Subspace Models for Bottleneck Features

Jun Qi[1], *Dong Wang*[2,3,4], *Javier Tejedor*[5]

[1] Department of Electronic Engineering, Tsinghua University, Beijing, 100084, China
[2] Center for Speech and Language Technologies, Division of Technical Innovation and Development, Tsinghua National for Information Science and Technology
[3] Center for Speech and Language Technologies, Research Institute of Information technology, Tsinghua
[4] Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China
[5] Human Computer Technology Laboratory, Universidad Autónoma de Madrid, Spain

qij08@mails.tsinghua.edu.cn, wangdong99@mails.tsinghua.edu.cn
javier.tejedor@uam.es

Abstract

The bottleneck (BN) feature, particularly based on deep structures, has gained significant success in automatic speech recognition (ASR). However, applying the BN feature to small/medium-scale tasks is nontrivial. An obvious reason is that the limited training data prevent from training a complicated deep network; another reason, which is more subtle, is that the BN feature tends to possess high inter-dimensional correlation, thus being inappropriate to be modeled by the conventional diagonal Gaussian mixture model (GMM). This difficulty can be mitigated by increasing the number of Gaussian components and/or employing full covariance matrices. These approaches, however, are not applicable for small/medium-scale tasks for which only a limited amount of training data is available.

In this paper, we study the subspace Gaussian mixture model (SGMM) for BN features. The SGMM assumes *full* but *shared* covariance matrices, and hence can address the inter-dimensional correlation in a parsimonious way. This is particularly attractive for the BN feature, especially on small/medium-scale tasks, where the inter-dimensional correlation is high but the full covariance modeling is not affordable due to the limited training data. Our preliminary experiments on the Resource Management (RM) database demonstrate that the SGMM can deliver significant performance improvement for ASR systems based on BN features.

Index Terms: bottleneck features, subspace Gaussian mixture model, automatic speech recognition

1. Introduction

Bottleneck (BN) features are generated from the 'bottleneck layer' of a multiple layer perceptron (MLP). The MLP is designed such that the input corresponds to certain primary features, and the output corresponds to the corresponding labels (e.g., context-independent phone states), so the bottleneck layer learns some latent patterns of the input features by training on a large amount of data. In prediction, the activations of the units at the bottleneck layer, which can be regarded as the coefficients of the input feature on the learned patterns, yield the BN feature. A multitude of research has demonstrated that the BN feature can deliver significant performance improvement for automatic speech recognition (ASR) [1, 2] when compared to the primary features, such as the Mel frequency cepstral coefficients (MFCC), the perceptual linear predictives (PLP) and Gammatone frequency cepstral coefficients (GFCC). Recent work shows that, with a deep network structure, the BN feature is even more promising [3, 4, 5].

A particular problem associated with the BN feature is that the feature vectors tend to be highly sparse, i.e., most of the probability mass concentrates on one or a few dimensions. This leads to significant correlation among different dimensions, thus causing serious problems in acoustic modeling based on Gaussian mixture models (GMMs), which often assumes diagonal covariance matrices for the Gaussian components. A widely used approach to dealing with the correlation is to project the BN feature onto orthogonal bases by the Karhunen–Loéve (K-L) transform [6] so that the projected feature is dimensional uncorrelated. The shortcomings of the KL approach are two-fold: first, the transform is linear, and second it is model-independent, which leads to limited effectiveness. A more systematic solution is to involve a large number of Gaussian components in the GMM and allow full covariance matrices for the Gaussians [7, 8]. This approach certainly addresses the dimensional correlation if the training data are abundant; for small/medium-scale tasks, however, such complex models are not affordable due to the limitation on training data.

In this paper, we employ the subspace Gaussian mixture model (SGMM) to tackle the problem. The SGMM assumes full but shared covariance matrices for the Gaussian components, and can deal with dimensional correlation in an approximated but parsimonious way [9]. We argue that this parsimonious solution is particularly attractive for the BN feature especially on small/medium-scale tasks, where the correlation is high and the full covariance modeling is not affordable.

