ISCA Archive
http://www.isca-speech.org/archive

INTERSPEECH 2004 - ICSLP
8th International Conference on Spoken Language Processing
ICC Jeju, Jeju Island, Korea
October 4-8, 2004

A New Nonlinear Feature Extraction Algorithm for Speaker Verification

M. Chetouani, M. Faundez-Zanuy †, B. Gas, J.L. Zarader

Laboratoire des Instruments et Systèmes d'Ile-De-France
Université Paris VI, Paris, FRANCE
(†) Escola Universitària Politècnica de Mataró , Barcelona, SPAIN
mohamed.chetouani@lis.jussieu.fr faundez@eupmt.es gas@ccr.jussieu.fr zarader@ccr.jussieu.fr

10.21437/Interspeech.2004-500

Abstract

In this paper we propose a new feature extraction algorithm based on nonlinear prediction: the Neural Predictive Coding model which is an extension of the classical LPC one. This model is applied to speaker verification by the Arithmetic-Harmonic Sphericity (AHS) method. Two different initialization methods are proposed for the coding method based on the Neural Predictive Coding (NPC): classical neural networks initialization and linear initialization. The first model obtains smaller rates. For the linear initialization, we obtain significant improvement in comparison to the most used methods (LPCC, MFCC). This study opens a new way towards different feature extraction schemes that offers better accuracy on speaker recognition tasks.

1. Introduction

Feature extraction plays a very significant role in speaker recognition systems. The extracted features have to modelize suitable characteristics of each speaker. In applications like language or speech recognition, on will rather seek to eliminate speakers characteristics. The most used coding methods are the LPCC (Linear Prediction Cepstral Coefficients) and the MFCC (Mel Frequency Cepstral Coefficients) ones. As it is noted in [1], the main drawback of these methods is that they do not guarantee best results in classification. The Independent Component Analysis (ICA) has been applied to feature extraction within the speaker recognition framework [2]. These features are extracted by a learning process and they are localized both in the time and the frequential domains. Another way for dynamic modelization is the Time-Frequency Principal Components (TFPC) [3].

The discriminant power of the features is also an important element. Discriminant criteria have been introduced by several approaches. The Linear Discriminant Analysis (LDA) is used for the design of the DCS parameters (Discriminant Components of the Spectrum) [3]. A different approach consists in the simultaneous training of both the feature extractor and the classifier by the same criterion based on the Minimization of the Classification Error (MCE) [4]. This method has been successfully applied to the design of filter banks in speech recognition [5]. Recently, the Discriminative Feature Extraction (DFE) has been used for the learning of a feature extractor based on mel scale estimation (different from the Mel scale) and GMM/VQ models [1].

In this paper we propose a new feature extraction algorithm: The Neural Predictive Coding (NPC) model described in section 2. We propose an initialization method based on speech knowledge. Next, we present the database and the other coding methods. Then, we briefly introduce the method used for speaker verification. Finally, we give some results and conclusions.

This work has been supported by COST-277, FEDER & CICYT TIC-2003-08382-C05-02

2. Neural Predictive Coding

The Neural Predictive Coding (NPC) [6] model (cf. Fig. 1) is a non-linear extension of the well-known LPC encoder. Like in the LPC framework with the AR model, the vector code is estimated by prediction error minimization. The main difference relies in the fact that the model is non-linear and it is a connectionnist model:

$$\hat{y}_k = F(\mathbf{y}_k) = \sum_j a_j \sigma(\mathbf{w}^T \mathbf{y}_k) \quad (1)$$

Where F is the prediction function realized by the neural model. $\hat{y}_k$ is the predicted sample and $\mathbf{y}_k$ the prediction context: $\mathbf{y}_k = [y_{k-1}, y_{k-2}, \ldots, y_{k-\lambda}]^T$ and λ the length of the prediction window. $\mathbf{w}$ and $\mathbf{a}$ represent the first and the output layer weights. σ is a non-linear activation function, the sigmoid function in our case.

