EXPERIMENTS WITH LINEAR FEATURE EXTRACTION IN SPEECH RECOGNITION

K. Beulen, L. Welling, H. Ney

Lehrstuhl für Informatik VI, RWTH Aachen – University of Technology,
D-52056 Aachen, Germany

ABSTRACT

In this paper we investigate Linear Discriminant Analysis (LDA) for the TI connected digit recognition task (TI task) and the Wall Street Journal large vocabulary recognition task (WSJ task). In addition to previous variants of LDA implementations, we avoided the explicit incorporation of derivatives in the acoustic vector. Instead a sliding window without derivatives was used. This large-sized vector was then taken to extract the features by an LDA transformation. Tests for this feature generation were performed both for Laplacian and Gaussian densities.

1. INTRODUCTION

It is a well known fact that the performance of a pattern recognition system depends heavily on the type of observations that are used in the system. Several methods are employed in practice which often consist of two stages: First the acoustic signal is transformed from time domain into frequency domain using a Fourier transformation or the like. Second the spectral components of the resulting acoustic vector are then transformed by a linear mapping.

In our baseline TI and WSJ systems, we use an augmented vector (spectral (WSJ system) or cepstral (TI system) components plus first-order and second-order derivatives) which is then transformed by Linear Discriminant Analysis (LDA). Because of this ad-hoc method which is used to add contextual information to the feature vector, one can argue that the LDA transformation, which is an optimal method for the derivation of features (corresponding to the separation criterion), should be able to supply more relevant information from the acoustic context to the feature vector. In this work we try to avoid the use of explicit derivatives in the feature vector. Instead we use a big 'spliced' vector which covers the adjacent vectors from which the derivatives were calculated, and an LDA transformation to automatically extract the additional information due to the derivatives.

To model the emission probabilities of the hidden Markov models (HMMs) of the recognizer, our baseline WSJ system has so far used Laplacian densities. The reason is that during the development of the WSJ system Laplacian and Gaussian densities were tested and Laplacian densities were found to perform better, given the boundary conditions of the system. In this paper, we made several tests to investigate the effect of combining Gaussian and Laplacian densities with LDA and derivatives for the TI task and the WSJ task. We found that, in combination with derivatives and LDA, Gaussian densities are superior to Laplacian densities.

Moreover we investigated two problems due to the LDA, the definiton of the LDA classes and the use of the silence observations to calculate the LDA matrix. We found that it is sufficient for good performance to take the states of the HMMs as classes for the LDA. The use of the silence observations for the calculation of the LDA matrix does not affect the recognition results.

This paper is organized in the following way. Section 2 describes the two speech recognition systems which were used in this work. Section 3 gives a brief introduction to the methods investigated, namely the explicit incorporation of derivatives and linear discriminant analysis. Section 4 describes the experimental results of the methods.

2. SYSTEM DESCRIPTION

Two speech recognition systems were used. The first system was developed for the TI task. In this system, the signal analysis is done the following way. The sampled speech signal is preemphasized. Then a Hamming window of 15 ms is applied to the signal every 10 ms. The short term spectrum is computed by a 8192-point fast Fourier transform. From this frequency range, the range of 0 to 5 kHz is used for further processing:

- A filter bank in which each filter has a triangle bandpass frequency response with bandwidth and spacing determined by a constant mel frequency interval is applied to the mel spectrum.
- For each filter the output is calculated as the logarithm of the sum of the weighted spectral magnitudes.
- The filter bank outputs are decorrelated by a discrete cosine transform [1]. 16 cepstrum coefficients c_m are computed from 20 filter bank outputs.
- For every utterance, a cepstral mean normalisation is carried out. Moreover the zeroth coefficient is shifted so that the maximum value within every utterance is zero (energy normalisation).

The acoustic modelling is based on the following properties:

- The word models are HMMs with continuous observation densities.
- The emission probabilities are modelled by Laplacian or Gaussian single densities with one variance vector per state.
- The transition probabilities are modeled as time distortion penalties for the skip and the loop transition. Their values are determined beforehand.