The rest of the paper is organized as follows: Section 2 presents the implementation of the BN feature, and Section 3 presents the SGMM for the BN feature. The experiments are reported in Section 4 and the paper is concluded in Section 5.

2. Bottleneck feature

The BN feature is derived from the bottleneck layer of an MLP structure. A typical bottleneck MLP is shown in Figure 1. There are 5 layers in total and 3 of them are hidden. The units at the input layer (at the bottom) correspond to a long-context feature vector that is generated by concatenating 31 consecutive frames of the primary feature followed by a discrete cosine transform (DCT). The dimension of the primary feature is 39, and hence the context window involves 1209 primary feature elements. After the DCT, the dimension is reduced to 624, which corresponds to the number of units at the input layer of the MLP.

The output layer (at the top) consists of 145 units, corresponding to 145 states of 48 context-independent phones. The three hidden layers are constructed following a 1024-39-1024 configuration, where the 39-unit layer (in the middle) is the 'bottleneck layer', and the activations of the units yield the BN feature. The configuration is selected to optimize the ASR performance on a development set.

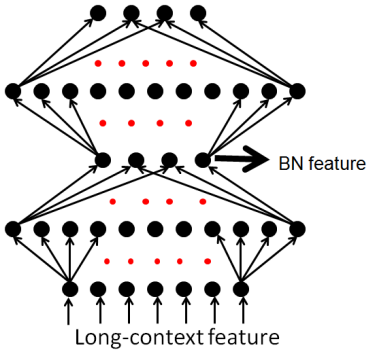


Figure 1: *Bottleneck feature extraction.*

The bottleneck structure shown in Figure 1 consists of a number of hidden layers and hence is regarded as 'deep'. A layer-by-layer growing approach can be employed to train such a network. In this study, we employ a popular pre-training approach based on the restricted Boltzmann machine (RBM) [5]. An RBM is an undirected graph model which consists of a hidden layer stacked on the top of a visible layer, as shown in Figure 2.

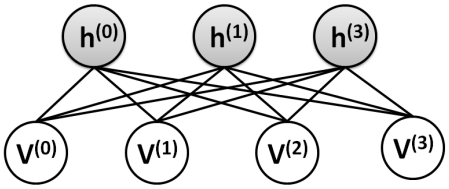
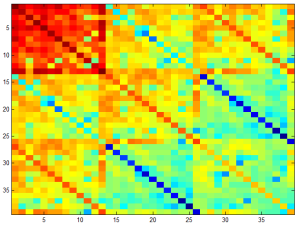


Figure 2: *Graphical representation of the RBM.*

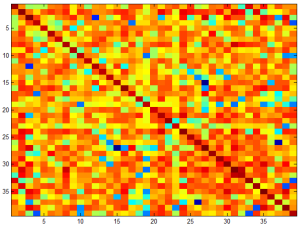
The RBM can be optimized in the sense of maximum-likelihood (ML) by the stochastic gradient descendent (SGD) approach. We use the contrastive divergence algorithm proposed by Hinton [5]. This algorithm, although an approximation to the SGD, can significantly speed up the optimization. Once the RBM is optimized, its parameters are fixed and the undirected edges are changed to direct edges. The fixed directed graph is then applied on the input feature as a stochastic transform to train the next RBM. This leads to an iterative approach that grows the network layer by layer, until the desired structure is obtained. In our experiments, the learning rate is set to 0.001 at the first iteration, and to 0.01 for the remaining iterations. In addition, the mini-batch size of the SGD is set to 512 and the momentum rate is set to 0.5.

Once the MLP structure is obtained by SGD, the conventional back-propagation [10] is employed to learn the parameters in a supervised and discriminative way. The learning rate is set in an auto-regularized manner: it is set to 4.0 at the first several iterations, and then is decreased by a factor of 2 at each of the remaining iterations. The maximum iteration is set to 20, and the convergence is tested on a development set.

