

Classifieur à prototypes et codage neuro-prédicatif pour l'extraction non linéaire de caractéristiques en classification de phonèmes

M. Chetouani, B. Gas, J.L. Zarader

Laboratoire des Instruments et Systèmes d'Ile-De-France
Université Paris VI
BP 164, Tour 22-12 2ème étage
4 Place Jussieu, 75252 Paris Cedex 05, France
mohamed.chetouani@lis.jussieu.fr gas@ccr.jussieu.fr zarader@ccr.jussieu.fr

ABSTRACT

Speech recognition is a special field of pattern recognition. In order to improve the performances of the systems, one can opt for several ways and among them the design of a feature extractor. This paper presents a new nonlinear feature extraction method based on the Learning Vector Quantization (LVQ) and the Neural Predictive Coding (NPC). The key idea of this work is to design a feature extractor, the NPC, by the introduction of discriminant constraints provided by the LVQ classifier. The performances are estimated on a phoneme classification task by several methods : GMM, MLP, LVQ. The phonemes are extracted from the NTIMIT database. We make comparisons with the most used coding methods (LPC, MFCC, PLP).

1. INTRODUCTION

L'extraction de caractéristiques joue un rôle fondamental dans tout processus de reconnaissance des formes. En reconnaissance de la parole, la difficulté est encore plus grande car la production des signaux de parole présente des phénomènes non linéaires (acoustique, aérodynamique, physiologiques, etc. ...) [6]. Cependant, la plupart des méthodes utilisées (LPC, MFCC, etc. ...) sont basées sur l'hypothèse de linéarité du processus de production de la parole [14].

Depuis quelques années, de nombreux travaux ont porté sur le développement de méthodes d'extraction de caractéristiques non linéaires. On peut citer par exemple l'espace des phases [11] qui permet une modélisation de la dynamique des signaux de parole. L'intégration des connaissances sur la production ou bien la perception humaine est aussi un axe de recherche [8]. Le codage MFCC (Mel Frequency Cepstral Coding) ou le codage perceptif PLP (Perceptual Linear Predictive coding) [7] en sont des exemples.

La nouvelle méthode d'extraction de caractéristiques présentée dans ce papier fait partie des méthodes discriminantes. En effet, une des qualités requise pour les caractéristiques est la discrimination : pour deux phonèmes appartenant à des classes différentes, le vecteur code ou caractéristique généré doit être le plus discriminant possible. Les vecteurs de caractéristiques idéales doivent minimiser l'erreur de classification [5]. Dans ce cas, le critère de classification le plus adapté est le critère de Bayes et les meilleures caractéristiques associées sont les probabilités *a posteriori* [5]. Compte tenu de la difficulté d'atteindre cet objectif, à savoir d'estimer les probabilités *a poste-*

riori, plusieurs approches sont possibles. L'analyse discriminante linéaire (Linear Discriminant Analysis LDA) permet par le biais de mesures de séparabilité d'extraire des caractéristiques avec un plus grand pouvoir discriminant. Cette approche a été utilisée pour l'estimation de fonctions de projection spectrale en remplacement des projections en cosinus (Discrete Cosinus Transform DCT) dans le cas de la reconnaissance de la parole [13].

La minimisation de l'erreur de classification peut se faire par l'apprentissage simultané de l'extracteur de caractéristiques et du classifieur par le biais du concept DFE (Discriminative Feature Extraction) basé sur le critère MCE (Minimum Classification Error) [10]. Ce critère est minimisé par la descente de gradient probabilistique généralisé (Generalized Probabilistic Descent GPD).

Dans ce papier, nous présentons une nouvelle méthode d'extraction de caractéristiques basée sur la coopération entre le codeur neuro-prédicatif NPC [3] et le réseau de neurones LVQ (Learning Vector Quantization). Dans un premier temps, nous présenterons le codage NPC, puis nous montrerons comment ce modèle permet l'extraction non linéaire de caractéristiques via une coopération avec le LVQ. Ensuite, nous présenterons les conditions expérimentales et les performances du système sur différents groupes phonétiques. Finalement, nous donnerons des conclusions et quelques perspectives sur le travail proposé.

