FEATURE SELECTION VS. FEATURE TRANSFORMATION IN REDUCING DIMENSIONALITY FOR SPEAKER RECOGNITION

Maider Zamalloa[1,2]*, L. J. Rodríguez-Fuentes*[1]*, Mikel Peñagarikano*[1]*, Germán Bordel*[1]*, Juan P. Uribe*[2]

(1) Grupo de Trabajo en Tecnologías del Software, DEE, ZTF/FCT
Universidad del País Vasco / Euskal Herriko Unibertsitatea
Barrio Sarriena s/n, 48940 Leioa, SPAIN

(2) Ikerlan – Technological Research Centre
Paseo J.M. Arizmendiarrieta 2, 20500 Arrasate-Mondragón, SPAIN

e-mail: maider.zamalloa@ehu.es

ABSTRACT

Mel-Frequency Cepstral Coefficients and their derivatives are commonly used as acoustic features for speaker recognition. Reducing the dimensionality of the feature set leads to more robust estimates of the model parameters, and speeds up the classification task, which is crucial for real-time speaker recognition applications running on low-resource devices. In this paper, a feature selection procedure based on genetic algorithms (GA) is compared to two well-known dimensionality reduction techniques based on linear transforms, namely Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). Evaluation is carried out for two speech databases, containing laboratory read speech and telephone spontaneous speech, and applying a state-of-the-art speaker recognition system. Results with GA-based feature selection suggest that dynamic features are less discriminant than static ones, since the low-size optimal subsets found by the GA did not include dynamic features. GA-based feature selection outperformed PCA and LDA when dealing with clean speech, but not for telephone speech, probably due to some noise compensation implicit in linear transforms, which cannot be accomplished just by selecting a subset of features.

1. INTRODUCTION

Mel-Frequency Cepstral Coefficients (MFCC) are commonly used as acoustic features for speaker recognition, since they convey not only the frequency distribution identifying sounds, but also information related to the glottal source and the vocal tract shape and length, which are speaker specific features. Additionally, it has been shown that dynamic information improves the performance of recognizers, so first and second derivatives are appended to MFCC. The resulting feature vector ranges from 30 to 50 dimensions. However, for applications requiring real-time operation on low-resource devices, high dimensional feature vectors do not seem suitable and some kind of dimensionality reduction must be applied, maybe at the cost of performance degradation.

A simple approach to dimensionality reduction is feature selection, which consists of determining an optimal subset of K features by exhaustively exploring all the possible combinations of D features. Most feature selection procedures use the classification error as the evaluation function. This makes exhaustive search computationally infeasible in practice, even for moderate values of D. The simplest method consists of evaluating the D features individually and selecting the K most discriminant ones, but it does not take into account dependencies among features. So a number of suboptimal heuristic search techniques have been proposed in the literature, which essentially trade-off the optimality of the selected subset for computational efficiency [1].

Genetic Algorithms (GA) suitably fit this kind of complex optimization problems. A major advantage of GA over other heuristic search techniques is that they do not rely on any assumption about the properties of the evaluation function. Multiobjective evaluation functions (e.g. combining the accuracy and the cost of classification) can be defined and used in a natural way [2]. GA can easily encode decisions (about selecting or not selecting features) as sequences of boolean values, allow to smartly explore the feature space by retaining those decisions that benefit the classification task, and simultaneously avoid local optima due to their intrinsic randomness. GA have been recently applied to feature extraction [3], feature selection [4] and feature weighting [5] in speaker recognition.

Alternatively, the problem of dimensionality reduction can be formulated as a linear transform which projects feature vectors on a transformed subspace defined by relevant directions. Among others, two well-known dimensionality reduction techniques, Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA), fall into this category.

This work has been jointly funded by the Government of the Basque Country, under projects S-PE06UN48, S-PE07UN43, S-PE06IK01 and S-PE07IK03, and the University of the Basque Country, under project EHU06/96.

In this paper, a feature selection procedure based on a GA-driven search is compared to PCA and LDA in a speaker recognition task. GA-based feature selection projects the original D-dimensional feature space into a reduced K-dimensional subspace by just selecting K features. PCA and LDA not only reduce but also scale and rotate the original feature space, through a transformation matrix A which optimizes a given criterion on the training data. From this point of view, PCA and LDA generalize feature selection, but the criteria applied to compute A (the highest variance in PCA, and the highest ratio of between to within class variances in LDA) do not match the criterion applied in evaluation (the speaker recognition rate). This is the strong point of GA, since feature selection is performed in order to maximize the speaker recognition rate on an independent development corpus.

