this paper is organized as follows: Section 2 describes three techniques for constructing regression class trees and the speaker clustering algorithm in MLLR transformation eigenspace; Section 3 provides details of the ASR system and tasks used in this work; Section 4 discusses the potential gains from the oracle assignment of speakers to regression class trees; Section 5 describes an algorithm that estimates weights, from adaptation data, for combining component MLLR transformations; Section 6 discusses the ASR performance improvements achieved using the weight estimation algorithm and presents an analysis of speaker-level ASR performance that shows the robustness across a range of adaptation conditions; Section 7 concludes by summarizing the key results and possible extensions.

2 Speaker Clustering for Regression Class Trees

2.1 Regression Class Trees

We investigated transformation sharing at the phone level [all hidden Markov model (HMM) states of all triphones with the same center phone are predetermined to share the same transformation] as well as at the state level (different states from the same phone can have different transformations, which may be useful when there are strong coarticulation effects). For RCT design in both cases, each hidden HMM state[2] is represented by multivariate Gaussian sufficient statistics trained from the data associated with that model. For the phone-based RCTs, the models are first grouped according the center phone of the triphone that the model is associated with, so the maximum number of classes is equal to the number of phones (45, in this case). Phone level Gaussian distributions are then estimated from the sufficient statistics of the HMM states belonging to a particular phone using an approach similar to that in Kannan et al. (1994) and Hwang and Huang (1998). For the state-

[2] Here HMM state refers to the set of unique HMM states in the SI acoustic model that are clustered to allow sharing of SI Gaussian distributions among triphone states within each cluster. (The actual model used in decoding is a Gaussian mixture.)

level RCTs, the models are not pre-grouped, and the maximum number of classes is equal to the number of leaves in the acoustic modeling tree (roughly 3000 here).

We experimented with two divisive clustering approaches for building regression class trees (RCTs). For phone-level transformation sharing, we explored both *constrained* and *unconstrained* clustering, and used *unconstrained* clustering for state-level sharing. In the constrained approach, we design a decision tree 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 (Young et al., 1994). In the unconstrained approach, we build a binary tree by splitting the group of models at a node into two clusters, using k-means clustering and a symmetric Kullback-Leibler (KL) distance measure[3] as in Leggetter (1995).

In all cases, trees are grown to their maximum size. For phone-based RCTs, both constrained and unconstrained, each leaf node corresponds to a single phone. Such RCTs, also referred to as phonetic RCTs, have been previously explored in Haeb-Umbach (2001). For state-level RCTs, the leaves of the tree represent the individual state-level distributions. Of course, in online adaptation, there is rarely enough data for robustly estimating a state-specific transformation. However, backoff techniques are used when there is insuffi-

[3] Given two Gaussian distributions described by means μ_1, μ_2 and covariances Σ_1, Σ_2, a symmetric KL distance between the two is given by

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

cient data, and including the fine grained models in the tree makes it possible to use very detailed transformations when there is a large amount of data from a speaker (e.g. for a news anchor).

The three methods for RCT design differ in the splitting criteria for constrained vs. unconstrained trees (i.e. change in likelihood of training data vs. symmetric KL distance between distributions) as well as in the allowed granularity of state tying (phone-level vs. triphone-state-level). These differences result in different branching structure for all three RCTs using the same data for building the trees. Given limited adaptation data, it is often the case that the estimated transformations correspond to internal nodes (non-root, nonleaf node), which then leads to different adaptation results as a function of RCT structure even for the two phone-based trees.

2.2 Speaker Clustering in MLLR Eigenspace

MLLR transformations represent speaker 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 (Huang et al., 2001) and as auxiliary features in mixtures-of-experts classifiers for speaker recognition (Ferrer et al., 2005). In Chen et al. (2000), faster speaker adaptation in ASR was achieved using such representations of MLLR transformations. In contrast, our use of such representations of MLLR transformations is to obtain adaptation-relevant speaker clusters.

We first estimate MLLR transformations for a large corpus of held-out training speakers, using a single constrained-type RCT that has R regression classes.

Given a d-dimensional feature vector used in recognition, we then vectorize the MLLR transformations (mean transform, offset vector) to produce a $d(d+1)$-length vector, and normalize each dimension to have zero mean and unit variance. Next, we perform principal component analysis (PCA) 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 (Mardia et al., 1979; R Development Core Team, 2005). The vectorized transforms are then projected onto the first N principal components, and we form an RN-dimensional supervector for each speaker by stacking together the PCA-reduced MLLR transforms for each of the R classes (excluding the nonspeech class).[4]

Finally, we use k-means clustering to partition the speakers into S clusters, using a Euclidean distance measure between the supervectors. The supervectors capture the speaker information present in MLLR transformations, and the clustering groups together speakers who share similar transform characteristics.

Given the speaker clusters, we then train a separate RCT for each speaker cluster, one of each type described in Section 2.1 using acoustic data of speakers in each cluster. 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, we hypothesize that the cluster-specific RCT will capture patterns representative of each speaker cluster in the structure of the RCT. This will

[4] R is chosen such that each speaker has sufficient data for estimating MLLR transformations for R regression classes.

produce diversity in RCT structures across clusters, in the sense that groups of models (SI Gaussian distributions) will be subdivided differently during tree building for each cluster. We also expect that choosing the appropriate MLLR RCT structure for every speaker should lead to improved ASR performance. In Section 4, we will show that this strategy does indeed lead to different RCT structures and improved ASR performance results when using the oracle RCT.

3 Task and System Description

The ASR system used for this work is SRI's Decipher$^{\text{TM}}$ large vocabulary engine (Stolcke et al., 2006). We used three versions of the system, each with performance comparable to the state of the art: i) English CTS; ii) English BN (Venkataraman et al., 2004), and iii) Mandarin BN and broadcast conversations (BC) (Hwang et al., 2006).

In the CTS version, for first-pass decoding, the system uses gender-dependent, word-internal triphones as HMMs (standard three state left-to-right HMMs), Mel-frequency cepstral coefficients (MFCCs) in the front end, a bigram language model, and phone-loop MLLR [5] (Sankar et al., 1996) to produce lattices. Subsequently, these lattices are expanded using a 4-gram language model and also processed using confusion networks to produce higher-quality hypotheses. The second pass of the system uses acoustic models based on crossword triphones and perceptual linear prediction (PLP) feature vectors as the front end. The PLP-based acoustic models also use, among other standard normalization techniques, speaker adaptive training based on constrained MLLR.

[5] Single transformation MLLR based on a phone-loop recognition pass.

The mean vectors and diagonal covariances of the Gaussian distributions of the PLP-based acoustic model are adapted to test speakers using MLLR and the hypotheses from the first stage. For the rest of the paper, all references to MLLR adaptation are for the second stage of this system. This is effectively cross-system adaptation since adaptation hypotheses from MFCC-based acoustic models are used to adapt PLP-based acoustic models.

The BN/BC version of the system uses a similar architecture, with some differences: it uses gender-independent acoustic models; it segments BN/BC shows into speaker "groups" using an unsupervised clustering algorithm (Sankar et al., 1995), 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 both recognition stages. The Mandarin BN/BC system used a tone-based phone set and MFCC-based features and pitch features for acoustic models in both stages.

The MLLR step for these ASR systems uses an RCT to estimate a full matrix transformation with an offset vector for the Gaussian means and a diagonal variance transformation vector. The baseline version of the systems uses a manually designed RCT that clusters triphone states with the same center phone into groups based on the knowledge of acoustic phonetics (e.g., vowels, fricatives, stops) and organizes them into a tree with 9 leaf classes. To achieve the best possible ASR performance the English CTS system used an empirically tuned threshold of 1200 frames resulting in 6 to 21 regression classes per speaker, while the BN/BC systems used a threshold of 4500 frames resulting in 1 to 9 regression classes per speaker, when using the *constrained* and *unconstrained* RCTs described in Section 2.1. The English CTS speakers have five minutes of adaptation data, on average, while the English BN speakers have adaptation data ranging from a few seconds (sound bites in news segments) to more than thirty minutes (for anchor speakers). This difference in distribution of adaptation data in the two domains is the reason for the difference in the average number of regression classes used per speaker for the CTS system compared to the BN/BC systems.