2. LE CODAGE NEURO-PRÉDICTIF

Le modèle neuro-prédicatif (Neural Predictive Coding NPC) est une extension du codage LPC. Comme dans le cas du codage LPC, il est basé sur un prédicteur de signaux de parole mais dans le cas du NPC, il est non-linéaire et il est réalisé par un réseau de neurones. Contrairement, au modèle proposé dans [3], le modèle présenté possède deux couches cachées.

Le modèle NPC est utilisé comme encodeur de signaux de parole, mais sa particularité est que seuls les poids de la seconde couche sont utilisés comme vecteur code ou caractéristique. Pour cela, la phase d'apprentissage est réalisée en deux temps. La phase de paramétrisation consiste en l'apprentissage de tous les poids du réseau par la minimisation de l'erreur de prédiction :

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

Avec y signal de parole, \hat{y} échantillon prédit et k index des échantillons et K le nombre d'échantillons.

A l'issue de cette phase, seuls les poids des premières couches $\mathbf{w}_{1,2}$ sont conservés, ils forment les paramètres du modèle NPC. Une fois paramétrisé, la seconde phase, appelée la phase de codage, consiste en l'estimation des poids de la seconde couche \mathbf{a} : le vecteur code ou caractéristique.

La fonction non linéaire de prédiction F est décomposée en deux fonctions $G_{\mathbf{w}_{1,2}}$ ($\mathbf{w}_{1,2}$ poids de la première et seconde couches) et $H_{\mathbf{a}}$ (\mathbf{a} poids de la couche de sortie) :

$$F_{\mathbf{w}_{1,2},\mathbf{a}}(\mathbf{y}_k) = H_{\mathbf{a}} \circ G_{\mathbf{w}_{1,2}}(\mathbf{y}_k) \quad (2)$$

Avec $\hat{y}_k = H_{\mathbf{a}}(\mathbf{z}_k)$ et $\mathbf{z}_k = G_{\mathbf{w}_{1,2}}(\mathbf{y}_k)$.

Les lois de modification des poids $\Delta \mathbf{a}^{Pred}$ et $\Delta \mathbf{w}_{1,2}^{Pred}$ sont proportionnelles au gradient des erreurs de prédiction $Q_{NPC} = \sum_i^M Q_i$, avec M le nombre de classes.

Un des objectifs du modèle NPC est l'extraction de caractéristiques discriminantes. Nous avons donc développé un critère discriminant adapté au modèle prédictif [2] mais qui n'intègre pas de lien direct avec la phase de classification. Dans la prochaine section, nous proposons une coopération entre le NPC et le classifieur à prototype LVQ (Learning Vector Quantization).

3. COOPÉRATION ENTRE LES MODÈLES NPC ET LVQ

Le classifieur à prototypes LVQ a été appliqué avec succès dans différent domaine comme la reconnaissance de l'écriture [12] ou bien la reconnaissance de la parole [15]. La procédure d'apprentissage consiste en l'ajustement des prototypes (représentants) afin de décrire les frontières optimales des classes.

L'apprentissage simultané des modèles NPC et LVQ peut se faire par le biais de la méthode DFE proposée par Biem [1]. Cette méthode permet d'introduire les discriminations imposées par le LVQ dans le modèle NPC. La méthode DFE considère l'extracteur de caractéristiques (NPC) et le classifieur (LVQ) comme un seul module décrit par $\Phi = (\mathbf{a}, \mathbf{m})$.

3.1. Extraction de caractéristiques discriminantes

La mise en œuvre de la méthode d'extraction de caractéristiques discriminantes DFE (Discriminative Feature Extraction) s'effectue en plusieurs étapes. Dans un premier temps, il faut définir une fonction discriminante décrivant la processus extracteur de caractéristiques et le classifieur. Dans le cas du LVQ, la fonction discriminante est la valeur négative du minimum de la distance entre le vecteur d'entrée (le vecteur caractéristique) et le prototype correct.

$$g_i(\mathbf{a}) = \min_{\tau} \|\mathbf{a} - \mathbf{m}_{i,\tau}\|^2 = \min_{\tau} d(\mathbf{a}, \mathbf{m}_{i,\tau}) \quad (3)$$

où $d(\mathbf{a}, \mathbf{m}_{i,\tau})$ est la distance Euclidienne entre le vecteur d'entrée \mathbf{a} (vecteur caractéristique) et un prototype $\mathbf{m}_{i,\tau}$ de la classe C_i .

