Modèles acoustiques pour la reconnaissance du locuteur

Anthony Larcher

Le Mans Université

ACADÉMIE DES PAYS DE LOIRE
LE MANS UNIVERSITÉ

Habilitation à Diriger des Recherches

présentée à l'Université du Mans

SPÉCIALITÉ : Informatique

École Doctorale MathSTIC
Unité de recherche EA 4023 LIUM

Modèles acoustiques pour la reconnaissance du locuteur

par

Anthony LARCHER

Soutenue publiquement le 7 décembre 2018 devant un jury composé de :

M.	Claude BARRAS	Maître de Conférences HDR, LIMSI, Paris	Rapporteur
M.	Guillaume GRAVIER	Directeur de recherches, IRISA/CNRS, Rennes	Rapporteur
M.	Jean-François BONASTRE	Professeur, LIA, Avignon	Examinateur
M.	Denis JOUVET	Professeur, LORIA, Nancy	Examinateur
M.	Sylvain MEIGNIER	Professeur, LIUM, Le Mans	Examinateur

lium

Sommaire

Remerciements

Introduction

Star Treck, Demolition Man, Alien Resurection, I Robot, Judge Dredd, la reconnaissance du locuteur peuple les œuvres de science-fiction depuis plus de 60 ans. Pourtant, la vérification du locuteur est encore loin de combler toutes nos attentes et d'atteindre les performances démontrées dans ces fictions. La vérification automatique du locuteur, qui est au cœur de ce manuscrit, consiste pour une machine à donner une réponse binaire à la question suivante :

Étant donné une personne connue et un échantillon vocal, l'échantillon vocal a-t-il été produit par cette personne ?

Depuis les années 1990, les progrès réalisés dans ce domaine ont permis de développer des systèmes automatiques utilisables pour des applications qui ne nécessitent pas un niveau de sécurité critique. La robustesse des systèmes au bruit ambiant, au canal de transmission et au manque de données a été grandement améliorée.

Depuis 15 ans, mes recherches se placent dans le cadre de la vérification du locuteur la plus générique (indépendante du texte), mais surtout de sa version contrainte pour laquelle l'utilisateur doit prononcer un texte déterminé à l'avance : la vérification du locuteur dépendante du texte. La contrainte ergonomique imposée à l'utilisateur se justifie par les performances des technologies actuelles : dans le cas où les échantillons vocaux collectés sont de courte durée (quelques secondes), contraindre le texte prononcé permet d'améliorer grandement les performances en réduisant la variabilité entre l'échantillon de référence appelé échantillon d'enrôlement et l'échantillon à comparer, appelé échantillon de test.

Ce document décrit ma vision des principales innovations du domaine depuis l'avènement des modèles Gaussiens en 1995 jusqu'à l'apparition de *x*-vecteurs en 2016, en replaçant les travaux que j'ai menés au LIA [1], à I^{2}R [2] et enfin au LIUM [3], dans leur contexte (cf. figure 1).

Depuis mon intégration au sein de l'équipe LST du LIUM, mes recherches se sont élargies à l'adaptation aux locuteurs des systèmes de reconnaissance de la parole dans le cadre de la thèse de Natalia Tomashenko que j'ai co-encadré avec Yannick Estève et Yuri Khokhlov [Tomashenko et al., 2016a,b]. Alors que la reconnaissance du locuteur dépendante du texte tend à intégrer une information lexicale pour vérifier l'identité d'un locuteur, les travaux de Natalia Tomashenko tendent à intégrer l'information liée au locuteur dans des systèmes de reconnaissance de la parole. On y retrouve ainsi le locuteur, l'information phonétique et les durées très courtes.

1. Laboratoire Informatique d'Avignon, FRANCE
2. Institute for Infocomm Research, SINGAPOUR
3. Laboratoire Informatique de l'Université du Mans, FRANCE

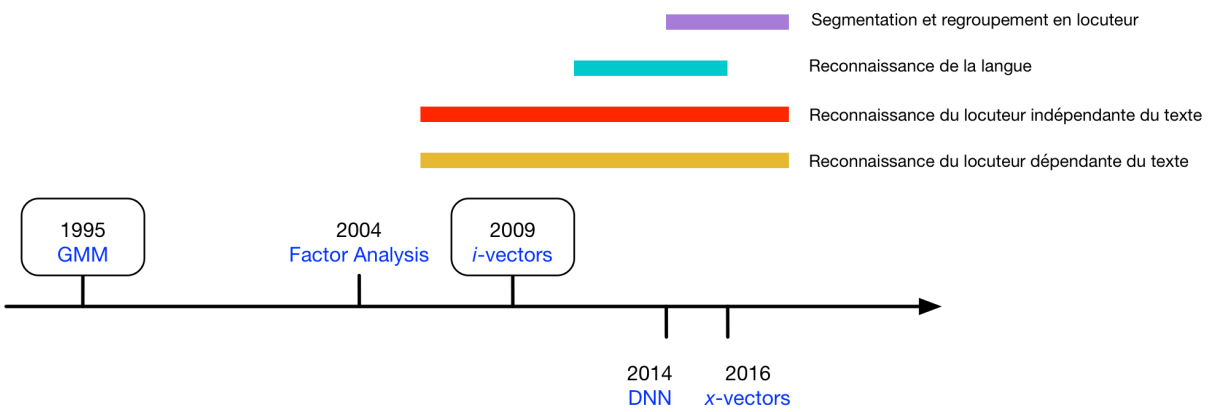

Figure 1 – Mes recherches en traitement de la parole dans le contexte des avancées principales de la reconnaissance du locuteur.

Mon intérêt pour la modélisation acoustique m'a également amené à travailler dans le domaine de la reconnaissance de la langue [Lee et al., 2016, 2011] et de la détection de parole ,dans le cadre de la thèse de Florent Desnous que je co-encadre avec Sylvain Meignier [Desnous et al., 2018]. Les travaux ne sont pas abordés dans ce document qui est centré sur la modélisation acoustique du locuteur pour la vérification du locuteur.

Dans le cadre de la thèse de Gaël Le Lan, co-encadré avec Sylvain Meignier et Delphine Charlet, nous avons étudié l'apprentissage non supervisé d'un système de segmentation et regroupement en locuteurs de collections [Le Lan, 2017; Le Lan et al., 2016, 2017, 2018, 2016a,b]. La caractérisation des locuteurs joue un rôle important pour cette tâche qui traite encore une fois de segments courts. La thèse de Gaël Le Lan a été l'occasion d'effectuer des travaux sur les systèmes neuronaux pour la vérification du locuteur et de les évaluer lors des évaluations NIST Lee et al. [2017]. Ces travaux sur l'adaptation de systèmes non supervisés m'ont permis d'obtenir un financement pour un projet européen dans lequel nous poursuivrons les travaux entrepris par Gaël.

L'organisation de ce manuscrit suit la logique d'évolution de mes travaux. Après avoir introduit les principales innovations du domaine, je présente mes travaux en vérification du locuteur indépendante du texte dans le contexte du paradigme GMM-UBM utilisant le *Factor Analyser* [Larcher et al., 2010a] ou du paradigme de variabilité totale *i*-vecteurs et du modèle PLDA [Jiang et al., 2012; Larcher et al., 2012b; Lee et al., 2013] qui ont toujours été accompagnés de participations aux campagnes internationales du NIST [Hautamaki et al., 2012; Hautomaki et al., 2011; Lee et al., 2017; Li et al., 2012; Saeidi et al., 2013].

Je décris ensuite mes travaux en reconnaissance du locuteur dépendante du texte. Mes recherches sont focalisées sur la prise en compte du contenu phonétique et de la structure temporelle dans la modélisation acoustique, dans le paradigme des modèles GMMs et dans le paradigme de variabilité totale (*i*-vecteurs). Ces travaux ont en grande partie été fi-

nancés par des projets académiques ou industriels. Tout d'abord le projet ANR *BIOBIMO* [Larcher et al., 2008a, 2010, 2013, 2008b,c,d, 2010b] avec Jean-François Bonastre, Corinne Fredouille, Christophe Lévy et Driss Matrouf, puis du projet domotique *HOME2015* [Larcher et al., 2012a,?, 2013, 2014c] et de projets industriels [Larcher et al., 2013a,b, 2014a,b,d] avec Lee Kong Aik, Li Haizhou et Ma Bin.

La dernière partie de ce document présente mes perspectives de recherche à court et moyen terme ainsi que les ambitions plus larges qui guident mes démarches

Première partie

Les modèles acoustiques et leurs utilisations

Cette partie introduit les modèles acoustiques utilisés en reconnaissance automatique du locuteur ces 20 dernières années. L'objectif est, ici, de souligner les hypothèses sous-jacentes à ces modèles et les similitudes existant entre les approches, notamment la non-prise en compte de la structure temporelle de la parole pour la vérification du locuteur qui, si elle peut se justifier lorsqu'on considère des durées de parole de quelques dizaines de secondes, nuit à la modélisation dans le cas de durée plus courtes comme nous le verrons dans partie III

CHAPITRE 1

Observations et modélisations acoustiques

1.1. Discrétisation et modélisation d'une source

Les différentes tâches du traitement automatique de la parole consistent à caractériser des phénomènes acoustiques en fonction de la source qui les a produits. La notion de source peut varier significativement, mais reflète un état acoustique stable plus ou moins long. Les sources qui nous intéressent ici sont les suivantes, associées aux tâches pour lesquelles elles interviennent :

- un locuteur produisant de la parole dont le contenu phonétique est indéterminé (reconnaissance du locuteur indépendante du texte, segmentation en locuteur) ;
- un locuteur produisant un certain phonème ou plus largement un contenu phonétique déterminé (reconnaissance du locuteur dépendante du texte, reconnaissance de la parole adaptée au locuteur) ;
- une source produisant de la parole (détection de parole/non-parole) ;
- une source produisant un signal ne contenant pas de parole : silence, bruit, musique (détection de parole/non-parole) ;
- une source produisant un phonème déterminé (reconnaissance de la parole, reconnaissance du locuteur dépendante du texte).

La caractérisation des phénomènes acoustiques est rendue difficile par les interactions complexes qui lient ces phénomènes et par leur durée très variable. Un locuteur peut parler pendant des secondes ou des dizaines de secondes sans interruption, la durée d'un

phonème est de l'ordre de la seconde, dans une conversation, les silences et les temps de parole des différents locuteurs varient énormément selon le contexte et les locuteurs. Un segment de parole comporte de nombreux phonèmes prononcés par un même locuteur. Les informations liées aux phonèmes, au contenu lexical, à la langue, au locuteur ou à la nature même de la parole se mélangent au sein d'un même signal et c'est le rôle de la modélisation acoustique de caractériser les différentes sources.

Les deux principaux types de modèles utilisés en traitement de la parole et dans mes travaux seront présentés dans les chapitres suivant :

— les modèles Gaussiens
— les réseaux de neurones

Avant de décrire ces modèles, la section suivante fournit un aperçu des méthodes de paramétrisation acoustique qui constituent la première étape du traitement de la parole.