The speaker clustering procedure of Section 2.2 was performed separately for each domain in each language, and for each gender when gender-dependent acoustic models were used. The corpus used for speaker clustering was drawn from three different sources: conversations of speakers from the Fisher Phase 2 corpus (Cieri et al., 2004) 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, and BN and BC shows in Mandarin (25 hours for each domain and language) from the corpus released by the Linguistic Data Consortium (LDC) in early 2006, for use with the English and Mandarin BN/BC systems, respectively. The corpus 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[6] (1 hour), and a 2005 Mandarin BC test set[7] (2.5 hours).

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

[7] Test set prepared by Cambridge University and used for internal evaluations; not released publicly at this time.

4 Oracle Cluster-Dependent Adaptation

4.1 Oracle Performance Improvements

Our initial studies 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 corpora in Section 3, we first trained speaker-clustered RCT for English CTS and BN. In the speaker-clustering algorithm, we represented each speaker by 8 vectorized MLLR transformations, since this produced stable clusters. We experimented with several different values of N, the number of principal components for projecting the vectorized MLLR transforms, and chose $N = 8$, since it produced the most diversity in structure among the cluster-specific RCT. Diversity of tree structures refers to the differences in the branching structure across the cluster-specific RCTs. In Section 2.2, it was mentioned that one of the goals of this research is to produce different tree branching structures for each cluster-specific RCT, which will result in different transformation sharing and thus different MLLR transformations being estimated based on each tree. Therefore, it is important to develop a quantitative notion of diversity in RCT structures. We used a simple measure that considered the number of splits that are different at any given level across the cluster-specific RCTs. Splits refer to the subdivision of clusters of phone-level Gaussian distributions, which may be based on the constrained question set or data driven. The splits near the top levels in the cluster-specific RCTs have the greatest impact on the subsequent tree branching structure. In the case of constrained RCTs the splits at each level in the RCTs are determined by choosing an appropriate linguistic question. The constrained cluster-specific RCTs are considered to be most diverse when approximately 3 out of 4 questions are different at two levels below the root, across the different cluster-specific RCTs.

The k-means-based clustering algorithm was set up to produce 4 clusters (or 4 clusters per gender) in each domain. This ensured that the speaker partitions had an adequate amount of data to train cluster-specific RCTs, and also maintained reasonable limits on computational costs of the transformation combination algorithm of Section 5. To estimate the potential maximum ASR system performance gains from using a cluster-specific RCT we follow the steps listed in Algorithm 1. We first partition the speaker population $\mathcal{S}$ into $\{\mathcal{S}_{\mathcal{T}}, \mathcal{S}_{\mathcal{H}}\}$, where $\mathcal{S}_{\mathcal{T}}$ is used for cluster refinement and $\mathcal{S}_{\mathcal{H}}$ is a held-out set for evaluation. The held-out subset of speakers $\mathcal{S}_{\mathcal{H}}$ for English CTS experiments was drawn from the recent NIST English CTS test sets (1998-2002). In Step 2 of Algorithm 1, speaker subset $\mathcal{S}_{\mathcal{T}}$ is used to train the cluster-specific RCT, and in Step 3 the held-out speaker subset $\mathcal{S}_{\mathcal{H}}$ is used for determining potential ASR performance gains. [8]

The results of evaluating ASR performance gains after MLLR adaptation using the cluster-specific unconstrained RCTs as detailed in Algorithm 1 are shown in Table 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. Since in step 3 of Algorithm 1, speakers are assigned to that cluster whose RCT in $\mathcal{T}_1$ produces the lowest WER after MLLR adaptation, the overall

[8] Approximately 1% of the speakers in $\mathcal{S}_{\mathcal{H}}$ had similar change in ASR performance after MLLR adaptation using any of the cluster-specific RCTs and were excluded from the error analysis in Steps 3 and 4 of Algorithm 1.

Algorithm 1 Procedure to compute oracle cluster-dependent WER

```
1: Speaker Clusters $\mathcal{C}(\mathcal{S})$: Perform speaker clustering on speaker population $\mathcal{S}$ to produce $k$ clusters. $\mathcal{C}(\mathcal{S})$ is the assignment of speakers in $\mathcal{S}$ to the $k$ clusters. Using $\mathcal{C}(\mathcal{S})$, derive cluster assignments $\mathcal{C}(\mathcal{S}_{\mathcal{H}})$ and $\mathcal{C}(\mathcal{S}_{\mathcal{T}})$ for speaker subsets $\mathcal{S}_{\mathcal{H}}$ and $\mathcal{S}_{\mathcal{T}}$, respectively.
2: Train constrained or unconstrained RCTs $\mathcal{T}_1$, one for each speaker cluster using $\mathcal{S}_{\mathcal{T}}$.
3: 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 whose RCT in $\mathcal{T}_1$ produces the lowest WER after MLLR adaptation.
4: Compute the overall WER for each of these new clusters in $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$.
```

word error rate (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. 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, we 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 RCTs and for brevity, are not presented here.

To estimate realistic ASR performance gains from using oracle cluster-specific RCTs, we excluded the speakers in the held-out test set $\mathcal{S}_{\mathcal{H}}$ from the speaker clustering training data $\mathcal{S}$ in Step 1 of Algorithm 1 in our experiments with English BN. The speaker subset $\mathcal{S}$ is the same as $\mathcal{S}_{\mathcal{T}}$, i.e., ($\mathcal{S} = \mathcal{S}_{\mathcal{T}}$) and is composed of 25 hours of English BN data released by LDC in 2006. The

	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**

Table 1

Oracle WER(%) for English CTS using unconstrained RCT

held-out speaker subset $\mathcal{S}_{\mathcal{H}}$ comprised the NIST 2004 English BN test set. On performing the steps described in Algorithm 1, we observed performance improvements similar to that of the English CTS case. The gains compared to the SI RCT vary from 0.5% to 0.8%, with the exception of one cluster, which showed no improvement. On closer examination, we found that the structure of the RCT for this cluster was similar to that of the SI RCT, which is the reason for no additional performance gains.

The cluster assignment of speakers in the held-out subset $\mathcal{S}_{\mathcal{H}}$ changed for many speakers between Steps 1 and 4 of Algorithm 1. Since the assignment of speakers to clusters in $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$ is based on minimum WER (rather than minimum

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

```
1: Given from Algorithm 1 the initial cluster assignments $\mathcal{C}(\mathcal{S})$ and $\mathcal{C}(\mathcal{S}_{\mathcal{T}})$ of the entire speaker population $\mathcal{S}$ and the training speaker population $\mathcal{S}_{\mathcal{T}}$, respectively (from Step 1), the RCTs $\mathcal{T}_1$ (from Step 2), and the updated cluster assignment $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$ of the held-out speaker population $\mathcal{S}_{\mathcal{H}}$ (from Step 3).
2: Produce new cluster assignments $\mathcal{C}_1(\mathcal{S}_{\mathcal{T}})$ 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. Number of speaker clusters remains unchanged from Algorithm 1.
3: Train *constrained* or *unconstrained* RCTs $\mathcal{T}_2$, one for each speaker cluster in $\mathcal{C}_1(\mathcal{S}_{\mathcal{T}})$, using the data of training speakers in that cluster.
4: Compute the overall WER for each speaker cluster $\mathcal{C}_1(\mathcal{S}_{\mathcal{H}})$ comprising speaker subset $\mathcal{S}_{\mathcal{H}}$ using the WER of each speaker in a cluster after MLLR adaptation using each cluster-specific RCT in $\mathcal{T}_2$.
```

squared error on transformations[9]), a re-clustering approach based on this criterion was explored, as detailed in Algorithm 2. A new assignment, $\mathcal{C}_1(\mathcal{S}_{\mathcal{T}})$,