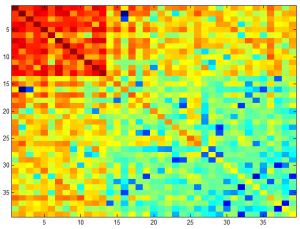
A particular property of the bottleneck MLP is that the bottleneck layer can learn some prominent patterns of the input speech signals in the training phase. In prediction, the BN feature represents the coefficients of the input primary feature based on the learned patterns. Since the patterns are discriminatively representative, the BN features are highly sparse, i.e., most of the mass of the feature concentrates on a few dimensions. This in turn leads to high correlation among the dimensions of the feature. In order to examine the correlation, we plot the covariance matrices of BN features and their corresponding primary features. Three primary features are selected: MFCC, PLP and GFCC. The results are shown in Figure 3-Figure 5 for the three types of features, where plots (a) and (b) present the correlation of the primary feature and the BN feature respectively. We observe that the BN feature is much more dimensional correlated than the primary feature, in all the cases of MFCC, PLP and GFCC. We also examine the contribution of the KLT, a widely used de-correlation approach. The covariance matrix of the KLT-applied BN features is shown in the plot (c) of Figure 3-Figure 5. We observe that the dimensional correlation is a little reduced by the KLT, though the impact is very marginal.



(a) MFCC



(b) MFCC-BN

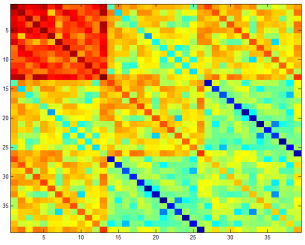


(c) MFCC-BN + KLT

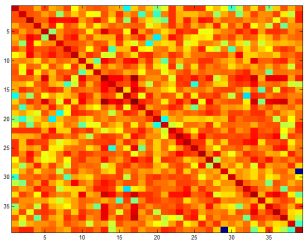
Figure 3: Covariance matrices of (a) MFCC, (b) MFCC-BN and (c) MFCC-BN + KLT.

3. SGMM for bottleneck feature

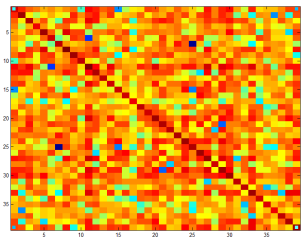
The conventional GMM-HMM framework for ASR assumes that the covariance matrices of the Gaussian components are diagonal. This assumption is obviously strong but it is necessary for small/medium-scale tasks which suffer from limited training data. For primary features such as MFCC, PLP and GFCC, this assumption partially holds due to the relatively dense distribution of the feature mass and the DCT at the end of the processing pipeline. For BN features, though, the high correlation among dimensions is difficult to remove, as we have seen in the previous section. The KLT is a useful tool for de-correlation, however its contribution to the BN feature is marginal. The limitation of the KLT is tow-fold: first, it is a linear orthogonal transform, while the correlation in the BN feature is clearly more complex; second, it is a global transform, while the correlation patterns of the phones/states are obviously different. For these reasons, the KLT is not enough to deal with the BN feature



(a) PLP

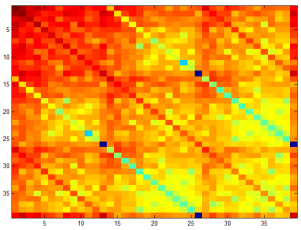


(b) PLP-BN

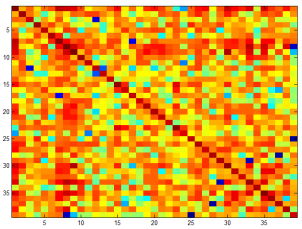


(c) PLP-BN + KLT

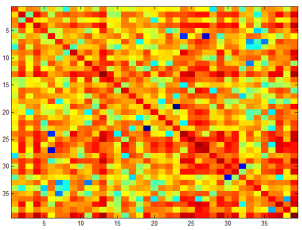
Figure 4: Covariance matrices of (a) PLP, (b) PLP-BN and (c) PLP-BN + KLT.