2. FEATURE SELECTION USING GENETIC ALGORITHMS

The GA-driven selection process begins by fixing the target size K of the reduced feature subspace. Then, an initial population of candidate solutions (K-feature subsets) is randomly generated. In this work, each candidate is represented by a D-dimensional vector of positive integers $R = \{r_1, r_2, \ldots, r_D\}$, ranging from 0 to 255 (8 bits), the K highest values determining what features are selected. To evaluate the K-feature subset $= \{f_1, f_2, \ldots, f_K\}$, the following steps are carried out (1) the acoustic vectors of the whole speech database are reduced to the components enumerated in ; (2) speaker models are estimated using the training corpus; (3) utterances in the development corpus are classified by applying the speaker models; and (4) the speaker recognition accuracy obtained for the development corpus is used to evaluate .

At the end of each iteration/generation, after all the K-feature subsets in the population are evaluated, some of them (usually the fittest ones), are selected, mixed and mutated in order to get the population for the next generation. Mutation is used to introduce small variations that help decrease the chances of getting local optima. On the other hand, *elitism* (copying some of the fittest individuals to the next generation) is applied to guarantee that the fitness function increases monotonically with successive generations. If that increase is smaller than a given threshold, or a maximum number of generations is reached, the algorithm stops and the optimal K-feature subset $\hat{} = \{\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_K\}$ is returned.

3. EXPERIMENTAL SETUP

3.1. Acoustic features

In this work, MFCC, energy and their first and second derivatives were taken as acoustic features. Speech was analysed in 25-millisecond frames, at intervals of 10 milliseconds. A Hamming window was applied and an FFT computed, whose length depended on the sampling frequency: 256 points for signals sampled at 8 khz and 512 points for signals sampled at 16 kHz. FFT amplitudes were then averaged in 20 (8 kHz) or 24 (16 kHz) overlapped triangular filters, with central frequencies and bandwidths defined according to the Mel scale. A Discrete Cosine Transform was finally applied to the logarithm of the filter amplitudes, obtaining 10 (8 kHz) or 12 (16 kHz) Mel-Frequency Cepstral Coefficients (MFCC). To increase robustness against channel distortion, Cepstral Mean Normalization was applied on an utterance-by-utterance basis. The first and second derivatives of the MFCC, the frame energy (E) and its first and second derivatives were also computed, thus yielding a 33-dimensional (8 khz) or a 39-dimensional (16 kHz) feature vector.

3.2. Speaker models

Most speaker recognition systems represent the distribution of feature vectors extracted from a speaker's speech by a linear combination of M multivariate Gaussian densities, known as *Gaussian Mixture Model* (GMM) [6], whose parameters are estimated from speaker samples by applying the *Maximum Likelihood* (ML) criterion. In this work, speaker recognition was performed using 32-mixture diagonal covariance GMMs as speaker models.

3.3. Speech databases

Two speech databases were used in this work: *Albayzín* (a phonetically balanced read speech database in Spanish, recorded at 16 KHz in laboratory conditions, containing 204 speakers) and *Dihana* (a spontaneous task-specific speech corpus in Spanish, recorded at 8 kHz through telephone lines, containing 225 speakers), each partitioned in three disjoint datasets: (1) the training set, used to estimate the speaker models and the PCA and LDA transforms; (2) the development set, used by the GA to compute the fitness function; and (3) the test set, used to evaluate the performance of the optimal K-feature subsets provided by GA, PCA and LDA.

3.4. GA, PCA and LDA Implementations

The well-known *Simple Genetic Algorithm* (SGA) [7], implemented by means of ECJ [8], was applied to search for the optimal feature set. Offspring was bred by first selecting and then mixing two parents in the current population. The first parent was selected according to the fitness-proportional criterion, by picking the fittest from seven randomly chosen individuals. The second parent was chosen the same way, but only from two randomly chosen individuals, to allow diversity and avoid local optima. One-point crossover was applied and the mutation probability was set to 0.01. Finally, the simplest case of elitism was applied by keeping the fittest individual for the next generation. The maximum number of generations was fixed to 40. A public domain software developed at the MIT Lincoln Laboratory, *LNKnet* [9], was used to perform PCA. Regarding LDA, a custom implementation was developed in Java.

Table 1. Optimal feature sets found by the GA in speaker recognition experiments for Albayzín and Dihana, for K = 30, 20, 13, 12, 11, 10, 8 and 6. Selected features are marked with a star ($\star$). Cells containing a dash (–) correspond to features not computed for Dihana.