[9] A test speaker in $\mathcal{S}_{\mathcal{H}}$ (Step 1 of Algorithm 1) can be assigned to that cluster whose RCT in $\mathcal{T}_1$ produces an MLLR mean transformation that is "closest" to a canonical MLLR mean transformation of that cluster, with respect to the squared distance between the two transformations (when each is vectorized). A canonical MLLR transformation for a speaker cluster in $\mathcal{C}(\mathcal{S}_{\mathcal{T}})$ can be estimated as the centroid of the MLLR transformations of the subset of speakers of $\mathcal{S}_{\mathcal{T}}$ assigned to that cluster and its corresponding RCT.

of speaker-clustering training speakers to that cluster whose tree produced the lowest WER was used, 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 2, do not exhibit patterns similar to those in Table 1. The difference in the two sets of results may also indicate that speaker variability for MLLR adaptation strategies can be better modeled by speaker clustering in the eigenspace of MLLR transformations, than by clustering speakers based on minimum WER, or it is perhaps the case that WER-based clustering criteria are too greedy.

	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

Table 2

Oracle WER(%) for English CTS using retrained unconstrained RCT. Lowest WERs in each row are highlighted.

4.2 Analysis of Regression Tree Structure

On manually examining the cluster-specific RCT we observed that a different sequence of questions was used by each constrained RCT, resulting in different structures of each tree. Since the structure of RCT describes similarities among clusters of phones (based on phone-level statistics), we conjecture 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, we found that the branches of the trees that split the acoustic units describing vowels (Figure 1) exhibited more differences in the hierarchical structure than the branches involving consonants, which is consistent with linguistic studies on regional variation in American English (Labov, 1996). For example, the relative proximity of “ao” and “aa” in the upper tree of Figure 1 compared to that in the lower tree is indicative of dialectal variation per Labov’s findings. 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 10 in the Appendix).

5 Soft Regression Class Trees

5.1 Maximum Likelihood Weights for Transform Combination

The experiments in Section 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. We compare 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 a speaker’s adaptation data.

Combinations of multiple MLLR transformations using ML weights have been proposed previously by Gales (1996); Boulis et al. (2001); Sankar et al. (1999). The component MLLR transformations have been estimated dynamically for test speakers using different nodes within a single RCT (Gales, 1996), or precomputed for speaker clusters using various techniques (Boulis et al., 2001), 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 (Gales, 1996; Boulis et al., 2001), or precomputed using an ML approach on a corpus of training speakers (Gales and Woodland, 1996). Our approach estimates the component MLLR transformations for each cluster-specific RCT (trained offline) using a test speaker's adaptation data. The optimal weights to combine the component transformations are estimated to maximize likelihood of the test data.

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 *constrained* and *unconstrained* RCT used in this work clusters all SI Gaussian distributions that belong to triphone states with the same center phone at each leaf node, resulting in each cluster-specific RCT to have the same number of leaf nodes, which is equal to the number of phones being used. This implies that for S cluster-specific RCTs, each with L leaf nodes, a set of S leaf nodes can be drawn (one from each RCT) such that each leaf node corresponds to the same phone. Such a set of S leaf nodes can be considered as an equivalence class since they would cluster the same set of SI Gaussian distributions corresponding to a particular phone.[10] This equivalence relationship can be represented by a mapping matrix, which has S rows and L columns. Each row corresponds to the S cluster-specific RCTs and each column the L leaf nodes. The S entries in the lth column would denote the indices of the leaf nodes, corresponding to each RCT, that form an equivalence class. Then the weights for combining MLLR mean transformations can be estimated for the S leaf nodes in an equivalence class, which we refer to as weights being tied at the leaves of the RCTs. Representing the weights $\alpha_s^{(l)}$ for the s-th RCT and the l-th equivalence class (leaf nodes) by

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

and using a procedure similar to that of Gales (1996), we define the auxiliary function of interest in Eqn. 1 for the expectation-maximization algorithm (EM) (Dempster et al., 1977)

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

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. Differentiating Eqn. 1 with respect to $\hat{\alpha}_s^{(l)}$ results in Eqn. 2

[10] The set of root nodes of each cluster-specific RCT also form an equivalence class since they cluster all the SI Gaussian distributions.

$$\left[\sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\hat{\mu}_m^{(s)T}\Sigma_m^{-1}\hat{\mathbf{M}}_m\right]\hat{\alpha}^{(l)} = \sum_{r=1}^{R}\sum_{c=1}^{C_r}\sum_{m=1}^{M_c}\sum_{\tau=1}^{T}\gamma_m(\tau)\hat{\mu}_m^{(s)T}\Sigma_m^{-1}\mathbf{o}(\tau) \ . \tag{2}$$

By differentiating Eqn. 1 with respect to each of the S weights $\hat{\alpha}_s^{(l)}$ a set of simultaneous equations can be generated as shown in Eqn. 3:

$$\mathbf{Z}^{(l)}\hat{\alpha}^{(l)} = \mathbf{V}^{(l)} \ , \tag{3}$$

where $\mathbf{Z}$ is a symmetric $S \times S$ matrix and $\mathbf{V}$ is an $S \times 1$ vector. Each row of $\mathbf{Z}$ and the corresponding element of $\mathbf{V}$ is computed by differentiating Eqn. 2 with respect to each of the S weights $\hat{\alpha}_s^{(l)}$. The weights $\hat{\alpha}_s^{(l)}$ can then be estimated by solving for $\hat{\alpha}^{(l)}$ using Eqn. 3 that requires the inversion of matrix $\mathbf{Z}$.

If the branching structure of the cluster-specific RCTs is such that the paths in two or more RCTs leading to leaf nodes (that form an equivalence class) are identical, then the cluster-specific MLLR transformations estimated at any node in such branches will be identical in each RCT. For such an equivalence class, matrix $\mathbf{Z}$ in Eqn. 2 will have numerically identical rows corresponding to the leaf nodes with identical branching structure. With some rows being identical, matrix $\mathbf{Z}$ will no longer have linearly independent rows and will not have full rank. In such cases, an infinite number of solutions will exist for the weights $\hat{\alpha}_s^{(l)}$. In our approach, we estimate the weights only for those clusters that have unique cluster-specific MLLR transformations, and assume the rest of the weights to be zero. Effectively, this approach ignores redundant cluster-specific MLLR transformations when estimating the transformation smoothing weights. As part of this procedure we introduce cluster-specific Lagrange multipliers, $\lambda^{(l)}$ and $\beta_s^{(l)}$ for the weights $\hat{\alpha}^{(l)}$ into the objective function of Eqn. 4 that places inequality constraints on the weights such that they are positive and an equality constraint that they sum to one.[11]

$$
\begin{aligned}
\mathcal{Q}(\mathcal{M}, \hat{\mathcal{M}}) = K - \frac{1}{2} \sum_{r=1}^{R} \sum_{c=1}^{C_r} \sum_{m=1}^{M_c} \sum_{\tau=1}^{T} \gamma_m(\tau) \quad \Big(\mathbf{o}(\tau) - \hat{\mu}_m\Big)^T \Sigma_m^{-1} \Big(\mathbf{o}(\tau) - \hat{\mu}_m\Big) \\
+ \sum_{l=1}^{L} \lambda^{(l)} (\sum_{s=1}^{S} \alpha_s^{(l)} - 1) + \beta_s^{(l)} (\alpha_s^{(l)} \geq 0) \ . \qquad (4)
\end{aligned}
$$

When an inequality constraint becomes active, it becomes an equality or the cluster-specific weight it corresponds to is zero, while the remaining weights are non-zero. To estimate the weights with inequality constraints, the procedure keeps the inequality constraint inactive for those clusters that have unique cluster-specific MLLR transformations and sets the constraint to be active for the rest of the clusters. The cluster-specific weights are then estimated under the constraint that they sum to one. It is straightforward to solve for the Lagrange multipliers, and the details are omitted here.

