Deep Neural Network Bottleneck Features
For Generalized Variable Parameter HMMs

Xurong Xie\textsuperscript{1,3}, Rongfeng Su\textsuperscript{1,3}, Xunying Liu\textsuperscript{1,2} \& Lan Wang\textsuperscript{1,3}

\textsuperscript{1}Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences
\textsuperscript{2}Cambridge University Engineering Dept, Trumpington St., Cambridge, CB2 1PZ U.K.
\textsuperscript{3}The Chinese University of Hong Kong, Hong Kong, China
xr.xie@siat.ac.cn, rf.su@siat.ac.cn, xl207@cam.ac.uk, lan.wang@siat.ac.cn

Abstract
Recently deep neural networks (DNNs) have become increasingly popular for acoustic modelling in automatic speech recognition (ASR) systems. As the bottleneck features they produce are inherently discriminative and contain rich hidden factors that influence the surface acoustic realization, the standard approach is to augment the conventional acoustic features with the bottleneck features in a tandem framework. In this paper, an alternative approach to incorporate bottleneck features is investigated. The complex relationship between acoustic features and DNN bottleneck features is modelled using generalized variable parameter HMMs (GVP-HMMs). The optimal GVP-HMM structural configuration and model parameters are automatically learnt. Significant error rate reductions of 48\% and 8\% relative were obtained over the baseline multi-style HMM and tandem HMM systems respectively on Aurora 2.

Index Terms: generalized variable parameter HMM, deep neural network, bottleneck features, robust speech recognition

1. Introduction

Recently deep neural networks (DNNs) have become increasingly popular for acoustic modelling in automatic speech recognition (ASR) systems \cite{1, 2, 3, 4, 5, 6, 7, 8}. In order to incorporate DNNs, or multi-layer perceptrons (MLPs) in general, into HMM based acoustic models, two approaches can be used. The first uses a hybrid architecture that estimates the HMM state emission probabilities using DNNs \cite{9}. The second approach uses an MLP or DNN as a feature extractor, trained to produce phoneme posterior probabilities. The resulting probabilistic features \cite{10}, or bottleneck features \cite{11} are used to train standard GMM-HMMs in a tandem fashion. As these features capture additional information complementary to standard front-ends, they are often combined in tandem systems.

One important issue associated with the tandem HMM approach is the appropriate method used to combine the conventional and bottleneck features. The precise nature of the relationship between the two is highly complex. Compared with the standard front-ends, bottleneck features provide a different view of the same speech signals. Certain correlation can therefore exist between the two. At the same time, complementary information characterizing the underlying hidden factors influencing the surface acoustic realization are also implicitly learnt by bottleneck features. They are propagated into HMMs as additional cues and constraints to improve discrimination. The standard approach augments the conventional front-ends with bottleneck features in a concatenated form. More advanced approaches that explicitly approximate the correlation between them using linear, affine transformations have also been proposed \cite{12, 13}.

In order to better capture the complex relationship between standard acoustic and bottleneck features, techniques motivated by speech production that can fully exploit the hidden variability in the bottleneck features may be used. Along this line, an alternative method to incorporate bottleneck features into a tandem system is proposed in this paper. DNN bottleneck features are used as influence factors to directly introduce controllability to the underlying generative acoustic models that are based on generalized variable parameter HMMs (GVP-HMMs) \cite{14, 15, 16, 17, 18}. The continuous trajectories of optimal HMM parameters against the time-varying hidden factors in the bottleneck features are modelled using polynomial functions. Their effects on the acoustic parameters are automatically learnt by locally optimized polynomial parameters and degrees. Using the proposed GVP-HMM tandem approach, significant error rate reductions of 48\% and 8\% relative were obtained over the multi-style baseline HMM and tandem HMM systems respectively on Aurora 2.

The rest of this paper is organized as follows. Generalized variable parameter HMMs and an associated efficient complexity control technique are introduced in section 2. Deep neural networks and bottleneck features are reviewed in section 3. A range of GVP-HMM systems using various modelling configurations are described in section 4. In section 5 various GVP-HMM systems using DNN bottleneck features are evaluated on Aurora 2. Section 6 is the conclusion and future research.

2. Generalized Variable Parameter HMMs

Generalized variable parameter HMMs (GVP-HMMs) \cite{14, 15, 16, 17} explicitly model the parameter trajectories of optimal Gaussian components, or more compact tied linear transformations, that vary with respect to some influence factors. In this paper, trajectories of Gaussian means and variances are used.