		E	c01	c02	c03	c04	c05	c06	c07	c08	c09	c10	c11	c12	dE	d01	d02	d03	d04	d05	d06	d07	d08	d09	d10	d11	d12	ddE	dd01	dd02	dd03	dd04	dd05	dd06	dd07	dd08	dd09	dd10	dd11	dd12
Albayzín	30	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆				⋆	⋆		⋆			⋆	⋆	⋆	⋆	⋆			⋆		⋆	⋆		⋆	⋆	⋆	⋆	⋆	⋆
	20	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆						⋆				⋆	⋆							⋆		⋆	⋆				⋆	
	13	⋆	⋆	⋆	⋆	⋆		⋆	⋆	⋆	⋆	⋆	⋆	⋆																			⋆							
	12	⋆	⋆	⋆		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆																										
	11	⋆		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆																										
	10	⋆	⋆	⋆	⋆	⋆		⋆			⋆	⋆	⋆	⋆																										
	8	⋆		⋆		⋆		⋆		⋆	⋆		⋆	⋆																										
	6	⋆				⋆	⋆			⋆		⋆	⋆																										
Dihana	30	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–	⋆			⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–
	20	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–			⋆	⋆	⋆	⋆	⋆	⋆		⋆	⋆	–	–										⋆		–	–
	13		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–						⋆		⋆	⋆			–	–												–	–
	12		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–				⋆				⋆				–	–												–	–
	11		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–								⋆				–	–												–	–
	10		⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	⋆	–	–												–	–												–	–
	8		⋆	⋆	⋆		⋆	⋆	⋆	⋆		⋆	–	–												–	–												–	–
	6		⋆				⋆	⋆	⋆	⋆		⋆	–	–												–	–												–	–

4. RESULTS

4.1. Feature selection using GA

The optimal K-dimensional feature sets provided by the GA for Albayzín and Dihana in speaker recognition experiments are shown in Table 1. The terms *cXX* and *E* correspond to the MFCC and the frame energy, and *dXX*/*dE* and *ddXX*/*ddE* to their first and second derivatives, respectively. As noted above, the computation of MFCC depends on the sampling frequency, so the dimension (D) of the full feature vectors is 39 (12 MFCC + energy + first and second derivatives) for Albayzín, and 33 (10 MFCC + energy + first and second derivatives) for Dihana.

Focusing on the results for Albayzín, note that the presence of a feature in the optimal subset of size K does not imply that the same feature will be present in the optimal subsets for larger values of K. For instance, *c05* appears in the optimal subset for $K = 6$, but not for $K = 8$ and $K = 10$. This suggests that optimal subspaces cannot be determined in an incremental way, by sequentially reducing its size. In other words, it seems that an exhaustive search is needed which explores all the feature combinations. Note also that the GA-optimal subsets for $K \leq 12$ consist of a number of MFCC plus the frame energy. Three of them, *E*, *c04* and *c11* are always selected by the GA; three other, *c02*, *c06* and *c09*, are selected always but for the smallest subset ($K = 6$). This suggests that static features are more relevant for speaker recognition than dynamic features.

The optimal subsets found by the GA for Dihana show an almost perfect sequential behaviour, contrasting with that obtained for Albayzín. Only two cases of non-sequential behaviour are found: *d08*, from $K = 20$ to $K = 13$; and *d03*, from $K = 13$ to $K = 12$. It would be worth to investigate this issue more deeply, since sequential optimization is much faster that an exhaustive search. On the other hand, the optimal sets for low values of K ($K \leq 10$) are composed exclusively of MFCC (the frame energy and dynamic features do not appear). Again, it seems that MFCC convey more relevant information about speaker characteristics than their derivatives. Interestingly, the frame energy does not appear in any of the optimal sets for $K \leq 13$, suggesting that this feature is not as robust for telephone spontaneous speech as for laboratory read speech.

4.2. Comparing GA to PCA and LDA

GA-based feature selection, PCA and LDA were tested in speaker recognition experiments over Albayzín and Dihana. First, D-dimensional feature vectors were transformed into reduced K-dimensional feature vectors, according to the optimal subset/transformation given by GA, PCA or LDA, then speaker models were estimated on the training corpus and finally speaker recognition experiments were carried out on the test corpus. Results are shown in Table 2.

Confidence intervals are shown to allow significant performance comparisons among different feature sets. This deserves a brief explanation. Model estimations start from random initializations. Preliminary experimentation showed that, fixed the set of features and the training database, random initializations led to slightly different model parameters after convergence, and therefore slight differences in speaker recognition performance were observed. This intrinsic uncertainty can be taken into account in performance comparisons by computing the confidence interval of an average error rate. It is assumed that the underlying distribution of error rates is Gaussian. So,

Table 2. Average error rates and 95% confidence intervals in speaker recognition experiments on test data for Albayzín and Dihana, using the optimal K-dimensional feature sets provided by GA, PCA nd LDA, for K = 6, 8, 10, 11, 12, 13, 20 and 30.

