SPEAKER VERIFICATION WITH ENSEMBLE CLASSIFIERS BASED ON LINEAR SPEECH TRANSFORMS

Jesper Ø. Olsen

Center for PersonKommunikation, Aalborg University,
Fredrik Bajers Vej 7A-6, DK-9220 Aalborg Øst, Denmark

email: jo@cpk.auc.dk,   http://www.kom.auc.dk/∼jo

ABSTRACT

For most classifier architectures realistic training schemes only allow classifiers corresponding to local optima of the training criteria to be constructed. One way of dealing with this problem is to work with classifier *ensembles*: multiple classifiers are trained for the same classification problem and combined into one "super" classifier. The problem addressed in this paper is text prompted speaker verification by means of phoneme dependent Radial Basis Function networks trained by gradient descent error minimisation. In this context ensemble techniques are introduced by combining different classifiers that classify feature vectors, which have been pre-processed using different linear transforms. Four different types of linear transforms are studied: the Fisher transform, the LDA transform, the PCA transform and the cosine transform. The verification system is evaluated on the Gandalf database, where the equal error rate is reduced from 3.6% to 3.2% when ensemble techniques are introduced.

1. INTRODUCTION

Speaker verification [1] is a binary classification problem: based on some evidence in the form of a speech signal, we need to accept or reject a specific identity claim. By definition, the optimal classifier with which to solve a classification problem is the Bayes classifier:

$$\text{Decide} \begin{cases} \text{accept} & \text{if } g_{\text{Bayes}}(\vec{\phi}) \geq \text{threshold} \\ \text{reject} & \text{otherwise} \end{cases} \tag{1}$$

where $g_{\text{Bayes}}(\vec{\phi})$ is the Bayes discriminant function:

$$g_{\text{Bayes}}(\vec{\phi}) = P(I|\vec{\phi}) - P(\neg I|\vec{\phi}) \tag{2}$$

and $\vec{\phi}$ is a feature vector representing a speech sample (e.g. a phoneme) and where $P(I|\vec{\phi})$ and $P(\neg I|\vec{\phi})$ represent respectively the probability of the target speaker (I) and the impostor speaker class ($\neg I$).

The problem when constructing a speaker verification system is basically that of obtaining a good approximation of equation 2. For most classifier architectures it is the case that the training algorithms used for determining the parameters of the classifiers are unable to choose parameters that guarantee the globally best possible approximation given the limitations of the architecture. This, for instance, is the case for neural networks trained by gradient descent error minimisation; the objective function has many local minima, and the training algorithm will only succeed in finding one of these. Interestingly, classifiers, $g_1(), ..., g_N()$, corresponding to different local minima are to some extent independent [5, 6]: each classifier approximates the Bayes function, but has an added bias:

$$g_i(\vec{\phi}) = g_{\text{Bayes}}(\vec{\phi}) + b_i(\vec{\phi}) \tag{3}$$

If the classifiers make errors independently the biases can be "removed" by averaging the outputs from the N classifiers:

$$g_{\text{ens}}(\vec{\phi}) = \frac{1}{N}\sum_{i=1}^{N} g_i(\vec{\phi}) = g_{\text{Bayes}}(\vec{\phi}) + \frac{1}{N}\sum_{i=1}^{N} b_i(\vec{\phi}) \tag{4}$$

Even if the individual classifiers are poor, equation 4 can be aribitrarily close to the Bayes function provided that N is large. This, however, assumes that the individual classifiers make errors independently, and this assumption is in practise difficult to justify, particularly when the individual classifiers used on their own are "good" approximations of the Bayes classifier: the classifiers will here tend to misclassify the same observations ($\vec{\phi}$). Hence, succesful application of ensemble techniques in practise depends on the ability to generate classifiers that are likely to have independent biases.

2. CLASSIFIER ARCHITECTURE

The speaker verification system used in this study was designed for *text prompted* speaker verification and relies on a two stage modelling approach [2, 3]. In the first stage Speaker Independent (SI) Hidden Markov Models (HMMs) are used for identifying the phoneme segments in the speech signal. This is done by forced alignment of the prompting text. In the second stage *phoneme vectors*, $\vec{\phi}$, are extracted: each phoneme segment is represented by a fixed dimensional vector. This is done by time sampling the individual phoneme segments; a fixed number of frame vectors (consisting of filter bank coefficients) are extracted and concatenated to form one long vector, which represents the entire phoneme segment. The phoneme vector is then subjected to a linear phoneme (Φ) dependent transformation:

