

FACTOR ANALYSIS SIMPLIFIED

Patrick Kenny, Gilles Boulianne, Pierre Ouellet and Pierre Dumouchel

Centre de recherche informatique de Montréal (CRIM)

{pkenny, gboulian, pouellet, pdumouch}@crim.ca

ABSTRACT

We show how the factor analysis model for speaker verification can be successfully implemented using some fast approximations which result in minor degradations in accuracy and open up the possibility of training the model on very large databases such as the union of all of the Switchboard corpora. We tested our algorithms on the NIST 1999 evaluation set (carbon data as well as electret). Using warped cepstral features we obtained equal error rates of about 6.3% and minimum detection costs of about 0.022.

1. INTRODUCTION

Factor analysis of speaker and channel effects in large corpora such as the Switchboard databases has led to the development of an effective method of compensating for inter-session variability in speaker verification [1]. However the computational requirements of this approach make it difficult to experiment with so we have found it necessary to develop some fast approximations for training and testing the factor analysis model.

Although it has appealing theoretical properties, joint estimation of speaker and channel variability as in [1] in training the factor analysis model is impractical on a large scale. In this paper we will present a simplified training procedure in which speaker and channel effects are decoupled. This is based on the assumption that for each training speaker there are sufficiently many recordings that channel effects can be averaged out by pooling statistics across all of the recordings of the speaker. Our experience has been that this simplified training procedure runs 2–3 times as fast as the exact procedure in [1] and performs just as well.

A more serious problem with the factor analysis model is that evaluation of the Bayes factor used to make verification decisions is computationally very expensive particularly if there are large numbers of t-norm speakers. We have implemented an approximation to this calculation which enables the computation to be shared among t-norm speakers. With 50 t-norm speakers and 11 hypothesized speakers per test utterance, this approximation results in a 25 fold increase in speed at a cost of a 5% (relative) degradation in

performance as measured by the NIST detection cost function.

2. SPEAKER AND CHANNEL FACTORS

We assume a fixed GMM structure containing a total of C mixture components. Let F be the dimension of the acoustic feature vectors. Our basic assumption is that a speaker- and channel-dependent supervector can be decomposed into a sum of two supervectors, one of which depends on the speaker and the other on the channel, and that speaker supervectors and channel supervectors are both normally distributed. The dimensions of the covariance matrices of these distributions are enormous ($CF \times CF$) so we begin by explaining briefly how these covariance matrices are modeled.

Let \mathbf{m} denote the universal background supervector and let $\mathbf{M}(s)$ be the speaker supervector for a speaker s . We assume that, for a randomly chosen speaker s ,

$$\mathbf{M}(s) = \mathbf{m} + \mathbf{v}\mathbf{y}(s) + \mathbf{d}\mathbf{z}(s) \quad (1)$$

where \mathbf{d} is diagonal, \mathbf{v} is a rectangular matrix of low rank and $\mathbf{y}(s)$ and $\mathbf{z}(s)$ are independent random vectors having standard normal distributions. In other words, $\mathbf{M}(s)$ is assumed to be normally distributed with mean \mathbf{m} and covariance matrix $\mathbf{v}\mathbf{v}^* + \mathbf{d}^2$. The components of $\mathbf{y}(s)$ are *speaker factors*. The *speaker space* is the affine space defined by translating the range of $\mathbf{v}\mathbf{v}^*$ by \mathbf{m} . If $\mathbf{d} = \mathbf{0}$ then all speaker supervectors are contained in the speaker space. We will use the term Principal Components Analysis (PCA) to refer to this case; in the general case ($\mathbf{d} \neq \mathbf{0}$) the term $\mathbf{d}\mathbf{z}(s)$ serves as a residual which compensates for the fact that it may not be possible in practice to estimate \mathbf{v} reliably [2]. (See Fig. 1).

