A COMPARATIVE STUDY OF LINEAR FEATURE TRANSFORMATION TECHNIQUES FOR AUTOMATIC SPEECH RECOGNITION

Thomas Eisele, Reinhold Haeb-Umbach, Detlev Langmann

Philips GmbH Forschungslaboratorien
Weißhausstr. 2
D-52066 Aachen, Germany
Email: {eisele, haeb, langmann}@ pfa.research.philips.com

ABSTRACT

Although widely used, there are still open questions concerning which properties of Linear Discriminant Analysis (LDA) do account for its success in many speech recognition systems. In order to gain more insight into the nature of the transformation we compare LDA with mel-cepstral feature vectors with respect to the following criteria: decorrelation and ordering property, invariance under linear transforms, automatic learning of dynamical features, and data dependence of the transformation.

1. INTRODUCTION

The structure of a typical continuous-speech recognizer consists of a front-end feature analysis block, followed by a statistical pattern classifier. The interface between these two, the feature vector, should ideally contain all the information of the speech signal relevant to subsequent classification, be insensitive to irrelevant variations (e.g. due to changes in the acoustic environment), and at the same time have a low dimensionality in order to minimize the computational demands of the classifier. Several types of feature vectors have been proposed. Here we study representations derived from a FFT analysis of the speech signal with a subsequent perceptually motivated bandpass filtering of the power density spectrum: log-spectrum coefficients, mel-frequency cepstral coefficients (MFCC) [1], and features obtained from a linear discriminant analysis (LDA) applied to either a log-spectral or a cepstral feature vector. Note that both MFCC and LDA features result from a linear transform of the log-spectrum feature vector, with very different properties, though. LDA has been proven to improve discrimination in the speech feature space. This has led to improvements in recognition performance both for small and large vocabulary speech recognition [2, 3]. Although widely used, there are, however, still open questions as to which properties of LDA do account for its success in many speech recognition systems. In order to gain more insight in the nature of the transformation we compare LDA with MFCC with respect to several criteria:

- Decorrelation and ordering of the resulting vector components
- Invariance under linear transforms
- Automatic learning of dynamical features
- Data dependence and robustness

2. PROPERTIES OF CEPSTRAL TRANSFORM AND LINEAR DISCRIMINANT ANALYSIS (LDA)

In this Section the various properties of the two transformation techniques are investigated. Experimental evidence will be given in *Section 3*.

2.1. Decorrelation

The initial log-spectral feature vector is computed by smoothing the spectrum with overlapping triangular kernels [1]. This leads to a relatively high correlation between the feature vector components as can be observed — for the later described SIETILL database — in the left image of *Figure 1*. Note that in order to concentrate on the characteristics of the different feature sets, we only show static features in *Figure 1* and *Figure 2*.

The log-spectral feature vector is the input for the two transformations. Assuming the correlation matrix of the input data exhibits Toeplitz structure and neglecting boundary effects , it can be shown that the cosine transform leads to decorrelated features. Indeed the measured correlation matrix of the cepstral feature vector is much more "diagonal-like" than that of the original log-spectral vector (*cf. Figure 1 left and middle*). By definition, LDA transformation delivers uncorrelated features. The complete decorrelation accomplished by LDA can be observed in *Figure 1 right*.

The importance of decorrelated feature sets for our speech recognizer is studied in Section 3.3.

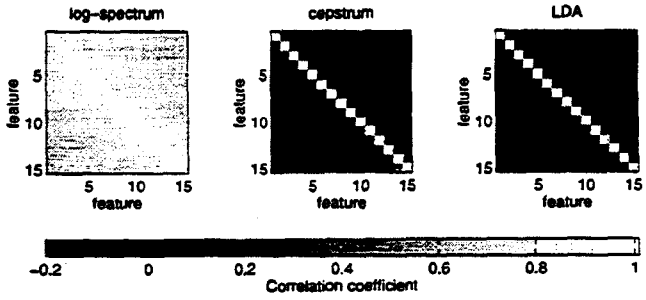

Figure 1: Correlation matrices of log-spectral, cepstral, and LDA feature set (measured on SIETILL database).