1.2. Paramètres acoustiques

Dans leur grande majorité, les systèmes de traitement automatique de la parole utilisent une analyse spectrale en bandes de fréquences pour obtenir une représentation temps/fréquence à court terme du signal audio [Kinnunen et Li, 2010; Lawson et al., 2011]. Le signal de parole est considéré comme stationnaire sur des fenêtres d'environ 30 millisecondes. Des vecteurs de paramètres sont extraits sur une fenêtre glissante dont la longueur varie généralement entre 20 et 30 millisecondes à une fréquence de 100 Herts (soit un vecteur toutes les 10 millisecondes). De nombreux traitements ont été proposés dans la littérature afin d'optimiser plusieurs critères dont :

— la décorrélation des coefficients (pour favoriser la compression de l'information)
— la robustesse au bruit ou aux dégradations
— la représentation des caractéristiques propres aux voix humaines

1.2.1 Mel-Frequency Cepstral Coefficients (MFCC) et Filter-Bank (FB)

Les MFCC (Mel- Frequency Cepstral Coefficients) dont une description détaillée est donnée dans [Rabiner et Juang, 1993] restent largement utilisés pour les tâches de reconnaissance de la parole, de la langue, du locuteur ou pour la segmentation en locuteur, malgré de très nombreux travaux proposant d'autres méthodes. Le processus d'extraction des MFCC est décrit par la figure 1.1. Le signal de parole est analysé localement à l'aide d'un fenêtrage temporel (souvent Hanning ou Hamming afin de réduire les effets

de bord qu'occasionne une fenêtre rectangulaire). La longueur de la fenêtre glissante (20-30 millisecondes) est choisie pour respecter l'hypothèse de stationnarité. Le décalage des fenêtres temporelles utilisées pour extraire deux segments consécutifs de signal est choisi de manière à ce que ces fenêtres se recouvrent en partie. Au segment de signal prélevé est ensuite appliquée une transformée de Fourier rapide (Fast Fourier Transform - FFT). Le module de son spectre est filtré par un banc de filtres qui permet de réduire la dimension du vecteur spectral en calculant la moyenne du spectre sur la bande de fréquence correspondant à chacun des filtres. Les fréquences centrales de chaque filtre sont fixées par l'échelle de MEL. Le logarithme de ces valeurs est calculé et multiplié par 20 pour obtenir l'enveloppe spectrale en décibels. Les coefficients acoustiques obtenus à cette étape peuvent être directement utilisés pour le traitement de la parole, on parlera dans ce cas de « banc de filtres logarithmique » ou *log filter-bank* (FB) en anglais. La dernière étape de la paramétrisation consiste à appliquer une transformée en cosinus discrète (DCT) d'où résultent les coefficients cepstraux (MFCC). La transformée en cosinus discrets est utilisée ici pour sa capacité à décorréler les données. Une information dynamique est ajoutée

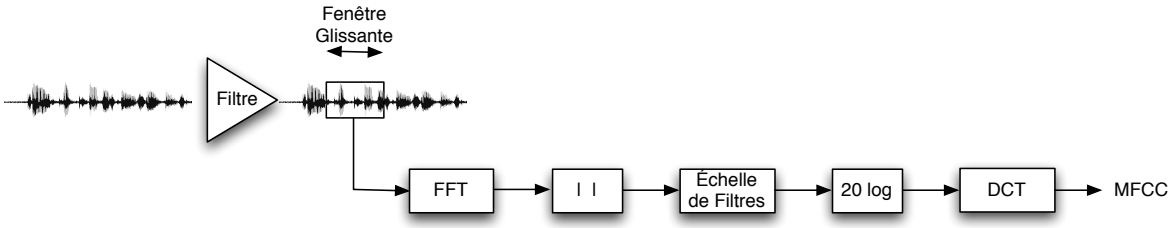

Figure 1.1 – Extraction des paramètres MFCC.

à ces coefficients en les concaténant à leurs dérivées temporelles premières et secondes [Fredouille, 2000; Furui, 1981].

1.2.2 Paramètres Bottleneck (BNF)

Les réseaux de neurones sont utilisés depuis longtemps pour extraire de l'information du signal de parole pour la tâche de reconnaissance de la parole [Hermansky et Sharma, 1999]. Cette paramétrisation a bénéficié des avancées récentes des réseaux de neurones profonds et il est maintenant courant d'utiliser des paramètres *bottleneck* (BNF pour BottleNeck Features) en entrée des systèmes de reconnaissance du locuteur ou des systèmes de segmentation en locuteurs [Lozano-Diez et al., 2016; McLaren et al., 2016, 2015]. Les meilleures performances étant généralement obtenues en concaténant ces BNF avec des MFCC [Alam et al., 2016]. L'extraction des BNF comporte deux étapes. Dans une première étape, les paramètres FB ou MFCC sont extraits du signal de parole. Les vecteurs successifs sont concaténés afin de fournir au réseau de neurones un contexte temporel plus

important (de l'ordre de 100 ou 200ms). Un fenêtrage de Hamming est éventuellement appliqué afin de renforcer l'importance du vecteur central. Dans un deuxième temps, ces vecteurs de grande dimension sont fournis en entrée d'un réseau de neurones qui a pour tâche de classifier ces vecteurs selon leur contexte phonétique (sénones) ou, plus rarement, selon le locuteur. Le réseau de neurones utilisé comporte généralement entre 3 et 7 couches cachées, dont une de dimension réduite (goulot ou *bottleneck*). Une fois entraînée pour la tâche choisie, le réseau est tronqué pour collecter la sortie de la couche *bottleneck* qui peut alors être utilisée comme vecteur de paramètres acoustiques.

Les BNF offrent l'avantage de prendre en compte un contexte temporel élargi grâce à la concaténation des vecteurs acoustiques en entrée du réseau de neurones. Ainsi il n'est pas nécessaire d'ajouter les dérivées premières ou secondes des paramètres acoustiques comme c'est le cas pour les MFCC ou les FB.

CHAPITRE 2

Distributions Gaussiennes et mixtures

Depuis les années 1990, les systèmes de traitement automatique de la parole reposent sur la description des sources acoustiques par des sources discrètes multidimensionnelles [Bimbot et al., 1995; Huang et al., 1990; Rabiner, 1989; Reynolds et Rose, 1995]. Les observations acoustiques provenant de ces sources sont les vecteurs de coefficients acoustiques décrits dans la section 1.2.

Les systèmes de traitement automatiques de la parole comparent une ou plusieurs sources acoustiques (c.-à-d. l'ensemble de ses observations) avec une ou plusieurs observations pour déterminer si ces observations ont été produites par l'une des sources connues. La comparaison entre séquences discrètes n'étant pas aisée, les sources connues sont modélisées par la distribution statistique de leurs observations. Modéliser une source acoustique par une distribution statistique continue offre quatre avantages principaux :

— un nombre limité de paramètres suffit à représenter la source (il n'est pas nécessaire de mémoriser toutes les observations qu'elle a produites) ;
— si la distribution statistique est bien choisie, l'utilisation de ce modèle nécessite un coût de calcul limité ;
— une distribution continue sur l'espace des paramètres acoustiques permet de comparer facilement n'importe quelle observation à cette distribution (c'est un effet d'interpolation) ;
— de ce dernier point découle le fait qu'on peut comparer deux modèles de sources ou une source et une séquence d'observations.

Les distributions Gaussiennes, qui offrent un modèle simple, ont été utilisées massivement durant ces vingt dernières années pour modéliser différentes sources acoustiques.

Du modèle mono-Gaussien proposé Bimbot et al. [1995] à l'espace de variabilité totale [Dehak et al., 2011a], ce chapitre retrace une partie des évolutions proposées.

2.1. Le modèle mono-gaussien

Un ensemble de N observations $\{\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t, ..., \mathbf{x}_N\}$ de dimension d est modélisé par une distribution Gaussienne unique. La distribution Gaussienne est définie par son vecteur moyen, $\boldsymbol{\mu}$, de dimension d et sa matrice de covariance, Σ, de dimension $d \times d$. La densité de probabilité d'une distribution Gaussienne est donnée par :

$$Pr(\mathbf{x}) = \frac{1}{(2\Pi)^{\frac{d}{2}}|\Sigma|^{\frac{1}{2}}} exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x}-\boldsymbol{\mu})\right] \tag{2.1}$$

et sera notée :

$$Pr(\mathbf{x}) = \mathcal{N}_x(\boldsymbol{\mu}, \Sigma) \tag{2.2}$$

Ce modèle Gaussien présente une forme générique, dont la matrice de covariance est pleine, et des simplifications qui sont décrites dans la suite de ce chapitre ainsi que dans le chapitre suivant.

2.1.1 Distribution Gaussienne à variance pleine

Description du modèle

La matrice Σ, présentée dans l'équation 2.1, est symétrique définie positive. Dans le cas générique, elle est pleine, définie par $\frac{d\times(d-1)}{2}$ coefficients. La figure 2.1 d illustre une distribution Gaussienne, de dimension deux, à matrice de covariance pleine.

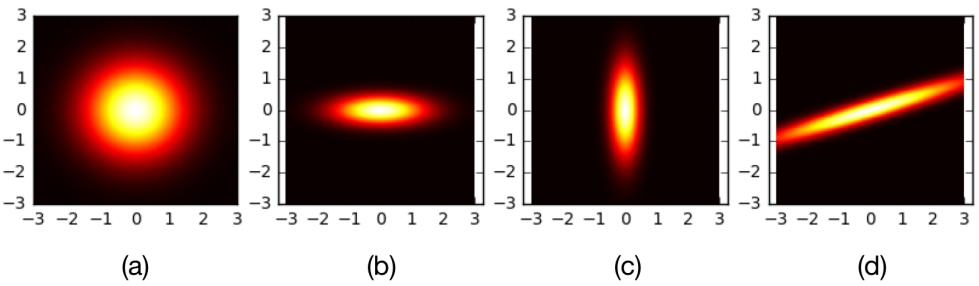

Figure 2.1 – Distributions Gaussiennes à covariance sphérique (a), diagonale (b et c) et pleine (d) .

Les $\frac{d\times(d-1)}{2} + d$ paramètres qui définissent une telle distribution sont estimés directement, grâce au critère de maximum de vraisemblance, de la façon suivante :

$$\begin{cases} \boldsymbol{\mu} = \frac{1}{N}\sum_{i=1}^{N}\mathbf{x}_i & (2.3) \\ \Sigma = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{x}_i - \mu)(\mathbf{x}_i - \mu)^T & (2.4) \end{cases}$$

Discussion

Ce modèle souffre de deux principales limitations :

— il ne permet de modéliser que des distributions mono-modales ;

— l'estimation de la matrice de covariance pleine nécessite un nombre d'observations proportionnel à la dimension des données : d ; ce modèle n'est pas adapté aux données de grandes dimensions

Les modèles mono-gaussiens à covariance pleine sont néanmoins utilisés dans le cadre de la reconnaissance de langue pour modéliser les distributions de scores (voir *Gaussian Backend* [Li et al., 2013]). Il est courant de partager la matrice de covariance entre tous les modèles de langues connues et de n'apprendre que le vecteur moyen μ pour chaque langue. Nous les avons utilisés dans le cadre des évaluations *NIST Language Recognition Evaluation* en 2011 et 2015 [Lee et al., 2016]. Ce modèle est également utilisé en segmentation du locuteur pour le Critère d'Information Bayesien (BIC) [Delacourt, 2000] et dans une version mono-dimensionnelle pour la détection de parole/non-parole [Bonastre et al., 2008].

2.1.2 Distribution Gaussienne à variance diagonale

Description du modèle

Afin de limiter la quantité de données nécessaire à l'estimation d'un modèle Gaussien, il est courant de limiter le nombre de paramètres à estimer en approximant la covariance par une matrice diagonale. La matrice Σ a alors la forme suivante :

$$\Sigma = \begin{bmatrix} \sigma_{0,0} & & 0 \\ & \ddots & \\ 0 & & \sigma_{d-1,d-1} \end{bmatrix} \quad (2.5)$$

où les termes diagonaux $\sigma_{i,i}$ sont estimés par maximum de vraisemblance comme suit :

$$\sigma_{i,i} = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_n^{(i)} - \boldsymbol{\mu}^{(i)})^2 \tag{2.6}$$

Discussion

Ce modèle souffre des limitations du modèle mono-Gaussien énoncées précédemment et ne permet,de plus, de modéliser que des distributions dont les axes principaux sont parallèles aux axes du repère utilisé dans l'espace acoustique. Cette limitation est illustrée par les figures 2.1 *b* et *c*. En reconnaissance de langue, ce modèle est utilisé pour modéliser les distributions de scores [Li et al., 2013]

Le cas particulier de la matrice de covariance sphérique (égale à la matrice identité multipliée par un paramètre scalaire) est très peu utilisé (cf. figure 2.1 *a*).

2.2. Les mélanges de Gaussiennes

Les modèles mono-Gaussien ne peuvent modéliser que des distributions mono-modales et la modélisation de distribution à covariance pleine nécessite un grand nombre d'observations. L'introduction des mélanges de Gaussiennes pour la modélisation acoustique [Reynolds et al., 2000] permet de modéliser des distributions complexes multimodales. L'utilisation dans le mélange de distributions Gaussiennes à covariance diagonale permet de représenter des distributions multimodales ou complexes en limitant le nombre de paramètres à estimer.

2.2.1 Le modèle et son apprentissage

Un mélange de Gaussiennes (GMM pour Gaussian Mixture Model) est une somme pondérée de C distributions Gaussiennes. La distribution est donnée par :

$$Pr(\mathbf{x}) = \sum_{c=1}^{C} \omega_c \, \mathcal{N}_{\mathrm{x}}^{c}(\boldsymbol{\mu}_c, \Sigma_c) \tag{2.7}$$

La figure 2.2 illustre la représentation d'une distribution multi- modale par un mélange de Gaussiennes en deux dimensions.

Il faut noter que ce modèle est défini par l'ensemble des vecteurs moyens et des matrices de covariance de ses composantes ainsi que par le poids ω_c de chacune de ces composantes dans la somme pondérée.

