STREAM DERIVATION AND CLUSTERING SCHEME FOR SUBSPACE DISTRIBUTION CLUSTERING HIDDEN MARKOV MODEL

Brian Mak Enrico Bocchieri
AT&T Labs - Research
180 Park Ave
Florham Park, NJ 07932, USA

Etienne Barnard
Oregon Graduate Institute
20000 NW Walker Rd
Portland, OR 97006, USA

Abstract - In [1], our novel subspace distribution clustering hidden Markov model (SDCHMM) made its debut as an approximation to continuous density HMM (CDHMM). Deriving SDCHMMs from CDHMMs requires a definition of multiple streams and a Gaussian clustering scheme. Previously we have tried 4 and 13 streams, which are common but ad hoc choices. Here we present a simple and coherent definition for streams of any dimension: the streams comprise the most correlated features. The new definition is shown to give better performance in two recognition tasks. The clustering scheme in [1] is an $O(n^2)$ algorithm which can be slow when the number of Gaussians in the original CDHMMs is large. Now we have devised a modified k-means clustering scheme using the Bhattacharyya distance as the distance measure between Gaussian clusters. Not only is the new clustering scheme faster, when combined with the new stream definitions, we now obtain SDCHMMs which perform at least as well as the original CDHMMs (with better results in some cases).

1 Introduction

In our SDCHMM debut paper [1], we presented a novel derivative of the continuous density HMM (CDHMM) which we call "subspace distribution clustering hidden Markov model" (SDCHMM). SDCHMMs are derived from CDHMMs by projecting mixture Gaussians of CDHMMs into disjoint subspaces, and the subspace Gaussians are then clustered into a small number of Gaussian prototypes. By exploiting the combinatorial effect of subspace Gaussian encoding, all mixture Gaussians can be represented by some combination of a small number of subspace Gaussian prototypes (or codewords). In our experience, 16 to 128 prototypes are generally adequate to give good accuracy. Consequently there is a great reduction in model parameters, and thus substantial savings in memory and computation. This renders SDCHMMs very attractive in practical implementation of acoustic models.

From the perspective of quantization, SDCHMM approximates CDHMM and achieves great data compression by Gaussian distribution quantization. From the perspective of parameter tying, SDCHMM may allow acoustic modeling to a greater detail without requiring more training data. Finally, SDCHMM unifies the theory of CDHMM and feature-level tying HMM [7]. This is because when there is only

0-7803-3698-4/97/$10.00 © 1997 IEEE.

one subspace, SDCHMM falls back to the conventional CDHMM, whereas if each subspace is one dimension of the feature space (scalar), it becomes the feature-level tying HMM.

SDCHMM requires a definition of the disjoint subspaces (or streams) and a Gaussian clustering scheme. We extend our previous work in [1] by:

- a simple and coherent definition for streams of any dimension;
- an $O(nkN)$ modified k-means subspace Gaussian clustering algorithm to replace the previous $O(n^2)$ clustering scheme where $kN \ll n$ in large vocabulary recognition system; and,
- SDCHMM recognition on two tasks, ATIS and HMIHY.

Both new techniques for the generation of SDCHMMs lead to better performance.

2 Review Of SDCHMM

Using the following notations:

$P(\mathbf{O})$: state output probability given observation $\mathbf{O}$
$\mathbf{O}_k$: k-th stream of observation $\mathbf{O}$
c_m : weight of the m-th mixture
$\mu_{\mathbf{m}}$: mean vector of the m-th mixture
$\sigma^{\mathbf{2}}_{\mathbf{m}}$: variance vector of the m-th mixture
c_{mk} : weight of the m-th mixture of the k-th stream
$\mu_{\mathbf{mk}}$: mean vector of the m-th mixture of the k-th stream
$\sigma^{\mathbf{2}}_{\mathbf{mk}}$: variance vector of the m-th mixture of the k-th stream

and assuming that observation pdfs of a CDHMM are Gaussians with diagonal covariances, the state output probability can be rewritten as follows:

$$\begin{aligned} P(\mathbf{O}) &= \sum_{m=1}^{M} c_m N(\mathbf{O}; \mu_{\mathbf{m}}, \sigma^{\mathbf{2}}_{\mathbf{m}}), \quad \sum_{m=1}^{M} c_m = 1 && (1) \\ &= \sum_{m=1}^{M} c_m \left(\prod_{k=1}^{K} N(\mathbf{O}_k; \mu_{\mathbf{mk}}, \sigma^{\mathbf{2}}_{\mathbf{mk}}) \right) && (2) \\ &\approx \sum_{m=1}^{M} c_m \left(\prod_{k=1}^{K} N^{tied}(\mathbf{O}_k; \mu_{\mathbf{mk}}, \sigma^{\mathbf{2}}_{\mathbf{mk}}) \right) && (3) \end{aligned}$$

The key observation is that Gaussians with diagonal covariances can be expressed as a product of subspace Gaussians as in Equation (2), each with diagonal covariance where the subspaces are disjoint and together span the original full feature space. The proposed SDCHMM as in Equation (3) is obtained by clustering the Gaussians in each subspace to a small number of subspace Gaussian prototypes. Thus SDCHMM can be considered as an approximation to conventional CDHMM.