The second system is described in [6] and is applied to the WSJ large vocabulary recognition task. In the acoustic analysis the following steps are performed. First a Hamming window is applied every 10 ms to a 25-ms segment. Then a 512-point FFT is performed on the 25-ms segment. 30 mel-frequency spectral intensities were computed and normalized with respect to their mean value. The resulting 30 intensities together with their energy form the acoustic vector. This vector is then normalized according to the long term spectrum of the sentence to counteract the influence of different recording conditions. The resulting 31-dimensional vector is then augmented by first-order and second-order derivatives and then (optionally) transformed by a LDA matrix resulting in a 35 component vector.

The acoustic modelling is almost the same as in the TI system, the only differences are the following:

- Instead of single densities, mixture densities are used to model the emission probabilities.
- The variance vector is pooled over all densities of all states.
- To capture the acoustic context dependencies of the phonemes, a set of 780 context dependent phoneme models is used.

Both systems estimate the parameters of the acoustic models using the maximum likelihood criterion and Viterbi training [5].

3. LINEAR TRANSFORMATION FOR FEATURE EXTRACTION

3.1. Derivatives

Adding derivatives is a simple but efficient method for incorporating contextual information into the feature vector. In both systems, first-order and second-order time derivatives are used to form a new acoustic vector of higher dimension. For a window size of $2\Delta t + 1$, the differences between the components of the feature vectors $x(t)$, $x(t-\Delta t)$ and $x(t+\Delta t)$ are computed to form the new vector $Y(t)$:

$$Y(t) = \begin{bmatrix} x(t) \\ \Delta x(t) \\ \Delta^2 x(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ x(t) - x(t-\Delta t) \\ x(t+\Delta t) - 2x(t) + x(t-\Delta t) \end{bmatrix}$$

This vector is then transformed by a method such as LDA or is directly used for training and recognition.

For the TI system, the augmented acoustic vector consists of the 16-dimensional original vector plus a 16-dimensional first-order derivatives vector and a 16-dimensional second-order derivatives vector. For the WSJ system, pairs of adjacent spectral energies are averaged and then used to calculate the time derivatives. Thus the augmented acoustic vector contains 15 first-order and 15 second-order derivatives and the first-order and the second-order derivative of the energy component in addition to the original 31 vector components. Both systems use a value of $\Delta t = 3$ frames for the calculation of the derivatives.

3.2. LDA

The LDA is a method introduced by Fisher [2] to reduce the number of dimensions for a given feature vector while keeping the classes as separate as possible. The basic idea of the LDA is to find a set of linear transformation functions for the initial feature vector and a ranking which indicates the separation capability of these functions. Then the m best functions are used to transform the initial feature vector yielding m features for classification. To estimate the class separability, the within-class scatter matrix W and the total scatter matrix T of the training set is computed and then used to calculate the criterion J:

$$J = det(W^{-1}T)$$

This criterion is then maximized using techniques of linear algebra. For the above criterion the solution is to calculate the eigenvectors and eigenvalues for the matrix product $W^{-1}T$ and then take the m eigenvectors with the largest eigenvalues to form the transformation matrix [4].

This transformation can be interpreted as a simultaneous diagonalization of the within-class scatter matrix and the total scatter matrix. First the within class scatter matrix is whitened which results in an approximative whitening transformation for each class. Then the total scatter matrix is diagonalized which means a proper rotation of the feature space.

The calculation of the within class scatter matrix involves the problem how to define these classes. Several possibilities are phonemes, states or densities [4]. For the TI system, the states of the word models were used as classes. For the WSJ system, we tested two definitions:

- Each context dependent phoneme state is a separate class; as a result there were 2338 classes for the LDA.
- The states of a set of 4644 triphones are tied together using a bottom-up clustering algorithm [3] to obtain a set of generalized triphones. Along with the 130 classes defined by the monophone states plus one silence state, we thus obtain 252 and 4494 classes for the LDA. The second definition takes the fact into account that during the recognition we only want to decide which phoneme has been uttered rather than the exact triphone.