5.2 Two-step ML Weight Estimation Strategy

For each test speaker, we apply a two-step ML weight estimation procedure. First, we estimate the mean and diagonal variance MLLR transformations for every cluster-specific RCT from HMM state occupancy statistics collected

[11] Note that the maximum likelihood solutions of weights obtained by solving Eqn. 3 do not have any constraints on the weights.

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 5.1) and the 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 RCT, we estimate the smoothing weights with inequality constraints (second step), and the corresponding diagonal variance transformation. Depending on the set of smoothing weights chosen, either from the first or the second step, the corresponding combined mean transformations and diagonal variance transformations are used to adapt the SI acoustic model.

5.3 Limitations of the Two-step ML Weight Estimation Strategy

Under conditions described in Sec. 5.1, the matrix $\mathbf{Z}$ in Eqn. 2 can have rows that are not linearly independent.[12] In such cases, we extended the ML procedure of Gales (1996) for estimating the cluster-specific weights by adding inequality constraints using Lagrange multipliers and estimating only the weights for those clusters that have unique cluster-specific MLLR transformations. This is one approach for robustly estimating the cluster-specific weights

[12] In our experiments, we did not encounter situations when matrix $\mathbf{Z}$ had full rank in theory, but in practice could not be inverted on finite-precision machines.

in a ML framework, but several other alternatives exist. These include generalized EM approaches that increase the likelihood of the objective function in Eqn. 1 without requiring inversion of the matrix $\mathbf{Z}$. Another possibility is to use an iterative approach to estimate each cluster-specific weight $\hat{\alpha}_s^{(l)}$ individually, keeping the others fixed, which also does not require inversion of the matrix $\mathbf{Z}$. The use of inequality constraints in our approach, which constrains the weights to be non-zero and sum to one, does not achieve the theoretical maximum that is possible without the constraints. However, the constrained weight estimation results in robust adaptation performance across a wide-range of speakers as shown in experimental evidence in Sections 6.4.[13]

5.4 Computation Requirements

Compared to standard MLLR adaptation with a single transformation (mean and variance), the computational cost of adaptation using multiple cluster-dependent RCT structures increases linearly with the number of clusters. With S clusters, $S+1$ transformations are estimated (including one for the SI RCT). For example, if there is only one iteration of MLLR estimation and the robust ML weight estimation strategy (Section 5.2) 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 Section 5.1, the ML weights are estimated only 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.

[13] An additional reason for using the constrained weight estimation scheme was the ease of its software implementation in our ASR engine.

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.

6 Recognition Experiments with Soft Regression Trees

6.1 Baseline Regression Class Tree Performance

We first present the baseline ASR system performance after MLLR adaptation (as described in Section 3) when using the different types of RCTs trained for this work. As mentioned earlier, we examined four different types of RCTs: *unconstrained*, *constrained*, and *unconstrained state-level* (described in Section 2.1) and a *manually* designed tree (using knowledge of acoustic phonetics as described in Section 3). Tables 3 and 4 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 3 and 4, "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").

	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 3

WER(%) after MLLR adaptation using four different RCT building schemes on the 2004 English BN test set

The results presented in Tables 3 and 4 show that the performance improve-

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

Table 4

WER(%) after MLLR adaptation using four different RCT building schemes on the 2003 English CTS test set

ments obtained from each of the three types of automatically built RCTs (*unconstrained*, *unconstrained state-level* and *constrained*) are similar. Given this evidence, it is unlikely that the use of the two-step ML procedure of Section 5.2 will yield any additional benefits with the *unconstrained state-level* type RCTs compared to using the two-step ML procedure with the *unconstrained* and *constrained* type RCTs. In addition, the *unconstrained state-level* type RCTs have higher degrees of freedom (or number of leaf nodes) since each leaf node represents a triphone HMM state, compared to the *unconstrained* and *constrained* type RCTs, which have one leaf per phone.[14] The higher degrees of freedom make it likely that the MLLR transformations estimated with the *unconstrained state-level* type RCT and the two-step ML procedure will over-train on errorful adaptation hypotheses. 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. Because of this, we will focus only on the *unconstrained* and *constrained* type RCTs for subsequent ASR experiments.

[14] 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.

6.2 Average ASR Performance Results

To evaluate the performance of the two-step ML procedure for combining MLLR transformations from speaker-clustered (SC) RCT, described in Section 5.2, we tested its performance on a range of standard NIST test sets. As mentioned in Section 4, for each domain we trained 4 cluster-specific RCTs, one for each speaker cluster, of both constrained and unconstrained types, and combined transformations using the two-step ML procedure. In Tables 6, 7 and 8, the columns denoted by "Unconstr." and "Constr." refer to the unconstrained and constrained RCT, respectively. The row labels refer to experiment configurations, which are explained in Table 5.

Experiment Name	Experiment Configuration
Adapt Hyps	WER of adaptation hypothesis
SI	speaker-independent automatically built 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 an extra weight for the case of no adaptation
Oracle SC	cluster-specific RCT that achieves the lowest WER

Table 5

Various configurations for ASR experiments

Table 6 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 *constrained* and *unconstrained* RCTs, using the two-step ML procedure, compared to using only one SI RCT. The improvement of 0.2% (absolute) for the configuration "Soft SC (leaves)" using the unconstrained RCT is significantly different from the "SI" case ($p < 0.01$).[15]

The ASR performance level from tying the ML weights at the root and at the leaves of the cluster-specific RCT indicates that both are equivalent in performance. 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 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 4 that the overall WER can be reduced significantly by choosing the optimal RCT.

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 7, 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% to 0.2% (absolute) for *constrained* and *unconstrained* RCTs, though these are not statistically significant.

We also tested the performance the of speaker-clustered RCT on the NIST

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

	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

Table 6

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

2006 Mandarin BN and 2005 BC (development) test sets using the Mandarin BN/BC ASR system. The results are shown in Table 8, where the constrained SI RCT achieved improvements of 0.3% (absolute) for both test sets, compared to the baseline (manually designed[16]) RCTs, which are statistically significant at the level of $p < 0.002$. The use of the 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.[17] As in the case of the speaker-clustered RCT for the English domains,

[16] The manually designed RCT had three leaf classes: vowels, consonants and non-speech organized into a tree.

[17] The linguistic question set used for building the constrained RCT is not as richly developed for Mandarin as is the case for English, which provides an explanation for

	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**

Table 7

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

the cluster-specific constrained RCT for Mandarin exhibited differences mainly in the vowel branches, and the structure of the cluster-specific unconstrained RCT shows more diversity compared to the constrained RCT.

Last, 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 6, 7 and 8. The motivation for this experiment is our observation in earlier work (Mandal et al., 2005, 2006) that 10% to 15% of 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 the 2003 English CTS test set and the

the better ASR performance when using the unconstrained RCT in the Mandarin case.

unconstrained type RCT for the 2006 Mandarin BN test set, compared to the "Soft SC" case.

	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

Table 8

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

6.3 Performance Analysis of Two-Step ML Weight Estimation

As mentioned earlier, this work extends the weight estimation framework of Gales and Woodland (1996) by introducing a robust two-step ML weight estimation procedure with or without inequality constraints. We compared the performance of the two-step procedure to the one-step procedure that used unconstrained ML weights for a combination of cluster-specific MLLR mean transformations. The results shown in Table 9 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.

