THE HMM ERROR MODEL

M.J.F. Gales

Cambridge University Engineering Department
Trumpington Street, Cambridge, CB2 1PZ, UK
mjfg@eng.cam.ac.uk

ABSTRACT

The most popular model used in automatic speech recognition is the hidden Markov model (HMM). Though good performance has been obtained with such models there are well known limitations in its ability to model speech. For these reasons, a variety of modifications to the standard HMM topology have been proposed including *factorial*, or multi-stream, HMMs. This paper describes a new form of HMM based on *transformation streams*. A particular form of transformation stream is described, the *HMM error model* (HEM). This model may be viewed as a *filter* model, the transformation stream, and a *residual* model. The filter model transforms the original data into a space in which all the data is "similarly" distributed. This normalised data is then modelled using the residual model. The HEM is evaluated on a standard large vocabulary speaker independent speech recognition task, SwitchBoard. On this task significant reductions in word error rate are obtained over standard HMM-based systems.

1. INTRODUCTION

In recent years there has been interest in using factorial HMMs [1] for speech recognition. Though there has not been a great deal of success in applying these to speech recognition tasks [2, 3], they are an interesting and highly flexible framework to build speech recognition systems. Recently, the standard version of factorial HMM has been extended to include the concept of *transformation* streams [4]. These streams affect either the observations or the model parameters used by other streams. The standard factorial HMM may be viewed as a highly restricted version of this where the transformation is restricted to a simple bias. This paper uses the concept of transformation streams to obtain a new form of acoustic model, the *HMM error model* (HEM).

The HEM is an extension to standard HMMs used for speech recognition. The model uses a transformation stream in conjunction with a simple, single state, "model" stream. Due to the nature of the model it may also be interpreted as a dynamic filter, the transformation stream, and a residual model. This allows a simple description of how the model might improve the modelling ability of a standard HMM. Standard HMMs may be viewed as a form of linear filter model (see section 2.3). In the normalised, filtered, domain the data is assumed to be zero mean, identity covariance matrix, Gaussian distribution. However, for many complex signals this is unlikely to be true. Instead the normalised data could be modelled using any standard probabilistic model. In this paper the use of Gaussian mixture models (GMMs) is investigated. One of the advantages of using a GMM is that it is possible to update all the model parameters in a consistent framework. In common with many other acoustic modelling schemes and factorial HMM schemes, the HEM may be described as a form of *soft* parameter tying. In soft tying schemes model parameters are "related" to one another. Standard tying schemes [5], in contrast, require that model parameters are either independent or identical. Examples of soft tying include soft state tying [6] and semi-tied covariance matrices [7].

This paper is organised as follows. The HEM is described in the next section. Re-estimation formulae for all the model parameters are given. Experiments on a large vocabulary speech recognition task are described which illustrate the advantage of HEMs over standard HMMs.

2. HMM ERROR MODEL

The HMM error model (HEM) is a 2-stream factorial model using a feature-space transformation stream. The HEM uses a very simple normalised-domain model, a GMM. As such it does not make full use of the power of a factorial HMM system, but does sit within the general class of factorial HMMs.

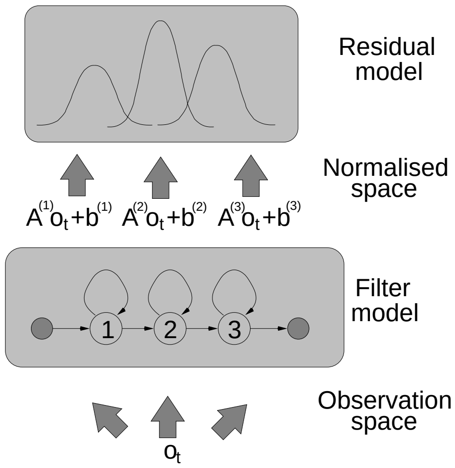

Fig. 1. HMM Error Model

Due to the relatively simple model used there is an intuitive interpretation of the HEM. Rather than considering it as a multiple stream system it could also be described in terms of a filter model and residual model. A HEM, with a three state transformation stream, is shown in figure 1. The first stream, a transformation stream, is a filter model. The aim of the filter model is to transform the data from the observation space into some normalised space such that the data is independent and identically distributed. The second stream, the residual model, models the data in this normalised space.