In order to incorporate inter-session effects, suppose we are given recordings $h = 1, \dots, H(s)$ of a speaker s . For each recording h , let $\mathbf{M}_h(s)$ denote the corresponding speaker- and channel-dependent supervector. We assume that the difference between $\mathbf{M}_h(s)$ and $\mathbf{M}(s)$ can be accounted for by a vector of channel factors $\mathbf{x}_h(s)$ having a standard normal distribution. That is, we assume that there

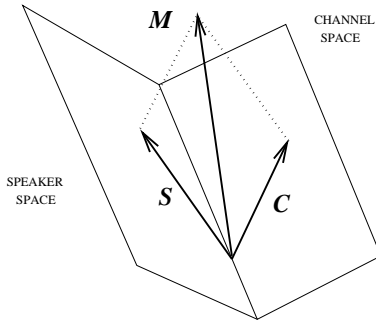


Fig. 1. In the PCA case, a speaker- and channel-dependent supervector M can be written as a sum of two supervectors one of which (indicated here by C) lies in the channel space and the other (S) lies in the speaker space. In the general case, speaker supervectors are assumed to be distributed in the neighbourhood of the speaker space.

is a rectangular matrix u of low rank such that

$$\left. \begin{aligned} M(s) &= m + vy(s) + dz(s) \\ M_h(s) &= M(s) + ux_h(s) \end{aligned} \right\} \quad (2)$$

for each recording $h = 1, \dots, H(s)$.

Thus we are assuming that channel supervectors are contained in a low-dimensional subspace of the supervector space, namely the range of uu^* , which we refer to as the *channel space*. The assumption of low dimensionality is clearly reasonable in the idealized case where all channels are linear and the acoustic features are cepstral coefficients. An empirical justification is provided by the fact that in all of our experiments with the factor analysis model we have found that the eigenvalues of uu^* drop off exponentially (as do the eigenvalues of vv^*).

If R_C is the number of channel factors and R_S the number of speaker factors, the factor analysis model is specified by a quintuple Λ of the form (m, u, v, d, Σ) where m is $CF \times 1$, u is $CF \times R_C$, v is $CF \times R_S$ and d and Σ are $CF \times CF$ diagonal matrices. To explain the role of Σ , fix a mixture component c and let Σ_c be the corresponding block of Σ . For each speaker s and recording h , let $M_{hc}(s)$ denote the subvector of $M_h(s)$ corresponding to the given mixture component. We assume that, for all speakers s and recordings h , observations drawn from mixture component c are distributed with mean $M_{hc}(s)$ and covariance matrix Σ_c .

The role of the hyperparameters m , v and d is to model inter-speaker variability but if we are given enrollment data for a target speaker s we can use speaker-dependent versions of these hyperparameters, namely $m(s)$, $v(s)$ and $d(s)$, to model the posterior distribution of the speaker's supervector $M(s)$ instead. This leads to a speaker-dependent version of the factor analysis model where we assume that,

for a given speaker s and recording h ,

$$\left. \begin{aligned} M(s) &= m(s) + v(s)y(s) + d(s)z(s) \\ M_h(s) &= M(s) + ux_h(s). \end{aligned} \right\} \quad (3)$$

Set $\Lambda(s) = (m(s), u, v(s), d(s), \Sigma)$. (We continue to treat u and Σ as speaker-independent because channel effects should not vary from one speaker to another.)

3. BUILDING A SPEAKER VERIFICATION SYSTEM

In order to build a speaker verification system using the factor analysis model we proceed as follows [1]:

1. **Train the UBM** The role of the universal background model is to extract the usual first and second order statistics from each utterance just as in the Baum-Welch algorithm. These are sufficient statistics for the factor analysis model.
2. **Train a PCA model** That is, estimate speaker-independent hyperparameters Λ from a large database in which each speaker is recorded in multiple sessions.
3. **Adapt this to the target speaker population** That is, given a target speaker population (as in one of the NIST evaluations), adapt the hyperparameters (m, v, d) to fit this population. (Adaptation is necessary because training a factor analysis model on a target speaker population is not possible if there is only one recording per speaker.) For computational reasons d is introduced in this step rather than in step 2.
4. **Enroll the target speakers** That is, estimate the speaker-dependent hyperparameters $\Lambda(s)$ for a given target speaker s by calculating the posterior distribution of the speaker's supervector $M(s)$ using the speaker's enrollment data and the speaker-independent hyperparameters.
5. **Test** That is, for a given test utterance \mathcal{X} and hypothesized speaker s , test the null hypothesis (that the test speaker is somebody other than s) against the alternative hypothesis (that the test speaker is s) using the log likelihood ratio statistic