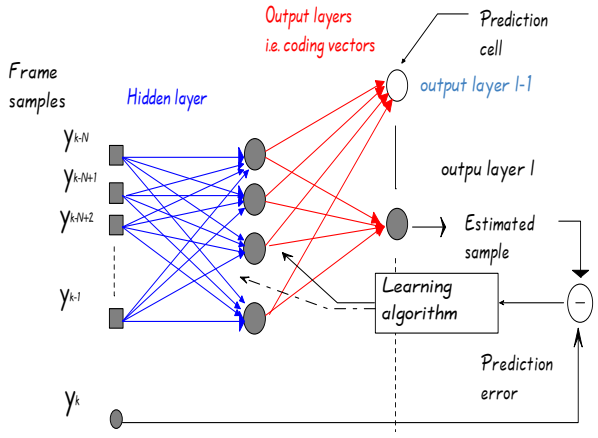


Figure 1: *Neural Predictive Coding (NPC) architecture: a connectionist model used as a non-linear predictor.*

The key idea is to use the NPC model as a non-linear autoregressive model. As in the LPC framework for the predictor coefficients, the NPC weights are the vector code. It is well-known that the weights can be consider as a representation of the input vector. A drawback of this method is that non-linear models have no clear physical meaning [7]. The solution weights can be very different for a same minimum of the prediction error. In our approach, we impose constraints on weights.

The NPC model has been successfully applied to phoneme classification [6, 8]. In this paper, we investigate feature extraction for

speaker verification.

2.1. Description

The NPC model is a Multi-Layer Perceptron (MLP) with one hidden layer. Only the output layer weights are used as coding vector instead of all the neural weights. For that we consider that the function F realized by the model, under convergence assumptions, can be decomposed into two functions: $G_{\mathbf{w}}$ ($\mathbf{w}$ first layer weights) and $H_{\mathbf{a}}$ ($\mathbf{a}$ output layer weights):

$$F_{\mathbf{w},\mathbf{a}}(\mathbf{y}_k) = H_{\mathbf{a}} \circ G_{\mathbf{w}}(\mathbf{y}_k) \tag{2}$$

With $\hat{y}_k = H_{\mathbf{a}}(\mathbf{z}_k)$ and $\mathbf{z}_k = G_{\mathbf{w}}(\mathbf{y}_k)$.

As one can note the NPC structure allows, contrary to the LPC one, a different prediction window's length independently to the coding vector size (cf. Fig. 1).

For that, the learning phase is realized in two times. First, the *parameterization phase* consists in the learning of all the weights by the prediction error minimization criterion:

$$Q = \sum_{k=1}^{K} (y_k - \hat{y}_k)^2 = \sum_{k=1}^{K} (y_k - F(\mathbf{y}_k))^2 \tag{3}$$

With y the speech signal, $\hat{y}$ the predicted speech signal, k the samples index and K the number of samples.

In this phase, only the first layer weights $\mathbf{w}$ which are the NPC encoder parameters are kept. Since the NPC encoder is set up by the parameters defined in the previous phase, the second phase, called the *coding phase*, consists in the computation of the output layer weights $\mathbf{a}$ (vector code). This is done also by prediction error minimization but only the output layer weights are updated. One can note that the output function is linear (cf. equation 1), so it can be done by the Levinson algorithm as for the LPC model. Here, for consistency with the *parameterization phase*, it is done by the backpropagation algorithm. In the next section, we discuss on initialization of the coding phase.

2.2. A new coding phase

Once the NPC model is parameterized, the *coding phase* consists in the estimation of the second layer weights by minimizing the prediction error. Like for all optimization processes, the initial weights are randomly chosen. Within the neural networks framework several studies [9, 10] have been carried out in order to try to improve the initialization phase. One of the major disadvantages of these methods is that they are not adapted to speech processing. Indeed, as the weights solutions minimizing the prediction error are multiple, one does not guarantee a continuity of their values in the frame-by-frame analysis specific to nonstationary signals like in speech processing. In order to guarantee this continuity but also in order to obtain a unique solution, we propose a new initialization method for the NPC *coding phase*.

We integrate speech knowledge for the initialization. We make the assumption that the NPC model is an extension of the LPC one. Consequently, one can consider as a linear initialization the NPC model initialized by the LPC model. This idea was used in [11] to initialize neuronal models (multi-layer perceptrons, recurrent networks, etc.) by equivalent linear models: Auto-Regressive (AR), Auto-Regressive model with eXternal input (ARX). The method suggested is based on matrices decomposition methods (QR, SVD, etc.).

During the *coding phase*, the NPC model is parameterized, the first layer weights $\mathbf{w}$ are known. The initialized second layer weights $\mathbf{a}$ (vector code) are determined by equivalence between NPC and LPC models. By neglecting biaises and removing the non-linear activation functions (for linear approximation), one obtains the following equivalence:

$$\Theta = \mathbf{w} \bullet \mathbf{a} \tag{4}$$

Where Θ are the LPC parameters of the speech signal, $\mathbf{w}$ first layer weights (determined in the previous phase: the *parameterization phase*) and $\mathbf{a}$ second layer weights (to determine).

