Tree-Based Estimation of Speaker Characteristics for Speech Recognition

Mats Blomberg, Daniel Elenius

Department of Speech, Music and Hearing, KTH/CSC, Stockholm, Sweden
{matsb, daniele}@speech.kth.se

Abstract
Speaker adaptation by means of adjustment of speaker characteristic properties, such as vocal tract length, has the important advantage compared to conventional adaptation techniques that the adapted models are guaranteed to be realistic if the description of the properties are. One problem with this approach is that the search procedure to estimate them is computationally heavy. We address the problem by using a multi-dimensional, hierarchical tree of acoustic model sets. The leaf sets are created by transforming a conventionally trained model set using leaf-specific speaker profile vectors. The model sets of non-leaf nodes are formed by merging the models of their child nodes, using a computationally efficient algorithm. During recognition, a maximum likelihood criterion is followed to traverse the tree. Studies of one- (VTLN) and four-dimensional speaker profile vectors (VTLN, two spectral slope parameters and model variance scaling) exhibit a reduction of the computational load to a fraction compared to that of an exhaustive grid search. In recognition experiments on children’s connected digits using adult and male models, the one-dimensional tree search performed as well as the exhaustive search. Further reduction was achieved with four dimensions. The best recognition results are 0.93% and 10.2% WER in TIDIGITS and PF-Star-Sw, respectively, using adult models.

Index Terms: speech recognition, speaker adaptation, VTLN, tree-based model selection

1. Introduction
Unsupervised and instantaneous speaker adaptation by means of model transformation controlled by speaker characteristic properties has the attractive property that the adapted models are guaranteed to be realistic if the used transforms are. Conventional adaptation techniques, on the other hand, put little constraint on the adapted models. This makes them sensitive to recognition errors and they require a sufficiently high initial accuracy in order to improve the quality of the adapted models.

Several speaker characteristic properties have been proposed for such adaptation. The most commonly used example is compensation for mis-match in vocal tract length, performed by Vocal Tract Length Normalization (VTLN) [1]. Other candidates are voice source quality, articulation clarity, speech rate, accent, emotion, etc.

However, in spite of the principal attractiveness of these features, there are at least two problems connected to the approach. One is to establish their acoustic relations and to formulate their influence on the models. The second problem is that the estimation process quickly becomes computationally heavy, since it is often implemented as an exhaustive grid search. This is particularly the case if there is more than one property to be jointly optimized. Two-stage techniques, e.g. [1] and [2], reduce the computational requirements, unfortunately to the prize of lower recognition performance, especially if the accuracy of the first recognition stage is low.

In this work, we approach the problem of excessive computational load by representing the range of the speaker profile vector as quantized values in a multi-dimensional tree. Each leaf node contains an individual value of the vector and a corresponding model set, which is created by a profile-controlled transformation of a conventionally trained set. Non-leaf node models are created by merging those of their child nodes. In this way, they represent regions of the profile vector, rather than sample points. The number of recognition iterations performed by a tree search is reduced substantially compared to an exhaustive search among the leaves.

A tree-based representation has previously been successfully used to structure the training and adaptation data, e.g. for speaker clustering [3] and for noisy speech recognition [4]. The feasibility of performing clustering based on vocal tract parameters as an alternative to acoustic observations has been shown by [5]. A general limitation of data-driven adaptation algorithms, though, is that they have difficulties in generalizing outside the adaptation data. In the current work, this limitation is removed by structuring the tree based on speaker characteristic properties rather than acoustic observations. If we know the acoustic effects of these properties, we can predict models of speaker profiles which don’t exist in the adaptation corpus. In this report, we evaluate this prediction performance by training the models on adult speech and evaluating the recognition performance on children’s speech. The results exhibit a substantial reduction in computational load while maintaining similar performance as an exhaustive grid search technique.