$$\log \frac{P_{\Lambda(s)}(\mathcal{X})}{P_{\Lambda}(\mathcal{X})} \quad (4)$$

where the numerator is the (factor analysis) likelihood of the test utterance calculated with the speaker-dependent hyperparameter set $\Lambda(s)$ (step 4) and the denominator is the likelihood calculated with the speaker independent hyperparameter set Λ (step 3).

Of these, steps 2 and 5 are by far the most computationally expensive.

3.1. Simplifying the training procedure

Note that \mathbf{m} can be estimated by Baum-Welch training and, in the case of a PCA model, if $\mathbf{u} = \mathbf{0}$ then the columns of \mathbf{v} can be interpreted as the eigenvoices of the training speaker population. Thus \mathbf{v} can be estimated by eigenvoice training or by cluster adaptive training. Note also that whereas the basic assumption in eigenvoice modeling is that speaker-supervectors are normally distributed with mean \mathbf{m} and covariance $\mathbf{v}\mathbf{v}^*$, we are also assuming that, for an arbitrary speaker s , speaker- and channel-dependent supervectors are normally distributed with mean $\hat{\mathbf{M}}(s)$ and covariance $\mathbf{u}\mathbf{u}^*$. Thus we can estimate \mathbf{u} by the same methods as \mathbf{v} .

To be more specific we have to define some statistics. For each recording h of a speaker s and for each mixture component c , let $N_{hc}(s)$ be the total number of observations for the given mixture component. For a supervector $\hat{\mathbf{M}}$, define centralized first and second order statistics by setting

$$F_{hc}(s, M_c) = \sum_t (X_t - M_c)$$

$$S_{hc}(s, M_c) = \text{diag} \left(\sum_t (X_t - M_c)(X_t - M_c)^* \right)$$

where the sum extends over all observations X_t aligned with the given mixture component, M_c is the c th block of $\hat{\mathbf{M}}$, and $\text{diag}()$ sets off-diagonal entries to 0. Let $\mathbf{N}_h(s)$ be the $CF \times CF$ diagonal matrix whose diagonal blocks are $N_{hc}(s)I$ (for $c = 1, \dots, C$) where I is the $F \times F$ identity matrix. Let $\mathbf{F}_h(s, \hat{\mathbf{M}})$ be the $CF \times 1$ vector obtained by concatenating $F_{hc}(s, M_c)$ (for $c = 1, \dots, C$). Similarly, let $\mathbf{S}_h(s, \hat{\mathbf{M}})$ be the $CF \times CF$ diagonal matrix whose diagonal blocks are $S_{hc}(s, M_c)$ (for $c = 1, \dots, C$).

In order to estimate \mathbf{v} using the algorithm in [2] we centralize the first and second order statistics using the UBM supervector \mathbf{m} , and for each training speaker we pool the statistics over all of the recordings for the speaker. Thus the input to the eigenvoice estimation algorithm is

$$\sum_{h=1}^{H(s)} \mathbf{N}_h(s), \sum_{h=1}^{H(s)} \mathbf{F}_h(s, \mathbf{m}), \sum_{h=1}^{H(s)} \mathbf{S}_h(s, \mathbf{m})$$

where s ranges over all of the training speakers and, for a given speaker s , $H(s)$ is the number of recordings of the speaker.