Par la suite, nous définissons une mesure de mauvaise classification :

$$\mu_i(\mathbf{a}) = g_i(\mathbf{a}) + \left[\frac{1}{M-1} \sum_{j \neq i} g_j(\mathbf{a}) \right]^{\frac{1}{\psi}} \quad (4)$$

où ψ est un nombre positif. Pour une grande valeur ψ , la mesure de mauvaise classification devient :

$$\mu_i(\mathbf{a}) = g_i(\mathbf{a}) + \bar{g}_i(\mathbf{a}) \quad (5)$$

$\bar{g}_i(\mathbf{a})$ est la fonction discriminante concurrente. Ceci revient à ne considérer que le premier prototype concurrent de la classe incorrecte [15] :

$$\bar{g}_i(\mathbf{a}) = \max_{j \neq i} g_j(\mathbf{a}) \quad (6)$$

La mesure de mauvaise classification $\mu_i(\mathbf{a})$ (4) doit être positive quand \mathbf{a} n'est pas bien classé et négative si ce n'est pas le cas :

$$\mu_r(\mathbf{a}) = d(\mathbf{a}, \mathbf{m}_{i,\tau}) - d(\mathbf{a}, \mathbf{m}_{j,v}) \quad (7)$$

où $\mathbf{m}_{i,\tau}$ est le plus proche prototype de la véritable classe alors que $\mathbf{m}_{j,v}$ est le plus proche prototype de la classe incorrecte.

La prochaine étape consiste en la définition du critère de minimisation de l'erreur de classification (MCE) reflétant les erreurs de classification :

$$l_i(\mathbf{a}) = l_i(\mu_i) = \frac{1}{1 + e^{-\zeta \mu_i}} \quad (8)$$

Le risque empirique total du critère MCE est le suivant :

$$L(\mathbf{a}, \mu) = \sum_{n=1}^N \sum_{i=1}^M l_i(\mathbf{a}_n) \delta_{C(\mathbf{a}_n) \neq i} \quad (9)$$

où $C(\mathbf{a}_n)$ est la classe d'appartenance du vecteur caractéristique \mathbf{a}_n et δ est le symbole de Kronecker qui vaut 1 lorsque $C(\mathbf{a}_n) = i$. N est le nombre de fenêtres d'analyse et M le nombre de classes.

La descente de gradient probabilistique généralisé GPD est appliquée pour la mise à jour des paramètres $\Phi = (\mathbf{a}, \mathbf{m})$:

$$\begin{aligned} \mathbf{a}_n &= \mathbf{a}_n - \beta(t) \frac{\partial l_i(\mathbf{a}_n)}{\partial \mathbf{a}_n} \\ \mathbf{m}_{i,\tau} &= \mathbf{m}_{i,\tau} + \alpha(t) \frac{\partial l_i(\mathbf{a}_n)}{\partial \mathbf{m}_{i,\tau}} \\ \mathbf{m}_{j,v} &= \mathbf{m}_{j,v} - \alpha(t) \frac{\partial l_i(\mathbf{a}_n)}{\partial \mathbf{m}_{j,v}} \end{aligned} \quad (10)$$

où $\alpha(t)$ et $\beta(t)$ sont les pas d'apprentissage du classifieur LVQ et du modèle NPC. Les pas d'apprentissage sont des fonctions décroissantes du nombre d'itération t .

Les lois d'adaptation des prototypes du LVQ sont données par :

$$\begin{aligned} \mathbf{m}_{i,\tau} &= \mathbf{m}_{i,\tau} + 2\alpha(t) l_i(\mathbf{a}_n) (1 - l_i(\mathbf{a}_n)) (\mathbf{a}_n - \mathbf{m}_{i,\tau}) \\ \mathbf{m}_{j,v} &= \mathbf{m}_{j,v} - 2\alpha(t) l_i(\mathbf{a}_n) (1 - l_i(\mathbf{a}_n)) (\mathbf{a}_n - \mathbf{m}_{j,v}) \end{aligned} \quad (11)$$

Et pour le modèle NPC, les vecteurs caractéristiques \mathbf{a}_n évoluent selon la loi suivante :

$$\Delta \mathbf{a}_n^{MCE} = 2\beta(t) l_i(\mathbf{a}_n) (1 - l_i(\mathbf{a}_n)) (\mathbf{m}_{i,\tau} - \mathbf{m}_{j,v}) \quad (12)$$

On peut remarquer que les caractéristiques évoluent en fonction de la distance entre les deux prototypes les plus proches, l'un de la véritable classe l'autre de la classe incorrecte. Elles évoluent donc dans le sens de maximisation la séparabilité entre ces deux classes.