2. Method
2.1. Tree generation
The tree is generated by first using a top-down split procedure in the speaker profile domain, followed by acoustic model generation at the leaf nodes, and finally performing a bottom-up merging process in the model domain. Initially, the root node is loaded with a full, sorted, list of values for each dimension in the speaker profile vector. The corresponding lists for its child nodes are achieved by splitting each dimension list into an upper and a lower half. Each combination of these sub-lists is then assigned to a child node. Hence, the number of child nodes to a mother node equals 2 raised to the power of the number of implemented speaker properties. The split process proceeds iteratively for each new child until the dimension lists at a node all have a single value, which defines a leaf node. The unique speaker profile vector at each leaf is then used to control the transformation of a conventionally trained original model set into a profile-specific set, which gets stored in the leaf node. This is followed by a merging procedure, in which leaf model sets with a common mother node are merged into a model set at the mother node. This is repeated upwards in the tree, until the
root model is reached. Each node in the tree now contains a model set corresponding to its lists of speaker profile values.

2.2. Gaussian Mixture Component Merging

In this work, the trained model set and, accordingly, all sets in the tree have a continuous-density, Gaussian mixture component HMM representation. Merging two or more model sets in this framework starts by inserting the mixture components of the corresponding source states at sibling nodes into the destination state at their mother node. This is followed by a merge procedure to make the resulting number of components per state the same as in each of the original models. This is done in order to avoid large memory requirements and long recognition response time. One existing technique to reduce the number of mixture components is the HTK command HHEd (MD subcommand), which performs a stepwise greedy algorithm [6]. The two mixture components with minimal merging cost are merged repeatedly until the specified number is reached. The algorithm is accurate and flexible but becomes slow if the number of original components is high.

In the approach used in this report, we account for the fact that the model sets to be merged are derived from the same original model set, and that adjacent nodes in the tree have small differences in their transformation feature values. It may therefore be an appropriate procedure to merge those components which have been created from the same original mixture component. The cost function computation and grouping optimization procedures are eliminated completely, which gives a large reduction of the computational cost.

A computationally efficient procedure to calculate the merged component is based on the second moments about the origin [7], [8]. The merged j-th mixture component \((\mu_j', \Sigma_j', c_j')\) is derived from the j-th component of each of the \(I\) source models, \((\mu_i, \Sigma_i, c_i)\), by

\[
\mu_j' = \sum_{i=1}^{I} c_{ij} \mu_{ij}
\]

and

\[
\Sigma_j' = EX_j^2 - \mu_j' \mu_j'^T,
\]

where \(EX^2\) is the second moment about the origin:

\[
EX_j^2 = \sum_{i=1}^{I} c_{ij} EX_{ij}^2,
\]

\[
EX_j^2 = \Sigma_{ij} + \mu_i \mu_j'^T
\]

The component weight, \(c_{ij}\), is \(c_{ij} \) normalized to sum to 1 over \(i\). In this work, the transformations do not change the original mixture component weights. Hence, \(c_{ij}\) is identical for all \(i\), leading to simple averaging in Eq. (1) and (3) and \(c_j' = c_j\).

2.3. Search procedure

The search for the model set to be used by the recognizer for a certain test utterance starts at the root of the tree. A recognition procedure is performed for each of its child nodes. The maximum-likelihood scoring child node is selected for further search. This is repeated until a stop criterion is met, which can be that the leaf level or a specified intermediate level is reached. Another selection criterion may be the maximum scoring node along the selected root-to-leaf path (path-max). This would account for non-stationary speaker properties in the test utterance, since a mother node covers a larger property range than each of its children.

3. Model transformations

We have selected a number of speaker properties to evaluate our multi-dimensional estimation approach. The current set contains a few basic properties, similar to our work in [9]. Its main purpose in this paper is to provide a number of features to evaluate the multi-dimensional model tree rather than the features themselves. They are chosen to operate with different transformations: rotation and scaling of spectral means and variances, and spectral bias. The composition of the set will be addressed in future work.