The calculation of the LDA matrix is done for both systems in a similar way:

- The feature vectors of the training data are time aligned with the states of the HMMs.

- The state labels are used to determine the classes of the feature vectors.
- Then a big 'spliced' vector

$$X(t) = \begin{bmatrix} x(t-\Delta t) \\ \vdots \\ x(t-1) \\ x(t) \\ x(t+1) \\ \vdots \\ x(t+\Delta t) \end{bmatrix}$$

  is formed from $2\Delta t + 1$ adjacent vectors. The class of the spliced vector is determined by the class of the central vector.
- In a first pass the mean vectors for the classes and the total mean vector are calculated.
- In a second pass the within-class scatter matrix W and the total scatter matrix T are calculated.
- Finally, the LDA matrix is calculated as the eigenvectors of $W^{-1}T$.

During training and recognition, this LDA matrix is used to calculate the final feature vector. For both the TI task and the WSJ task, a window size of $2\Delta t + 1 = 11$ frames was used when derivatives were not explicitly incorporated. Otherwise only 3 adjacent vectors were used (only for the WSJ system). The final feature vector contains 48 components for the TI system, 35 for the WSJ system.

4. EXPERIMENTAL RESULTS

All tests described in this section were performed on the TI connected digit recognition test set (28583 spoken words) for the TI system and on the WSJ November 92 development test set (6779 spoken words) for the WSJ system.

The TI system consists of 11 word models. Per gender we have 357 states plus one for silence. Each state consists of one mixture with a single Gaussian or Laplacian density. The WSJ system is based on an inventory of 780 context dependent models plus one model for silence. Each phoneme model is divided into 3 segments with 2 states per segment and one mixture per segment. The silence model consists of only one state with one mixture. The number of densities per mixture is approximately 50 so that the whole acoustic models contain about 110,000 Gaussian or Laplacian densities.

4.1. Derivatives and LDA

Our first test addresses the question how accurate the contextual information is captured by derivatives or LDA. For both systems the following parameters were chosen:

- a time delay Δt of 3 frames for the derivatives,
- a window size of 11 frames for the LDA with no derivatives,
- a time delay Δt of 3 frames and three adjacent acoustic vectors for combined derivatives and LDA.

Table 1: Word error rates [%] on WSJ0, Nov.'92 (Dev-5k) and on TI digits for derivatives and LDA (Laplacian densities).

Features		TI		WSJ	
Deriv.	LDA	DEL-INS	WER	DEL-INS	WER
Yes	No	49-22	0.63	–	12.4
No	Yes	67-54	0.82	115-81	11.2
Yes	Yes	72-37	0.72	114-83	10.3

Table 1 shows the word error rate (WER) for 3 different combinations of derivatives and LDA (Laplacian densities). For the WSJ system, the LDA gave an improvement of about 10% over the vector incorporating the derivatives. For the TI system, the vector incorporating the derivatives reduced the error rate by 1/4 compared to the LDA. The best results (for Laplacian densities) are achieved by a combination of LDA and derivatives.

4.2. Effect of Density Models

In a second test, two different density models were tested for derivatives and LDA. Laplacian and Gaussian densities were compared with each other, the results are shown in Table 2. It is obvious that for LDA-transformed features the Gaussian densities perform slightly better than the Laplacian densities. For pure derivatives, the Laplacian densities gave slightly better results than the Gaussian densities for the TI task.

4.3. LDA Class Definition

A third experiment was carried out to find out a suitable definition for the LDA classes as discussed in Section 3.2. We calculated three different LDA matrices, one for the set of 2338 phoneme states (triphones and monophones) which were also used for training and recognition, and two LDA matrices for a set of 382 and 4624 phoneme states (generalized triphones and monophones) which were derived from a set of 4644 triphones using a

Table 2: Word error rates [%] on WSJ0, Nov.'92 (Dev-5k) and on the TI task for different density models and derivatives and LDA.