Ces modifications doivent être associées aux modifications classiques du modèle NPC qui sont en fonction de l'erreur de prédiction : $\Delta \mathbf{w}_{1,2}^{Pred}$ et $\Delta \mathbf{a}^{Pred}$.

3.2. Coopération

L'objectif de la coopération entre les deux modèles NPC et LVQ est l'introduction de contraintes discriminantes optimales dans la phase de paramétrisation du NPC. Par exemple, on pourrait opter pour une minimisation sous contrainte comme dans [4] où l'apprentissage simultané des classifieurs est effectué par le biais du formalisme lagrangien. Ici, nous choisissons une autre approche. Les deux procédés de minimisation sont modérés par le biais d'un coefficient θ .

La modification résultante des vecteurs caractéristiques \mathbf{a}_n est :

$$\Delta \mathbf{a} = \theta \Delta \mathbf{a}^{Pred} + (1 - \theta) \Delta \mathbf{a}^{MCE} \quad (13)$$

La seconde étape de la coopération consiste à modifier les poids des premières couches pour maximiser la séparabilité des classes. Cependant, la relation entre les poids des premières couches $\mathbf{w}_{1,2}$ et le critère MCE (8) n'est pas aussi direct que dans le cas des poids de la couche sortie, c'est-à-dire les vecteurs caractéristiques. En considérant l'objectif du modèle NPC dans la coopération, cela revient à rapprocher les caractéristiques de leurs prototypes adéquates et de les éloigner des prototypes incorrects. En d'autres mots, le vecteur caractéristique $\mathbf{a}_{i,n}$ produit par le modèle NPC pour la fenêtre d'analyse $\mathbf{y}_{i,n}$ (appartenant à la classe C_i) doit être proche de l'un des prototypes $\mathbf{m}_{i,\tau}$.

Pour cela, on introduit une nouvelle étape dans le modèle NPC. Pour la fenêtre $\mathbf{y}_{i,n}$, on détermine les deux modifications nécessaires pour :

- Rapprochement des caractéristiques vers le prototype $\mathbf{m}_{i,\tau}$: minimisation de l'erreur de prédiction sous la contrainte que la couche de sortie soit $\mathbf{m}_{i,\tau}$. On obtient la modification des premières couches $\Delta \mathbf{w}_{1,2}^{mod}$.
- Éloignement des caractéristiques du prototype $\mathbf{m}_{j,v}$: maximisation de l'erreur de prédiction sous la contrainte que la couche de sortie soit $\mathbf{m}_{j,v}$. On obtient la modification des premières couches $\Delta \mathbf{w}_{1,2}^{disc}$.

Lors de ces deux procédés, on estime les modifications nécessaires pour maximiser la séparabilité des classes.

La loi de modification des premières couches est une modulation de ces deux effets :

$$\Delta \mathbf{w}_{1,2} = \theta \Delta \mathbf{w}_{1,2}^{mod} + (1 - \theta) \Delta \mathbf{w}_{1,2}^{disc} \quad (14)$$

On peut remarquer que cette loi de modification n'intègre pas la modification du modèle NPC $\Delta \mathbf{w}_{1,2}^{Pred}$. En effet, cette modification n'est plus utile car la contribution $\Delta \mathbf{w}_{1,2}^{mod}$ permet de tenir compte de la partie modélisation nécessaire au processus LVQ-NPC.

4. CONDITIONS EXPÉRIMENTALES

4.1. Base de données

La base de donnée NTIMIT [9] est utilisée dans les expériences de ce travail. Cette base est composée de 10 phrases prononcées par 630 locuteurs de 8 régions des États-Unis. Une de ces particularités est que l'on se place dans un contexte de reconnaissance en qualité téléphonique.