3.1. VTLN

Vocal Tract Length Normalization (VTLN) is an obvious candidate as one element in the speaker profile vector. If the distribution is Gaussian and the transformation is linear and constrained, then the transformed mean vector and the covariance matrix can be computed [10] as

\[
\hat{\mu} = A^{-1} \mu \quad \text{and} \quad \hat{\Sigma} = A^{-1} \Sigma A^{-1T}
\]

where \(A\) is an \(n \times n\) matrix for transformation of the input cepstral feature vector of size \(n\).

A phoneme-independent transformation of mel cepstra (MFCC) is used in the current work. A two-segment piece-wise linear warping function projects the original model spectrum into its warped spectrum. The conversions between the cepstral and the spectral domains, between the linear and the mel frequency scales, as well as the warping function are integrated into one linear transformation of the original cepstrum, realised as a matrix multiplication, as in [11].

One effect that has to be accounted for during frequency warping is cepstral smoothing [11]. This is implicitly performed during normal training by retaining only a limited number of cepstrum coefficients in the stored models. For warping factors larger than 1.0 (spectral expansion), the highest order coefficients in the warped cepstrum will be empty or have low values, since they depend on non-existing coefficients in the original cepstral vector. In order to ensure a certain number of coefficients during recognition, we use a larger number of coefficients during training, as in [9].

3.2. Spectral slope

Our main intention with this feature is to compensate for difference in the voice source spectrum. However, since the operation currently is performed on all models, also unvoiced phone models will be affected. The feature will, thus, perform an overall compensation of mis-match in spectral slope, whether caused by voice source or the transmission channel.

We use a first order pole to approximate the gross spectral shape of the voice source function. In order to correctly modify a model in this feature, it is necessary to remove the characteristics of the training data and to insert those of the test speaker. A transformation of this feature thus involves two parameters: an inverse pole for the training data and a pole for the test speaker. In this work, the pole bandwidths are varied between 100 and 4000 Hz.

This two-stage normalization technique gives us the theoretically attractive possibility to use separate transformations for the vocal tract transfer function and the voice source spectrum (at least in these parameters). After the inverse filter, there remains (in theory) only the vocal tract transfer function. Performing frequency warping at this position will thus not
affect the original voice source of the model. The new source characteristics are inserted after the warping and are also unaffected. In contrast, conventional VTLN implicitly warps the voice source spectrum identically to the vocal tract transfer function. That procedure is, to our knowledge, not supported by speech production research.

3.3. Model variance

An additional source of difference between adults’ and children’s speech is the larger intra- and inter-speaker variability of the latter category [12]. We account for this effect by increasing the model variances. This feature will also compensate for mismatch which can’t be modeled by the other profile features. Universal variance scaling is implemented by multiplying the diagonal covariance elements of the mixture components by a factor ranging between 1.0 and 3.0.

4. Experiments

4.1. Speech corpora

For evaluation, connected digits recognition in the mismatched case of child test data using adult training data was selected. The Swedish PF-Star children’s corpus (PF-Star-Sw) [13] and TIDIGITS [14], were used for this purpose.

PF-Star-Sw consists of 198 children aged between 4 and 8 years. In the digit subset, each child was aurally prompted for ten 3-digit strings. Recordings were made in a separate room at day-care and after school centers. Downsampling and re-quantization of the original specification of PF-Star-Sw was performed from 24 bits / 32 kHz to 16 bits / 16 kHz.

Since PF-Star-Sw does not contain adult speakers, the training data was taken from the adult Swedish part of the SPEECON database [15]. In that corpus, each speaker uttered one 10 digit-string and four 5 digit-strings, prompted on a computer screen. Recordings were made with an 80 Hz high-pass filter and 16-bits / 16 kHz sampling. The same type of head-set microphone was used in the two corpora.