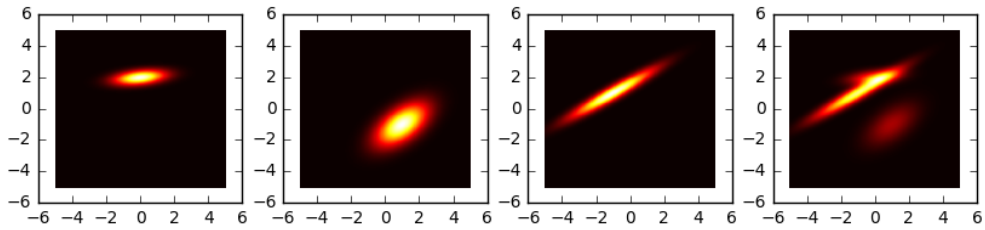

Figure 2.2 – Exemple de mélange de Gaussiennes en dimension 2. La quatrième distribution (à droite) est une somme pondérée des trois premières (poids respectifs de (0,2 ; 0,6 ; 0,2).

Les paramètres du mélange de Gaussiennes ne peuvent être estimés directement et nécessitent un algorithme EM (Expectation Maximization). À chaque itération de cet apprentissage, les paramètres de moyenne et variance de chacune des composantes Gaussienne du mélange sont estimés comme dans le cas d'un modèle mono-Gaussien si ce n'est que chaque observation est pondérée afin de calculer la moyenne et la variance de chaque distribution. Le poids de chaque observation est déterminé en prenant en compte l'ensemble des composantes du mélange. Une description détaillée de cet algorithme est donnée dans [Bimbot et al., 2004; Larcher, 2009; Prince, 2012].

2.2.2 Paradigme UBM-GMM et super-vecteurs

Les modèles GMM permettent une modélisation relativement simple et précise de la distribution des vecteurs acoustiques produits par une source et offrent un cadre simple pour comparer une source (un GMM) et une séquence de vecteur acoustique (vecteurs de MFCC) [Bimbot et al., 2004]. Pour cette raison, les GMMs ont été utilisés en vérification du locuteur et toutes ses dérivées (identification, segmentation, regroupement) en suivant le paradigme UBM-GMM entre 1995 et 2009. La vérification du locuteur fait un usage intensif du rapport d'hypothèses afin de répondre à la question suivante : la séquence de vecteurs acoustiques $\mathcal{S}$ a-t-elle été produite par le locuteur cible : hypothèse $\mathcal{H}_0$, ou par un autre locuteur : hypothèse $\mathcal{H}_1$.

$$\mathcal{LR}(\mathcal{S}, \mathcal{H}_0, \mathcal{H}_1) = \frac{p(\mathcal{H}_0|\mathcal{S})}{p(\mathcal{H}_1|\mathcal{S})} \tag{2.8}$$

Ce rapport de vraisemblances nécessite deux modèles : le modèle du locuteur cible et le modèle de l'*autre* locuteur. Cet *autre* locuteur est modélisé de façon générique en utilisant les données d'un grand nombre de locuteurs qui sont censés ne jamais apparaître dans les tests afin que le résultat de ceux-ci ne soit pas biaisé. Ce modèle est appelé *modèle du monde* et est censé représenter tous les locuteurs sauf le locuteur cible.

Si l'estimation du modèle du monde est relativement aisée grâce à l'algorithme EM, on manque la plupart du temps d'observations du locuteur cible pour estimer les paramètres

du GMM de façon robuste. Il est donc classique d'estimer les paramètres d'un locuteur particulier en adaptant ceux du modèle du monde qui par hypothèse représente la « *voix moyenne* ». Le modèle d'un locuteur est alors adapté, généralement avec le critère du *Maximum A Posteriori* (MAP). Ce critère consiste à adapter chaque distribution d'un mélange de Gaussiennes en considérant un modèle a priori (en général le modèle du monde). Dans un premier temps, les observations sont attribuées à une ou plusieurs distributions du modèle avant que ces distributions ne soient adaptées indépendamment les unes des autres en fonction des observations qui leur sont attribuées. La première étape de cette adaptation s'apparente à l'étape d'*Estimation* de l'algorithme EM alors que la seconde étape ré estime les paramètres comme une somme pondérée de l'a priori et de l'estimation par maximum de vraisemblance de cette distribution. Le critère MAP permet d'adapter chaque distribution de façon précise, mais nécessite une quantité relativement importante pour obtenir une estimation robuste du modèle. Ce ne sera pas le cas pour l'extraction des *i*-vecteurs (cf. chapitre 3.3.1) qui peut également être vu comme une adaptation du modèle du monde.

Ce paradigme a été largement utilisé du fait que les GMMs utilisent des distributions à covariance diagonale qui simplifient les calculs et que l'adaptation MAP utilisée ne modifie que les paramètres de moyenne des distributions, laissant les poids des distributions et les matrices de covariances inchangées d'un modèle à l'autre. De ce fait, les paramètres spécifiques à un locuteur donné se limitent aux moyennes des distributions de son GMM, soit un ensemble de C vecteurs de dimension d qui sont représentés sous une forme concaténée appelée super-vecteur [Kinnunen et Li, 2010].

Chaque locuteur est ainsi représenté par un super-vecteur dans un espace de grande dimension dont le centre est le super-vecteur du modèle du monde. Il est possible dans cet espace de classifier les locuteurs ou tout autre phénomène acoustique modélisé par un mélange de Gaussienne [Campbell et al., 2006; Dehak et al., 2009].

2.3. Discussion

Depuis les années 1980, les modèles Gaussiens simples ou en mélange ont été utilisés très largement dans tous les domaines de traitement automatique de la parole pour modéliser des phénomènes variés (sénones, phonèmes, parole, non-parole, musique, locuteur, langue...)

Les principaux inconvénients de ces modèles sont les suivants :

1. l'estimation de leurs paramètres nécessite un algorithme EM ;

2. le nombre de distributions Gaussiennes nécessaires pour modéliser un phénomène est difficile à estimer ;
3. le paradigme UBM-GMM implique que tous les phénomènes à comparer aient été modélisés avec la même complexité.
4. Les modèles Gaussiens et les GMMs ne permettent pas de modéliser la structure temporelle du signal de parole et il est nécessaire d'utiliser des modèles de Markov (cf. Partie III) ;
5. l'estimation robuste de leurs nombreux paramètres requiert une grande quantité de données ;
6. la modélisation est très sensible aux perturbations induites par le bruit ambiant ou par les distorsions dues au canal de transmission.

Le point 4 est discuté dans la partie III. Le Factor Analsyer, approximation de la distribution Gaussienne à covariance pleine, et ses dérivés qui permettent de palier aux points 5 et 6 est décrit dans la suite de cette partie.

CHAPITRE 3

Un compromis : le Factor Analyser

Le *Factor Analyser* a marqué la reconnaissance du locuteur depuis 2004 et plusieurs approches se sont succédées depuis les *Eigen Voices* [Kuhn et al., 1998] utilisées pour la reconnaissance de la parole, les *Eigen Channels* [Kenny et Dumouchel, 2004; Matrouf et al., 2007] utilisés pour supprimer la variabilité liée au canal, les *i*-vecteurs qui offrent une représentation en dimension réduite d'un segment audio et les versions complètes du *Joint Factor Analysis* [Kenny et al., 2007a,b] ou de l'analyse linéaire discriminante probabiliste (PLDA) [Prince et Elder, 2007] qui visent à séparer le locuteur du canal et du bruit.

Cette partie présente brièvement la théorie du *Factor Analyser* et de deux de ses applications en reconnaissance du locuteur : la PLDA et l'espace de variabilité totale (*Total Variability Space*) des *i*-vectors. La description suit une logique de complexité plutôt que temporelle et ne traite que des deux méthodes sur lesquelles ont porté mes travaux.

3.1. Le modèle du Factor Analyser mono-Gaussien

Le *Factor Analyser* est un compromis entre la distribution Gaussienne à covariance pleine qui nécessite un nombre important d'observations pour estimer les $\frac{d\times(d-1)}{2}$ paramètres libres de sa matrice de covariance, et la distribution à covariance diagonale très contrainte.

3.1.1 Description du modèle

Dans le compromis du *Factor Analyser* , la matrice de covariance de la distribution Gaussienne est décrite par $(r+1) \times d$ paramètres, où $r << d$. Le nombre de paramètres à estimer est donc compris entre d (matrice diagonale) et $\frac{d(d-1)}{2}$ (matrice pleine). La matrice de covariance est de la forme : $\Phi\Phi^T + \Sigma$ où Φ est une matrice portrait de rang r et de dimension $d \times r$ et Σ est une matrice diagonale. L'équation qui décrit la distribution Gaussienne devient alors :

$$Pr(\mathbf{x}) = \mathcal{N}_x(\boldsymbol{\mu}, \Phi\Phi^T + \Sigma) \tag{3.1}$$

Dans ce modèle, on considère que la variabilité réside principalement dans un sous espace linéaire de dimension réduite : r. La distribution Gaussienne est expliquée par une variable cachée de dimension r, qui réside dans ce sous-espace linéaire. Cette variable est projetée dans l'espace des observations par une opération linéaire : la multiplication par la matrice Φ à laquelle s'ajoute un bruit Gaussien dont la distribution est centrée, de covariance Σ. Un modèle de graphique du *Factor Analyser* est donné dans la figure 3.1.

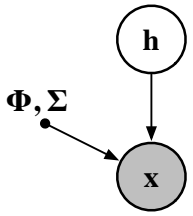

Figure 3.1 – Modèle graphique du *Factor Analyser* .

La description mathématiques complète du *Factor Analyser* mono-Gaussien est donnée en annexe A.

3.1.2 Discussion

Le *Factor Analyser* a été largement exploité dans le cadre de la reconnaissance du locuteur [Dehak et al., 2009; Kenny et al., 2007a; Kenny et Dumouchel, 2004; Prince et Elder, 2007] ou de la langue [Dehak et al., 2011b]. Ce modèle assure un bon compromis entre complexité et précision et présente deux intérêts principaux :

- il peut être utilisé pour de la réduction de dimension ;
- ses dérivés permettent de développer des modélisations discriminantes (voir 3.2) ou multi-Gaussiennes (voir 3.3).

Ce modèle est utilisé dans les *Eigen Voices* [Kenny et al., 2005a] et le *Joint Factor Analysis* [Kenny et al., 2007a] qui ne seront pas discutés dans ce manuscrit par souci de concision,

car mes travaux ne portent pas sur ces modèles. Il a également permis le développement des *Eigen Channels* (cf. section 3.3.3) et des *i*-vectors (cf. 3.3). Enfin le *Factor Analyser* est également à l'origine de l'analyse linéaire discriminante probabiliste (PLDA) qui est décrite ci-après.

3.2. Factor Analyser discriminant et mono-Gaussien

3.2.1 L'analyse linéaire discriminante probabiliste (PLDA)

Dans le paradigme de la PLDA, une observation $\mathbf{x}$ est vue comme la somme d'une composante due au locuteur, d'une composante due au canal et d'un bruit aléatoire. Le tout se présente sous la forme d'un *Factor Analyser* . Le modèle graphique de la PLDA est décrit par la figure 3.2, l'équation générative correspondante est la suivante :

$$\mathbf{x}_{i,s} = \boldsymbol{\mu} + \mathbf{F}\mathbf{h}_i + \mathbf{G}\mathbf{w}_{i,s} + \boldsymbol{\epsilon}_{i,s} \tag{3.2}$$

Une observation $\mathbf{x}_{i,s}$ obtenue pour la session s d'un locuteur i est la somme d'une observation moyenne $\boldsymbol{\mu}$, d'une composante liée au locuteur i, $\mathbf{F}\mathbf{h}_i$, d'une composante liée à la session s du locuteur i, $\mathbf{G}\mathbf{w}_{i,s}$ et d'un bruit propre à cette même session, $\boldsymbol{\epsilon}_{i,s}$.

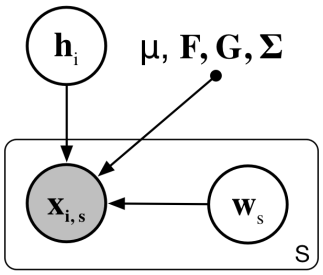

Figure 3.2 – Modèle graphique de l'analyse linéaire discriminante probabiliste (PLDA). Les variables cachées $\mathbf{h}_i$ qui représente l'identité du locuteur et $\mathbf{w}_{i,s}$ qui représente la session courante expliquent l'observation $\mathbf{x}_{i,s}$.