(a) GFCC



(b) GFCC-BN



(c) GFCC-BN + KLT

Figure 5: Covariance matrices of (a) GFCC, (b) GFCC-BN and (c) GFCC-BN + KLT.

inter-dimensional correlation.

We thus resort to a more systematic approach, i.e., model the correlation using non-diagonal Gaussians. The ideal solution is to employ a large number of Gaussian components with full covariance matrices. This approach, however, is not suitable for small/medium-scale tasks for which the training data are usually limited. A subspace model is a more feasible choice, which on the one hand relaxes the diagonal covariance assumption and thus can model the correlation, and on the other hand, it assumes some shared structures, thus modeling the correlation in a parsimonious way.

There are some subspace models, such as EMLLT [11], S-PAM [12]. In this study, we choose the subspace Gaussian mixture model [13]. Basically, the SGMM assumes all phonetic states share a common Gaussian mixture model structure, and the means and mixture weights vary in a subspace of the total parameter space. This compact representation allows full covariance matrices for the shared Gaussian components, and thus models the dimensional correlation with a comparable number of parameters as the regular GMM.

The probability of a feature $\boldsymbol{x}$ at the j-th state is given as follows:

$$p(\boldsymbol{x}|j) = \sum_{i=1}^{I_j} c_{ji} \sum_{m=1}^{M} w_{jim} \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}_{jim}, \boldsymbol{\Sigma}_m) \qquad (1)$$

where i indexes the sub-states, and m indexes the mixtures. The weight of the i-th sub-state is denoted by c_{ji} and the weight of the m-th mixture in the i-th sub-state is denoted by w_{jim}. The mean vector $\boldsymbol{\mu}_{jim}$ spans in a subspace as follows:

$$\boldsymbol{\mu}_{jim} = \boldsymbol{M}_m \boldsymbol{v}_{ji} \qquad (2)$$

where v_{ji} represents a state-specific vector at the mixture j of sub-state i, and $\boldsymbol{M}_m$ defines the subspace. Note that the covariance matrix $\boldsymbol{\Sigma}_m$ is full and is shared across states and mixtures, and therefore the model is able to model non-diagonal covariances with a small number of parameters. The parameter set $\{c_{ji}, w_{jim}, \boldsymbol{M}_m, \boldsymbol{v}_{ji}, \boldsymbol{\Sigma}_m\}$ can be estimated by an expectation-maximization (E-M) procedure as with the conventional GMM [13].

The entire system architecture that models the BN feature with the SGMM is shown in Figure 6, where a conventional GMM-HMM system is first constructed, and then a universal background model (UBM) is created by clustering the Gaussian components. Afterwards, the SGMM is initialized by copying the UBM, and then trained by an E-M algorithm similar to the GMM [13].

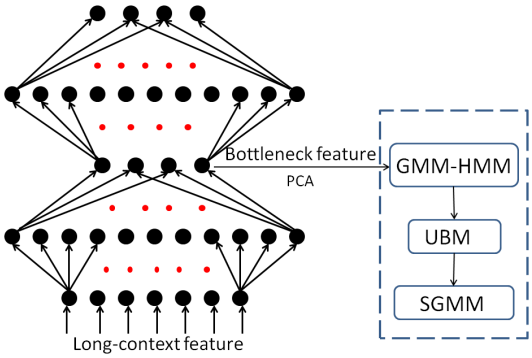


Figure 6: *An illustration of the SGMM approach with the BN feature.*

4. Experiments

4.1. Data profile

Our experiment is conducted on the DARPA Resource Management (RM) continuous speech corpus. The RM-SI dataset (3990 sentences) is used to train the bottleneck MLP and the GMM/SGMM models. From the training set, 1000 sentences are chosen to train an initial monophone system; the same data are also used for cross-validation in the MLP training. We note that the training data are rather limited, and thus applying advanced techniques such as large scale deep learning and full covariance

modeling is highly challenging.