	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

Table 9

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

6.4 WER Distribution Analysis

Since the overall gains from using the speaker-clustered RCT are small, we analyzed the results of 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, or consistent, improvements across speakers.

In Figures 2 and 3, we have ordered the speakers in the NIST 2004 English BN (234 speakers) and 2003 English CTS (144 speakers) test sets by duration of adaptation data.[18] In Figure 4, we show speaker-level analysis of WER change from adaptation, relative to the unadapted case, with the same ordering of speakers as in Figure 2, using both the SI RCT (left) and speaker-clustered RCT (right) for speakers in the NIST 2004 English BN test set. The plots in Figure 4 indicate that, contrary to expectations, not all speakers benefit

[18] Total length of waveforms after segmentation

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. We can also see in Figure 4 that fewer speakers have performance losses (relative WER change > 0) when the speaker-clustered RCTs 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.

Based on the error reductions with the oracle cluster-specific RCT in Section 4 and the trends seen in Figure 4, we conclude that using multiple speaker-clustered RCTs leads to more robust adaptation strategies, compared to a single SI RCT.

We present quantitative evidence for this conclusion in Tables 10 and 11, which show the difference between the percentage of speakers who benefit from adaptation and the percentage of speakers who have degraded ASR system performance due to adaptation (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 RCTs (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 $p < 0.15$.[19] 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 10, 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 our observation that the optimal RCT structure varies across speakers. Table 10 also shows that using multiple speaker-clustered RCTs, a greater net percentage of speakers benefits from adaptation, compared to both the "SI" and "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 is shown graphically in Figures 5 and 6 with the ordering of speakers the same as in Figures 2 and 3, respectively. Again, we observe that fewer speakers have degraded ASR system performance due

[19] 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 that this is a simple, weaker significance test than the matched pairs test used with earlier results.

to adaptation with “Soft SC”. In Figure 5, in the graph plotting only speakers with significant performance changes from the speaker-clustered RCT in adaptation, there is one speaker who shows a large relative performance loss (59%). On examining the performance patterns for this speaker, we 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 RCT configuration with which we experimented. This speaker is particularly disfluent, but the unusually poor performance most likely occurs because he is grouped in a cluster with another speaker who has a lot of background noise (e.g., keyboard clicks) that can negatively affect the adaptation transformations.

We performed the same analysis 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 11, 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 small amounts of adaptation data.

We present the information shown in Figure 4 again in Figure 8, with the only difference being that the speakers are now ordered by their unadapted WER (ordering shown in Figure 7). The graphs in Figure 8 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

	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 10

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	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

Table 11

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

SC" cases, respectively. In Figure 9, we show only those speakers whose relative WER change from adaptation is significant (at the level of $p < 0.15$) with the same ordering of speakers as in Figure 7. It can be seen from both Figure 8 and Figure 9, that the average performance loss (WER increase) is lower (11% vs. 18%) when using the speaker-clustered RCT with MLLR adaptation than when using the SI RCT.

6.5 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 12. 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 are shown in row 3 ("SI") of Table 12. Then, we explored two possibilities to smooth the initial MLLR mean transformations with 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 12 and row 6, where "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 of "Soft SC" is the same. More important, 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 improvement is achieved by the

"Soft SC" case compared to the "SI" case for these 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 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 [20] in conversations.

	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

Table 12

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

7 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. We have

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

presented evidence 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 fine-grained speaker-specific adaptation strategies. We have introduced a speaker clustering algorithm 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, we noticed that more diversity was exhibited by the vowel branches than the consonant branches, which we conjecture to be indicative of dialectal variations in the training speaker population.

To take advantage of the speaker-clustered RCT in evaluating ASR systems, we use a procedure 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. From a visual inspection, [21] we found that the differences in the structure of the cluster-specific unconstrained RCT for Mandarin BN was more marked than the English BN

[21] Examining the phone clusters produced at each split in the RCT.

case. Thus, in the Mandarin BN case, the cluster-specific MLLR transformations used in linear combination are more different than in the English case.

Further, we 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. We also observed that the speaker-clustered RCT benefited the majority of the speakers who had degraded ASR system performance due to MLLR adaptation with a single SI RCT, and reduced the average performance loss for those speakers who had degraded ASR system performance due to MLLR adaptation.

In future work, we plan to explore auxiliary speaker-level features that are more relevant for predicting the optimal RCT structure to use for individual speakers, similar to the way we predicted the optimal number of regression classes in Mandal et al. (2005). Also, we want 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, we hope to capture more diverse structures in the ensemble of RCTs, further improving robustness of the proposed method.

Acknowledgments

A shorter version of this paper, with English CTS experiments only, appeared as Mandal et al. (2006). The authors thank Ramana Gadde and Jing Zheng for their helpful suggestions with MLLR implementation details in the Decipher system. This work was funded by DARPA under contract No. HR0011-06-C-0023. Distribution is unlimited. The views herein are those of the authors and do not reflect the views of the funding agencies.

References

Anastasakos, T., McDonough, J., Makhoul, J., 1997. Speaker adaptive training: A maximum likelihood approach to speaker normalization. In: Proc. of ICASSP. Vol. 2. pp. 1043–1046.

Bocchieri, E., Digalakis, V., Corduneanu, A., Boulis, C., 1999. Correlation modeling of MLLR transform biases for rapid HMM adaptation to new speakers. In: Proc. of ICASSP. Vol. 2. pp. 773–776.

Boulis, C., Diakoloukas, V., Digalakis, V., 2001. Maximum likelihood stochastic transformations adaptation for medium and small data sets. Computer Speech & Language 15 (3), 257–287.

Chen, K. T., Liau, W. W., Wang, H. M., Lee, L. S., 2000. Fast speaker adaptation using eigenspace-based maximum likelihood linear regression. In: Proc. of ICSLP. Vol. III. pp. 742–745.

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

Dempster, A., Laird, N., Rubin, D., 1977. Maximum likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society 39 (1), 1–38.

Digalakis, V., Rtischev, D., Neumeyer, L., 1995. Speaker adaptation using constrained estimation of Gaussian mixtures. IEEE Transactions on Speech and Audio Processing 3 (5), 357–366.

Ferrer, L., Sönmez, K., Kajarekar, S., 2005. Class-based score combination for speaker recognition. In: Proc. of Eurospeech. pp. 2173–2176.

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

Gales, M., 1997. Transformation smoothing for speaker and environmental adaptation. In: Proc. of Eurospeech. Vol. 4. pp. 2067–2070.

Gales, M., 1998. Maximum likelihood linear transformations for HMM-based speech recognition. Computer Speech & Language 12, 75–98.

Gales, M., 2000. Cluster adaptive training of hidden Markov models. IEEE Transactions on Speech and Audio Processing 8 (4), 417–428.

Gales, M., Woodland, P., 1996. Mean and variance compensation within the MLLR framework. Computer Speech & Language 10, 249–264.

Haeb-Umbach, R., 2001. Automatic generation of phonetic regression class trees for MLLR adaptation. Speech and Audio Processing, IEEE Transactions on 9 (3), 299–302.

Huang, C., Chen, T., Li, S., Chang, E., Zhou, J., 2001. Analysis of speaker variability. In: Proc. of Eurospeech. Vol. 2. pp. 1377–1380.

Hwang, M.-Y., Huang, X., 12-15 May 1998. Dynamically configurable acoustic models for speech recognition. In: Acoustics, Speech, and Signal Processing, 1998. ICASSP '98. Proceedings of the 1998 IEEE International Conference on. Vol. 2. pp. 669–672vol.2.