If the NPC vector code dimension is ρ and the prediction window is λ then the LPC vector code dimension has to be set to λ. The second layer weights $\mathbf{a}$ are given by:

$$\mathbf{a} = \mathbf{w}^{+} \bullet \Theta \tag{5}$$

Where $\mathbf{w}^{+}$ is the pseudo-inverse of $\mathbf{w}$.

The initialized weights are determined by a simple LPC analysis with an order λ. Once the initialization is accomplished, the coding process (prediction error minimization) proceeds as well as the original NPC *coding phase* by the backpropagation algorithm.

3. Speaker-dependent feature extraction

Currently in speaker recognition, the speech feature extraction is carried out in a same way for all the speakers. Most of the efforts have been made in the second stage which is the model of the speaker.

In our system, the speaker is obviously modelized by the second stage (reference model) but the first stage (feature extractior) is also specialized in the processing of this speaker. The speaker-dependent features are extracted by the NPC model (cf. Fig. 2). Each NPC model is specialized in the processing of only one speaker.

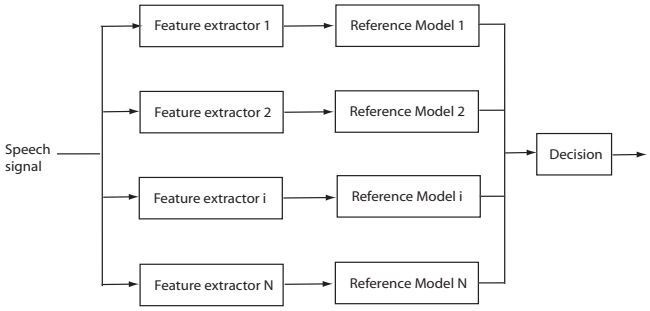


Figure 2: *Speaker dependent feature extraction principle.*

Traditionally, the speaker recognition task is composed by two phases: the enrollment phase and test phase. Due to our approach, these two phases are modified.

3.1. Enrollment phase

During the enrollment phase or training phase, each speaker from the database has to provide samples of their speech. Those speakers who belong to the database are known as client. Then, a reference model is computed from these samples. In our approach, contrary to other methods, a speaker model is composed by a feature extractor and a reference model. Consequently, the enrollment phase is carried out in two stages. First, the feature extractor is trained. This step is the NPC *parameterization phase* described previously. Secondly, from these features, a reference model is computed. One can note that the reference model can be computed by various methods (Gaussian Mixtures Models, Neural Networks, etc.). Once this

phase is accomplished, one obtains a NPC model (feature extractor) and a reference model for each speaker.

A sentence of one minute is used for the enrollment phase. For the NPC *parameterization phase*, we extract a part of the sentence (only 12 seconds). Once the NPC is parameterized, the whole sentence is coded. Then different methods are carried out in order to compute the reference model. This procedure is repeated for all the speakers belonging of the database.

The reference models are estimated by covariance matrices. We compare the covariance matrices by the Arithmetic-Harmonic Sphericity (AHS) [12] method.

3.2. Test phase

During the test phase, the speaker claims his identity. Then the speech signal is coded by the appropriate feature vector. From these features, a decision is done by comparison of the reference models: reference stored model and the reference model computed from the features.

The feature extractor is learned only on speech signals from the genuine speaker. When speech signals from impostors are encoded, the generated features will be very different from the learned features by the reference model.

4. Experimental conditions

In order to evaluate the NPC performances, we apply it in a speaker verification task. We have successfully applied this new feature extractor to speaker identification [13]. In this paper, we focus only on speaker verification. This task is very interesting because it measures the modelization and the discrimination power of our approach.

We compare the proposed coding method based on Neural Predictive Coding (NPC) with the classical ones:

- Linear Predictive Coding (LPC) is based on a linear modelization of the vocal tract. The all-pole auto-regressive (AR) coefficients model of the spectrum captures the vocal tract properties.
- Linear Predictive Cepstral Coding (LPCC) computes a LPC spectral envelope, before converting it into cepstral coefficients. The LPCC are LP-derived cepstral coefficients. The LPCC is the most used coding method in speaker recognition.
- Mel Frequency Cepstral Coding (MFCC) is based on signal decomposition with the help of a filter bank, which uses the Mel scale. The MFCC results of a discrete cosine transform of the real logarithm of the short-term energy expressed on the Mel-frequency scale. The MFCC has shown good performances in speech recognition but also in speaker recognition.
- Perceptual Linear Predictive (PLP) [14] is an example of knowledge integration resulting from psychoacoustics in the estimation of auto-regressive (AR) models. Indeed, this method integrates critical bands, equal loudness preemphasis and intensity-to-loudness compression. The PLP is based on the nonlinear Bark scale. It was originally designed to speech recognition with the removing of speaker dependent characteristics.