The test data involve 6 datasets from the RM corpus, including Feb-89, Feb-91, Mar-87, Sep-92, Oct-87 and Oct-89. This amounts to 1460 utterances in total.

We study three popular features in the experiments: MFCC, PLP and GFCC. These 'primary features' are used to build the baseline systems. For a fair comparison, all the primary features are derived from the same frequency range (80-5000 Hz) with the same frame rate (100 frames per second). Each feature involves 13 static components (12 static + C0) plus their first and second order derivatives, resulting in 39-dimensional feature vectors. Cepstral mean subtraction (CMS) is applied to compensate for channel variation.

The Kaldi toolkit[1] is utilized to train the acoustic models, to build the decoding graphs and to conduct the decoding. The TNet toolkit[2] is used to train the MLPs. The GFCC feature is generated by a public toolkit developed by the authors[3].

4.2. Baseline results

We first build the baseline systems which are based on the conventional diagonal GMM. Each context-dependent phone (triphone) is modeled as a hidden Markov model (HMM) which consists of three left-to-right non-skipping states. The probability distribution of each state is modeled by a GMM that comprises 8 Gaussian components. The states belonging to the same phone are tied by a decision tree, and there are 2700 tied states in total. This leads to overall 9000 Gaussian components. A tri-gram model with 1000 words in the lexicon is used as the language model in decoding.

The experiments are conducted with the three primary features and the corresponding BN features based on them. All the features are of 39 dimensions. For the BN features, we study their original forms and the forms with the KLT applied. The experimental results are reported in Table 1 in terms of the word error rate (WER).

	WER%		
	MFCC	PLP	GFCC
Primary	4.13	4.25	3.84
BN	3.91	4.03	3.66
BN + KLT	3.56	3.68	3.42

Table 1: WER of the GMM baseline systems.

We first observe that the BN features outperform the corresponding primary features in a consistent way. The relative WER reduction is about 5%. This demonstrates that even with a small training set, the BN features based on deep learning still contribute. Furthermore, we find that the KLT provides additional performance gains for the BN features (relative EER reduction of 6.3% in average), confirming its contribution as a decorrelation method. When comparing MFCC, PLP and GFCC, we see clear advantage with GFCC. This is consistent with our previous study in [14] and can be attributed to the capability of GFCC in simulating the auditory process of humans.

4.3. SGMM results

In the second experiment, we study the SGMM modeling for the primary features and the BN features. In order to construct the SGMM, a UBM was first created by clustering all the Gaussian components of the GMM system. The final UBM is composed of 400 Gaussian components in our experiment. The full covariance matrices of the Gaussian components are then trained via an E-M procedure. Afterwards, the SGMM is constructed by copying the UBM followed by an E-M full training [13]. The dimension of the state-specific vector $\boldsymbol{v}_{ji}$ is set to 40, which leads to an almost square mean projection matrix $\boldsymbol{M}_m$ (refer to (1) and (2)). The final number of sub-states is 7495 in our experiment.

The WER results of the SGMM systems are presented in Table 2. We find that the SGMM leads to substantial performance improvement over the GMM baseline with both the primary and BN features. More significantly, this subspace approach provides much more performance gains for the BN features than for the primary features (31.9% vs. 20.9%). This supports our argument that subspace models are more important for the BN features than for the primary features.

If we regard the SGMM and the KLT as two de-correlation approaches, and compare their contribution, we find the SGMM (the 'BN' row of Table 2) is much more efficient than the KLT (the 'BN+KLT' row of Table 1), as we conjectured in Section 3. Finally, we find that the SGMM and the KLT can be combined to deliver further performance improvement, though very marginal. This is shown in the last row of Table 2.