Hwang, M.-Y., Lei, X., Wang, W., Shinozaki, T., 2006. Investigation on Mandarin broadcast news speech recognition. In: Proc. of ICSLP. pp. 1233–1236.

Imamura, A., 1991. Speaker adaptive HMM-based speech recognition with a stochastic speaker classifier. In: Proc. of ICASSP. Vol. 2. pp. 841–844.

Kannan, A., Ostendorf, M., Rohlicek, J. R., July 1994. Maximum likelihood clustering of gaussians for speech recognition. IEEE Transactions in Speech and Audio Processing 2 (3), 453–455.

Kosaka, T., Sagayama, S., 1994. Tree structured speaker clustering for fast speaker adaptation. In: Proc. of ICASSP. Vol. 1. pp. 245–248.

Kuhn, R., Junqua, J., Nguyen, P., Niedzielski, N., 2000. Rapid speaker adaptation in eigenvoice space. IEEE Transactions on Speech and Audio Processing 8 (6), 695–707.

Labov, W., 1996. The organization of dialect diversity in North America. In: Fourth International Conference on Spoken Language Processing.

Leggetter, C., 1995. Improved acoustic modelling for HMMs using linear transformations. Ph.D. thesis, University of Cambridge.

Leggetter, C., Woodland, P., 1995. Maximum likelihood linear regression for speaker adaptation of HMMs. Computer Speech & Language 9, 171–185.

Mak, B., Hsiao, R., 2004. Improving eigenspace-based MLLR adaptation by kernel PCA. In: Proc. of ICSLP. Vol. I. pp. 13–16.

Mandal, A., Ostendorf, M., Stolcke, A., 2005. Leveraging speaker-dependent variation of adaptation. In: Proc. of Eurospeech. pp. 1793–1796.

Mandal, A., Ostendorf, M., Stolcke, A., 2006. Speaker clustered regression-class trees for MLLR adaptation. In: Proc. of ICSLP. pp. 1133–1136.

Mardia, K., Kent, J., Bibby, J., 1979. Multivariate Analysis. Academic Press.

Padmanabhan, M., Bahl, L., Nahamoo, D., Picheny, M., 1998. Speaker clustering and transformation for speaker adaptation in speech recognition systems. IEEE Transactions on Speech and Audio Processing 6 (1), 71–77.

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

Sankar, A., Beaufays, F., Digilakis, V., 1995. Training data clustering for improved speech recognition. In: Proc. of Eurospeech. Vol. 1. pp. 502–505.

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

Sankar, A., Neumeyer, L., Weintraub, M., 1996. An experimental study of acoustic adaptation algorithms. In: Proc. of ICASSP. Vol. 2. pp. 713–716.

Stolcke, A., Chen, B., Franco, H., Gadde, R., Graciarena, M., Hwang, M. Y., Kirchoff, K., Lei, X., Mandal, A., Morgan, N., Ng, T., Ostendorf, M., Sonmez, K., Venkataraman, A., Vergyri, D., Wang, W., Zheng, J., Zhu, Q., 2006. Recent innovations in speech-to-text transcription at SRI-ICSI-UW. IEEE Transactions on Audio, Speech and Language Processing 14 (5), 1729–1744.

Stolcke, A., Ferrer, L., Kajarekar, S., Shriberg, E., Venkataraman, A., 2005. MLLR transforms as features in speaker recognition. In: Proc. of Eurospeech. pp. 2425–2428.

Venkataraman, A., Stolcke, A., Wang, W., Vergyri, D., Gadde, V., Zheng, J., 2004. SRI's 2004 broadcast news speech to text system. In: EARS RT04 Workshop.

Woodland, P., Gales, M., Pye, D., 1996. Improving enviromental robustness in large vocabulary speech recognition. In: Proc. of ICASSP. Vol. 1. pp. 65–68.

Young, S., Odell, J., Woodland, P., 1994. Tree based state tying for high accuracy modelling. In: Proc. ARPA Spoken Language Technology Workshop. pp. 405–410.

Appendix

Unconstrained Trees for English CTS

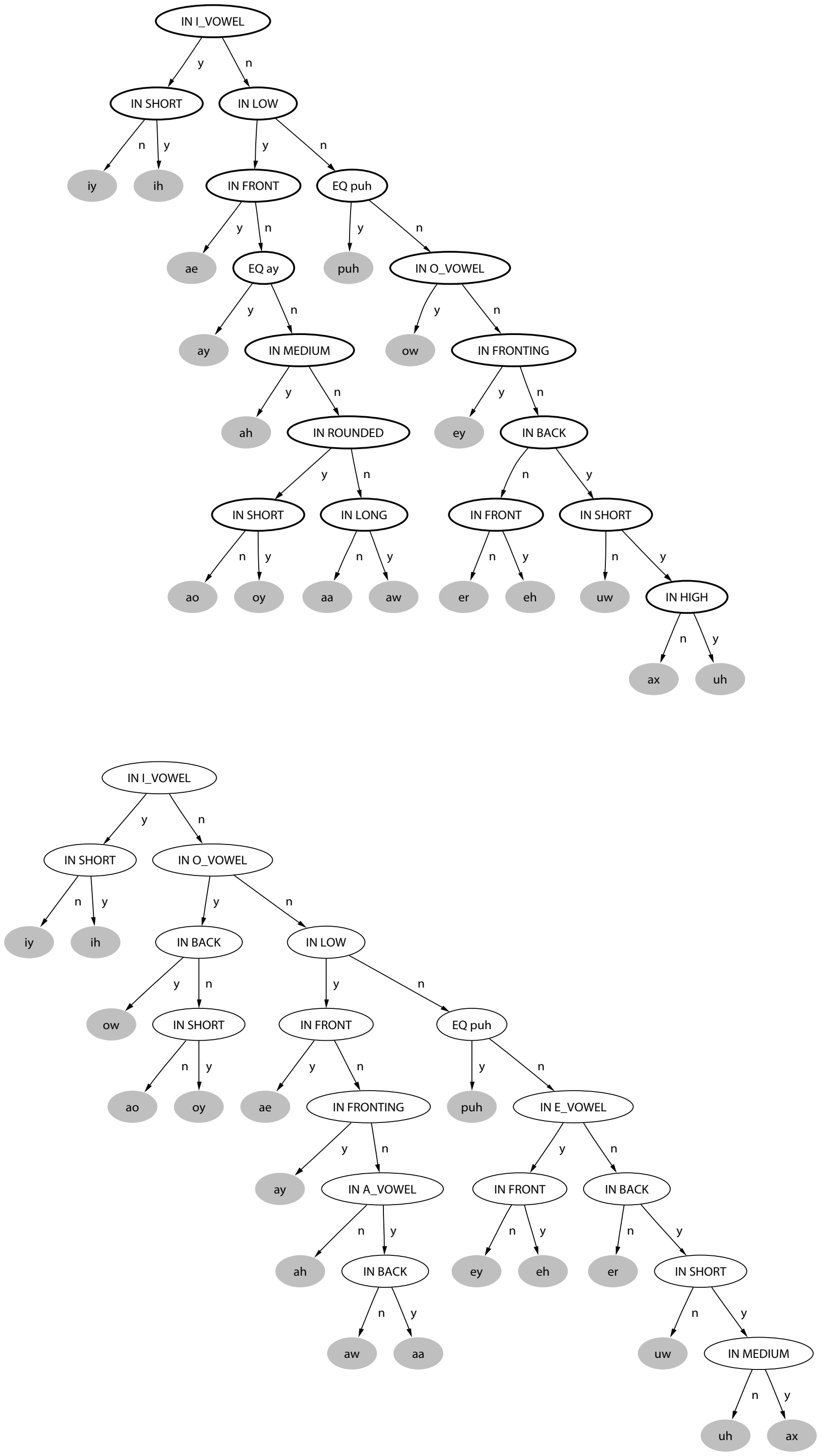

Fig. 1. Vowel branches of the constrained RCT for two different English CTS Male clusters