For this work a mixture of linear feature-space transformations is associated with each state of the filter model (a mixture of linear transformations has previously been used for speaker adaptation [8]). If the filter model is initialised using HMMs trained with ML estimation, the data in the normalised space will have zero mean and an identity matrix covariance matrix. However the higher order statistics are not constrained in anyway. Furthermore, for speech recognition diagonal covariance matrices are commonly used. In this case the only the leading diagonal of the covariance matrix of the normalised data is constrained to consist of ones. The off-diagonal terms are not restricted. By using either a GMM, or more complex covariance matrix Gaussian distribution, as the residual model, it is possible to more accurately model the normalised data correlations and higher order moments.

A model similar to the HEM has previously been investigated for variance compensation in speaker adaptation [9]. The filter model was constrained to have diagonal transforms and the residual model a single speaker-specific full covariance matrix multivariate Gaussian distribution. This allowed feature vector correlations to be modelled. The model described in [9] differs from the HEM in a couple of ways. First the residual model is trained per speaker. The HEM residual model is trained on all the training data (though may, of course, be adapted when speaker adaptation is being used). Second the filter model was not adaptively trained. A standard HMM was simply trained and used as a filter model (see section 2.3 for the relationship between standard HMM parameters and filter model parameters).

This section describes the likelihood calculation for a HEM and how the model may be trained using ML estimation. For more details of the optimisation and some implementation issues see [4].

2.1. Likelihood Calculation

The likelihood of the HEM generating an observation $\mathbf{o}_t$ is calculated by computing the likelihood of the normalised observation using the residual model. Appropriate scaling is required to account for the component weights of the filter model and the effects of the state-dependent feature transformation of the filter model. Consider a single state q of an HMM, which is modelled with M Gaussian components. A K-component GMM residual model is used. The likelihood of the state q may be written as

$$p(\mathbf{o}_t|q_t = q) = \sum_{m=1}^{M^{(q)}} \sum_{k=1}^{K} c^{(m)} c^{(k)} |\det(\mathbf{A}^{(m)})| \mathcal{N}(\mathbf{W}^{(m)}\boldsymbol{\zeta}_t; \boldsymbol{\mu}^{(k)}, \boldsymbol{\Sigma}^{(k)}) \quad (1)$$

where state q has $M^{(q)}$ components, $1, \ldots, M^{(q)}$,

$$\mathbf{W}^{(m)} = \begin{bmatrix} \mathbf{A}^{(m)} & \mathbf{b}^{(m)} \end{bmatrix} \quad (2)$$

$$\boldsymbol{\zeta}_t = \begin{bmatrix} \mathbf{o}_t \\ 1 \end{bmatrix} \quad (3)$$

Various forms of transformation matrix $\mathbf{A}^{(m)}$ may be used. These range from simple diagonal transforms to full matrix transforms. The most common form of covariance model used in speech recognition is diagonal. In terms of viewing the HEM as a filter and residual model this corresponds to a diagonal transform. This will be the form of model primarily discussed. The likelihood may also be expressed in terms of a constrained estimation of HMM parameters (a form of *soft* tying) [4].

Equation 1 illustrates one of the major disadvantages of the HEM. For the model described MK Gaussian components must be calculated. In contrast to some tying schemes there is no simple scheme for reducing this requirement. However, Gaussian selection schemes may be used to reduce this cost [10].

2.2. Parameter Estimation

This section describes how both the filter model and the residual model may be trained using expectation maximisation (EM). As with other factorial HMM schemes multiple levels of indicator variables are required. $\gamma_t^{(mk)}$ is the probability of being in component m of the filter model and residual component k at time t and M is the *total* number of transform components in the filter model. To simultaneously update both the filter model parameters and residual model parameters is highly complex (and for most tasks impractical). Instead a simple iterative maximisation scheme is used. First the filter model parameters are estimated given the current estimates of the residual model. Second the residual model parameters are updated given the filter model. The two stages are described below.