2.2. Ordering and Compactness of Feature Set

In order to come up with a feature set as small as possible, feature transformations should be able to concentrate the relevant information for classification in only few features. In addition they should order the transformed features according to their class separability, resulting in an easy reduction of the feature set by simply leaving out the last ones in the list.

For the measurement of class separability we use a measure originally introduced as a criterion for the computation of the LDA, namely the *trace criterion* [4, *p. 446*]:

$$J_{tr} = tr\left(S_w^{-1} S_b\right)$$

where S_w denotes the within-class and S_b the between-class scatter matrix. The matrix product $S_w^{-1} S_b$ stands for the ratio of between-class variance to variance inside the classes. The higher this ratio, the better is the class separation. To convert the matrix product to a scalar, which is necessary for a well-defined optimization problem, we can use the trace, which equals the sum of the eigenvalues of $S_w^{-1} S_b$. Thus the elements on the main diagonal are a measure for the contribution of each single feature to class separation. *Figure 2* illustrates this measure for the different feature sets — again for the later described SIETILL database.

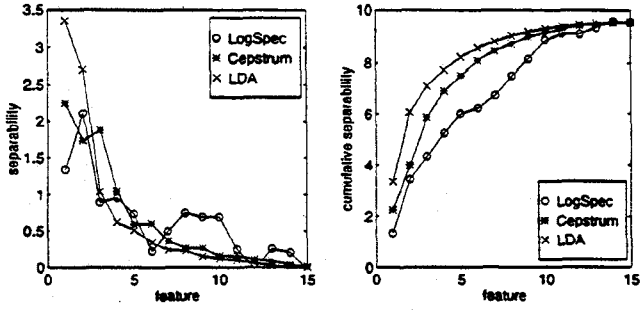

Figure 2: Class separation of individual features of log-spectral, cepstral, and LDA feature sets (measured on SIETILL database).

The left picture of *Figure 2* demonstrates the ordering of the feature set achieved by LDA and — to a lesser extent — by cepstrum transformation. The separability measures are cumulated in the right picture, thus illustrating the fact that LDA is able to concentrate more separability in less features than cepstrum and — much worse — log-spectrum. We can conclude that LDA should be able to allow recognition with fewer features than cepstrum. Both order the features according to separability, thus making feature reduction easy.

2.3. Invariance Under Linear Transforms

The LDA transformation matrix is invariant under linear transforms [4]. Indeed, the same features result no matter whether the input is a cepstral vector obtained by a linear transformation of the log-spectral vector or the original log-spectral vector. We verified that this property also holds approximately in the presence of a feature vector augmented by time differences: the resulting word error rate was very similar for a log-spectral and a cepstral input vector of the LDA, both vectors augmented by time differences in their respective domains before applying the LDA transform.

2.4. Automatic Learning of Dynamical Features

Time derivatives are generally included in the feature set in order to measure spectrum variations more directly. Since LDA is able to incorporate several frames and transform them optimally into one feature vector, it has often been argued that LDA should be able to compute derivatives implicitly or even invent better measures for variations in the spectrum [5]. We compared the use of a vector of only static features to the explicit inclusion of time derivatives. The latter turned out to perform only slightly better.

2.5. Data Dependence and Robustness to Channel Mismatch

In contrast to cepstrum, the LDA transformation matrix is data-dependent. Siohan [6] has reported experiments on the sensitivity of LDA to SNR mismatches in training and test. We carried out cross-test experiments where the transformation matrix had been determined in acoustic environments different from the testing data. The performance of LDA-transformed features dropped below that of cepstrum features, even for multiple input frames.

3. EXPERIMENTAL RESULTS

3.1. Structure of the Recognizer Used

For the tests described in the following we filtered and sampled the speech signal at 8 kHz. After a subsequent preemphasis a 256-point FFT is computed every 16 ms from a Hamming-windowed 32 ms portion of the speech signal. 15 power-spectral density coefficients are determined from a convolution of the power density spectrum with triangular kernels with a nonlinear frequency spacing. The logarithm is computed and the resulting 15 coefficients are the basis for the different feature vectors used:

log-spectral: The average of the 15 coefficients is subtracted from each channel and included as an additional component. From these 16 features first-order time derivatives are computed (by subtracting the features 2 frames in the past) and appended to the feature vector, resulting in 32 features.

cepstrum: The 15-component log-spectral feature vector is transformed using a discrete cosine transform and augmented by time derivatives as described above. Since the first cepstral coefficient c_0 already contains the average of the 15 log-spectral channels, we come up with only 30 features.