K	Albayzín			Dihana		
	GA	PCA	LDA	GA	PCA	LDA
6	**5.71±0.09**	14.37±0.15	8.11±0.14	34.23±0.16	**33.23±0.12**	35.52±0.14
8	**1.81±0.09**	5.86±0.12	2.64±0.09	**23.90±0.14**	24.19±0.13	25.06±0.13
10	**0.94±0.04**	2.73±0.12	1.21±0.06	19.70±0.12	20.67±0.12	**19.43±0.12**
11	**0.35±0.04**	1.61±0.07	1.12±0.06	19.32±0.14	20.27±0.13	**18.10±0.13**
12	**0.30±0.04**	0.94±0.06	0.79±0.06	19.27±0.14	19.75±0.16	**18.18±0.12**
13	**0.33±0.05**	0.56±0.05	0.88±0.04	19.12±0.11	19.63±0.10	**17.66±0.10**
20	**0.16±0.02**	0.19±0.02	0.39±0.04	19.99±0.11	17.61±0.13	**17.24±0.11**
30	**0.13±0.02**	0.15±0.03	0.33±0.04	19.10±0.14	**15.97±0.15**	18.17±0.12

in order to compute the average error rate and the 95% confidence interval, the whole process of training speaker models and carrying out speaker recognition experiments was repeated 20 times for each feature set.

In the case of Albayzín, neither PCA nor LDA outperformed GA. PCA yielded lower error rates than LDA for $K > 12$. For $K \leq 12$, LDA outperformed PCA. However, the error rates are too low and the differences in performance too small for these conclusions to be statistically significant.

Error rates for Dihana were much higher, because it was recorded through telephone channels in an office environment and a large part of it consists of spontaneous speech. The presence of channel and environment noise in Dihana makes PCA and LDA more suitable than GA, because feature selection cannot compensate for noise, whereas linear transforms can do it to a certain extent. This may explain why either PCA or LDA outperformed GA in all cases but for $K = 8$. LDA was the best approach in most cases (for K = 10, 11, 12, 13 and 20). GA was the second best approach for K = 6, 10, 11, 12 and 13. Finally, the lowest error rate (15.97%) was obtained for $K = 30$ using PCA.

In summary, GA-based feature selection seems to be competitive only when dealing with clean speech, though it performs quite well even for noisy speech when the target K is small. Authors that argue against GA optimization say that it is too costly, since it requires iteratively evaluating candidate solutions in classification experiments over a development dataset. It must be noted, however, that GA optimization is done off-line, so the computational cost is not an issue in practice. Moreover, during recognition, feature selection is less costly than feature transformation.

5. CONCLUSIONS

Feature selection based on GA suggests that static features are more discriminant than dynamic features for speaker recognition applications. In the case of telephone speech, the smallest feature subsets ($K \leq 13$) did not include the frame energy, which reveals that channel and/or environment noise is distorting the information it conveys. Summarizing, if a reduced set of features had to be selected (due to storage or computational restrictions), MFCC would be the best choice, augmented with the frame energy when dealing with clean-laboratory speech.

GA outperformed PCA and LDA only when dealing with clean speech, whereas PCA and LDA outperformed GA in most cases when dealing with telephone speech, probably due to some noise compensation implicit in linear transforms, which cannot be accomplished just by selecting a subset of features. In any case, since applying a linear transform is more costly than selecting a subset of features, depending on the target K, the gain in performance might not be worth the additional effort.

6. REFERENCES

[1] A. K. Jain, R. P. W. Duin, y J. Mao, "Statistical Pattern Recognition: A Review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, January 2000.

[2] L. S. Oliveira, R. Sabourin, F. Bortolozzi, y C. Y. Suen, "A Methodology for Feature Selection Using Multiobjective Genetic Algorithms for Handwritten Digit String Recognition," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 17, no. 6, pp. 903–929, 2003.

[3] C. Charbuillet, B. Gas, M. Chetouani, y J. L. Zarader, "Filter Bank Design for Speaker Diarization Based on Genetic Algorithms," in *Proceedings of the IEEE ICASSP'06*, Toulouse, France, 2006.

[4] M. Zamalloa, G. Bordel, L. J. Rodriguez, y M. Peñagarikano, "Feature Selection Based on Genetic Algorithms for Speaker Recognition," in *IEEE Speaker Odyssey: The Speaker and Language Recognition Workshop*, Puerto Rico, June 2006, pp. 1–8.

[5] M. Zamalloa, G. Bordel, L. J. Rodríguez, M. Peñagarikano, y J. P. Uribe, "Using Genetic Algorithms to Weight Acoustic Features for Speaker Recognition," in *Proceedings of the ICSLP'06*, Pittsburgh (USA), September 2006.

[6] D. A. Reynolds y R. C. Rose, "Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 72–83, January 1995.

[7] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, 1989.

[8] ECJ 16, ," http://cs.gmu.edu/ eclab/projects/ecj/.

[9] R. P. Lippmann, L. Kukolich, y E. Singer, "LNKnet: Neural Network, Machine Learning and Statistical Software for Pattern Classification," *Lincoln laboratory Journal*, vol. 6, no. 2, pp. 249–268, 1993.