2.2.1. Filter Model Estimation

The filter model parameters $c^{(m)}$ and $\mathbf{W}^{(m)}$ for each component m of the filter must be estimated given the current estimate of the residual model parameters. In the trivial case where the residual model is a single diagonal covariance matrix Gaussian component the filter model parameter estimation is identical to the standard HMM estimation schemes (see the initialisation section for an interpretation of the model parameters). For the more interesting GMM residual model case, the form of the auxiliary function is very similar to the estimation of a constrained MLLR transform given in [9]. Only the case where the covariance matrices of the residual components are diagonal is considered, though the extension to full-covariance models is possible. The ML estimate of the i^{th} row of $\mathbf{W}^{(m)}$, $\mathbf{w}_i^{(m)}$, may be shown to be

$$\mathbf{w}_i^{(m)} = \left(\alpha\mathbf{p}_i + \mathbf{k}^{(i)}\right) \mathbf{G}^{(i)-1} \quad (4)$$

where $\mathbf{p}_i$ is the extended row vector $\begin{bmatrix} p_{i1} & \ldots & p_{in} & 0 \end{bmatrix}$, ($p_{ij} = \text{cof}(\mathbf{A}_{ij}^{(m)})$),

$$\mathbf{G}^{(i)} = \sum_{k=1}^{K} \frac{1}{\sigma_i^{(k)2}} \sum_{t=1}^{T} \gamma_t^{(mk)} \boldsymbol{\zeta}_t \boldsymbol{\zeta}_t^T \quad (5)$$

$$\mathbf{k}^{(i)} = \sum_{k=1}^{K} \frac{1}{\sigma_i^{(k)2}} \mu_i^{(k)} \sum_{t=1}^{T} \gamma_t^{(mk)} \boldsymbol{\zeta}_t^T \quad (6)$$

and α satisfies

$$\alpha^2 \mathbf{p}_i \mathbf{G}^{(i)-1} \mathbf{p}_i^T + \alpha \mathbf{p}_i \mathbf{G}^{(i)-1} \mathbf{k}^{(i)T} - \beta^{(m)} = 0 \quad (7)$$

where $\beta^{(m)} = \sum_{k=1}^{K} \sum_{t=1}^{T} \gamma_t^{(mk)}$. The optimisation scheme is iterative since the estimation of each matrix row is influenced by the cofactors of the complete matrix. For the simple case where the transformation matrix $\mathbf{A}^{(m)}$ is diagonal, the cofactors are zero so a closed form solution is possible.

In addition to the transform parameters the weights of the components of the filter model are required. These may be found using

$$c^{(m)} = \frac{\sum_{k=1}^{K} \sum_{t=1}^{T} \gamma_t^{(mk)}}{\sum_{i \in \mathcal{J}^{(m)}} \sum_{k=1}^{K} \sum_{t=1}^{T} \gamma_t^{(ik)}} \quad (8)$$

where $\mathcal{J}^{(m)}$ is the set of transform components belonging to the same filter model state as component m.

2.2.2. *Residual Model Estimation*

Once the filter model has been trained the estimation of the residual model is very similar to the standard GMM training. It is simple to show that the mean of residual component k is given by

$$\boldsymbol{\mu}^{(k)} = \frac{\sum_{m=1}^{M} \sum_{t=1}^{T} \gamma_t^{(mk)} \mathbf{W}^{(m)} \boldsymbol{\zeta}_t}{\sum_{m=1}^{M} \sum_{t=1}^{T} \gamma_t^{(mk)}} \quad (9)$$

and similarly for the covariance matrix

$$\boldsymbol{\Sigma}^{(k)} = \frac{\sum_{m=1}^{M} \sum_{t=1}^{T} \gamma_t^{(mk)} (\mathbf{W}^{(m)} \boldsymbol{\zeta}_t - \boldsymbol{\mu}^{(k)})(\mathbf{W}^{(m)} \boldsymbol{\zeta}_t - \boldsymbol{\mu}^{(k)})^T}{\sum_{m=1}^{M} \sum_{t=1}^{T} \gamma_t^{(mk)}} \quad (10)$$

The estimation of residual component weights follows the standard GMM optimisation. Thus

$$c^{(k)} = \frac{\sum_{m=1}^{M} \sum_{t=1}^{T} \gamma_t^{(mk)}}{\sum_{m=1}^{M} \sum_{k=1}^{K} \sum_{t=1}^{T} \gamma_t^{(mk)}} \quad (11)$$

2.3. Parameter Initialisation

In common with other EM-based schemes it is important to have reasonable estimates to initialise the training. The initialisation of the HEM model parameters has two distinct stages. First the filter model parameters are estimated assuming that a single, zero mean, identity covariance matrix Gaussian residual model is used. As previously mentioned this requires training a standard HMM. The conversion of the standard HMM parameters into the filter model parameters is achieved using

$$\mathbf{A}^{(m)} = \mathbf{C}^{(m)-1} \quad (12)$$

$$\mathbf{b}^{(m)} = -\mathbf{A}^{(m)} \boldsymbol{\mu}^{(m)} \quad (13)$$

where

$$\mathbf{C}^{(m)} \mathbf{C}^{(m)T} = \boldsymbol{\Sigma}^{(m)} \quad (14)$$

Note that in equation 1 the "transform" $\mathbf{A}^{(m)}$ is a full matrix. However from the standard full covariance HMM it can be initialised with an upper triangular matrix from the Choleski factorisation of the covariance matrix. The need for a full transformation matrix when there is more than one component in the residual GMM (i.e the set of components K), is the same as for full semi-tied transforms [7].