La validation du modèle LVQ-NPC se fait sur les voyelles antérieures (/ih/, /ey/, /eh/, /ae/) et les occlusives voisées (/b/, /d/, /g/) et non voisées (/p/, /t/, /k/) de la première région DR1. Ce choix peut se justifier par le fait que ces phonèmes sont connus pour être difficile à classer. Nous utilisons la configuration proposée dans la base NTIMIT pour les ensembles d'apprentissage (24 hommes et 14 femmes) et de test (7 hommes et 4 femmes). Le tableau 1 décrit le nombre de phonèmes utilisés.

TAB. 1: Composition des bases d'apprentissage et de test

Phonème	Apprentissage	Test
ih	316	104
ey	189	54
eh	297	83
ae	346	79
b	183	59
d	300	90
g	167	44
p	215	48
t	320	93
k	390	103

4.2. Classification

Le travail proposé consiste en l'évaluation d'une nouvelle méthode d'extraction de caractéristiques : le LVQ-NPC. Par conséquent, nous réalisons des comparaisons avec les méthodes les plus utilisées : le codage linéaire prédictif LPC, le codage MFCC (Mel Frequency Cepstral Coding), le codage perceptuel PLP (Perceptual Linear Predictive (PLP)) [7]. Pour l'ensemble des méthodes, la dimension du vecteur code est fixée à 12.

Une manière efficace pour évaluer les méthodes est la classification. La classification est faite trame par trame sans contexte. De plus, elle doit être effectuée par plusieurs types de classifieurs afin de réellement mesurer les capacités discriminantes de chacune des méthodes de codage.

Mélanges de gaussiennes (Gaussians Mixture Models GMM) Ce modèle est basé sur l'estimation des densités de probabilités de chacune de classes. Les GMMs sont entraînés par le biais de l'algorithme EM (Estimation-Maximization) avec l'hypothèse de matrices de covariance diagonales. Ce classifieur est sensible à l'initialisation, les paramètres initiaux sont initialisés par le biais de l'algorithme des k -moyennes (10 itérations) avec $k = 16$. La classification se fait selon le critère du maximum de vraisemblance.

Classification par prototypes LVQ Le modèle LVQ (Learning Vector Quantization) est un classifieur à prototypes. L'apprentissage et le test se font par le biais de la distance euclidienne. Cette méthode est également sensible à l'initialisation, nous utilisons donc également les k -moyennes avec $k = 50$. Il est à noter que ce classifieur est différent de celui entraîné pendant la phase de coopération LVQ-NPC.

Réseaux de neurones Les réseaux de neurones sont des modèles à base de fonctions discriminantes non linéaires. Le modèle possède une couche cachée de 10 neurones et l'apprentissage se fait par le biais de l'algorithme de Levenberg-Marquardt.

LVQ-NPC Le nombre de prototypes pour le modèle LVQ-NPC est fixé à 25. Le paramètre modérateur θ suit la loi décroissante suivante :

$$\theta(t) = \theta_0 \left(1 - \frac{t}{N}\right) \quad (15)$$

où N est le nombre d'itération. La valeur de θ_0 diffère selon le groupe phonétique traité : 0.6 pour les voyelles, 0.7 pour les occlusives voisées et non voisées.

Une telle loi d'évolution montre qu'il est nécessaire de commencer par modéliser les classes puis à augmenter au fur et à mesure la discrimination.

5. CLASSIFICATION DE PHONÈMES

Dans cette section, nous présentons les résultats en classification de phonèmes. Les taux de classification présentés sont ceux de la base de test.

Le tableau 2 présente les taux de classification pour les voyelles avec les différentes méthodes de codage et de classification. La première remarque que l'on peut faire est que la méthode de codage proposée, le LVQ-NPC, permet une augmentation des scores de classification. L'introduction d'une modélisation non linéaire par réseaux de neurones et la discrimination permet un gain de 4% par rapport au codage MFCC.