Les variables $\mathbf{h}$ et $\mathbf{w}$ sont des variables cachées qui suivent une loi de probabilité normale $\mathcal{N}(O, \mathbf{I})$ (les indices sont abandonnés ici pour plus de lisibilité). Le bruit $\boldsymbol{\epsilon}$ suit une distribution Gaussienne de moyenne nulle et de covariance Σ tandis que $\mathbf{F}$ et $\mathbf{G}$ sont des matrices rectangulaires portait qui définissent les sous-espaces locuteur et canal dans l'espace acoustique des observations. La matrice Σ est diagonale de rang égal à la dimension de l'espace des observations. Selon ce modèle, les observations suivent une distribution de probabilité Gaussienne de moyenne $\boldsymbol{\mu}$ et de covariance $\mathbf{F}\mathbf{F}^T + \mathbf{G}\mathbf{G}^T + \Sigma$ notée $P(\mathbf{x}_{i,s}) = \mathcal{N}(\mathbf{x}_{i,s}|\boldsymbol{\mu}, \mathbf{F}\mathbf{F}^T + \mathbf{G}\mathbf{G}^T + \Sigma)$.

Le paradigme PLDA permet d'apprendre les paramètres du modèle *Factor Analyser* tout en considérant que certaines observations proviennent du même locuteur, c.-à-d. partagent la même variable cachée $\mathbf{h}_i$ et offre ainsi un potentiel discriminant puisque le modèle apprend à regrouper les observations par locuteur dans le sous-espace dédié.

3.2.2 Discussion

Le modèle PLDA, implémenté *brutalement*, ne permet pas d'exploiter un grand nombre de sessions par locuteur et mes travaux avec Kong Aik Lee et Jiang Ye pour remédier à ce problème sont présentés dans le chapitre 6.

Le modèle PLDA offre l'avantage de pouvoir calculer facilement un rapport de vraisemblances entre hypothèses. Les cas de la vérification et de l'identification en milieu fermé, illustrés par les figures 3.3 et 3.4 sont décrits dans [Prince, 2012] et mes travaux avec Kong Aik Lee étendent et simplifient le calcul des scores PLDA pour le cas d'une identification du locuteur en milieu partiellement ouvert.

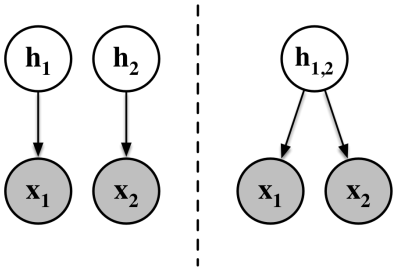

Figure 3.3 – Modèle graphique correspondant aux hypothèses de la tâche de vérification. Dans la première hypothèse, les deux observations $\mathbf{x}_1$ et $\mathbf{x}_2$ proviennent de deux locuteurs distincts, $\mathbf{h}_1$ et $\mathbf{h}_2$, tandis que dans la deuxième hypothèse ils proviennent d'un même locuteur $\mathbf{h}_{1,2}$.

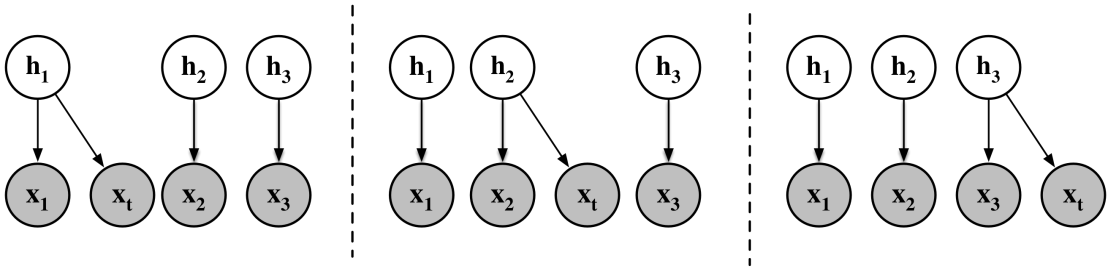

Figure 3.4 – Modèle graphique correspondant aux hypothèses de la tâche d'identification en milieu fermé. Les représentations graphiques successives représentent le cas où l'observation $\mathbf{x}_t$ est produite par les locuteurs $\mathbf{h}_1$, $\mathbf{h}_2$ et $\mathbf{h}_3$.

3.3. Extension multi-Gaussienne du Factor Analyser

Nous avons vu dans la section 2.2.2 que les modèles GMMs obtenus par adaptation MAP d'un modèle du monde dont on n'adapte que les paramètres de moyennes sont entièrement décrits par leur super-vecteur qui a souvent une dimension de l'ordre de 10000 paramètres (p. ex., pour un modèle GMM à 512 distributions Gaussiennes et une dimension de vecteur acoustique de 50, le super-vecteur est de dimension 25100).

Il est tentant de procéder à une classification de locuteurs ou des langues dans cet espace vectoriel où la représentation des segments acoustiques est un vecteur et non plus un modèle statistique complexe. De nombreux travaux ont été menés dans les années 2000 et les machines à vecteurs supports (SVM) on démontré leur potentiel pour ces tâches [Campbell et al., 2006, 2007, 2008; Fauve et al., 2007b]. Cependant, la grande dimension de l'espace des super-vecteurs et le nombre limité d'observations par classe acoustique rendent la tâche difficile.

En 2009, Dehak et al. [2011a] propose d'utiliser le *Factor Analyser* pour réduire la dimension de représentation des locuteurs en passant de dimension de l'ordre 10^4 à 10^2. L'espace de représentation est baptisé espace de variabilité totale et est décrit et discuté dans ce chapitre.

3.3.1 Hypothèses et modèles de variabilité totale

Comme nous l'avons vu dans la section 3, le *Factor Analyser* est une approximation de distribution Gaussienne mono-modale. L'équation génératrice du modèle *Factor Analyser* à une Gaussienne est la suivante :

$$\mathbf{x} = \boldsymbol{\mu} + \Phi\mathbf{h} + \epsilon \tag{3.3}$$

Une représentation graphique de ce modèle est donnée par la figure 3.5. Cette figure illustre le fait que la variable cachée $\mathbf{h}$ qui décrit la classe acoustique est de dimension réduite par rapport à l'observation $\mathbf{x}$. Le *Factor Analyser* peut donc être utilisé pour compresser l'information.

Une Gaussienne seule, même à matrice de covariance pleine, n'est pas suffisante pour modéliser la complexité de la distribution de vecteurs acoustiques produits par un locuteur ou dans une langue. Les GMMs permettent d'obtenir une représentation plus précise de cette distribution, mais produisent des super-vecteurs de dimension trop importante. Comme expliqué dans la section 2.2.2, les super- vecteurs sont la concaténation des vecteurs moyens des distributions Gaussiennes d'un GMM. En supposant que chaque Gaus-

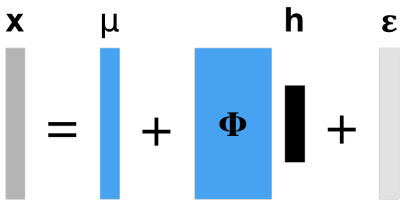

Figure 3.5 – Interprétation graphique du modèle *Factor Analyser* à une Gaussienne. L'observation **x** est la somme d'un vecteur moyen $\boldsymbol{\mu}$ avec une composante liée à la classe acoustique qui génère l'observation : **h** multipliée par la matrice de facteurs Φ plus un vecteur de bruit ϵ.

sienne de ce GMM a une matrice de covariance approximée par un *Factor Analyser* , on obtiendrait le modèle représenté par la figure 3.6. Soit un super- vecteur $\mathcal{M}$ représentant un GMM. Chacune des C composantes du modèle GMM se différencie du modèle du monde par son vecteur moyen $\mathcal{M}_c$ qui est décrit par un modèle *Factor Analyser* comme la somme d'un vecteur moyen $\boldsymbol{\mu}_c$ avec le terme $\Phi_c\mathbf{h}_c$ où la matrice Φ_c est la matrice de facteurs du *Factor Analyser* pour la c^{ieme} Gaussienne du GMM et $\mathbf{h}_c$ est la variable cachée qui décrit la classe acoustique. Enfin, le terme de bruit ϵ_c qui suit une distribution de probabilité Gaussienne $P(\epsilon_c) = \mathcal{N}(\epsilon_c|O, \Sigma_c)$ s'ajoute à la somme.

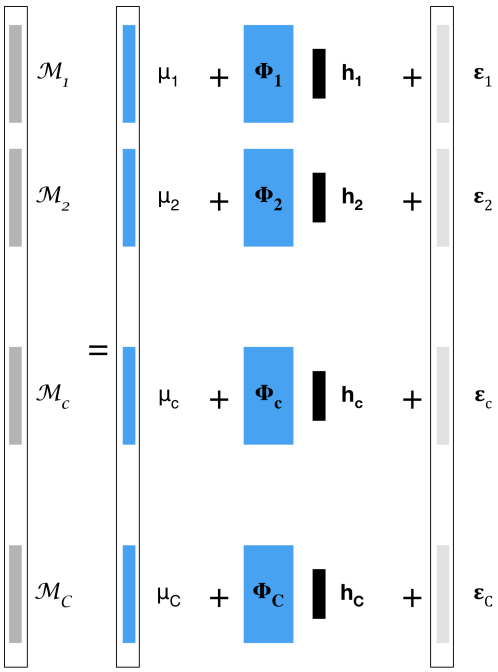

Figure 3.6 – Interprétation graphique du modèle *Factor Analyser* pour un GMM. L'observation $\mathcal{M}$ est un super-vecteur, concaténation de C vecteurs moyens. Chaque vecteur $\mathcal{M}_c$ est la somme d'un vecteur moyen $\boldsymbol{\mu}_c$ avec une composante liée à la classe acoustique qui génère l'observation : $\mathbf{h}_c$ multipliée par la matrice de facteurs Φ_c, plus un vecteur de bruit ϵ_c.

Le modèle décrit ci-dessus est une mixture de *Factor Analyser* . Dans le cadre de la reconnaissance du locuteur, on souhaite que le super-vecteur $\mathcal{M}$ soit dépendant d'une unique variable cachée **h** qui expliquerait de façon unique l'ensemble des distributions du

modèle GMM. Cette nouvelle hypothèse correspond à la figure 3.7 dans laquelle le vecteur moyen de chaque Gaussienne du GMM : $\mathcal{M}_c$ est obtenu grâce à un unique vecteur caché **h** qui ne dépend plus de la distribution, mais est commun à l'ensemble des distributions du modèle.

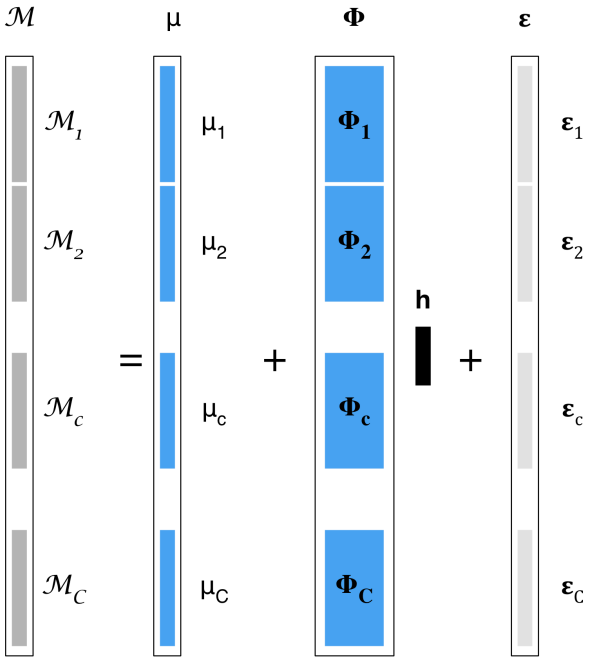

Figure 3.7 – Utilisation du *Factor Analyser* en partageant la variable cachée entre toutes les distributions du modèle GMM. L'observation $\mathcal{M}$ est un super-vecteur, concaténation de C vecteurs moyens. Chaque vecteur $\mathcal{M}_c$ est la somme d'un vecteur moyen $\boldsymbol{\mu}_c$ avec une composante unique liée à la classe acoustique qui génère l'observation : **h** multipliée par la matrice de facteurs Φ_c plus un vecteur de bruit ϵ_c. La variable cachée **h** est partagée entre toutes les distributions du GMM.

Dans ce modèle, nous pouvons utiliser des notations groupées : $\mathcal{M}$ pour le super-vecteur du GMM, $\boldsymbol{\mu}$ pour le super-vecteur de moyennes, Φ pour la matrice de facteurs qui est la concaténation des C matrices Φ_c et $\boldsymbol{\epsilon}$ pour le super-vecteur de bruit. Nous retrouvons ici une forme semblable au *Factor Analyser* mono-Gaussien de la figure, 3.5 mais il faut noter que les vecteurs $\mathcal{M}$ et $\boldsymbol{\epsilon}$ ne suivent pas une distribution Gaussienne comme c'est le cas dans le, *Factor Analyser* mais sont décrits par un GMM. Dans ce modèle, la variable cachée **h** est partagée entre toutes les distributions Gaussiennes du GMM.