2.1. Model Definition

For a $D$ dimensional observation $o_t$, emitted from Gaussian mixture component $r_o$, assuming $P$th order polynomials modelling a total of $N$ regression variables are used, the form of
GVP-HMMs considered in this paper is given by
\[ Q^{(i)}(\theta, \hat{\theta}) = \sum_{m,t} \gamma_{m,t}(\theta) \log p \left( o^{(i)}, \mu^{(m)}(v_i), \Sigma^{(m)}(v_i) \right) \]
(5)
where \( \gamma_{m,t}(\theta) \) is the posterior probability of frame \( o_t \) being emitted from component \( m \) at a time instance \( t \).

Combining the above with equations (1) and (4), the corresponding parts of the above auxiliary function associated with the polynomial coefficient vectors of the Gaussian mean and variance trajectories respectively can be re-arranged into convex quadratic forms,
\[
Q^{(\nu^{(m)})}_{\text{ml}}(\hat{\theta}, \tilde{\theta}) = -\frac{1}{2} \nu^{(m)}(v_i) \Sigma^{(m)}(v_i) \nu^{(m)}(v_i) \tag{6}
\]
and
\[
Q^{(\sigma^{(m)})}_{\text{ml}}(\hat{\theta}, \tilde{\theta}) = -\frac{1}{2} \sigma^{(m)}(v_i) \Sigma^{(m)}(v_i) \sigma^{(m)}(v_i) \tag{6}
\]
where the constant terms independent of the coefficient vectors \( \nu^{(m)} \) and \( \sigma^{(m)} \) can be ignored. Setting the above gradients against the respective polynomial coefficient vectors to zero, the following ML solutions of the coefficient vectors can then be derived
\[
\hat{\nu}^{(m)}(v_i) = U(v_i) \Sigma^{(m)}(v_i) \tag{7}
\]
and the sufficient statistics are
\[
U(v_i) = \sum_{t} \nu^{(m)}(v_i) \Sigma^{(m)}(v_i) \nu^{(m)}(v_i) \tag{7}
\]
\[
k(v_i) = \sum_{t} \nu^{(m)}(v_i) \Sigma^{(m)}(v_i) \nu^{(m)}(v_i) \tag{7}
\]
\[
U^{(m)}(v_i) = \sum_{t} \nu^{(m)}(v_i) \Sigma^{(m)}(v_i) \nu^{(m)}(v_i) \tag{7}
\]
\[
k^{(m)}(v_i) = \sum_{t} \nu^{(m)}(v_i) \Sigma^{(m)}(v_i) \nu^{(m)}(v_i) \tag{7}
\]

2.3. Model Complexity Control for GVP-HMMs

An important issue associated with GVP-HMMs is the appropriate polynomial degree to use. The use of higher degree polynomials can result in severe over-fitting and oscillation [25]. In addition, the precise form of individual parameter trajectories should be in line with the nature of the distinct effects imposed on them by the influencing factors. In order to more flexibly capture these complex, potentially locally varying effects and improve robustness, the optimal polynomial degrees of Gaussian mean and variance trajectories can be automatically determined at low level using complexity control techniques [18].

In Bayesian learning, when no prior knowledge over model structures \( \mathcal{M} \) is available, the optimal model structure or complexity, is determined by maximizing the evidence,
\[
p(O|W, \mathcal{M}) = \int p(O|\theta, W, \mathcal{M}) p(\theta|\mathcal{M}) d\theta \tag{9}
\]
where \( \theta \) is a parameterization of \( \mathcal{M} \), \( O = \{ o_1, ..., o_T \} \) is a training data set of \( T \) frames and \( W \) the reference transcription.

For standard HMMs and GVP-HMMs, it is computationally intractable to directly compute the evidence in equation (9). To handle this problem, an efficient approximation using the BIC style first order asymptotic expansion [26] of a lower lower bound [18, 27, 28, 29] of the evidence integral can be used. The optimal model complexity is determined by
\[
\mathcal{M} = \arg \max_{\mathcal{M}} \left\{ Q_{\text{ml}}^{(\mathcal{M})}(\hat{\theta}, \tilde{\theta}) - \beta \cdot \frac{k}{2} \log T \right\} \tag{10}
\]
where the ML auxiliary functions associated with Gaussian mean and variance trajectory parameters given in equation (6) evaluated at the optimal model parameters \( \hat{\theta} \) using the statistics given in equations (7) and (8), \( k \) denotes the number of free parameters in \( \mathcal{M} \) and \( \beta \) is a tunable penalty term [30].