$$\vec{\phi}' = \mathbf{A}_{\Phi}^{T}\vec{\phi} \tag{5}$$

before being input to a Radial Basis Function (RBF) network, which computes the function:

$$g_{\Phi}(\vec{\phi}') = \tanh\left\{ S \sum_i w_i \exp\left(C_i \frac{(\vec{\phi}' - \vec{\mu}_i)^2}{\vec{\sigma}_i^2} \right) \right\} \tag{6}$$

where $\vec{\mu}_i$ and $\vec{\sigma}_i^2$ make up a codebook of centroids and corresponding variance vectors, S is the scaling factor of the activation function ($\tanh()$), C_i a set of basis function scales and finally w_i a set of basis function weights. A gradient descent based error minimisation algorithm [3] is used for training the RBF networks to associate the output values $+1$ and -1 with respectively class I and class $\neg I$. As a result, it can be shown [10] that the RBF networks approximate the Bayes optimal discriminant function:

$$g_{\Phi}(\vec{\phi}') \approx P(I|\vec{\phi}') - P(\neg I|\vec{\phi}') \tag{7}$$

In the test situation a local verification score is computed for each phoneme segment using the relevant RBF network (equation 6). These local verification scores are then averaged into a global verification score, which is used for decision making.

2.1. Linear Speech Transforms

The classifiers in an ensemble model should ideally make independent classification decisions, and in order to promote this, the training algorithm must be able to produce multiple classifiers that are as "different" as possible, without compromising too much on the classification accuracy of the individual classifiers. Different classifiers of the same architecture can be trained for instance by training each classifier from different partitions of the training data [4], or by using gradient descent training where different parameter initialisations are used for the different classifiers [5, 6]. In this study a different approach is taken: different classifiers are constructed by using different speech transforms (equation 5) for preprocessing the phoneme vectors. Five different linear transforms are considered: the Fisher transform [2], the LDA transform [7], the PCA transform [8] and the cosine transform [9].

The Fisher Transform
The Fisher transform [2] is a target-speaker dependent discriminative transform based on Fisher's linear discriminant function. The basis vectors in the transform (the columns of $\mathbf{A}_{\Phi}$) are computed using a number of cohort speakers (training impostors) equal to the desired number of coefficients in the transformed feature space, i.e. one basis vector is computed for every cohort speaker:

$$\vec{a}_i = \mathbf{U}_i^{-1}(\vec{\mu}_1 - \vec{\mu}_{2,i})^T \tag{8}$$

where $\vec{\mu}_1$ is the mean phoneme vector for the target speaker, $\vec{\mu}_{2,i}$ the mean phoneme vector for impostor speaker number i, and $\mathbf{U}_i$ the pooled phoneme vector covariance matrix for the target speaker and impostor speaker number i.

The LDA Transform
The LDA transform [7] is a discriminative transform. The transform is estimated from a set of speakers — of which one here is the target speaker — and the idea is to compute the basis vectors of the transform in such a way that the between speaker variance is maximised relative to the within speaker variance. For an N dimensional transform, this is in practise done by constructing a transform where the basis vectors are formed by the N eigenvectors corresponding to the N largest eigenvalues of the matrix $\mathbf{W^{-1}B}$, where $\mathbf{W}$ and $\mathbf{B}$ are respectively the average within speaker covariance and the covariance of the speaker means.

The PCA Transform
The PCA transform [8] is a non-discriminative transform. The idea is to extract a number of coefficients, which explain most of the feature variations. For an N dimensional transform, this is done by constructing a transform where the basis vectors are formed by the N eigenvectors corresponding to the N largest eigenvalues of the average within speaker covariance matrix, $\mathbf{W}$.

The Cosine Transform
The cosine transform [9] is a data independent transform. The coefficients of the i'th basis vector, $\vec{a}_i$ are given by

$$a_{i,k} = \cos\left(i(k-0.5)\frac{\pi}{K}\right), \;\; k = 1, \ldots, K \tag{9}$$

where K is the dimensionality of the untransformed feature space. The cosine transform is sometimes referred to as a "cheap" approximation of the PCA transform, because for speech processing, cepstral coefficients turn out to be relatively uncorrelated (compared to filter bank coefficients).