Enfin dans le cadre de la reconnaissance du locuteur ou de la langue, les observations sont des séquences de vecteurs acoustiques $\mathcal{X} = \{\mathbf{x}_t\}_{t\in[1,N]}$. La dernière hypothèse qui conduit au paradigme de l'espace de variabilité totale consiste à partager l'unique variable cachée **h** à travers toute la séquence de vecteurs observés au cours du temps. Dans ce modèle, le vecteur moyen $\boldsymbol{\mu}$, la matrice de facteurs Φ ne sont pas modifiés. Le super-vecteur est obtenu par adaptation MAP du modèle du monde, et de ce fait exploite l'information de l'ensemble des vecteurs acoustiques de la séquence.

Dans le modèle de variabilité totale, la variable cachée $\mathbf{h}$ est partagée par les C distributions Gaussiennes du GMM et par les N vecteurs acoustiques de la séquence temporelle. La représentation obtenue, appelée i-vecteur correspond au Maximum A Posteriori de la variable cachée $\mathbf{h}$. Pour illustrer l'effet de compression du modèle de variabilité totale, considérons un segment d'une minute de parole représenté par 6000 vecteurs acoustiques de dimension 50. Le i-vecteur de dimension 500 (dimension standard en reconnaissance du locuteur) compresse l'information des 300000 coefficients acoustiques de la séquence, soit un taux de compression de $1,6\dot{1}0^{-}4$.

3.3.2 Estimation des paramètres et extraction des i-vecteurs

Sans rentrer dans les détails, nous donnons ici un aperçu de la procédure d'apprentissage des paramètres du modèle de variabilité totale qui sera utile dans la suite.

Estimation du modèle de variabilité totale

Estimer le modèle de variabilité totale consiste à estimer le super-vecteur moyen $\boldsymbol{\mu}$, la matrice Φ et la matrice de covariance Σ qui est diagonale par bloc et où chaque bloc est la matrice de covariance Σ_c décrite précédemment. L'estimation des paramètres est effectuée grâce à un algorithme EM qui nécessite au début de chaque itération de déterminer, pour chaque observation (c.-à-d. chaque vecteur acoustique), le poids de celle-ci dans l'estimation des paramètres $(\boldsymbol{\mu}_c, \Phi_c, \epsilon_c)$ correspondant à une Gaussienne fixée. Le poids d'une observation pour une distribution Gaussienne fixée, appelé occupation, correspond à la vraisemblance du vecteur acoustique pour cette Gaussienne divisée par la somme des vraisemblances pour toutes les distributions du GMM (comme dans le cas de l'apprentissage d'un modèle GMM présenté au chapitre 2).

Il y a donc deux quantités importantes pour l'algorithme EM du modèle de variabilité totale : l'occupation et l'observation (vecteur acoustique) pondérée par cette occupation. On s'intéresse plus exactement, et parce que la variable cachée du modèle TV est partagée par une séquence d'observation, à la somme des occupations des observations de la séquence, appelé statistique d'ordre 0 et à la somme pondérée des observations, appelée statistiques d'ordre 1. Si on note $\gamma_{t,c}$ l'occupation de l'observation $\mathbf{x}_t$ pour la Gaussienne c :

$$\gamma_{t,c} = \frac{P(\mathbf{x}_t|\boldsymbol{\mu}_c, \Sigma_c)}{\sum_{i=1}^{C} P(\mathbf{x}_t|\boldsymbol{\mu}_i, \Sigma_i)} \tag{3.4}$$

Notés respectivement $\mathbf{N}_c^{(i)}$, et $\mathbf{F}_c^{(i)}$, les statistiques d'ordre 0 et 1 pour une séquence i

et une Gaussienne c sont obtenus par les équations suivantes :

$$\mathbf{N}_c^{(i)} = \frac{1}{T}\sum_{t=1}^{T}\gamma_{t,c} \tag{3.5}$$

$$\mathbf{F}_c^{(i)} = \frac{1}{T}\sum_{t=1}^{T}\gamma_{t,c}\mathbf{x}_t \tag{3.6}$$

où T est le nombre d'observations de la séquence i.

Notons que $\mathbf{N}_c^{(i)}$ est un scalaire et $\mathbf{F}_c^{(i)}$ est un vecteur de même dimension que les observations. Ces quantités peuvent être regroupées en les concaténant pour obtenir un vecteur $\mathbf{N}^{(i)}$, de dimension égale au nombre de distributions dans le GMM et un vecteur $\mathbf{F}^{(i)}$ de dimension égale à celle des super-vecteurs. Notons de plus que les quantités $\mathbf{N}^{(i)}$ et $\mathbf{F}^{(i)}$ sont dépendantes du GMM utilisé pour les calculer. La modification de $\boldsymbol{\mu}$ et Σ nécessite le recalcul de $\mathbf{N}^{(i)}$ et $\mathbf{F}^{(i)}$. Afin d'alléger le calcul, les paramètres $\boldsymbol{\mu}$ et Σ sont fixés et seule la matrice Φ est ré estimée à chaque itération de l'algorithme EM. Les $\boldsymbol{\mu}_c$ et Σ_c sont initialisés avec les vecteurs moyens et les matrices de covariance du modèle du monde.

Pour plus de commodité, les statistiques sont centrées de la façon suivante :

$$\mathbf{F}_c^{(i)} \leftarrow \mathbf{F}_c^{(i)} - \mathbf{N}_c^{(i)}\boldsymbol{\mu}_c \tag{3.7}$$

$$\boldsymbol{\mu}_c \leftarrow 0 \tag{3.8}$$

En utilisant ces notations, on peut dériver (de façon non triviale) l'algorithme EM qui permet d'estimer le modèle de variabilité totale [Kenny et al., 2007a; Kenny et Dumouchel, 2004; Matrouf et al., 2007]. Durant l'étape E on calcule :

$$E\left[\mathbf{h}_i\right] = (\mathbf{I} + \Phi\mathbf{N}^{(i)}\Sigma^{-1}\Phi)^{-1}\Phi^T\Sigma^{-1}\mathbf{F}^{(i)} \tag{3.9}$$

$$E\left[\mathbf{h}_i^T\mathbf{h}_i\right] = E\left[(\mathbf{h}_i - E\left[\mathbf{h}\right])(\mathbf{h}_i - E\left[\mathbf{h}_i\right])^T\right] + E\left[\mathbf{h}_i\right]E\left[\mathbf{h}_i\right]^T \tag{3.10}$$

$$= (\mathbf{I} + \Phi\mathbf{N}\Sigma^{-1}\Phi)^{-1} + E\left[\mathbf{h}\right]E\left[\mathbf{h}\right]^T \tag{3.11}$$

On accumule alors les statistiques pour l'ensemble des segments de parole i disponibles dans les matrices $\mathbf{A}_c$ définies pour chaque distribution c et une matrice $\mathbf{C}$ de dimension $r \times d\dot{C}$ où r est le rang de Φ et d la dimension des observations.

$$\mathbf{C} = \sum_i E\left[\mathbf{h}_i\right]\mathbf{F}^{(i)} \tag{3.12}$$

$$\mathbf{A}_c = \sum_i \mathbf{N}_c^{(i)} E\left[\mathbf{h}_i^T\mathbf{h}_i\right] \tag{3.13}$$

Et à l'étape M, la nouvelle matrice Φ est obtenue par :

$$\Phi = \mathbf{C}\mathbf{A}^{-1} \tag{3.14}$$

où $\mathbf{A}$ est la concaténation des matrices $\mathbf{A}_c$.

Discussion sur l'extraction des i-vecteurs

Le modèle de variabilité totale permet d'extraire pour chaque segment de parole un *i*-vecteur qui est utilisé pour comparaisons. Sans en décrire les détails, cette section a pour but de souligner deux aspects de l'extraction des *i*-vecteurs .

1. Un *i*-vecteur est calculé selon l'équation suivante :

$$\mathbf{w}_i = (\mathbf{I} + \Phi\mathbf{N}^{(i)}\Sigma^{-1}\Phi)^{-1}\Phi^T\Sigma^{-1}\mathbf{F}^{(i)} \tag{3.15}$$

   L'extraction d'un *i*-vecteur nécessite donc l'inversion d'une matrice de dimension $r \times r$ qui s'avère coûteuse dans le cas d'applications temps réel. Cet aspect sera discuté dans la section 5.1.

2. La figure 3.8 schématise le processus d'extraction des *i*-vecteurs. Chaque observation acoustique issue de la séquence d'entrée est caractérisée grâce au modèle du monde GMM. Les occupations γ_c calculées lors de cette étape décrivent l'appartenance de l'observation à chacune des classes correspondant aux différentes distributions du modèle du monde. Ces occupations, ou statistiques d'ordres zéro, permettent de calculer les statistiques d'ordre 1. Ces statistiques (d'ordre 0 et 1) sont accumulés pour l'ensemble de la séquence acoustique considérée. Le modèle de variabilité totale, dérivé d'un *Factor Analyser* est alors utilisé pour compresser l'information contenue dans ces statistiques et produire un *i*-vecteur, représentation de la séquence acoustique de dimension réduite.

   Nous pouvons noter ici que les classes utilisées pour caractériser chaque observation acoustique (distributions du modèle du monde) ainsi que le processus de réduction de dimension (variabilité totale) sont entièrement non supervisés et utilisent le critère de maximum de vraisemblance qui n'est pas en lien avec la tâche de reconnaissance du locuteur. Cet aspect sera discuté dans les sections 4.1 et 4.2.1.

3.3.3 Le Factor Analyser pour supprimer la variabilité session

Comme dans le cas monomodal de la PLDA, le *Factor Analyser* multi-Gaussien peut être utilisé pour modéliser séparément les composantes liées au locuteur et à la variabi-

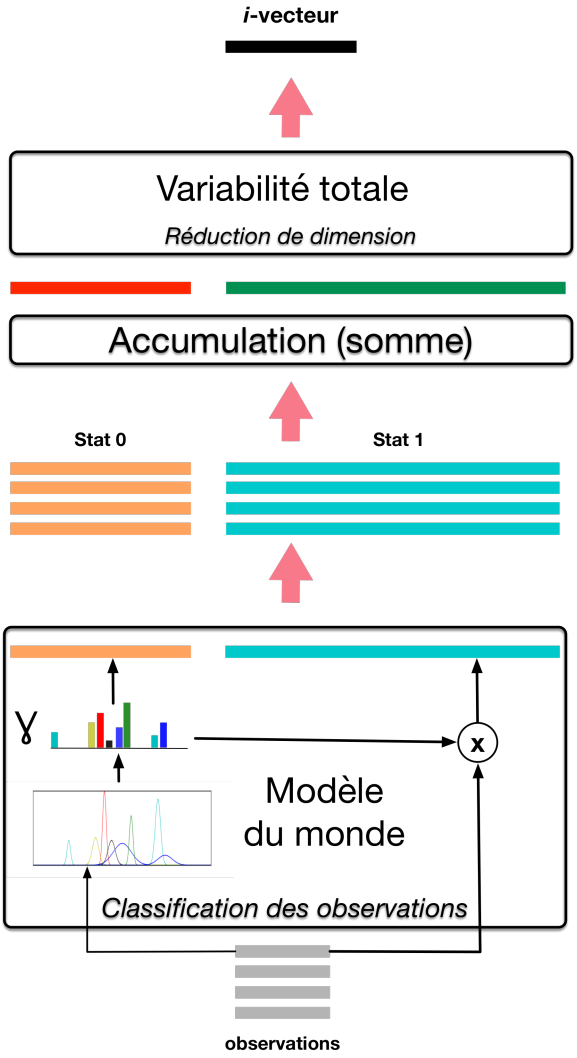

Figure 3.8 – Illustration des différentes étapes du processus d'extraction des *i*-vecteurs.

lité du canal (canal de transmission et bruit ambiant). Entre 2004 et 2009, différentes modélisations ont été proposées pour cet effet [Kenny et al., 2005b, 2007a; Kenny et Dumouchel, 2004; Matrouf et al., 2007]. Je ne décrirai ici que le modèle des *Eigen Channels* qui s'inscrit dans les travaux menés avec Christophe Lévy, Jean-François Bonastre et Driss Matrouf [Larcher et al., 2010a]. L'équation générative de ce modèle est donnée par :

$$\mathcal{M} = \boldsymbol{\mu} + \mathbf{Uh} + \mathbf{Dz} + \boldsymbol{\epsilon} \tag{3.16}$$

où $\mathcal{M}$ est le super-vecteur obtenu pour une session donnée d'un locuteur, $\boldsymbol{\mu}$ est un super-vecteur moyen, $\mathbf{U}$ est la matrice des *Eigen Channels*, matrice rectangulaire portait de dimension réduite dont les colonnes sont les axes du sous-espace contenant la variabilité canal. $\mathbf{h}$ est la composante canal résidant dans le sous-espace décrit précédemment, $\mathbf{D}$ est une matrice diagonale et $\mathbf{z}$ est un vecteur de même dimension que le super-vecteur $\mathcal{M}$ qui décrit la composante liée au locuteur dans l'espace des super-vecteurs. $\boldsymbol{\epsilon}$ est un

super-vecteur de bruit. Une description détaillée de ce modèle est donnée dans [Matrouf et al., 2007].