Training and evaluation sets were formed by 60 speakers, resulting in a training data size of 1800 digits and a children’s test data of 1650 digits. The latter size is due to the failure of some children to produce all the three-digit strings.

TIDIGITS is recorded with 16 bits / 16 kHz. The adult training set consists of 55 men and 57 women. The child test set consists of the 25 boys and 25 girls in the full test set. Their ages range from around 6 up to 15 years.

The differences between training and testing conditions in PF-Star-Sw, combined with the lower age of the children, make it a more challenging task than TIDIGITS.

4.2. Pre-processing and model configuration

A phone model representation of the vocabulary has been chosen in order to allow phoneme-dependent transformations. A continuous-density HMM system with word-interval, three-state triphone models is used. In TIDIGITS, we split diphthongs into a sequence of two vocalic phones to account for their dual-target character. The output distribution is modeled by 16 and 32 diagonal covariance mixture components for PF-Star-Sw and TIDIGITS, respectively, reflecting the different amounts of training data in the corpora.

The cepstral coefficients are derived from a 38-channel mel filterbank with 0-7600 Hz frequency range, 10 ms frame rate and 25 ms analysis window. The original models are trained with 18 MFCCs plus normalized log energy, and their delta and acceleration features. After model transformation, cepstral smoothing is performed by removing the 6 highest MFCCs, resulting in a standard 12-MFCC, 39-element vector.

4.3. Test conditions

The word insertion penalty was adjusted separately for the test data in PF-Star-Sw and TIDIGITS, but within each corpus it was unchanged over different test conditions.

In the single dimension speaker vector case (VTLN), the frequency warping factor was quantized into 16 log-spaced values between 1.0 and 1.7. Values below 1.0 were not used since the training and test speakers were all adults and children, respectively. The tree consists of 5 levels, 16 leaf nodes and in total 31 nodes.

In the 4-dimensional case, we kept 16 values for the warping factor, while the two voice source factors and the variance scaling factor, being judged as less informative, were quantized to 8 log-spaced values. This reduction made it possible to store all 12561 sets on the computer’s hard disk. The tree has 5 levels and 8192 leaves. The exhaustive grid search was not performed due to prohibitive computational requirements.

The node selection criterion during the tree search was varied to stop at different levels. An additional rule (path-max) was to select the maximum-likelihood node of the traversed path from the root to a leaf node. These were compared against an exhaustive grid search among all leaf nodes and a baseline with non-transformed models.

Training and recognition experiments were conducted using HTK [6]. Separate software was developed for the transformation and the model tree algorithms.

5. Results

Results for the one- and four-element speaker profiles are presented in Table 1 for different search criteria. One observation in the one-dimensional case is that even the profile-independent root node provides substantial improvement compared to the baseline results. Ending the search at two levels below the root gives equal or better performance than the exhaustive search at 25% of the computational load. The changes by extending the search below this level are marginal. The path-max criterion works better in TIDIGITS than in PF-Star-Sw but is not better than the second level below the root.

Also for the four-dimensional speaker profile, an optimum compromise between error rate and number of iterations is indicated at two levels below the root. Four features yield consistent improvements over the single feature, except for the root criterion. The computational load at the optimum stop-level is around 0.4% of that of the exhaustive search.

Table 2 presents the two merging algorithms regarding processing time to build the one-dimensional model tree, including the transformation of the leaf models. Recognition error rate is also shown for the adult-child case with a leaf-level stop criterion. The proposed second moment algorithm is much faster with no visible performance degradation.

6. Discussion

The accuracy of the one-dimensional tree-based search was as high as that of the exhaustive search at a fraction of the computational load. This result is especially positive, regarding that the latter search is guaranteed to find the global maximum-likelihood speaker vector. The optimum compromise between error rate and computational load occurs with a stop criterion around two levels below the root.
Table 1. Number of recognition iterations and WER (%) for single (1-D: VTLN) and four-dimensional (4-D) profile estimation.