	WER%		
	MFCC	PLP	GFCC
Primary	3.12	3.32	3.21
BN	2.43	2.96	2.51
BN + KLT	2.38	2.82	2.45

Table 2: WER of the SGMM systems.

5. Conclusions

This paper studied subspace modeling for the BN feature. The motivation is that BN features are highly dimensional correlated and therefore they are difficult to be modeled by the conventional diagonal GMM. In addition, the limited training data in small/medium-scale tasks prohibit the full covariance modeling. Subspace models such as the SGMM are therefore the most suitable to trade-off the model complexity and the training data. Our experiments on the RM corpus showed that the SGMM approach is highly effective to model the dimensional correlation for the BN feature, leading to significant performance gains over the conventional diagonal GMM. More interestingly, the gain with the SGMM is much more significant for the BN features than for the primary features (20.9% vs. 31.9%). This supports our argument that BN features should be used together with subspace modeling, particularly in small/medium-scale tasks. Future work involves studying BN features based on heterogeneous primary features, and the SGMM approach for other highly correlated features, such as sparse representations.

6. Acknowledgements

This work was partially supported by Tencent Ltd. under the contract No. 20132000294. Special thanks to Jenny Chen, Li Lu and Eryu Wang from Tencent Ltd. for the initial discussion and close collaboration.

[1] https://kaldi.svn.sourceforge.net/svnroot/kaldi/trunk kaldi-trunk

[2] http://speech.fit.vutbr.cz/software/neural-network-trainer-tnet

[3] http://homepages.inf.ed.ac.uk/v1dwang2/public/tools/index.html

7. References

[1] Q. Zhu, Y. Chen, and N. Morgan, "On using MLP features in LVCSR," in *Proc. Interspeech'04*, 2004, pp. 921–924.

[2] F. Grezl, M. Karafiat, S. Kontar, and J. Cernocky, "Probabilistic and bottleneck features for lvcsr of meetings," in *ICASSP'07*, 2007, pp. 757–760.

[3] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition," *IEEE Signal Processing Magazine*, vol. 28, no. 6, pp. 82–97, 2012.

[4] D. Yu and M. L. Seltzer, "Improved bottleneck features using pretrained deep neural networks," in *Proc. Interspeech'11*, 2011, pp. 237–240.

[5] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, pp. 504–507, 2006.

[6] J. Frankel, D. Wang, and S. King, "Growing bottleneck features for tandem ASR," in *Proc. Interspeech'08*, 2008, p. 1549.

[7] P. Bell, "Full covariance modelling for speech recognition," Ph.D. dissertation.

[8] P. Olsen, V. Goel, and S. Rennie, "Discriminative training for full covariance models," in *Proc. ICASSP'11*, 2011, pp. 5312–5315.

[9] D. Povey, "Subspace Gaussian mixture models for speech recognition," in *Proc. ICASSP'10*, 2010, pp. 4330–4333.

[10] M. Christopher, *Pattern Recognition and Machine Learning*. New York, Inc. Secaucus, NJ, USA: Springer, 2007.

[11] P. Olsen and R. A. Gopinath, "Modeling inverse covariance matrices by basis expansion," in *Proc. ICASSP'02*, 2002, pp. 945–948.

[12] N. K. Goel and A. G. Andreou, "Heteroscedastic discriminant analysis and reduced-rank HMMs for improved speech," *Speech Communication*, vol. 26, no. 4, pp. 283–297, 1998.

[13] D. Povey, "A tutorial introduction to subspace Gaussian mixture models for speech recognition," MSR-TR-2009-11, Microsoft Research, Tech. Rep., 2009.

[14] J. Qi, D. Wang, Y. Jiang, and R. Liu, "Auditory feature based on gammatone filters for robust speech recognition," in *the IEEE International Symposium on Circuits and Systems (ISCAS)*, 2013.