Le modèle *Eigen Channels* est utilisé pour chaque session afin d'estimer la composante, **h**, liée au canal et soustraire **Uh** du super-vecteur $\mathcal{M}$ qui est utilisé par la suite pour recréer le modèle GMM du locuteur utilisé pour la vérification. Ce modèle souffre des mêmes problèmes que le modèle de variabilité totale : il nécessite l'estimation de matrices de grandes dimensions et une quantité importante de données pour un apprentissage robuste. Ce dernier point fait l'objet des travaux présentés dans la section 5.2.

CHAPITRE 4

Réseaux de neurones

L'introduction du deep learning dans les années 2000 a relancé l'utilisation des réseaux de neurones en traitement automatique de la parole [Hinton et al., 2012, 2006]. Malgré de nombreuses tentatives, les premiers résultats positifs obtenus en reconnaissance du locuteur datent de 2014 [Lei et al., 2014]. Nous avons déjà rapporté une utilisation des réseaux de neurones profonds pour l'extraction de paramètres acoustiques dans la section 1.2 et nous ne décrivons ici que les deux approches qui ont marqué une avancée flagrante pour la reconnaissance du locuteur indépendante du texte. Ces deux approches, qui s'insèrent dans le paradigme de variabilité totale ou reproduisent l'intégralité d'un système *i*-vecteur, se différencient principalement par le critère qu'elles optimisent et que nous discuterons au cours de ce chapitre.

4.1. Extraction des statistiques par un réseau de neurones

Cette section donne une interprétation du paradigme de variabilité totale et du travail réalisé par Lei et al. [2014]. Je tente ici de donner un sens aux modèles complexes utilisés afin d'offrir un cadre de réflexion et non une réalité mathématique. Le paradigme utilisé est ensuite discuté.

4.1.1 Description du système

Comme décrit dans la section 3.3.2, l'extraction des *i*-vecteurs nécessite le calcul des statistiques d'ordre zéro utilisant le modèle du monde : une mixture de Gaussiennes. Ces statistiques sont obtenues en calculant l'appartenance de chaque vecteur de paramètres acoustiques aux différentes Gaussiennes du modèle du monde. On peut interpréter ce processus comme une comparaison locale des paramètres acoustiques. Chaque vecteur de paramètres contribue à la description du locuteur dans une zone de l'espace acoustique (déterminée par une distribution Gaussienne du modèle du monde) à proportion de son occupation pour cette Gaussienne. Les statistiques d'ordre 0 fournissent l'information sur le nombre d'observations appartenant à chaque « zone » de l'espace et les statistiques d'ordre 1 fournissent une représentation locale moyenne du locuteur dans cette zone de l'espace.

Cette interprétation a posteriori du modèle permet d'introduire un « sens physique » dans le modèle de variabilité totale et il est tentant de vouloir déterminer ce pavement de l'espace selon des critères liés au langage, ce qui n'est pas le cas dans le paradigme de variabilité totale où les distributions Gausiennes sont déterminées par le critère de maximum de vraisemblance et où chaque distribution ne revêt aucune signification interprétable.

Ainsi, dans [Lei et al., 2014], les auteurs proposent de remplacer la mixture de Gaussiennes (modèle du monde) par un réseau de neurones utilisé en reconnaissance de la parole, qui classifie les observations selon leur appartenance à une sénone, sous-état acoustique d'un phonème en contexte. Dans ce travail, les auteurs introduisent un sens physique dans les classes fournissant les statistiques d'ordre 0 et 1 ; les locuteurs sont comparés localement selon la façon dont ils produisent les différents phonèmes.

Le remplacement du modèle du monde (GMM) par un réseau de neurones a permis d'améliorer sensiblement les performances des systèmes de reconnaissance du locuteur [Lei et al., 2014; McLaren et al., 2015, 2014] et de la langue [McLaren et al., 2016]. L'architecture correspondante est représentée par la figure 4.1 par analogie à la figure 3.8 décrivant l'extraction des *i*-vecteurs classiques.

4.1.2 Réflexion sur la structure de l'espace acoustique

L'utilisation des modèles GMM en reconnaissance du locuteur est motivée par leur capacité à représenter des distributions multimodales. Le rôle structurant du modèle du monde [Scheffer, 2006] tel que décrit précédemment n'est cependant pas pris en compte lors de son apprentissage. Le nombre de classes (distributions Gaussiennes du modèle) est

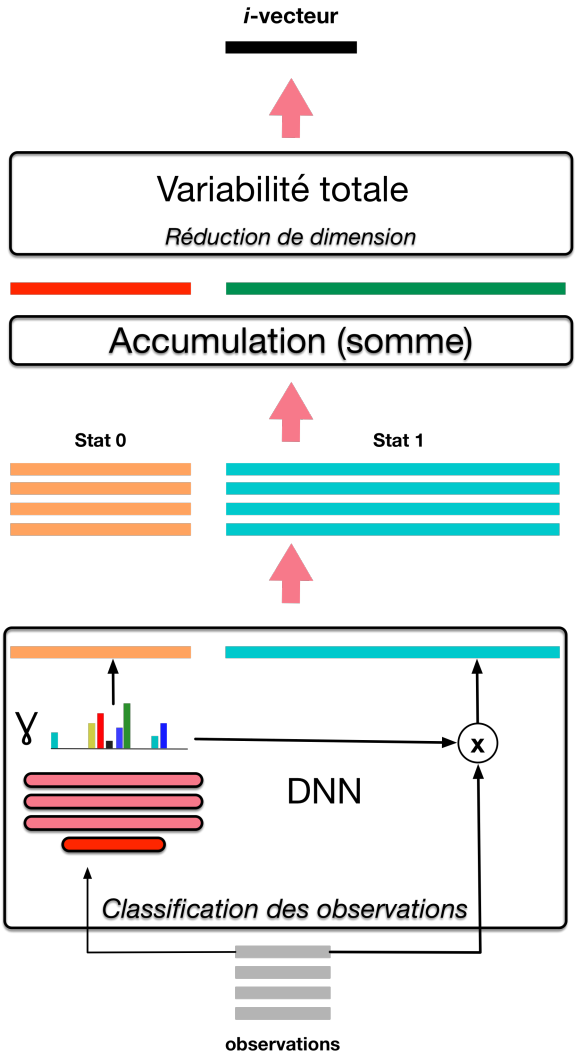

Figure 4.1 – Illustration des différentes étapes du processus d'extraction des *i*-vecteurs en remplaçant le GMM du modèle du monde par un réseau de neurones entraîné pour la reconnaissance de parole.

déterminé de manière empirique et le critère d'apprentissage (maximum de vraisemblance) ne garantit pas la séparation optimale des locuteurs dans l'espace acoustique.

L'utilisation d'un réseau de neurones optimisé pour la reconnaissance de la parole peut sembler inappropriée pour calculer les statistiques d'ordre 0 comme dans [Lei et al., 2014]. En effet, le réseau de neurones utilisé est optimisé pour « supprimer » ou en tout cas atténuer l'information liée au locuteur et ne tenir compte que de l'information phonétique portée par les observations. Il semble pertinent de choisir un critère de classification des vecteurs de paramètres acoustiques en lien avec la tâche de caractérisation du locuteur. La difficulté en reconnaissance du locuteur réside dans l'impossibilité d'entraîner un système discriminant pour l'ensemble des locuteurs du monde entier, car ils sont trop nombreux et que nous ne disposons pas de données pour chacun. Les approches les plus récentes, décrites ci-dessous, répondent à ce problème en se calquant sur les systèmes *i*-vecteurs.

4.2. Reproduction d'un système i-vecteur par un réseau de neurones

La nature de la tâche de reconnaissance du locuteur, et plus précisément de la vérification du locuteur, rend difficile l'utilisation de réseaux de neurones *end-to-end*. La solution actuelle consiste à entraîner des réseaux de neurones pour une tâche d'identification en milieu fermé avec un grand nombre de locuteurs et d'exploiter des *embeddings* produits par ces réseaux pour représenter les segments audio. : c.-à-d. des projections de ces segments dans un espace de dimension réduite, comme le sont les *i*-vecteurs. Cette approche a d'abord montré des résultats encourageants pour la reconnaissance du locuteur dépendante du texte où la variabilité acoustique est moindre avec les *d*-vecteurs [Heigold et al., 2016; Variani et al., 2014] et plus récemment en reconnaissance du locuteur indépendante du texte avec les *x*-vecteurs [Snyder et al., 2017, 2016].

Notons que Snyder et al. [2016] ou Heigold et al. [2016]; Variani et al. [2014] proposent des solutions *end-to-end* dans lesquelles la sortie du réseau de neurones est un score de vérification assimilable à un rapport de vraisemblances, mais ces approches nécessitent une quantité de données très importante qui n'est pas accessible facilement à l'heure actuelle. Le coût de calcul de l'apprentissage des réseaux, directement lié à la quantité de données, est également prohibitif et dans [Snyder et al., 2017] les auteurs montrent que pour le cas de la reconnaissance du locuteur indépendante du texte, il est possible d'entraîner un système neuronal surpassant les systèmes *i*-vecteurs dans la plupart des conditions pour un coût limité (en données et temps de calcul) si on limité le rôle du réseau de neurones à la production d'*embeddings* et que la comparaison finale est faite grâce à un modèle PLDA classique (cf. section 3.2).

4.2.1 Les systèmes x-vecteurs

Un système *x*-vecteur actuel pour la vérification du locuteur se compose d'un réseau de neurones produisant des *embeddings* et d'un modèle PLDA pour leur comparaison. Le réseau de neurones utilisé est composé de trois parties : un TDNN (*time-delay neural network*) qui prend en entrée les observations acoustiques (MFCC ou FB) avec leur contexte temporel, une couche de *pooling* qui accumule les sorties du TDNN pour l'ensemble de la séquence temporelle considérée (généralement équivalente à quelques secondes) et produit un vecteur unique correspondant à la moyenne et la déviation standard des sorties du TDNN concaténées pour la séquence temporelle et un réseau de neurones *feed-forward* simple qui prend en entrée pour chaque segment audio l'unique vecteur résultant de la couche de *pooling*. La dernière couche de ce réseau est un *softmax* qui donne l'index du

locuteur le plus probable parmi tous les locuteurs utilisés pour l'apprentissage. Ce réseau est donc entraîné pour une tâche d'identification du locuteur en milieu fermé. La première partie du réseau comporte généralement 5 couches convolutionnelles, et la dernière partie 3 couches simples. Les *embeddings* utilisés sont les sorties des couches intermédiaires de cette dernière partie du réseau.