<table>
<thead>
<tr>
<th>Search alg.</th>
<th>No. Iterations</th>
<th>PF-Star-Sw Adult-child (%)</th>
<th>TIDIGITS Adult-Child (%)</th>
<th>TIDIGITS Male-Child (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>1</td>
<td>32.2</td>
<td>3.28</td>
<td>45.94</td>
</tr>
<tr>
<td>1-D</td>
<td>16</td>
<td>11.5</td>
<td>1.13</td>
<td>3.51</td>
</tr>
<tr>
<td>4-D</td>
<td>8192</td>
<td>13.9</td>
<td>3.05</td>
<td>7.33</td>
</tr>
<tr>
<td>Root</td>
<td>1</td>
<td>1.9</td>
<td>1.42</td>
<td>3.78</td>
</tr>
<tr>
<td>Level 1</td>
<td>2</td>
<td>11.2</td>
<td>1.12</td>
<td>3.51</td>
</tr>
<tr>
<td>Level 2</td>
<td>4</td>
<td>10.2</td>
<td>1.00</td>
<td>3.41</td>
</tr>
<tr>
<td>Level 3</td>
<td>6</td>
<td>10.2</td>
<td>1.09</td>
<td>3.46</td>
</tr>
<tr>
<td>Leaf</td>
<td>8</td>
<td>10.4</td>
<td>1.14</td>
<td>3.45</td>
</tr>
<tr>
<td>Path-max</td>
<td>9</td>
<td>11.6</td>
<td>1.01</td>
<td>3.35</td>
</tr>
</tbody>
</table>

Table 2. Processing time to build the model tree and leaf-level error rate for the two merging algorithms. Single process on an Intel Q6600 @ 2.4GHz CPU.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>PF-Star-Sw TIDIGITS GMM-16</th>
<th>TIDIGITS GMM-32</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greedy</td>
<td>439</td>
<td>962</td>
</tr>
<tr>
<td>2nd Mom.</td>
<td>29</td>
<td>54</td>
</tr>
</tbody>
</table>

A possible explanation to the lack of error reduction below this level could be that the advantage of higher quantization resolution at levels closer to the leaves is counteracted by increasingly narrow models. The narrower models have lower ability to capture non-stationarity in the speaker profile vector, such as phoneme-specific values.

Even the root level set provides high recognition rate, especially in the one-dimensional case. Since there is no estimation procedure involved, this saves considerable computation while still providing substantial performance improvement over the baseline with the same model complexity. A possible explanation to the higher root node error rate for four dimensions might be that excessive spectral tilt compensation was not penalized due to the use of uniform speaker profile distribution.

For all the other stop criteria, the use of four properties lowered the single-property error rate further. Clearly, vocal tract length is very important, but spectral slope and variance scaling also have positive contribution. Further work will be aimed at improving the set. Candidates are the spontaneous/read speech dimension, speech rate, regional accent, etc. Ongoing work on phone-specific factors is likely to benefit from the reduced search space provided by the tree-based model approach.

Further size reduction of the tree and efficiency improvement could be achieved by merging leaf nodes whose models are acoustically similar, even if the nodes are non-adjacent. This may occur if certain changes in the speaker profile have little effect on the transformed model. Standard clustering techniques could be used for this purpose.

The accuracy of the simple algorithm for merging mixture components was equal to that of the more general greedy procedure at a fraction of the processing time. Evidently, the special conditions in this case do not require the general capacity in the latter algorithm.

7. Conclusion
A tree-based search in the speaker profile space provides recognition accuracy similar to an exhaustive search at a fraction of the computational load and makes it possible to perform joint estimation in a larger number of speaker profile dimensions. Even the root model sets, without any estimation process, achieve high recognition rate. A simple mixture component merging algorithm is shown to give as high accuracy as a more advanced greedy algorithm at a fraction of the computational requirement.

8. Acknowledgements
This work was financed by the Swedish Research Council.

9. References