LDA: The 32-component log-spectral vector is transformed using a LDA matrix computed beforehand using an existing

segmentation from a log-spectral training and using the HMM states as class definition.

The connected-word recognition algorithm is based on whole-word hidden Markov models, the emission probabilities of which are modeled by continuous Laplacian densities (approx. 16 per state) with a single 'standard deviation' vector pooled over all states of all models. The transition probabilities $a(s|s')$ from state s' to state s are not trained but instead are given fixed a priori values that are non-zero only for loop, skip and forward transitions.

We employ the Viterbi approximation in both training and recognition, i.e. the probability of a word is replaced by the probability of its most likely state sequence. Details of the acoustic modeling can be found in [7].

3.2. Speech Corpus

We conducted our experiments on the SIETILL digits corpus. This corpus contains German digit strings recorded over telephone lines. Each of the 379 male and 339 female speakers uttered 31 to 34 three-digit, two five-digit and up to two variable-length strings. 191 male and 171 female speakers were arbitrarily chosen for the training corpus, the rest was used as recognition corpus. The SIETILL digits are characterized by a large variety of line and speaker characteristics. For the 11-word vocabulary we use 23 models, i.e. a separate model for each word and gender, and a single background noise model.

3.3. Decorrelation

In our acoustic modeling we make an approximation by assuming decorrelated feature vector components. Hence we suppose that a feature set exhibiting low correlation should lead to a better performance.

In order to verify this conjecture, we tested the three feature vector types described above using their full length. Since no input information has been discarded by reducing the number of features, the differences in the error rate should solely stem from the different amount of decorrelation of the features (*Table 1*).

Corpus	log-spectrum	cepstrum	LDA
SIETILL	3.65%	3.64%	3.14%

Table 1: Word error rates using full-length vectors.

The results show that decorrelation achieved by cepstral transformation is not sufficient to exhibit a significant decrease of the error rate, whereas LDA leads to an improvement of 14% rel. The disappointing performance by cepstrum may be due to the inclusion of time derivatives after cepstrum transform, but before LDA, so that the cepstrum transform is not able to remove correlations between features and their derivatives.

3.4. Ordering and Compactness of Feature Set

In order to test ordering property and compactness of the different feature sets, we ran our recognizer with different numbers of features. The selected features were chosen according to their separability measure as described in 2.2, but now including time derivatives. For cepstrum and LDA we could simply remove the last features, but the log-spectral features required choices out of order (time derivatives of channels 0...2 and 8...12 for 24 features). The results are summarized in *Table 2*:

# features	log-spectral	cepstrum	LDA
32	*16 + 16 Δ* 3.65%	—	3.14%
30	—	*15 + 15 Δ* 3.64%	—
24	*16 + 8 Δ* 4.22%	*12 + 12 Δ* 3.25%	3.07%
20	—	*11 + 9 Δ* 3.30%	3.14%
18	—	*10 + 8 Δ* 3.51%	—
16	—	*9 + 7 Δ* 3.66%	3.20%
12	—	*7 + 5 Δ* 3.99%	3.62%

Table 2: Word error rates using different numbers of features.

Cepstral features perform better than or as good as 32 log-spectral features down to a number of features of 16. LDA reduces this number to 12. In addition, if we look at the performance with e. g. 24 features, cepstrum is 23% rel. better than log-spectrum features, while LDA is even 5.5% rel. better than cepstrum.

3.5. Invariance Under Linear Transforms

In chapter 2.3. we stated that the features produced by LDA should be nearly independent of the feature space at the input of LDA, even if time derivatives are included. In order to verify this, we used two different input vectors to LDA: The 32-component log-spectral vector and the 30-component cepstral vector, both as described above. The output consisted of the 24 best LDA features. The results are given in *Table 3*.