4.1. Database

Our experiments have been computed over 49 speakers from the Gaudi database [15] that has been obtained with a microphone connected to a PC. The speech signal has been down-sampled to 8 kHz, pre-emphasized by a first order filter whose transfer function is H(z)=1-0.95z-1. A 30ms Hamming window is used, and the overlapping between adjacent frames is 2/3. A parameterized vector of order 16 was computed. One minute of read text is used for training, and 5 sentences for testing (each sentence is about 2-3 seconds long).

4.2. Speaker verification algorithm

In order to measure the performances of each coding method, we used the Arithmetic-Harmonic Sphericity (AHS) [12]. The AHS is a statistical method based on second order measures. The AHS method has demonstrated good performances on speaker recognition.

A covariance matrix (CM) is computed for each speaker, and an Arithmetic-Harmonic Sphericity (AHS) measure is used in order to compare matrices [12]:

$$\mu(\mathbf{C}_j, \mathbf{C}_{test}) = \log\left(\mathrm{tr}(\mathbf{C}_{test}\mathbf{C}_j^{-1})\mathrm{tr}(\mathbf{C}_j\mathbf{C}_{test}^{-1})\right) - 2log(P) \quad (6)$$

Where tr is the trace of the matrix, P is the feature vector dimension ($P = 16$). The number of parameters for each speaker model is $\frac{P^2+P}{2}$ (the covariance matrix is symmetric). For the CM model, more parameters imply a higher dimensional feature vectors.

We have applied the following equation in order to convert the distance measure μ into a probability measure p: $p = e^{-0.5\mu}$. The system has been evaluated using the DET curves [16], with the following Detection Cost Function (DCF):

$$DCF = C_{miss} \times P_{miss} \times P_{true} + C_{fa} \times P_{fa} \times P_{false} \quad (7)$$

Where C_{miss} is the cost of a miss, C_{fa} is the cost of a false alarm. P_{true} is the a priori probability of the target, $P_{false} = 1 - P_{true}$. The optimal value is indicated in each plot is indicated with a "o" mark. We have used $C_{miss} = C_{fa}$.

5. Verification results

In this section, we show and discuss the results of feature extraction methods for speaker verification.

Figure 3 shows the DET curves for the speaker verification task. This representation allows to show all the possible operating points. This curve reveals that the MFCC and the LPCC behaviors are very close. These two speech coding methods are the most used in speaker recognition. The PLP method gives bad results and it can be explained by the fact that it was originally designed for speaker characteristics suppression.

Concerning the NPC model, very different results are obtained. For the random initialization, the results are worst. In contrast with this initialization, the NPC linear initialization gives best results.

In order to compare the speech coding methods, we have reported (see Table 3) for each method the minimum of the Detection Cost Function (DCF). One can see that the LPC, LPCC and the MFCC results are very close. According to the initialization method, the NPC model allows to improve the results compared to the most used speech coding methods (LPCC and MFCC).

6. Conclusion

We have presented a new approach for feature extraction in speaker verification. The main idea is to extract speaker-dependent features. For that, we proposed a new feature extraction architecture. Traditionally, the feature extraction is carried out in a same way for all the speakers. In this paper, the feature extractor forms an integral part of the speaker model as much as the reference model.

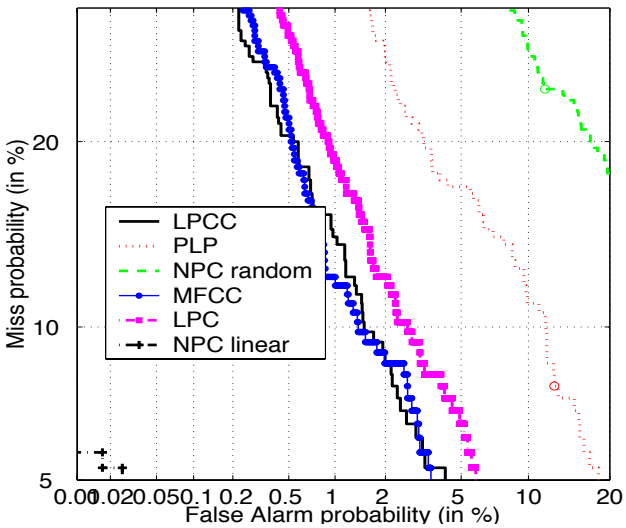


Figure 3: *DET curves for different speech coding methods.*

Table 1: *Experimental results for different speech coding methods.*

Speech coding method	Minimum DCF (%)
LPC	5.23
LPCC	4.16
MFCC	3.84
PLP	10.22
NPC (random initialization)	17.67
NPC (linear initialization)	0.16