Since it has been proved by years of research that CDHMM is a good model for speech recognition, a carefully designed approximation to the CDHMM formulation — SDCHMM — should, in principle, also deliver high performance.

3 Issue I: Subspaces Definition

In our previous paper [1], SDCHMMs were tested with "common" stream definitions which are designed in an ad hoc fashion. Here we use the heuristics that correlated features, by definition, should tend to cluster in a similar manner, and we require each stream to comprise the most correlated features. Intuitively this criterion should result in smaller distortions for the clustered subspace Gaussians. It also gives a single coherent definition for *any* arbitrary number of streams of *any* dimension.

While the correlation between 2 features, ρ_{ij}, is commonly measured by Pearson's moment product correlation coefficient,

$$\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \sigma_j}, \tag{4}$$

multiple correlation measures among 3 or more features are less well-defined. In this paper, we define multiple correlation which will be denoted as R as

1 $-$ determinant of correlation matrix of the features.

That is, the multiple correlation R among k features is,

$$R = 1 - \begin{vmatrix} 1 & \rho_{12} & \rho_{13} & \cdots & \rho_{1k} \\ \rho_{21} & 1 & \rho_{23} & \cdots & \rho_{2k} \\ \rho_{31} & \rho_{32} & 1 & \cdots & \rho_{3k} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \rho_{k1} & \rho_{k2} & \rho_{k3} & \cdots & 1 \end{vmatrix} \tag{5}$$

In particular, when there are only 2 features,

$$R = 1 - \begin{vmatrix} 1 & \rho_{ij} \\ \rho_{ji} & 1 \end{vmatrix} = \rho_{ij}^2$$

Hence, in the case when there are only two features, R equals the square of the moment product correlation coefficient. Since the correlation matrix is symmetric, its determinant is equal to the product of its eigenvalues. Therefore,

$$R = 1 - \lambda_1 \lambda_2 \cdots \lambda_k \tag{6}$$

where λ_j is the j-th eigenvalue of the correlation matrix. Equation (6) gives a geometrical interpretation to the multiple correlation measure. When the features are highly correlated, the correlation matrix corresponds to an elongated ellipsoid with most eigenvalues except one being small, giving a small value for their product and thus a high value of R. When the features are less correlated, the matrix is more spherical, giving a higher value for the eigenvalue product and smaller value of R. It can also easily be shown that R has the following desirable properties of a correlation measure:

- $0 \leq R \leq 1$
- when all features are correlated, i.e. $\forall i, j,\ \rho_{ij} = 1, R = 1$
- when all features are uncorrelated, i.e. $\forall i, j,\ \rho_{ij} = 0, R = 0$

Practically we apply a greedy algorithm to obtain the most correlated streams as depicted in Algorithm 1. It is simple to modify the algorithm in cases when F is not a multiple of K.

Algorithm 1 Selection of most correlated streams

Goal: Given F features, define K n-dimensional streams with $F = nK$.

Step 1. Compute the multiple correlation among *any* n features.

Step 2. Sort the multiple correlation values in descending order, each tagged by an n-feature-tuple indicating the features it computes from.

Step 3. Starting from the top, an n-feature-tuple is moved from the sorted list to the "solution list" if *none* of its features already appear in any feature-tuples of the solution list.

Step 4. Repeat Step 3 until all features appear in the solution list.

Step 5. The feature-tuples in the "solution list" are the K-stream definition.

4 Issue II: Subspace Gaussian Clustering

Previously in [1], subspace Gaussians were clustered by a bottom-up agglomerative clustering scheme of $O(n^2)$ complexity in a similar way as in [2] in which two Gaussians are merged if they result in minimum distortion(scatter) increase. To avoid an otherwise $O(n^3)$ complexity, we introduced in [2] the heuristic that at each iteration, the Gaussian corresponding to the smallest training ensemble must be merged first. Algorithm 2 shows a newly devised $O(nkN)$ modified k-means clustering algorithm which derives the subspace Gaussian prototypes without using such heuristics, where N is the number of required subspace Gaussians per stream and k is the number of iterations in going through the k-means clustering algorithm. Usually $kN \ll n$ for large systems.

Since the entities to cluster are Gaussians, we adopt as distance measure the classification-based Bhattacharyya distance which has been shown to perform well in speech-related tasks [5, 6]. The Bhattacharyya distance captures both the first and the second order statistics, and is expected to give better clustering results than the previous distortion measure.