3. SPEECH DATA

In this work, the Swedish Gandalf database [11] was used. The database contains speech recorded over the public telephone network; The target speaker set consists of 58 speakers (23 female + 35 male) recorded over a one year period, and the impostor set consists of 77 speakers (28 female + 49 male). The speech items from Gandalf that were used in these experiments consist of digit strings. For enrolment purposes, the target speakers were prompted for 25 5-digit utterances in a single session (session 1). The test trials (sessions 2–28) were based on 4-digit utterances: each speaker verification decision was based on one such utterance. The results reported in section 5 were based on the so called "favorite handset" part of Gandalf, where the target speakers used the same telephone handset as in the enrolment session.

The speech data was parameterised as the logarithmic energy outputs of a filter bank with 24 triangular filters spaced linearly along the logarithmic mel scale; each filter overlapped 50% with each of its two neighbours. Feature vectors were extracted using a 25.6 ms Hamming window and a 10 ms frame period. Phoneme segments were identified by forced Viterbi decoding using SI HMMs. The phoneme segments were represented by extracting three feature vectors from each phoneme segment (one feature vector per emitting state in the HMM phoneme models). Hence, the resulting phoneme vectors were 72 dimensional. In order to eliminate the signal gain, the phoneme vectors were normalised to have norm one [3].

4. TRAINING ENSEMBLE CLASSIFIERS

Classifier ensembles can be constructed by combining classifiers trained using the four different transform types under consideration, but each of these transforms can also be used on their own as the basis for constructing classifier ensembles and it is mainly this option that will be investigated here.

The Fisher, LDA and the PCA transform have in common that they have to be estimated from training speech. Hence, for each transform different "versions" of the transform can be computed by estimating it from different partitions of the training data. In particular, this will here be done by creating partitions corresponding to different sets of cohort speakers. A total of 83 cohort speakers was available (for each target speaker the remaining 57 target speakers were included in the set of possible cohort speakers).

The cosine transform does not have to be estimated, since it is completely specified from equation 9. Different classifiers can here be trained by extracting cepstral coefficients of a specific order, and training the classifiers on different subsets of the coefficients, e.g. if 30 cepstral coefficients are computed, three different classifiers can be trained on respectively $(c_1, \ldots, c_{10})$, $(c_{11}, \ldots, c_{20})$ and $(c_{21}, \ldots, c_{30})$.

5. RESULTS & DISCUSSION

A number of experiments were conducted in order to compare the four different transforms and their usefulness when constructing ensemble classifiers. The dimensionality of the feature vectors input to the RBF classifiers is an important parameter. Ensemble classifiers can be introduced as an alternative to simply increasing the input dimensionality of the individual classifiers, i.e. instead of training a 30 dimensional classifier, one can train two 15 dimensional, or three 10 dimensional, etc. This is the principle that was used here when generating ensemble classifiers; all the ensemble classifiers have a total complexity which is comparable to using single classifiers that take 30 dimensional phoneme vectors as input.

Table 1 summarises the Equal Error Rates (EERs) for each of the four transforms when classifiers are used that take between 1 and 60 dimensional transformed phoneme vectors as input (column #Dim). The EERs were computed by adjusting the false rejection and false acceptance error rates a posteriori using a single SI decision threshold. The impostor attempts were simulated by using only same sex impostor speakers. The EERs reported for the 1 – 15 dimensional classifiers are the average error rates for the individual classifiers in each ensemble. The EERs are shown graphically in figure 1 and here error bars are used to indicate the EERs of respectively the worst and the best performing classifier in each ensemble. Also shown in table 1 and figure 1 are the EERs for the classifier ensembles; column #Ens in table 1 indicates the number of members in each classifier ensemble.

For all four transforms it is possible to train classifiers that have EERs below 6% even when ensemble techniques are not used. There is, however, a big difference between the different transforms concerning how many coefficients are needed in order to obtain the optimal performance. The Fisher transform needs 15 (3.7%), the LDA transform 10 (4.1%), the PCA transform 20 (5.2%) and the cosine transform 40 (5.3%) coefficients.

#Dim	#Ens	Equal Error Rates			
		Fisher	**LDA**	**PCA**	**Cosine**
1	1	15.7%	15.3%	30.3%	31.4%
3	1	7.7%	6.1%	17.3%	19.0%
5	1	5.4%	5.6%	12.0%	14.3%
10	1	4.0%	4.1%	7.7%	10.3%
15	1	3.7%	4.1%	6.2%	8.4%
20	1	3.9%	4.3%	**5.2%**	6.9%
30	1	3.7%	4.5%	5.3%	5.7%
40	1	3.7%	5.0%	5.4%	**5.3%**
50	1	3.6%	5.1%	5.2%	5.4%
60	1	3.8%	5.2%	5.9%	5.4%
1	30	8.8%	8.6%	25.6%	6.1%
3	10	5.6%	4.3%	15.2%	9.6%
5	6	4.3%	3.7%	10.7%	10.5%
10	3	3.4%	**3.5%**	7.0%	9.6%
15	2	**3.3%**	3.8%	5.9%	7.8%

Table 1: EERs for the four different transforms. The test set contained a total of 5115 (favorite handset) target speaker trials and 9413 same sex impostor speaker trials.