For the work presented here a scheme similar to the HTK *mixing-up* routine was used [5] to increase the number of filter model components. Here

$$\begin{bmatrix} \mathbf{A}^{(m)} & \mathbf{b}^{(m)} \end{bmatrix} \rightarrow \left\{ \begin{array}{l} \begin{bmatrix} \mathbf{A}^{(m)} & (\mathbf{b}^{(m)} + \epsilon) \end{bmatrix} \\ \begin{bmatrix} \mathbf{A}^{(m)} & (\mathbf{b}^{(m)} - \epsilon) \end{bmatrix} \end{array} \right. \quad (15)$$

where ϵ is the perturbation value. For the experiments used in this paper ϵ is set 0.2. The component weight is evenly divided between the two new transforms. Once the filter model parameters have been estimated, the residual model can be initialised using any standard scheme for initialising GMMs.

3. RESULTS

The performance of the HEM was evaluated on a standard large-vocabulary speaker-independent speech recognition task. Hub5, or SwitchBoard, is a telephone bandwidth spontaneous speech recognition task. The acoustic training data is obtained from two corpora: Switchboard-1 (Swb1) and Call Home English (CHE). The full training corpus consists of an 180 hour training set, 4482 sides from Swb1 and 235 sides from CHE. For the experiments performed in this section a subset of this was used. A total of 68 hours was chosen to include all the speakers from Swb1 in h5train00 as well as a subset of the available CHE sides. 862 Swb1 sides and 92 CHE sides were used in this subset. This is the "h5train00sub" training set described in [6]. The speech waveforms were coded using perceptual linear prediction cepstral coefficients derived from a Mel-scale filterbank (MF-PLP) covering the frequency range from 125Hz to 3.8kHz. A total of 13 coefficients, including c_0, and their first and second order derivatives were used. Cepstral mean subtraction and variance normalisation were performed for each conversation side. Vocal tract length normalisation (VTLN) was applied in both training and test. A gender-independent cross-word-triphone mixture-Gaussian tied-state HMM system was built

For all the experiments presented here diagonal covariance matrices are used for all Gaussian components (including the residual model where used). The transforms of the filter model were also constrained to be diagonal. The same decision tree clustering was used for determining the context dependent standard HMM states and the HEM transformation states. Hence the number of model parameters for a standard HMM system and a diagonal transform HEM system are about the same for the same number of components in the filter model. The model sets generated were trained in the standard HTK fashion using mixing-up. State-based decision tree clustering was used to define a total of 6165 distinct speech states. The performance of the system was examined at various stages of the mixing up process. For all experiments a trigram language model was used, built as in [6]. The filter model transform parameters were initialised with a single component standard HMM system as described in section 2.3. The three component residual model was initialised to means at 0.2, 0.0 and -0.2 for each dimension and identity covariance matrices. A three Gaussian component residual model was used. In addition to a global residual model, phone specific residual models were also built.

The performance of a standard system, a global three-component residual-model HEM system and a phone-based, three-component residual-model, HEM system are shown in figure 2. The performance of the global HEM residual model was consistently better than the standard HMM system. The twelve component system performance of 46.1% was significantly better at the 95% level than the performance of standard system 46.7%. The phone residual model performance was consistently slightly better than the global residual model system. It is interesting to note that the 8-component phone level residual model HEM system out performs the 12-component standard system. For comparison the number of components of the standard system was increased to 14. This increased the error rate to 46.8%, indicating that simply increasing

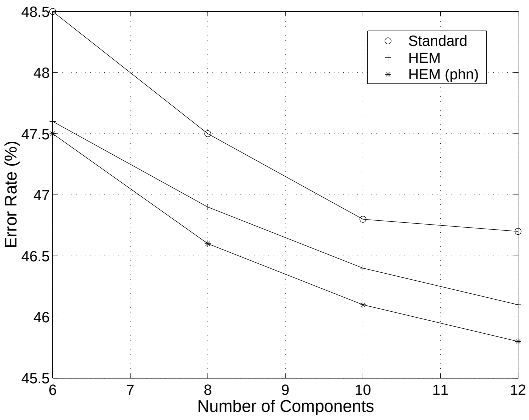

Fig. 2. 1998 Switchboard evaluation performance

the number of components in a standard system does not improve the performance on this task.