La structure de ce réseau, représentée par la figure 4.2, est identique à celle d'un extracteur de *i*-vecteurs :

Étape	Système *i*-vecteurs	Système *x*-vecteurs
1	Chaque observation (vecteur acoustique) est utilisé pour calculer les statistiques d'ordre 0 et 1 pour le modèle du monde	Chaque observation, avec son contexte temporel est traitée par la première partie du réseau pour produire un vecteur unique de plus grande dimension
2	Les statistiques d'ordre 0 et 1 sont sommés pour l'ensemble de la séquence acoustique	La couche de *pooling* accumule les sorties de la première partie du réseau pour chaque observation et produit la moyenne et la déviation standard pour l'ensemble de la séquence acoustique
3	Un modèle de variabilité totale est utilisé pour compresser l'information contenue dans la somme des statistiques d'ordre 0 et 1 dans un *i*-vecteur de dimension réduite	La dernière partie du réseau produit un *embedding* de dimenson réduite

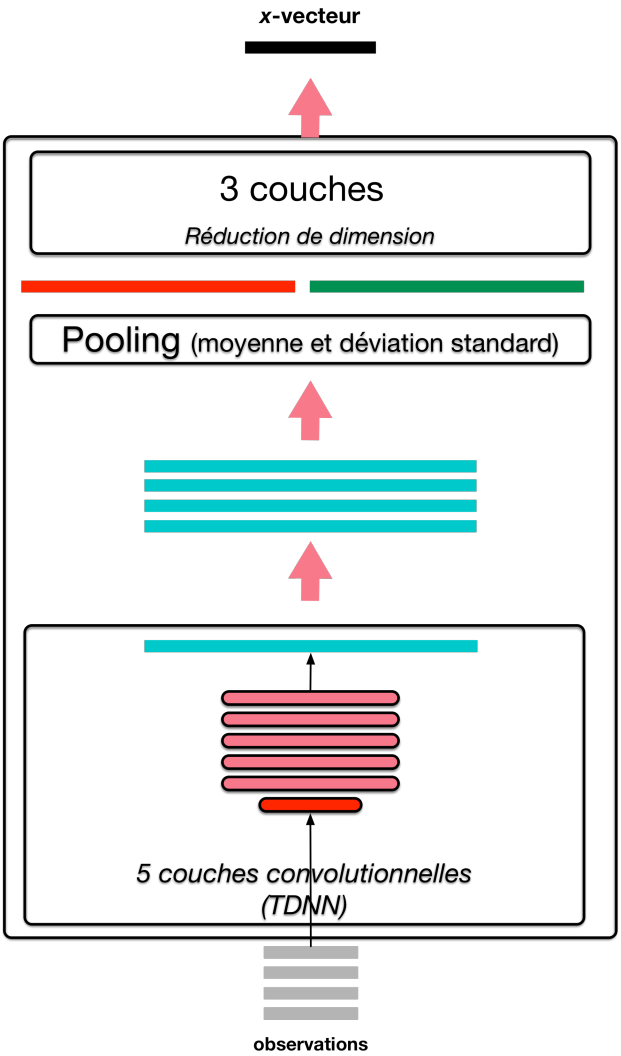

Figure 4.2 – Illustration de la structure du réseau de neurones utilisé pour extraire des *x*-vecteurs. La structure est calquée sur la structure du système *i*-vecteurs (cf. figure 3.8).

Discussions

Ce retour sur les avancées majeures de ces vingt cinq dernières années en reconnaissance du locuteur pose certaines questions.

La paramétrisation acoustique

Malgré les nombreux travaux dans le domaine de l'extraction de paramètre acoustiques, les MFCCs restent dominants dans les domaines de la reconnaissance du locuteur. Pourtant initialement développés pour la reconnaissance de la parole, ces coefficients restent souvent plus efficaces en reconnaissance du locuteur que des paramètres pensés pour cette application. Les BNF extraites par des réseaux de neurones phonétiques ont également montré leur fort potentiel et il est encore plus frappant que ces paramètres censés concentrer l'information liée au contenu phonétique, et ainsi probablement diminuer l'effet du locuteur, puissent être aussi efficaces dans ce domaine.

Il est possible que les bonnes performances des BNF phonétiques en reconnaissance du locuteur proviennent d'un ajout d'information jusque là indisponible pour les systèmes classiques. L'ajout d'une information phonétique dans les systèmes ajoute une information pertinente qui permet probablement de comparer les observations acoustiques « en contexte ». Les travaux récents en reconnaissance du locuteur [Liu et al., 2018], mais également en reconnaissance de la parole Ma et al. [2018]; Senior et Lopez-Moreno [2014] confirment le bénéfice d'un apprentissage commun pour différentes tâches.

Les modèles Gaussiens

Avec le retour récent des réseaux de neurones, les modèles Gaussiens connaissent une baisse d'activité. Pourtant ils présentent encore de nombreux intérêts comme on peut le voir

dans le fait que la PLDA est encore utilisée massivement en sortie de réseaux x-vecteurs. Ces modèles nécessitent moins de données que les systèmes neuronaux et présentent une grande robustesse aux variabilités compte tenu de la quantité de données limitée qu'ils exploitent. Dans l'avenir, il est possible que les réseaux de neurones soient exploités pour analyser le comportement des modèles Gaussiens afin de déterminer le type et la quantité de données nécessaires à l'apprentissage de ces derniers.

Tout comme il est probable que les modèles neuronaux bayésiens [Hernández-Lobato et Adams, 2015] bénéficient au traitement de la parole, il pourrait être intéressant de combiner les approches neuronales et Gaussiennes dans des architectures innovantes.

Les réseaux de neurones en reconnaissance du locuteur

En comparaison d'autres domaines, les résultats des réseaux de neurones profonds se sont fait attendre en reconnaissance du locuteur. À l'heure actuelle, peu d'architectures se sont montrées performantes et les rares qui surpassent les modèles Gaussiens reproduisent l'architecture de ces derniers.

Cependant, ces premières avancées ont permis aux réseaux de neurones profonds d'occuper en à peine quelques années une place centrale dans les systèmes de reconnaissance du locuteur et de la langue. Je propose en quelques points un bilan succinct de l'utilisation des réseaux de neurones en reconnaissance du locuteur qui sera rediscuté dans la partie IV dans le cadre de mes travaux en cours.

— L'architecture des systèmes neuronaux est similaire à celle des systèmes *i*-vecteurs .
— À performances équivalentes ou supérieures, les systèmes neuronaux nécessitent une quantité de données plus importante que les systèmes *i*-vecteurs [McLaren et al., 2018]
— Tout comme les *i*-vecteurs, les *x*-vecteurs ne prennent pas en compte la structure temporelle du signal de parole puisque l'accumulation des statistiques ou le *pooling* temporel détruisent cette structure. Cependant, les *x*-vecteurs prennent en entrée un contexte temporel plus important que dans le paradigme de variabilité totale.
— Comparé à un système *i*-vecteurs pour lequel le modèle du monde, le *Factor Analyser* et la PLDA sont appris séquentiellement avec des critères différents, un extracteur de *x*-vecteurs est appris de façon globale avec un critère en lien avec la tâche finale. Cet aspect laisse envisager un fort potentiel qui sera discuté dans la Partie IV.

Deuxième partie

Variantes sur la modélisation acoustique en reconnaissance du locuteur indépendante du texte

Les travaux présentés dans cette partie, menés à partir de 2010, sont divisés en trois catégories : les approches déterministes, les approches probabilistes et les approches neuronales. Tous ces travaux se situent dans le cadre du paradigme *i*-vecteurs, à différents niveaux de la chaîne de traitement. Les travaux sur les approches déterministes [Bousquet et al., 2012; Larcher et al., 2012b, 2010a] ont principalement été menés au Laboratoire Informatique d'Avignon (LIA) avec Jean-François Bonastre, Christophe Lévy, Pierre-Michel Bousquet et Driss Matrouf tandis que les approches probabilistes [Jiang et al., 2012; Lee et al., 2013] ont été explorées à l'Institute for Infocomm Research de Singapour avec Kong Aik Lee, Jiang Ye, Ma Bin et Haizhou Li. Enfin les approches neuronales [Le Lan, 2017] ont été développées au Laboratoire Informatique de l'Université du Mans avec Sylvain Meignier, Gaël Le Lan et Delphine Charlet d'Orange Lab.

CHAPITRE 5

Approches déterministes

Les travaux présentés dans cette partie ont été réalisés avec différentes motivations. La première étude sur la réduction de dimension par Analyse en Composantes Principales vise à accélérer le processus d'extraction des *i*-vecteurs en réduisant la mémoire utile pour cette étape afin de permettre l'extraction des *i*-vecteurs sur des plateformes aux ressources limitées. La seconde proposition est une étude sur la scalabilité du *Factor Analyser* utilisé pour les *Eigen Channels* en vue de faciliter d'un point de vue computationnel l'entraînement d'un système de grandes dimensions. Enfin les travaux sur la normalisation des *i*-vecteurs ont pour but d'optimiser le traitement appliqué à ces vecteurs en prenant en compte les modèles utilisés pour la comparaison de locuteur.

5.1. Réduction de dimension déterministe

Comme présenté dans la section 3.3, l'extraction des *i*-vecteurs qui suit le paradigme de l'espace de variabilité totale peut être vue comme un processus de compression des super-vecteurs de moyennes des modèles GMM. L'extraction des *i*-vecteurs peut être décomposée en deux étapes :

1. calcul et accumulation des statistiques
2. compression du super-vecteur

À ces deux étapes s'ajoute une étape de normalisation des *i*-vecteurs qui vise à réduire la variabilité due au canal de transmission ou au bruit Bousquet et al. [2011]; Garcia-Romero et Espy-Wilson [2011].

5.1.1 Motivations et Analyse en Composantes Principales

L'extraction des *i*-vecteurs intervient lors des deux phases de l'authentification : à l'enrôlement pour produire une référence de ce locuteur et lors de la phase de test où un échantillon vocal est collecté et comparé à une référence connue de locuteur. Si durant l'enrôlement, les ressources et le temps nécessaires ne sont pas une contrainte, ils le deviennent lorsqu'il s'agit d'authentifier une personne en temps réel en disposant de ressources matérielles plus ou moins réduites.

L'extraction d'un *i*-vecteur $\mathbf{w}$, décrite par l'équation 5.1 nécessite l'inversion d'une matrice de grande dimension (plusieurs centaines de dimensions) : $(\mathbf{I} + \Phi\mathbf{N}^{(i)}\Sigma^{-1}\Phi)$

$$\mathbf{w} = (\mathbf{I} + \Phi\mathbf{N}^{(i)}\Sigma^{-1}\Phi)^{-1}\Phi^{T}\Sigma^{-1}\mathbf{F}^{(i)} \tag{5.1}$$

Cette matrice doit être inversée pour chaque, *i*-vecteur car elle dépend des statistiques d'ordre 0 de la session de test. D'autres méthodes de réduction de dimension sont fréquemment utilisées dans d'autres domaines, parmi celles-ci, l'analyse en composantes principales (ACP) [Jolliffe, 2002; Tipping et Bishop, 1999] est une des plus simples et présente un lien avec le *Factor Analyser* qui peut être intéressant pour modifier le processus d'extraction des *i*-vecteurs sans le remettre en cause entièrement.

L'analyse en composantes principales permet de trouver une base orthonormale d'un sous-espace, appelé espace propre, qui maximise la variance des données après projection et minimise également l'erreur quadratique moyenne. Soit un ensemble de super-vecteurs de dimension dC et de matrice de covariance Σ. La matrice Σ peut s'écrire :

$$\Sigma = \mathbf{P}\mathbf{D}\mathbf{P}^{t} \tag{5.2}$$

où $\mathbf{D}$ est une matrice diagonale dont les termes, appelés valeurs propres, sont rangés par ordre décroissant et $\mathbf{P}$ est la matrice des vecteurs propres de C rangés en colonnes.

La réduction de dimension est obtenue grâce à la matrice rectangulaire $\mathbf{P}$, de rang k (et donc de dimensions $k \times dC$) dont les lignes sont les k vecteurs propres de Σ correspondant aux k plus grandes valeurs propres de $\mathbf{P}$. La projection $\mathbf{w}$ d'un super-vecteur $\mathbf{M}$ est obtenue par :

$$\mathbf{w} = \mathbf{P} \cdot \mathcal{M} \tag{5.3}$$

L'analyse en composantes principales présente deux avantages :

— la réduction de dimension d'un super-vecteur est obtenue par un simple produit matriciel, moins complexe que l'inversion de matrice nécessaire pour extraire les *i*-vecteurs ;

— la matrice **P** correspond à un minimum local du modèle de *Factor Analyser* et offre donc un lien avec le paradigme de variabilité totale (Le *Factor Analyser* est assimilable à une ACP qui prend en compte les variances par distribution Gaussienne et les effectifs ou occupations des observations. Cela revient à pondérer les *i*-vecteurs par la fiabilité de leur estimation).

5.1.2 Performances de l'Analyse en Composantes Principales

Dans cette section, les vecteurs obtenus par ACP sont comparés aux *i*-vecteurs [Larcher et al., 2012b] sur le plan des performances et de temps de calcul pour la vérification du locuteur.

Systèmes et protocoles

Les différents systèmes considérés dans cette étude ont été évalués sur la partie homme de la tâche principale de NIST-SRE08 [1].

Tous les systèmes proposés partagent des composantes communes :

— les paramètres acoustiques sont composés de 13 coefficients PLP ainsi que de leurs dérivées premières et secondes ;

— un unique modèle du monde à 512 distributions a été appris sur des données téléphone et microphone provenant de NIST-SRE04 et NIST-SRE05 ;

— ces mêmes bases de données augmentées de NIST06 et SwitchBoard ont été utilisées pour l'apprentissage des différents paramètres des systèmes ;

— tous les vecteurs extraits sont centrés, réduits et divisés par leur norme euclidienne comme décrit dans [Garcia-Romero et Espy-Wilson, 2011].