Algorithm 2 Modified K-means algorithm for clustering subspace Gaussians

Goal: To derive a K-stream SDCHMM with N subspace Gaussian prototypes for each stream.

Step 1. *Initialization:* First train a 1-stream Gaussian mixture model with N mixture components. Project each of the N mixture Gaussians into the K subspaces according to the given K-stream specification. The resultant KN subspace Gaussians will be used as initial subspace Gaussian prototypes.

Step 2. Similarly project each Gaussian pdf in the original CDHMM into the K subspaces.

Step 3. For each stream, repeat Step 4 & 5 until some convergence criterion is met.

Step 4. *Membership*: Associate each subspace Gaussian of CDHMM in each stream with its nearest prototype as determined by their Bhattacharyya distance.

Step 5. *Update*: Merge all subspace Gaussians which share the same nearest prototype to become the new subspace Gaussian prototypes.

5 Recognition Evaluation

Two recognition tasks are used for evaluation: ARPA–ATIS [3] and ATT–HMIHY [4]. ATIS (Airline Travel Information Service) is a task containing spontaneous speech for air travel information queries, with a 1532-word vocabulary. In these experiments, we have used word-class bigrams language model with a perplexity of about 20. HMIHY (How May I Help You) is an AT&T speech corpus containing spontaneous responses to the open-ended prompt of "How may I help your?" over the telephone. It has a vocabulary of about 3600 words and a phrase bigram language model of perplexity of 18. The disfluencies in HMIHY speeches make word recognition very difficult.

In all experiments, speech is represented by a 39-dimensional feature vector consisting of 12 MFCCs and power, their first and second order time derivatives at a frame rate of 10ms. Cepstral mean subtraction is applied and recognition is done using a one-pass beam search with appropriate pruning thresholds.

5.1 Improved Results With Streams of Correlated Features and MKM Gaussian Clustering

Fig. 1 and Fig. 2 show incremental improvements in recognition performance on ATIS and HMIHY obtained by 13-stream context-independent SDCHMMs with the four combinations of using the old or the new 13-stream definitions, and the old or the new clustering schemes. (The common 13-stream definition is defined as in [1] as follows: 12 streams of triplets $\{\text{cep}, \Delta\text{cep}, \Delta\Delta\text{cep}\}$ and 1 stream composed of $\{\text{power}, \Delta\text{power}, \Delta\Delta\text{power}\}$.)

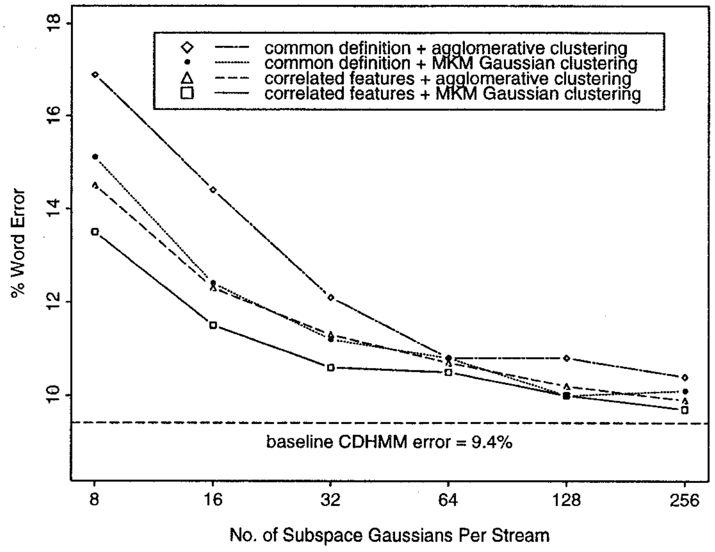

Figure 1: Recognition accuracy of 13-stream context-independent SDCHMMs on ATIS with various stream definitions and clustering schemes

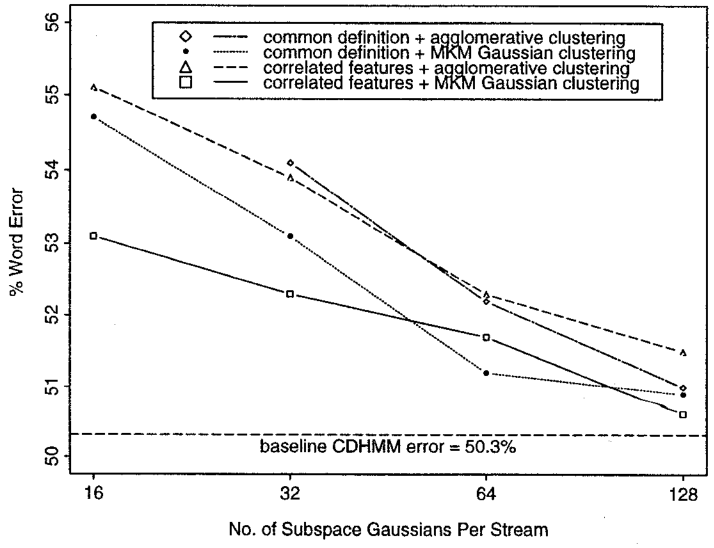

Figure 2: Recognition accuracy of 13-stream context-independent SDCHMMs on HMIHY with various stream definitions and clustering schemes

Notice that the improvement is bigger with fewer prototypes and this is desirable as fewer prototypes usually translate into faster recognition.