The absolute gain obtained using the phone-based HEM models was small (0.9% absolute), though it was significant at the 95% confidence level. The performance of the HEM systems may be compared with other soft tying schemes[1] An implementation of the state soft-tying scheme as described in [6], using the same training data, gave an error rate of 46.2% compared to the 45.8% obtained using a phone level HEM[2]. One interpretation of the residual model is that it is, crudely, modelling the correlations of the feature vector. A standard soft tying approach to this is to use semi-tied covariance matrices [7]. On this task a global semi-tied transform yielded an error rate of 46.1%. Thus the use of a simple global residual model achieves about the same performance as a global semi-tied system on this task.

4. CONCLUSIONS

This paper has described a new form of acoustic model for speech recognition, the HMM error model. This model uses an general extension to standard factorial HMMs, transformation streams. The model has an intuitive representation of a non-linear filter model and residual model. Re-estimation formulae and initialisation schemes were described. The new model was evaluated on a large vocabulary speech recognition tasks. On this task the new form of model performed significantly better than standard HMMs. Similar significant reductions in error rate were observed on the Wall Street Journal task [4].

Very few of the possible options for the HEM, or more generally the use of transformation streams, have been investigated in this paper. In particular the form of the HEM model used here concentrates the complexity on the filter model rather than for example using a more complex residual model (possibly using the full power of factorial HMMs with transformation streams). An interesting option would be to use a distribution modelling scheme such as Gaussianization [11]. The model has also not been assessed in terms of how it performs with speaker adaptation (nor whether it is better to simply adapt the residual model rather than the filter model). Finally the use of discriminative training techniques is popular in state-of-the-art speech recognition. It is interesting to know whether HEMs are suitable for discriminative training.

[1] Due to the nature of the HEM, which uses a GMM for the residual model, it is more appropriate to compare the performance with soft tied systems rather than factorial HMM systems. To the author's knowledge there are no performance figures for factorial HMMs on systems of this size.

[2] In the published paper [6] the soft results quote on the larger 180 hour training data set. It should also be noted that due to minor differences in the initialisation of the standard systems the baseline performance for the h5train00sub training was 46.6%, compared to the 46.7% obtained in these experiments.

5. REFERENCES

[1] Z Ghahramani and M I Jordan, "Factorial hidden Markov models," *Machine Learning*, vol. 29, pp. 245–275, 1997.

[2] H Nock, *Techniques for modelling Phonological Processes in Automatic Speech Recognition*, Ph.D. thesis, Cambridge University, 2001.

[3] S Matsoukas and G Zavaliagkos, "Convolutional density estimation in hidden Markov models for speech recognition," in *Proc. ICASSP*, 1999.

[4] M J F Gales, "Transformation streams and the HMM error model," Tech. Rep. CUED/F-INFENG/TR416, Cambridge University, 2001, Available from: svr-www.eng.cam.ac.uk/˜mjfg.

[5] S J Young, J Jansen, J Odell, D Ollason, and P Woodland, *The HTK Book (for HTK Version 2.0)*, Cambridge University, 1996.

[6] T Hain, P C Woodland, G Evermann, and D Povey, "The CU-HTK March 2000 HUB5E transcription system," in *Proceedings of the Speech Transcription Workshop*, 2000.

[7] M J F Gales, "Semi-tied covariance matrices for hidden Markov models," *IEEE Transactions Speech and Audio Processing*, vol. 7, pp. 272–281, 1999.

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

[9] M J F Gales, "Maximum likelihood linear transformations for HMM-based speech recognition," *Computer Speech and Language*, vol. 12, pp. 75–98, 1998.

[10] M J F Gales, K M Knill, and S J Young, "State-based Gaussian selection in large vocabulary continuous speech recognition," *IEEE Transactions Speech and Audio Processing*, 1999.

[11] S S Chen and R A Gopinath, "Gaussianization," in *Proceedings Neural Information Processing Systems*, 2000, pp. 423–429.