TAB. 2: Taux de classification pour les voyelles

	LPC	MFCC	PLP	LVQ-NPC
GMM	34.94	47.07	43.05	51.32
LVQ	35.27	42.55	39.71	50.89
MLP	40.30	45.39	42.57	50.49

Du fait que les occlusives soient des phonèmes voisés, nous retrouvons un comportement similaire pour le LVQ-NPC (cf. tab. 3). La classification des occlusives non voi-

TAB. 3: Taux de classification pour les occlusives voisées

	LPC	MFCC	PLP	LVQ-NPC
GMM	53.52	58.05	57	65.22
LVQ	51.01	56.26	56.23	63.67
MLP	54.58	55.19	57.07	62.73

sées est intéressante car ce sont des phonèmes non voisés, ce qui dans un premier temps semble être pénalisant pour les modèles prédictifs comme LVQ-NPC. Cependant, la discrimination permet de surmonter ce problème et d'obtenir de meilleurs résultats quelque soit la méthode de classification (cf. tab. 4).

TAB. 4: Taux de classification pour les occlusives non voisées

	LPC	MFCC	PLP	LVQ-NPC
GMM	43.21	49.52	46.12	51.15
LVQ	42.93	47.16	45.83	49.76
MLP	44.49	47.67	45.52	47.16

6. CONCLUSIONS

Nous avons présenté une nouvelle méthode de codage : le LVQ-NPC qui est basée sur l'apprentissage simultané

d'un extracteur de caractéristiques et un classifieur. Le modèle est adapté au traitement de la parole qui semble nécessiter une modélisation non linéaire mais aussi une discrimination adéquate. Les résultats expérimentaux en classification de phonèmes issus de la base NTIMIT montre l'intérêt de la méthode. Les performances sont de plus à évaluer par des classifieurs aux comportements différents. Nos prochains travaux consistent en une validation sur un plus grand nombre de phonèmes mais aussi sur la coopération avec d'autres classifieurs.

RÉFÉRENCES

- [1] A. E. Biem. *Extraction de paramètres discriminatifs : Application à la reconnaissance de la parole*. PhD thesis, Paris 6, 1997.
- [2] M. Chetouani, B. Gas, and J.L. Zarader. Maximization of the modelisation error ratio for neural predictive coding. *Proc. of NOLISP*, 2003.
- [3] M. Chetouani, B. Gas, J.L. Zarader, and C. Chavy. Extraction de caractéristiques par codage neuro-prédictif. *Proc. of JEP*, pages 85–88, 2002.
- [4] X. Driancourt. *Optimisation par descente de gradient stochastique de systèmes modulaires combinant réseaux de neurones et programmation dynamique*. PhD thesis, Université Paris XI Orsay, 1994.
- [5] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification*. Wiley-Interscience Publication, 2001.
- [6] M. Faundez. *Modelado predictivo no lineal de la señal de voz aplicado a codificación y reconocimiento de locutor*. PhD thesis, Universitat politècnica de catalunya (España), 1998.
- [7] H. Hermansky. Perceptual linear predictive (plp) analysis of speech. *The Journal of the Acoustical Society of America*, pages 1738–1752, 1990.
- [8] H. Hermansky. Should recognizers have ears? *Proc. ESCA Tutorial and Research Workshop on Robust Speech Recognition for Unknown Communication Channels*, pages 1–10, 1997.
- [9] C. Jankowski, A. Kalyanswamy, S. Basson, and J. Spitz. Ntimit : A phonetically balanced, continuous speech, telephone bandwidth speech database. *ICASSP*, 1 :109–112, 1990.
- [10] S. Katagiri. *Handbook of Neural Networks for Speech Processing*. Artech House eds., 2000.
- [11] A. C. Lindgren, M. T. Johnson, and R. J. Povinelli. Speech recognition using recognition using reconstructed phase space features. *ICASSP*, 1 :61–63, 2003.
- [12] C.-L. Liu and M. Nakagawa. Evaluation of prototype learning algorithms for nearest-neighbor classifier in application to handwritten character recognition. *Pattern Recognition*, 34 :601–615, 2001.
- [13] N. Malayath and H. Hermansky. Data-driven spectral basis functions for automatic speech recognition. *Speech Communication*, 70 :449–466, 2003.
- [14] J. Mariani. *Analyse, synthèse et codage de la parole-Traitement automatique du langage parlé 1*. Hermès Science Pulications, 2002.
- [15] E. McDermott. *Discriminative Training for Speech Recognition*. PhD thesis, Waseda University (Japan), 1997.