When determining the optimal order for a particular polynomial associated with the ith dimension of the ith Gaussian component in the system, \( \nu^{(m)}(\cdot) \), for example, the above statistics in equation (8) are accumulated for the highest order \( P_{\text{max}} \) being considered. The corresponding statistics for any other order \( 0 \leq P_{\text{nu}}^{(m)}(\cdot) < P_{\text{max}} \) can be derived by taking the associated submatrices or subvectors from the full matrix statistics accumulated for \( P_{\text{max}} \). Using these statistics and the ML
3. DNN Bottleneck Features

Bottleneck features are normally generated from a narrow hidden layer of an MLP that is trained to predict phonemes or phoneme states. Compared with the size of other layers, this hidden layer has a significantly smaller number of hidden units [11]. This narrow layer introduces a constriction in the network while retaining the information useful to classification in the resulting low dimensional features extracted via a non-linear and discriminative transformation.

In this paper the bottleneck features used for tandem HMM systems are extracted from deep neural network (DNN) multi-layer perceptrons (MLP) [1, 2, 3, 4]. DNNs are MLPs with many hidden layers. The inputs are formed from a stacked set of adjacent frames of the acoustic feature for each time instance. Within each hidden layer, the input to each unit is computed as a linearly weighted sum of the outputs from the previous layer. Each hidden node transforms its input with a sigmoid activation function to achieve non-linearity. An softmax output activation function is used at the output layer to compute the posterior probability of phonemes or phoneme state targets. In all the experiments of this paper, a pretrained DNN consisting of six hidden layers is used. The first five layers have a total of 512 hidden nodes while the last bottleneck layer has 26 units. The network is trained on inputs formed by splicing 11 frames of 39 dimensional MFCC features together. The layer-by-layer RBM based pre-training implemented in the Kaldi toolkit [31] was used.

Following DNN training 26 dimensional bottleneck features are extracted and decorrelated using PCA. For the baseline tandem HMM systems, they are appended to standard MFCC features to form the tandem feature vector. Prior to recognition, tandem GMM-HMMs are then trained based on the new concatenated tandem features. For GVP-HMM systems, these are used as the input factor vectors at each frame to estimate continuous trajectories of Gaussian mean and variance parameters. An extended version of the HTK toolkit [32] was used to train various GVP-HMM systems.

4. Using DNN Bottleneck Features In GVP-HMMs and Tandem GVP-HMMs

In order to adjust the trade-off between modelling resolution, robustness and computational efficiency, a range of GVP-HMM configurations may be considered to incorporate DNN bottleneck features. Description of these GVP-HMM variant systems’ configurations and the number of parameters used for the standard Aurora 2 task are shown in table 1. 39 dimensional standard MFCC features including the first and second order differentials were used. All the baseline GVP-HMMs with no complexity control used 2nd degree polynomials for all parameter trajectories, as suggested in [21, 22]. The penalty term in the complexity control metric of equation (10) was fixed as $\rho = 1$ in all experiments. For all parameter polynomials the range of candidate degree to consider is $[0, 5]$.

**Baseline HMM and tandem HMM systems:** In the first 3 lines of table 1, the number of parameters for the multi-style [33] trained baseline HMM system and two tandem HMM systems are shown. The second tandem HMM system, “tandem†”, used 18 Gaussians per state thus has a model complexity comparable to the other complexity controlled GVP-HMM systems in the table. The Gaussian parameters of these baseline HMM or tandem HMM systems were trained on standard MFCC or tandem features while no parameter trajectory modelling was used.

**GVP-HMM systems:** In the second section of table 1, a total of four GVP-HMM modelling configurations, denoted as “mean” and “mv” respectively, which use trajectory modelling for Gaussian component means using the DNN bottleneck features as the factor input in equations from (1) to (3), with the further options of using variance trajectories conditioned on the SNR variable, and with or without applying the model selection technique presented in section 2.3, are shown from the 4th to 7th line in table 1. As expected, using the standard GVP-HMMs with no complexity control on the 26 dimensional bottleneck features results in a massive increase in model parameters. Determining the optimal degrees for parameter trajectory polynomials using the model selection method of section 2.3 significantly reduced the model complexity by over to 80%.

**Tandem GVP-HMM systems:** In the last section of table 1, four comparable tandem variants of the above four GVP-HMM systems are shown. In these tandem GVP-HMM systems, the DNN bottleneck features are not only used as the input factor vectors to estimate the continuous trajectories of Gaussian parameters in the acoustic feature subspace, but also used as normal features to train the standard mean and variance parameters in the bottleneck feature subspace. For example, the final mean vector of component $m$ at time instance $t$ is thus computed as

$$\hat{\mu}_m^{(m)} = [\hat{\mu}_m^{(m)}(v_B^\text{BN}) \cdot \hat{\mu}_m^{(m)}].$$

where the $\hat{\mu}_m^{(m)}(v_B^\text{BN})$ is the mean subvector trajectory taking a Vandermonde vector input $v_B^\text{BN}$ constructed using the 26 dimensional DNN bottleneck features, as described in section 2.1.
in this paper. The complementary information characterizing the hidden factors influencing the surface acoustical realization implicitly learnt by bottleneck features are exploited to improve controllability and robustness. The proposed technique significantly reduced the error rate by 48% and 8% relative over the baseline multi-style HMM and tandem HMM systems respectively on Aurora 2. Future research will focus on using bottleneck features to model the trajectories of more efficient feature space transforms [17].