In addition to producing an estimate of \mathbf{v} , the eigenvoice estimation algorithm in [2] also produces, for each training speaker s , a point estimate of $\hat{\mathbf{M}}(s)$ which we denote by $\hat{\mathbf{M}}(s)$. In order to estimate the matrix \mathbf{u} , we eliminate speaker effects by centralizing the first and second order statistics for each training speaker s using $\hat{\mathbf{M}}(s)$ and

present the eigenvoice estimation algorithm with the following input:

$$N_h(s), F_h(s, \hat{\mathbf{M}}(s)), S_h(s, \hat{\mathbf{M}}(s)) \quad (h = 1, \dots, H(s))$$

where s ranges over all of the training speakers.

3.2. Simplifying the verification decision

T-norm score normalization obviates the need to calculate the denominator of (4) but requires evaluating the numerator for all of the t-norm speakers in addition to a hypothesized speaker s . It would be a straightforward matter to evaluate the likelihood $P_{\Lambda(s)}(\mathcal{X})$ if the values of the hidden variables $\mathbf{x}_1(s)$, $\mathbf{y}(s)$ and $\mathbf{z}(s)$ were given because we would then be able to write down the speaker- and channel-dependent supervector $\mathbf{M}_1(s)$ in accordance with (2). (The subscript 1 here indicates that we are working with a single recording i.e. $h = 1$ in (2).) Since the values of the hidden variables are not given we have to evaluate the integral

$$\int P_{\Lambda(s)}(\mathcal{X} | \mathbf{x}_1, \mathbf{y}, \mathbf{z}) N(\mathbf{x}_1, \mathbf{y}, \mathbf{z} | \mathbf{0}, \mathbf{I}) d\mathbf{x}_1 d\mathbf{y} d\mathbf{z} \quad (5)$$

where $N(\cdot | \mathbf{0}, \mathbf{I})$ is the standard Gaussian kernel. By arguments similar to those used in demonstrating Proposition 2 in [2], the value of this integral can be expressed in terms of the inverse of the matrix $\mathbf{L}(s)$ defined by

$$\mathbf{L}(s) = \mathbf{I} + \mathbf{V}^*(s) \mathbf{\Sigma}^{-1} \mathbf{N}_1(s) \mathbf{V}(s)$$

where $\mathbf{V}(s) = \begin{pmatrix} \mathbf{v}(s) & \mathbf{u} & \mathbf{d}(s) \end{pmatrix}$.

By taking advantage of the fact that $\mathbf{d}(s)$ is diagonal, the calculation can be reduced to evaluating the Cholesky decomposition of a matrix of dimension $(R_S + R_C) \times (R_S + R_C)$ constructed from $\mathbf{v}(s)$ and \mathbf{u} . (Recall that R_C is the rank of \mathbf{u} and R_S is the rank of \mathbf{v} and of $\mathbf{v}(s)$.) But if the number of speaker and channel factors is large the cost of these Cholesky decompositions may be prohibitive, particularly if there are many t-norm speakers.

In [1] we dealt with this problem by reducing the rank of $\mathbf{v}(s)$ by suppressing the minor eigenvalues of $\mathbf{v}(s)\mathbf{v}^*(s)$. The fact that Bayes factor scoring does not seem to be very helpful with conventional GMM's suggests that a more extreme expedient might work, namely setting $\mathbf{d}(s) = \mathbf{0}$ and $\mathbf{v}(s) = \mathbf{0}$. Since $\mathbf{N}_1(s)$ depends on \mathcal{X} but not on s (because the sufficient statistics for the factor analysis model are extracted using the UBM), it follows that in this case $\mathbf{L}(s)$ is independent of s so all that is required is a single Cholesky decomposition of an $R_C \times R_C$ matrix in order to evaluate $P_{\Lambda(s)}(\mathcal{X})$ for any speaker s including all of the t-norm speakers.