5.2 Summary of Best Results

Using the two novel techniques, the best results for various SDCHMM configurations (with different number of streams and different number of prototypes) are summarized in Table 1 and Table 2. Context-independent ATIS SDCHMMs give an insignificant(absolute) 0.1% drop in accuracy, while ATIS context-dependent SDCHMMs and HMIHY models give slightly better accuracies than their original CDHMMs.

Table 1: Summary of the best ATIS results (K = #streams, n = #tied Gaussians, CI = context independent, CD = context dependent, WER = word error rate, TIME is relative to that of the baseline system, PR = parameter reduction and MS = memory savings. For PR, figures in parenthesis takes into account the mappings of subspace Gaussians to the full-space Gaussians. For MS, 1-byte mappings are assumed)

CI/CD	K	n	WER	TIME	PR	MS
CI	1	2302	9.4	1.00	1	1
CI	13	256	9.7	0.72	8 (3.5)	6.1
CI	20	128	9.5	0.70	15 (3.1)	7.6
CI	39	32	9.5	0.70	38 (1.9)	6.7
CD	1	76725	5.2	1.00	1	1
CD	4	256	5.8	0.42	63 (15)	35
CD	13	128	5.2	0.44	70 (5.6)	18
CD	20	64	5.0	0.50	74 (3.8)	13
CD	39	16	5.0	0.71	78 (2.0)	7.3

Table 2: Summary of the best HMIHY results (refer Table 1 for notations)

CI/CD	K	n	WER	TIME	PR	MS
CI	1	2829	50.3	1.0	1	1
CI	13	128	50.6	0.63	17 (4.5)	10
CI	20	64	50.2	0.65	29 (3.5)	10
CI	39	16	50.0	0.81	55 (2.0)	7

6 Conclusion

We show that by properly projecting mixture Gaussians of accurate CDHMMs into subspaces and carefully tying the resultant subspace Gaussians, performance of SDCHMMs can be greatly improved, especially in those cases with few Gaussian prototypes per stream. Using the two new techniques, we now can achieve recognition results on ATIS and HMIHY that are at least as good as the baseline CDHMM results; yet the total computation time is reduced by 30% – 60% while HMM memory usage is decreased by a factor of 10 – 20. For example, on ATIS, a 20-stream CD SDCHMM system with only 64 subspace Gaussian prototypes (for each stream) is more accurate than a context-dependent CDHMM system containing 76725 mixture Gaussians, yet it runs at twice the speed with 75-fold (13-fold if encoding information is also counted and represented by 1 byte) decrease in memory usage. Similarly, on the HMIHY task, a 20-stream SDCHMM system with 64 subspace Gaussian prototypes (for each stream) performs better than the original CDHMM system it derives from, but saves 35% computation time and 96% (70% when encoding information is also counted) of the acoustic parameters.

We are now investigating if re-training will further improve the performance.

References

[1] E. Bocchieri and B. Mak. "Subspace Distribution Clustering for Continuous Observation Density Hidden Markov Models". In *Proc. of Eurospeech*, volume 1, pages 107–110, 1997.

[2] E. Bocchieri and G. Riccardi. "State Tying of Triphone HMM's for the 1994 AT&T ARPA ATIS Recognizer". In *Proc. of Eurospeech*, pages 1499–1502, 1995.

[3] D. Dahl et al. "Expanding the Scope of the ATIS Task: The ATIS-3 Corpus". *Proc. of ARPA Human Language Technology Workshop*, March 1994.

[4] A. Ljolje G. Riccardi, A.L. Gorin and M. Riley. "A Spoken language system for automated call routing". In *Proc. of ICASSP*, pages 1143–1146, 1997.

[5] P.C. Loizou and A.S. Spanias. "High-Performance Alphabet Recognition". *IEEE Trans. on Speech and Audio Processing*, 4(6):430–445, Nov 1996.

[6] B. Mak and E. Barnard. "Phone Clustering using the Bhattacharyya Distance". In *Proc. of ICSLP*, volume 4, pages 2005–2008, 1996.

[7] S. Takahashi and S. Sagayama. "Effects of Variance Tying for Four-Level Tied Structure Phone Models". In *Proc. of ASI Conference*, volume 1-Q-23, pages 141–142, 1995 (in Japanese).