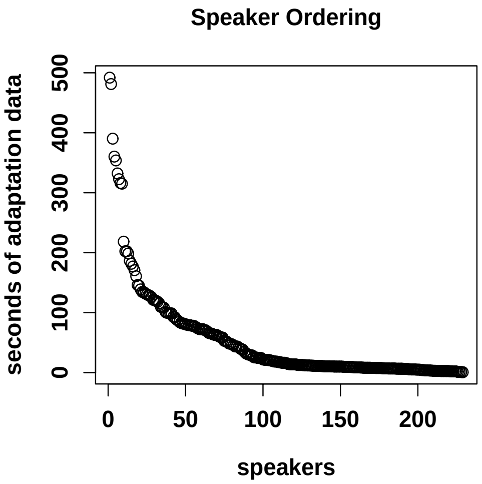

Fig. 2. Speakers ordered by amount of adaptation data in seconds (2004 English BN test set)

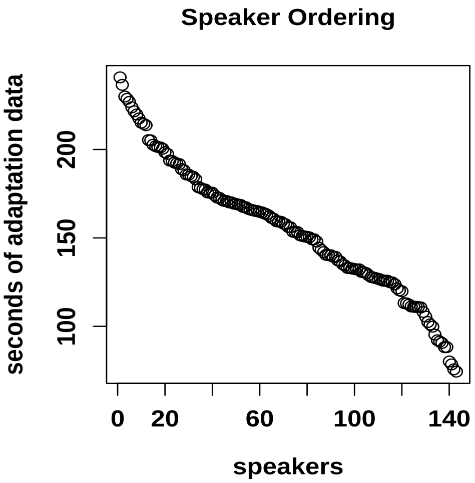

Fig. 3. Speakers ordered by amount of adaptation data in seconds (2003 English CTS test set)

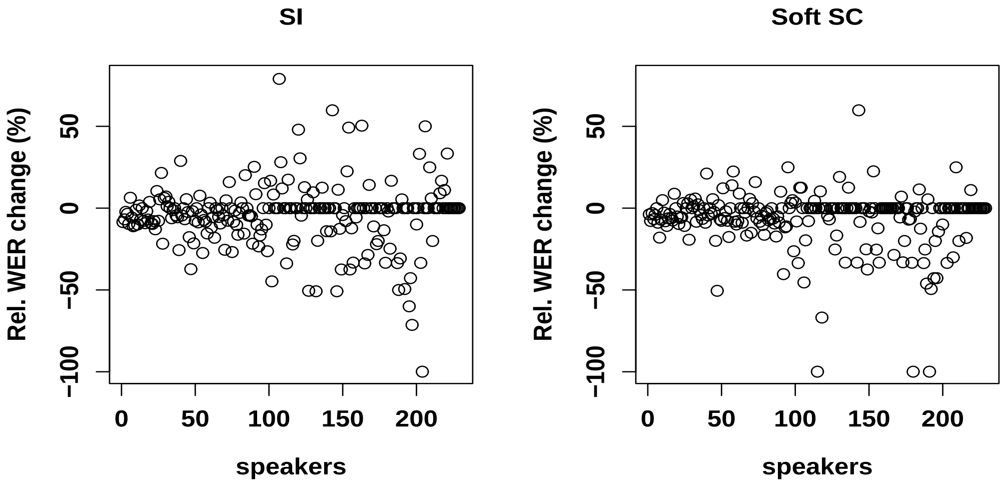

Fig. 4. Relative change in WER for all speakers in the NIST 2004 English BN test set, ordered by decreasing amount of adaptation data

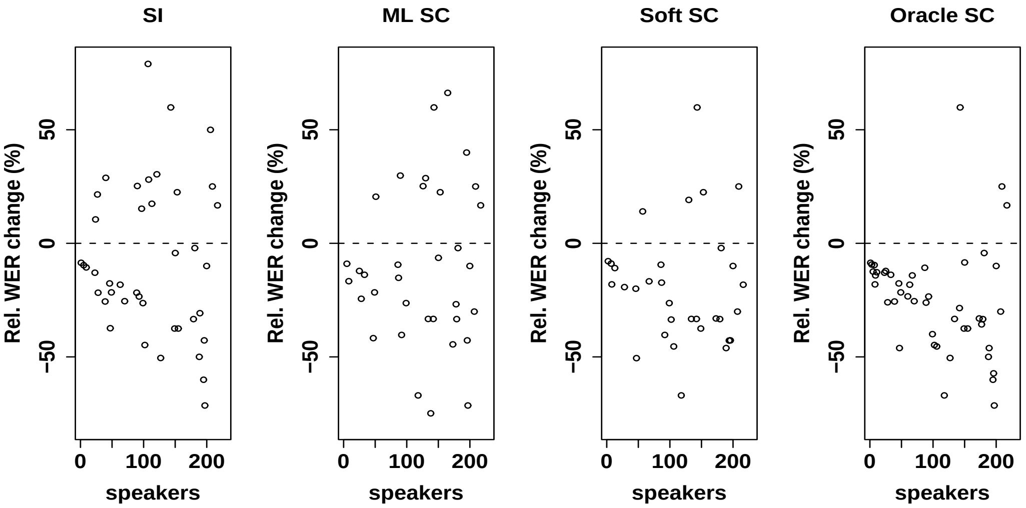

Fig. 5. Significant ($p < 0.15$) performance changes of speakers from adaptation with various tree configurations (NIST 2004 English BN test set)

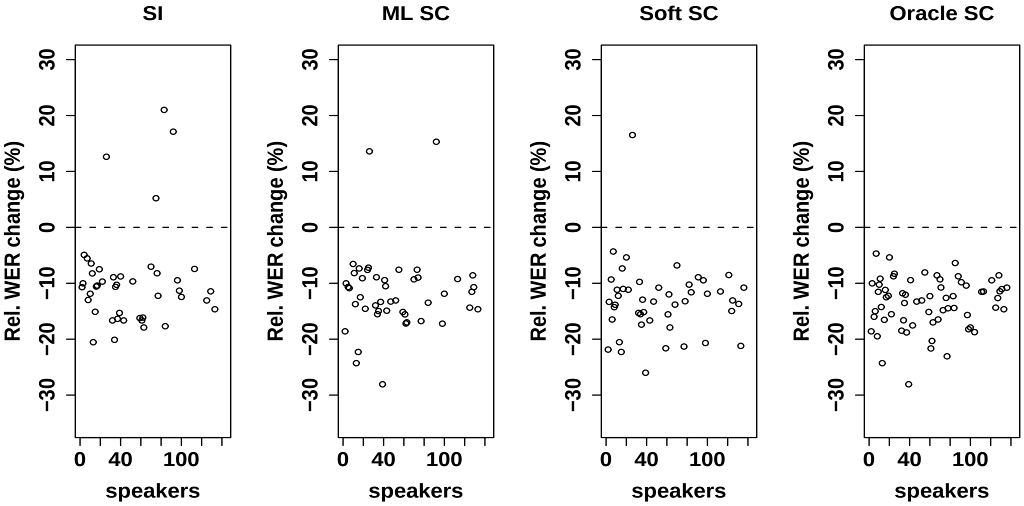

Fig. 6. Significant ($p < 0.15$) performance changes of speakers from adaptation with various tree configurations (NIST 2003 English CTS test set)