Corpus	log-spectral input	cepstral input
SIETILL	3.07%	3.18%

Table 3: Word error rates for different inputs to LDA

There is only a small increase by 3.6% rel. if cepstral input is used. This may be due to the different number of input features to LDA.

3.6. Automatic Learning of Dynamical Features

We adjoined n frames ($n = 3, 5, 7$) without time derivatives to a single vector at the input to the LDA and let it transform to one feature vector of length 24. For comparison we adjoined $n-2$ frames *with* time derivatives to one vector at the input of the LDA. The reduction by 2 comes from the fact that our derivatives are computed by subtracting the next-to-last frame from the actual one, so they incorporate a larger time period. For a meaningful comparison we have to keep the number of input frames the LDA is seeing the same for both cases. The results are shown in *Table 4*.

n	$n-2$ frames with derivatives	n frames without derivatives	rel. change
3	3.07%	2.88%	−6.2%
5	2.50%	2.60%	+4.0%
7	2.58%	2.71%	+5.0%

Table 4: Word error rates using multiple frames as input to LDA.

For few input frames LDA is able to compute time derivatives implicitly, and even better than normally done by the explicit method. But if we use five or more input frames, the inclusion of time derivatives in the input performs better.

3.7. Data Dependence and Robustness to Channel Mismatch

In order to test the data dependency of LDA, we computed a LDA transformation matrix out of a different corpus. It has the same vocabulary as SIETILL, but has been recorded in the car through a hand set and a dashboard-mounted microphone. 20 different speakers each uttered approx. 200 sentences containing 1-, 3-, and 7-digit strings. In effect, we computed the LDA for the same acoustic modeling, but with data recorded through a completely different channel and with different SNR. We computed the LDA for one frame and three frames with derivatives using the car database. With the resulting LDA transformation matrix we carried out the same training and recognition runs on the SIETILL data as above, using the 24 best features. The results — including cepstrum and LDA trained on same environment for comparison — are given in *Table 5*. The results confirm our hypothesis that LDA transformation heavily depends on the data used for computing its transformation matrix. Even if we include 3 frames with derivatives in the input, it performs worse than a simple cepstral feature vector.

Cepstrum	LDA same environment		LDA cross environment	
	1 frame	3 frames	1 frame	3 frames
3.25%	3.07%	2.50%	4.02%	3.58%

Table 5: Word error rates using a different LDA transformation matrix.

4. CONCLUSIONS

Properties of two widely used linear transforms of log-spectral feature vectors, cepstrum and linear discriminant analysis (LDA), have been studied in view of approximations due to an implementation in a real-life speech recognizer. It was shown that LDA delivers more compact and informative features yielding better recognition accuracy with a shorter feature vector. The big drawback of LDA is its data-dependence which makes it more sensitive to changes in the acoustic environment.

5. REFERENCES

1. Davis, S. B., and Mermelstein, P., "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences", *IEEE Trans. ASSP* 28: 357–366, *1980*.

2. Hunt, M., and Lefèbvre, C., "A Comparison of Several Acoustic Representations for Speech Recognition with Degraded and Undegraded Speech", *in Proc. ICASSP'89, Glasgow, UK, May 1989, pp 262-265.*

3. Haeb-Umbach, R., and Ney, H., "Linear Discriminant Analysis for Improved Large Vocabulary Continuous Speech Recognition", *in Proc. ICASSP '92, San Francisco, CA, March 1992, pp I.13-I.16.*

4. Fukunaga, K., "Introduction to statistical pattern recognition", *Academic Press, Boston 1990*.

5. Beulen, K., Welling, L., Ney, H., "Experiments with linear feature extraction in speech recognition", *in Proc. EUROSPEECH '95, Madrid, Spain, Sept. 1995, pp 1415-1418.*

6. Siohan, O., "On the Robustness of Linear Discriminant Analysis as a Preprocessing Step for Noisy Speech Recognition", *in Proc. ICASSP '95, Detroit, May 1995, pp 125-128.*

7. Aubert, X., Haeb-Umbach, R., Ney, H., "Continuous Mixture Densities and Linear Discriminant Analysis for Improved Context-Dependent Acoustic Models", *in Proc. ICASSP '93, Minneapolis, MN, 1993, Vol. II, pp 648-651.*