For that, we have applied the Neural Predictive Coding (NPC) model. This model is an extension of the well-known LPC model by a connectionist model . We have also proposed a new initialization method for the NPC *coding phase*. This initialization is referred as a linear initialization but it differs from conventional neural networks initializations. The proposed initialization exploits speech knowledge. We used the LPC coding method for the initialization of the nonlinear coding model. By this way, one can see the NPC as a nonlinear feature extractor initialized by a linear model.

The NPC model with a linear initialization gives significative improvements (DCF=0.16%) compared to the random initialization (DCF=17.67%) or traditional methods (MFCC: 3.84%, LPCC: 4.16%). We used the Arithmetic-Harmonic Sphericity (AHS) method.

From the results, on can do some conclusions. First, the feature extraction plays an important role in the speaker recognition process. Secondly, non-linear feature extraction has to be carried out while taking into account the speech signal characteristics.

7. References

[1] C. Miyajima, H. Watanabe, K. Tokuda, T. Kitamura, and S. Katagiri, "A new approach to designing a feature extractor in speaker identification based on discriminative feature extraction," *Speech Communication*, vol. 35, pp. 203–218, 2001.

[2] G.-J. Jang, T.-W. Lee, and Y.-H. Oh, "Learning statistically efficient features for speaker recognition," *Neurocomputing*, vol. 49, pp. 329–348, 2002.

[3] I. Magrin-Chagnolleau, G. Gravier, M. Seck, O. Boeffard, R. Blouet, and F. Bimbot, "A further investigation on speech features for speaker characterization," in *Proceedings of ICSLP 2000*, October 2000, beijing, China.

[4] S. Katagiri, *Handbook of Neural Networks for Speech Processing*. Artech House eds., 2000.

[5] A. E. Biem, "Discriminative feature extraction applied to speech recognition," Ph.D. dissertation, Paris 6, 1997.

[6] B. Gas, J. Zarader, C. Chavy, and M. Chetouani, "Discriminant neural predictive coding applied to phoneme recognition," *Neurocomputing*, vol. 56, pp. 141–166, 2004.

[7] W. B. Kleijn, "Signal processing representations of speech," *IEICE Trans. Inf. and Syst.*, vol. E86-D, no. 3, pp. 359–376, March 2003.

[8] M. Chetouani, B. Gas, and J. Zarader, "Modular neural predictive coding for discriminative feature extraction," *Proc. of ICASSP*, vol. II, pp. 33–36, 2003.

[9] L. Wessels and E. Barnard, "Avoiding false local minima by proper initialization of connections," *IEEE Trans. on Neural Networks*, vol. 3, pp. 899–905, 1992.

[10] L. Bottou, "Une approche théorique de l'apprentissage connexioniste: Applications à la reconnaissance de la parole," Ph.D. dissertation, Université Paris XI (Orsay), 1991.

[11] T. L. Burrows, "Speech processing with linear and neural networks models," Ph.D. dissertation, Cambridge, 1996.

[12] B. Bimbot and L. Mathan, "Text-free speaker recognition using an arithmetic-harmonic sphericity measure," *EUROSPEECH*, pp. 169–172, 1991.

[13] M. Chetouani, M. Faundez-Zanuy, B. Gas, and J. Zarader, "A new nonlinear speaker parameterization algorithm for speaker identification," *Proc. of Odyssey: The speaker and language recognition workshop*, 2004.

[14] H. Hermansky, "Perceptual linear predictive (plp) analysis of speech," *The Journal of the Acoustical Society of America*, pp. 1738–1752, 1990.

[15] J. Ortega-Garcia and al., "Ahumada: a large speech corpus in spanish for speaker identification and verification," *ICASSP*, vol. 2, pp. 773–776, 1998.

[16] A. Martin, G. Doddington, T. Kamm, M. Ordowski, and M. Przybocki, "The det curve in assessment of detection performance," *EuroSpeech*, vol. 4, pp. 1895–1898, 1997.