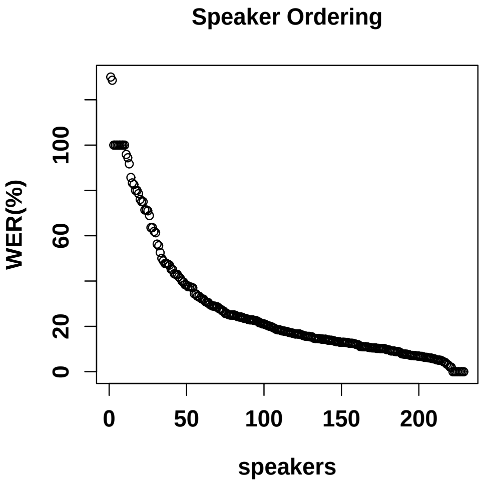

Fig. 7. Speakers ordered by unadapted WER (2004 English BN test set)

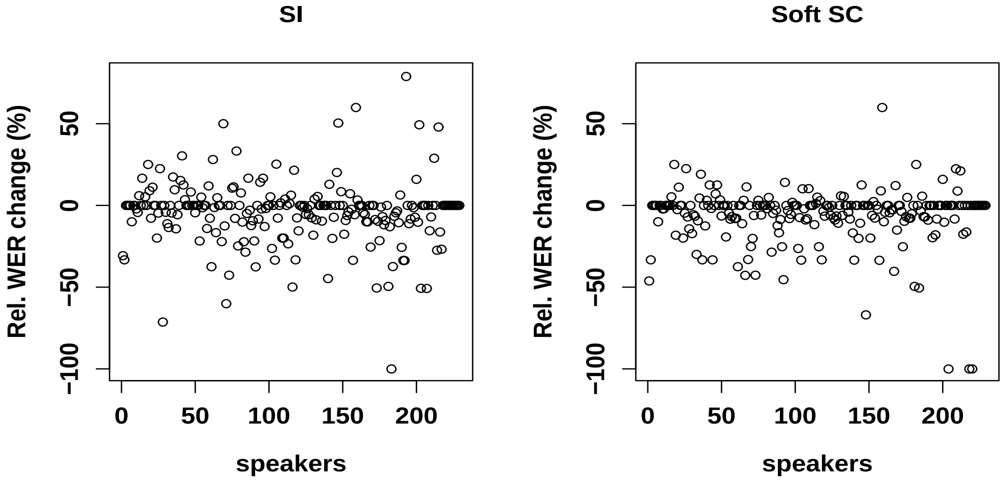

Fig. 8. Effect of unadapted WER on adaptation success (2004 English BN test set)

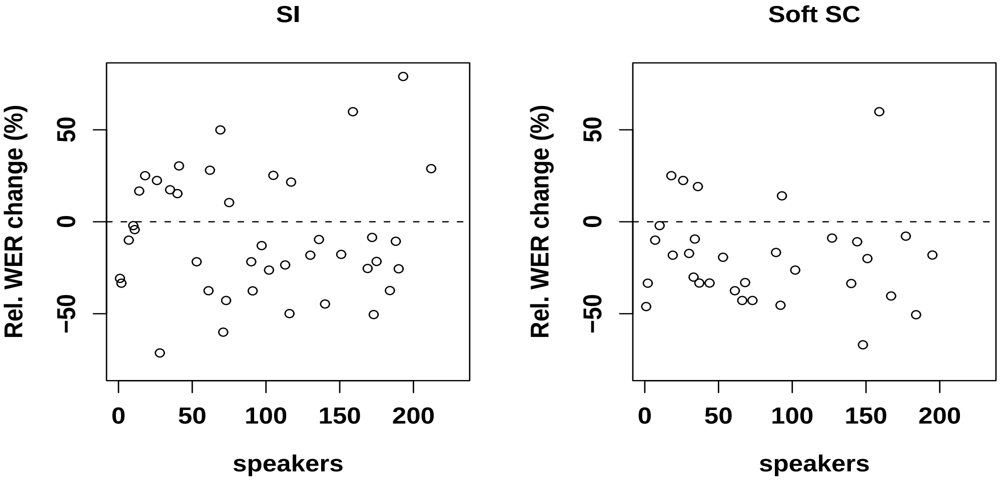

Fig. 9. Speakers with significant ($p < 0.15$) performance change from adaptation (NIST 2004 English BN test set)

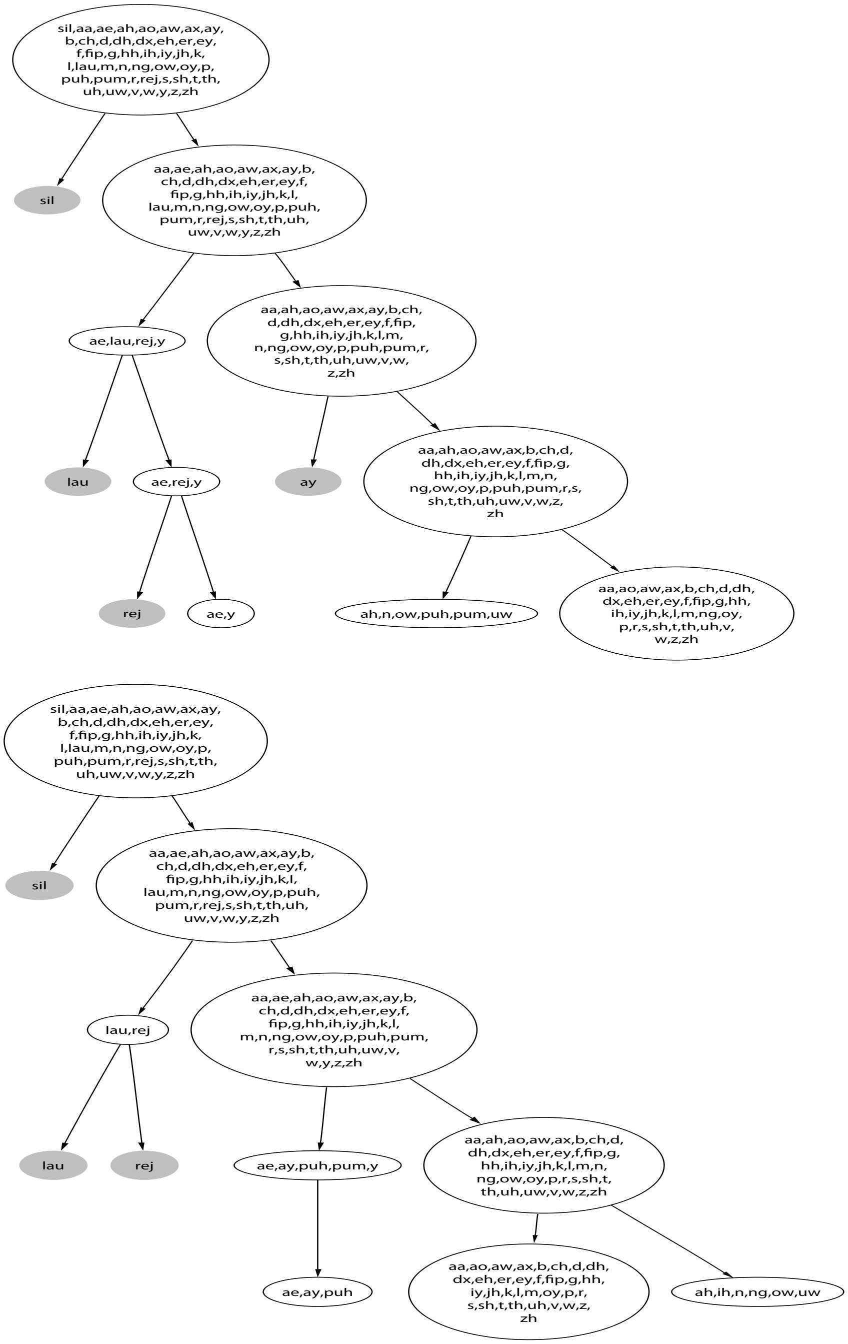

Fig. 10. Top-levels of unconstrained RCT for two different clusters of English CTS (Female)