4. EXPERIMENTS

4.1. Databases and signal processing

We used NIST 1999 evaluation set in its entirety (carbon data as well as electret) to test our algorithms. We used Switchboard II, Phases 1 and 2 for training the factor analysis model (step 2 in Section 3), and we used the enrollment data provided by NIST to train the UBM (step 1), to adapt the PCA model (step 3) and to enroll the target speakers (step 4). Since the 1999 evaluation data was drawn from Switchboard II, Phase 3 this experimental setup ensures that the NIST protocol is respected (although there is a geographical mismatch between the training and test speaker populations [1]).

After excising silences, the female portion of the training set consisted of 230 hours of data (700 speakers, 8400 conversation sides) and the male portion consisted of 180 hours of data (625 speakers, 7400 conversation sides) — about twice as much data as we used for our experiments on the 1999 test set in [1].

Speech was sampled at 8 kHz and 12 liftered mel frequency cepstral coefficients and the log energy were calculated at a frame rate of 10 ms using a 25 ms Hamming window. The acoustic feature vector consisted of these 13 parameters together with their first derivatives. Feature warping (Gaussianization) was applied as in [3].

4.2. Training and testing

For each gender we used a GMM having 2,048 Gaussians as a UBM and we used 300 speaker factors and 100 channel factors.

In testing, we used 50 t-norm speakers per test utterance. In order to evaluate the likelihood ratio statistic (4) we reduced the rank of $v(s)$ from 300 to 100 for each target speaker s . This is the exact decision rule referred to in the third column of Table 1; the simplified decision rule consists in setting $d(s)$ and $v(s)$ to zero as in Section 3.2.

4.3. Results

The results of our experiments on the female portion of the 1999 test set are summarized in Table 1. Line 1 gives the benchmark result obtained by training with the maximum likelihood and minimum divergence estimation algorithms described in [1] ('exact' training) and the exact decision rule.

Comparing Line 1 with Line 2 and Line 3 with Line 4 shows that simplified training has a minimal effect on both the minimum detection cost (DCF) and the equal error rate (EER). Comparing Line 1 with Line 3 and Line 2 with Line 4 shows that the simplified decision rule has a minimal effect on the DCF but there is a non-negligible degradation in

	Training Algorithm	Decision Rule	Feature Warping	DCF	EER
1	Exact	Exact	Yes	0.021	6.2%
2	Simplified	Exact	Yes	0.022	6.3%
3	Exact	Simplified	Yes	0.022	7.3%
4	Simplified	Simplified	Yes	0.023	7.1%
5	Simplified	Exact	No	0.029	8.4%
6	Simplified	Simplified	No	0.041	14.2%

Table 1. Results of speaker verification experiments on the NIST 1999 evaluation set, female speakers. DCF denotes the minimum value of the NIST detection cost function, EER the equal error rate.

the EER.

Results on the male portion of the test set are similar. Replicating the experiment in Line 2 gave a DCF of 0.020 an EER of 5.9% and replicating the experiment in Line 4 gave a DCF of 0.021 and an EER of 6.4%.

The results in Lines 5 and 6 were obtained without feature warping. Comparing them with the results in Lines 2 and 4 show that that feature warping is very effective. It seems likely that the reason for this is that Gaussianization reinforces the modeling assumptions in Section 2.

4.4. Conclusions

Our results on the 1999 test set are substantially better than those reported in [1] thanks to doubling the amount of training data for the factor analysis model and to using Gaussianized acoustic features. We found that the simplified training procedure works as well as the exact training procedure and achieves a 2–3 fold speed up but the simplified decision rule is not as effective as the exact decision rule at least as measured by EER's (although it does have the virtue of greatly reducing the turn-around time for experiments).

5. REFERENCES

- [1] P. Kenny and P. Dumouchel, "Experiments in speaker verification using factor analysis likelihood ratios," in *Proc. Odyssey 2004*, Toledo, Spain, June 2004.
- [2] P. Kenny, G. Boulianne, and P. Dumouchel, "Eigen-voice modeling with sparse training data," *IEEE Trans. Speech Audio Processing*, May 2005.
- [3] J. Pelecanos and S. Sridharan, "Feature warping for robust speaker verification," in *Proc. Speaker Odyssey*, Crete, Greece, June 2001.