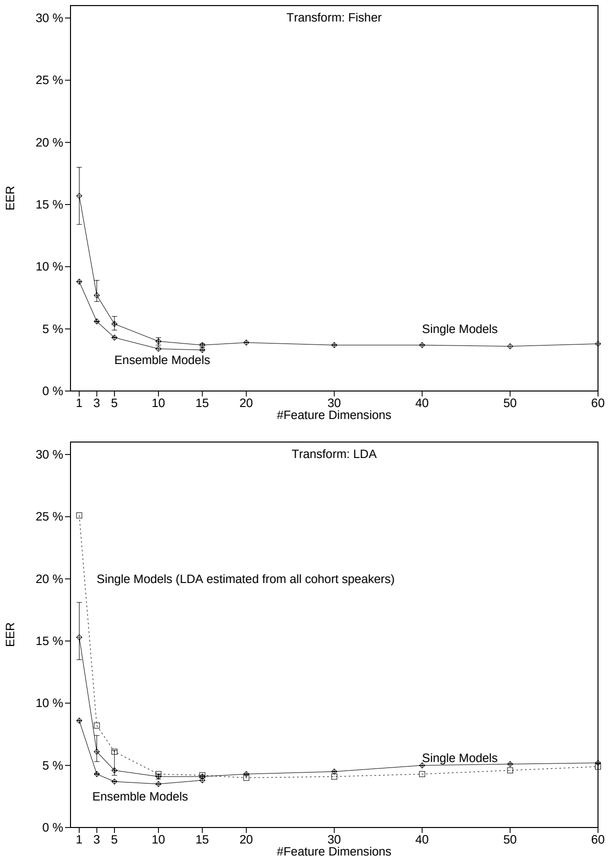

Figure 1: EERs for the Fisher and LDA based classifiers.

In all the experiments the ensemble classifiers outperform even the best performing individual classifier in the classifier ensemble. For the PCA transform, however, the improvements are relatively small indicating that the PCA transforms computed from different subsets of speakers are quite similar: the same principle components characterise all speakers, and consequently the resulting classifiers are not independent. In this respect the PCA transform is quite different from the cosine transform, which performs the best when an ensemble of 30 one dimensional classifiers, corresponding to $c_1, \ldots, c_{30}$, are used.

Overall the best performance is achieved when using the Fisher and LDA transforms. For the one dimensional case the two transforms are almost identical and as expected the EERs here are almost identical. For multidimensional transforms, the Fisher transform generally performs slightly better than the LDA transform. It may seem unfair to compare the LDA ensemble classifiers to single LDA classifiers where the transforms have been computed from only a subset of the training data, since estimating the LDA transforms from all the available cohort speakers could improve the quality of the transforms. However, due to a “deficiency” in the LDA criteria [2], this turns out not to be the case. This exemplified in figure 1, where for the LDA transform, the EERs are shown when the transforms have been computed from all the available cohort speakers.

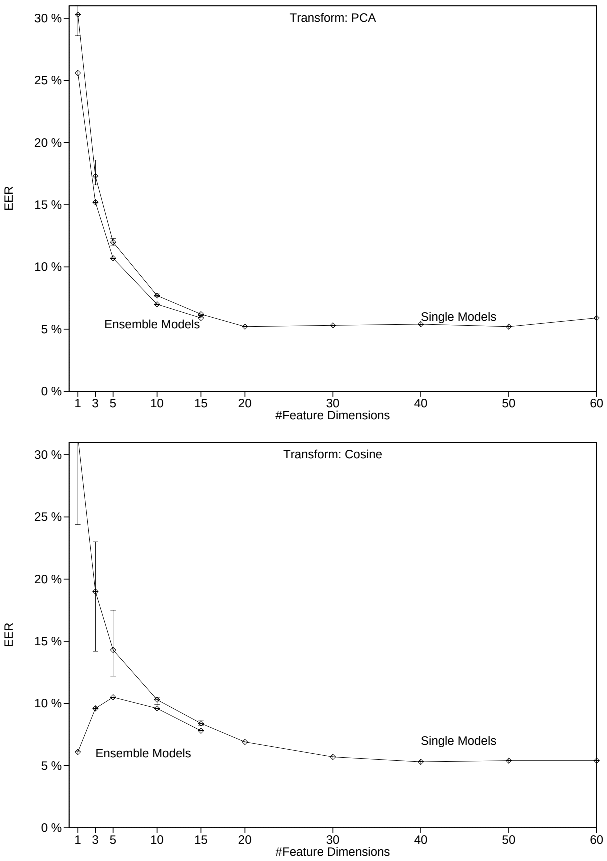

Figure 2: EERs for the PCAand cosine based classifiers.