6. Conclusion

An alternative approach to incorporate bottleneck features into a tandem system using generalized variable parameter HMMs is investigated in this paper. The complementary information characterizing the hidden factors influencing the surface acoustical realization implicitly learnt by bottleneck features are exploited to improve controllability and robustness. The proposed technique significantly reduced the error rate by 48% and 8% relative over the baseline multi-style HMM and tandem HMM systems respectively on Aurora 2. Future research will focus on using bottleneck features to model the trajectories of more efficient feature space transforms [17].

Table 3: WER performance of tandem GVP-HMM systems using DNN bottleneck features on Aurora 2 test set A. All systems used the same naming conventions as in table 1.

<table>
<thead>
<tr>
<th>Noise Type</th>
<th>System Ctrl</th>
<th>0dB</th>
<th>5dB</th>
<th>10dB</th>
<th>15dB</th>
<th>20dB</th>
<th>Ave</th>
</tr>
</thead>
<tbody>
<tr>
<td>subway</td>
<td>mcond</td>
<td>12.98</td>
<td>5.99</td>
<td>3.01</td>
<td>2.12</td>
<td>1.23</td>
<td>5.41</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>12.84</td>
<td>4.61</td>
<td>2.15</td>
<td>1.47</td>
<td>1.26</td>
<td>4.75</td>
</tr>
<tr>
<td></td>
<td>mv</td>
<td>20.94</td>
<td>8.58</td>
<td>4.91</td>
<td>3.72</td>
<td>3.84</td>
<td>8.39</td>
</tr>
<tr>
<td></td>
<td>mcond</td>
<td>30.47</td>
<td>12.09</td>
<td>6.53</td>
<td>4.59</td>
<td>4.08</td>
<td>11.55</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>26.96</td>
<td>8.04</td>
<td>4.63</td>
<td>3.19</td>
<td>3.45</td>
<td>7.29</td>
</tr>
<tr>
<td></td>
<td>mv</td>
<td>38.42</td>
<td>14.33</td>
<td>6.38</td>
<td>4.35</td>
<td>3.51</td>
<td>11.40</td>
</tr>
<tr>
<td></td>
<td>mcond</td>
<td>22.88</td>
<td>9.42</td>
<td>4.29</td>
<td>3.58</td>
<td>2.78</td>
<td>9.55</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>17.72</td>
<td>9.00</td>
<td>4.08</td>
<td>2.40</td>
<td>1.99</td>
<td>7.20</td>
</tr>
<tr>
<td></td>
<td>mv</td>
<td>26.01</td>
<td>14.86</td>
<td>6.05</td>
<td>4.10</td>
<td>3.26</td>
<td>10.75</td>
</tr>
<tr>
<td></td>
<td>mcond</td>
<td>21.46</td>
<td>10.27</td>
<td>4.91</td>
<td>3.39</td>
<td>2.72</td>
<td>9.39</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>15.22</td>
<td>6.29</td>
<td>3.05</td>
<td>2.13</td>
<td>1.88</td>
<td>5.71</td>
</tr>
<tr>
<td></td>
<td>mv</td>
<td>16.20</td>
<td>6.60</td>
<td>3.15</td>
<td>1.54</td>
<td>1.03</td>
<td>5.71</td>
</tr>
</tbody>
</table>

Table 2: WER performance of GVP-HMM systems using DNN bottleneck features on Aurora 2 test set A of four noise types. All systems used the same naming conventions as in table 1.

The WER performance of the multi-style HMM baseline, “mcond”, and various GVP-HMM systems shown from the 4th to 7th line of table 1 are given in table 2. The following trends can be found in the table. First, the use of DNN bottleneck features gave significant WER reductions for all GVP-HMM modelling configurations across various noise types over the “moncon” HMM baseline. Second, as expected, using the model selection technique of section 2.3, in addition to the model size compression shown previously in table 1, an average WER reduction of 2.41% absolute (29% relative) was obtained over various standard GVP-HMM systems with no complexity control. Third, combined with model complexity control, the use of variance trajectory polynomials gave further improvements over using mean trajectory modelling only. Using the best GVP-HMM systems highlighted in bold in table 2, an average WER reduction of 3.64% absolute (40% relative) over the multi-style MFCC feature trained baseline “mcond” HMM system was obtained. However, all of these four GVP-HMM systems were outperformed by the baseline tandem HMM system shown in the 1st line of each noise specific section in table 3.
7. References