Features	Model	TI		WSJ	
		DEL-INS	WER	DEL-INS	WER
Deriv., No LDA	Lapl.	132-70	0.80	–	12.4
	Gauss	156-93	1.01	–	–
No Deriv., LDA	Lapl.	67-54	0.82	115-81	11.2
	Gauss	70-38	0.71	124-68	10.0
Deriv., LDA	Lapl.	72-37	0.72	114-83	10.3
	Gauss	–	–	123-60	9.9

Table 3: Word error rates [%] on WSJ0, Nov.'92 (Dev-5k) for different sets of phoneme models.

Model Set	Nr. of LDA Classes	DEL-INS	WER
Triphones	2338	114-83	10.3
Gen. Triphones	382	117-87	10.9
Gen. Triphones	4624	101-77	10.3

bottom-up clustering algorithm [3]. The LDA matrix was calculated with these two different class definitions and then training and recognition was performed using the conventional set of 780 context dependent models (Table 3). The results indicate that a LDA matrix which is consistent with the models used in the recognizer performs best.

Finally we tested the influence of the silence observations on the LDA matrix. Because silence covers about 20% of the training material of WSJ0, it is not clear whether including the silence observations in the calculation of the LDA matrix affects the discrimination of the non-silence classes (Table 4). We found that excluding the silence observations has no effect on the word error rate (Table 4).

5. CONCLUSIONS

We investigated two linear transformation methods, namely derivatives and LDA, on the TI task and the WSJ task. Each of them was able to improve the performance of the recognizer. The best results were achieved by a combination of the methods. The experimental tests indicated the following results:

- For Gaussian densities, we are able to get the same results with LDA as with the combination of LDA and derivatives.
- For Laplacian densities, the combination of LDA and derivatives performs slightly better than LDA without derivatives.
- For LDA, Gaussian mixture densities gave slightly better results than Laplacian mixture densities.
- The best class definition for LDA seems to be the states of the recognizer.
- The application of the silence observations for calculating the LDA matrix does not affect the recognition performance.

The unsatisfactory result is that the LDA is not able to produce more efficient features from the context than the features which are supplied by the augmented vector.

Table 4: Word error rates [%] on WSJ0, Nov.'92 (Dev-5k) for LDA with and without a silence class.

LDA training	DEL-INS	WER
With silence	114-83	10.3
Without Silence	112-78	10.3

There are some other questions about LDA which were not addressed in this paper such as the optimal number of components for the transformed vector, suitable criteria of class separability and a combination of several LDA matrices with various class definitions.

References

1. S.B. Davis, P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuous spoken sentences," *IEEE Trans. on Acoustic, Speech and Signal Processing*, Vol. ASSP-28, No. 4, August 1980.
2. R.O. Duda, P.E. Hart, "Pattern Classification and Scene Analysis," J. Wiley, New York, NY, 1973.
3. C. Dugast, R. Kneser, X. Aubert, S. Ortmanns, K. Beulen, H. Ney, "Continuous Speech Recognition Tests and Results for the NAB'94 Corpus," *Proc. ARPA Spoken Language Technology Workshop*, Austin, TX, pp. 156-161, January 1995.
4. R. Haeb-Umbach, H. Ney, "Linear Discriminant Analysis for improved Large Vocabulary Continuous Speech Recognition," *Proc. Int. Conf. on Acoustics, Signal and Speech Processing*, San Francisco, CA, March 1992.
5. H. Ney, "Acoustic Modelling of Phoneme Units for Continuous Speech Recognition", Proc. Fifth Europ. Signal Processing Conf., pp 65-72, Barcelona, September 1990.
6. V. Steinbiss, H. Ney, R. Haeb-Umbach, B.-H. Tran, U. Essen, R. Kneser, M. Oerder, H.G. Meier, X. Aubert, C. Dugast, D. Geller, "The Philips Research System for Large-Vocabulary Continuous-Speech Recognition," *Proc. Europ. Conf. on Speech Communication and Technology*, pp. 2125-2128, Berlin, September 1993.
7. L. Welling, H. Ney, A. Eiden, C. Forbrig, "Connected Digit Recognition Using Statistical Template Matching," in *Proc. Europ. Conf. on Speech Communication and Technology*, Madrid, September 1995.