5.1. Combining Different Transform Types

The EERs reported in table 1 can be further reduced by constructing even larger classifier ensembles. The best performing system configurations for the four different transform are indicated in bold in table 1. Table 2 shows the EERs for the classifier ensembles that result when these classifier ensembles are combined. The EER is slightly improved each time new classifiers are added to the pool; this is so even when the relatively poor performing PCA and cosine based classifiers are added. Ideally the different classifiers in an ensemble should not be combined simply by averaging, because this does not take into account that the individual classifiers have different classification accuracies and dependencies. A better solution would be to use a weighted average, but unfortunately it is in practise very difficult to estimate these weights a priori.

#Dim	#Ens	EER
Fisher	2	3.30%
+LDA	5	3.25%
+PCA	6	3.23%
+cosine	7	3.20%

Table 2: EERs when merging the best performing classifier ensembles for each of the four transform types.

6. CONCLUSIONS

Four different linear transforms, the Fisher, the LDA, the PCA and the cosine transform, were used as the basis for constructing classifier ensembles. Used for single classification systems, the Fisher and the LDA transforms were the most effective, both in terms of the number of coefficients needed to obtain the optimal performance and in terms of the achieved EER. Of the two transforms, the Fisher transform was the slightly more effective, no doubt owing to the fact that it is target speaker dependent. In order to be effective, the LDA transform needs to be computed from a relatively small number of cohort speakers, otherwise the highest ranking basis vectors will not be able to discriminate well between similar speakers. The PCA and the cosine transforms both needed a relatively large number of coefficients in order to perform well. This is not surprising, since they are both non-discriminative. The lowest EERs obtained using these two transforms were significantly higher that those of the Fisher and LDA transforms. The PCA and the cosine transform proved not to be useful for constructing ensemble classifiers. None of the ensemble classifiers could here outperform a single classifier verification system of the same total complexity. The performance of the Fisher and LDA based classifier ensembles turned out to be comparable, which is interesting since the Fisher transform is target speaker dependent, whereas the LDA transform depends mostly on the cohort speaker set. This indicates that basis vectors that in general discriminate well between pairs of speakers (equation 8) are useful. In principle this idea can be used for producing ensemble sets, which are significantly larger than the ones constructed here.

7. REFERENCES

[1] S. Furui, *Recent Advances in Speaker Recognition*, Lecture Notes in Computer Science 1206, Springer Verlag, pp. 237–252, 1997

[2] J. Olsen, *A two-stage procedure for phone based speaker verification*, Pattern Recognition Letters (Elsevier) 18:889–897, 1997

[3] J. Olsen, *Phoneme based speaker recognition*, PhD Thesis, Aalborg University, 1997.

[4] A. Krogh and J. Vedelsby, *Neural Network Ensembles, Cross Validation and Active Learning*, Advances in NIPS, Volume 7, pp. 231–238, 1995.

[5] L.K. Hansen, P. Salamon, *Neural network ensembles*, IEEE Transactions on Pattern Analysis and Machine Inteligence, 12(10):993–1001, 1990

[6] M.P. Perrone, L.N. Cooper, *When networks disagree: ensemble methods for hybrid neural networks*, Chapter 10 of *Artificial Neural Networks for Speech and Vision*, R.J. Mammone (Ed.), Chapman & Hall, pp. 126–142, 1994

[7] D. Hand, *Discrimination and Classification*, John Wiley & Sons, 1981

[8] I. Jolliffe, *Principle Component Analysis*, Springer Verlag, 1986

[9] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, Prentice Hall, 1993

[10] D.W. Ruck et al., *The Multilayer perceptron as an approximation to a Bayes optimal discriminant function*, IEEE Trans. on Neural Networks, 1(4):296–298, 1990

[11] H. Melin, *GANDALF - A Swedish telephone speaker verification database*, Proc. of ICSLP, 1996, pp. 1954–1957