Quatre systèmes sont comparés :

IV-PLDA est un système *i*-vecteurs classique. Le *Factor Analyser* est utilisé pour apprendre une matrice de rang 500. Les *i*-vecteurs extraits selon l'équation 5.1 sont normalisés avant que les scores de vérifications ne soient calculés grâce à un modèle PLDA. Ce système constitue le système de référence ;

IV-Mahalanobis est un système identique au précédent excepté que les scores de vérifications sont calculés grâce à une distance de Mahalanobis ;

ACP-Mahalanobis pour chaque session, les super-vecteurs sont estimés par une adaptation MAP du modèle du monde et projetés dans un sous-espace propre de dimension 500 grâce à une ACP. Les vecteurs obtenus sont normalisés de la même façon

1. http://www.itl.nist.gov/iad/mig/tests/spk/2008/index.html vu le 15/10/2018

que les *i*-vecteurs et les scores de vérifications sont calculés grâce à une distance de Mahalanobis ;

IV-ACP-Mahalanobis des *i*-vecteurs sont extrait selon l'équation, 5.1 mais la matrice Σ utilisée est la matrice de projection de l'ACP calculée pour le système *ACP-Mahalanobis*. Les *i*-vecteurs obtenus sont normalisés comme pour les autres systèmes et les scores sont calculés grâce à une distance de Mahalanobis.

Excepté pour le système de référence qui correspond au système état-de-l'art à l'époque où les travaux ont été réalisés [Garcia-Romero et Espy-Wilson, 2011], les autres systèmes comparés utilisent tous une distance de Mahalanobis en raison des bonnes performances obtenues avec cette métrique pour le système qui nous intéresse ici : *ACP-Mahalanobis*. De plus, l'utilisation de la distance de Mahalanobis permet de supprimer pour ce système toute utilisation du *Factor Analyser* et d'estimer les performances d'un système entièrement déterministe.

Performances pour la vérification du locuteur

Les taux d'égales erreurs (EER) obtenus par les 4 systèmes considérés sont présentées dans le tableau 5.1 pour les 8 conditions de l'évaluation homme NIST-SRE08.

Table 5.1 – Performances des différents systèmes pour les 8 conditions de l'évaluation NIST-SRE08 (tests hommes) en termes de taux d'égales erreurs (% EER).

	Systèmes			
Condition	IV-PLDA	IV-Mahalanobis	IV-ACP-Mahalanobis	ACP-Mahalanobis
det 1	5.15	4.95	**3.77**	4.90
det 2	0.81	**0.62**	1.21	1.04
det 3	5.37	5.13	**3.91**	5.11
det 4	**3.73**	4.10	3.87	3.95
det 5	**3.14**	3.61	4.06	3.64
det 6	**4.12**	5.03	4.35	5.56
det 7	**1.37**	1.96	**1.37**	2.49
det 8	0.56	1.32	**0.44**	1.19

Conformément à nos attentes, l'approche IV-PLDA obtient les meilleures performances dans 4 des 8 conditions. Il est cependant intéressant de noter que dans les deux conditions 1 et 3, ce même système est moins bon que tous les autres systèmes présentés.
Le système IV-Mahalanobis surpasse le système IV-PLDA dans 3 des 8 conditions.

Cependant, les taux d'égales erreurs obtenus dans les conditions incluant des données téléphoniques (conditions 4 à 8) laissent penser que la classification par distance de Mahalanobis est moins robuste a la variabilité du canal de transmission que l'Analyse Lineaire Discriminante Probabiliste.
Le système IV-ACP-Mahalanobis utilisant la matrice calculée par ACP pour extraire des *i*-vecteurs obtient les meilleures performances dans 4 des 8 conditions tout en étant relativement proche du système IV-PLDA dans les 4 autres conditions. D'après ces résultats, il semble que l'utilisation de la matrice obtenue par Analyse en Composantes Principales sur les super-vecteurs peut être utilisée directement afin d'extraire les *i*-vecteurs.
Le système ACP-Mahalanobis utilisant directement la projection orthogonale des super-vecteurs sur la matrice obtenue par PCA ne se distingue dans aucune des conditions. En revanche, il obtient des taux d'erreurs plus faibles que le système IV-PLDA dans 2 des 8 conditions. Ces résultats sont d'autant plus intéressants que ce système est totalement déterministe.

Comparaison des temps de calcul

Le temps de calcul des deux types d'extracteurs de vecteurs utilisés est présenté dans le tableau 5.2. Dans le cas de l'analyse en composantes principales (ACP), le temps de calcul de la matrice correspond au temps total tandis que pour le *Factor Analyser* le temps donné ne correspond qu'à une itération de l'algorithme EM alors qu'il est fréquent de faire plusieurs itérations. Les résultats présentés précédemment sont donnés avec 3 itérations d'EM. Pour la dimension du système considéré, l'estimation de la matrice d'ACP est 4 fois plus rapide qu'une unique itération d'EM pour le *Factor Analyser* . Même si ce gain de temps n'est pas primordial dans notre cas, car la matrice peut être calculée « hors ligne » sans contrainte de temps, il est intéressant de voir qu'estimer la matrice d'ACP est plus rapide, d'autant que cette matrice peut être utilisée pour extraire des *i*-vecteurs.

Table 5.2 – Temps de calcul (en secondes) mesuré sur un CPU *Xeon 3.47GHz* pour des extracteurs de vecteur utilisant le *Factor Analyser* ou une ACP, de rang 500.

Extracteur	Calcul de la matrice	Extraction des vecteurs
Factor Analyser (1 iteration)	18,930	2.59
ACP	4,498	5.7×10^{-4}

Comme escompté, l'extraction des vecteurs par ACP est beaucoup plus rapide que le calcul des *i*-vecteurs : 4500 fois plus rapide dans la configuration choisie pour cette

étude. L'utilisation d'une ACP pour réduire la dimension des super-vecteurs peut aussi être considérée si les performances ne sont pas critiques pour l'application considérée.

À la suite de ces travaux, plusieurs approches ont été proposées pour accélérer le temps de calcul des *i*-vecteurs et réduire les ressources nécessaires, notamment en ce qui concerne la mémoire, [Glembeck et al., 2011] certaines préservant les performances des systèmes *i*-vecteurs [Cumani et Laface, 2013a,b]. Ces approches reposent sur une compression complexe de la matrice de variabilité totale et une approximation des *i*-vecteurs qui utilise un algorithme d'optimisation par descente de gradient.

5.2. Scalabilité du Factor Analyser

L'étude décrite se place dans le cadre du paradigme des *Eigen Channels* qui est décrit dans la section 3.3.3.

5.2.1 Motivations et principe

Comme discuté dans la section 4.1.1, le modèle du monde permet de segmenter (paver) l'espace des observations acoustiques et déterminer l'appartenance des observations aux différentes zones de l'espace correspondant aux distributions du GMM. Cette interprétation laisse penser que le pouvoir discriminant du modèle de *Factor Analyser* est lié à la précision de classification du modèle du monde et donc à sa taille. Il est notable que les performances des différents systèmes impliquant le *Factor Analyser* s'améliorent lorsque la taille du modèle du monde (c.-à-d. le nombre de ses distributions) augmente [Matejka et al., 2011].

Le nombre de distributions du modèle du monde est cependant limité par les ressources nécessaires à l'apprentissage du *Factor Analyser* correspondant. En effet, la dimension des vecteurs de statistiques utilisés dans cette estimation (cf. section 3.3) est directement liée au nombre de distributions du modèle du monde. Ainsi un nombre de distributions trop important entraîne des besoins de mémoire et de temps de calcul très importants. Pour cette raison, les modèles du monde utilisés dans la littérature sont généralement limités à 2048 distributions.

L'approche dont Jean-François Bonastre est à l'origine [Larcher et al., 2010a] consiste à décorréler la taille de la matrice du *Factor Analyser* du nombre de distributions du modèle du monde en considérant que la variabilité canal liée à différentes zones de l'espace acoustique (distributions Gaussiennes du modèle du monde) est décrite par une même matrice d'*Eigen Channels*.

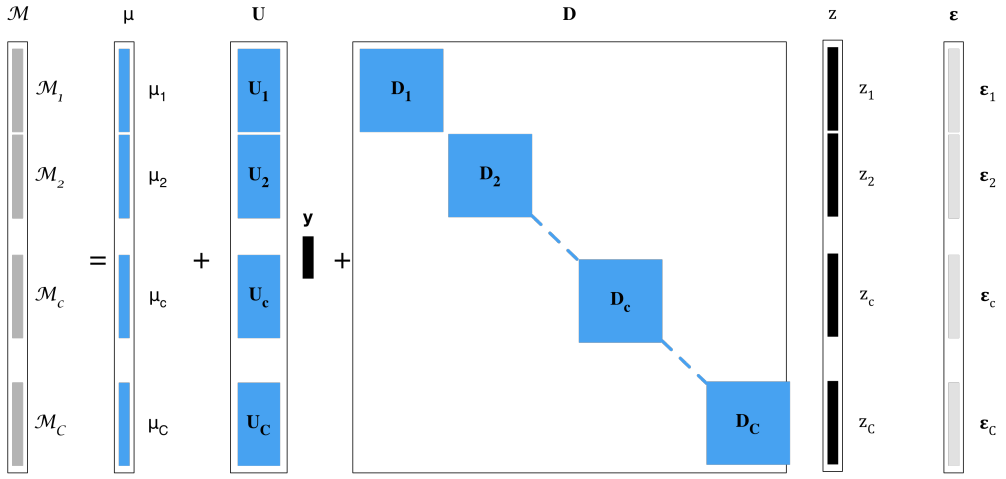

Figure 5.1 – Interprétation graphique du modèle *Eigen Channels* pour un GMM. L'observation $\mathcal{M}$ est un super-vecteur, concaténation de C vecteurs moyens. Chaque vecteur $\mathcal{M}_c$ est la somme d'un vecteur moyen $\boldsymbol{\mu}_c$ avec une composante liée à la composante canal, $\mathbf{U}_c\mathbf{y}$, au locuteur, $\mathbf{D}_c\mathbf{z}_c$, plus un vecteur de bruit $\boldsymbol{\epsilon}_c$.

Afin de décorréler le nombre de distributions du modèle du monde de la dimension de la matrice de facteurs, $\mathbf{U}$, l'apprentissage est réalisé en 4 étapes :

1. un modèle du monde, UBM^{grand}, est appris avec un grand nombre de distributions $\mathbf{C}_{grand}$
2. un second modèle GMM, UBM^{petit} est obtenu en fusionnant les distributions du modèle UBM^{grand}. Les distributions sont fusionnées 2 à 2 jusqu'à obtention du nombre fixé au préalable. Les distributions fusionnées sont les 2 distributions, $\mathcal{N}_1(\boldsymbol{\mu}_1, \Sigma_1, w_1)$ et $\mathcal{N}_2(\boldsymbol{\mu}_2, \Sigma_2, w_2)$ les plus proches selon la distance :

$$D(\mathcal{N}_1, \mathcal{N}_2) = \frac{w_1}{w_1 + w_2} \log\left(\frac{\sqrt{\Sigma}}{\sqrt{\Sigma_1}}\right) + \frac{w_2}{w_1 + w_2} \log\left(\frac{\sqrt{\Sigma}}{\sqrt{\Sigma_2}}\right) \tag{5.4}$$

   et Σ est la variance de la distribution Gaussienne entre les deux distributions. La distribution $g'(c', \boldsymbol{\mu}', \Sigma')$ résultant de la fusion de $g_i(c_i, \boldsymbol{\mu}_i, \Sigma_i)$ et $g_j(c_j, \boldsymbol{\mu}_j, \Sigma_j)$ est donnée par :

$$c' = c_i + c_j \tag{5.5}$$

$$\mu' = \frac{c_i * \mu_i + c_j * \mu_j}{c_i + c_j} \tag{5.6}$$

$$\Sigma' = \frac{c_i}{c_i + c_j}\Sigma_i + \frac{c_j}{c_i + c_j}\Sigma_j + \frac{c_i * c_j}{(c_i + c_j)^2}(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)(\boldsymbol{\mu}_i - \boldsymbol{\mu}_j)^{tr} \tag{5.7}$$

   Toutes les étapes de fusion sont enregistrées dans un arbre (cf. figure 5.2).
3. une matrice d'*Eigen Channels*, $\mathbf{U}^{petit}$, est apprise en utilisant le modèle du monde UBM^{petit} ;
4. La matrice $\mathbf{U}^{petit}$ est étendue pour obtenir une matrice $\mathbf{U}grand$ en correspondance avec le modèle du monde UBM^{grand}. L'arbre de fusion des distributions est parcouru

en sens inverse et toutes les distributions du UBM^{grand} sont associées à la matrice UBM_c^{petit} dont elles héritent en respectant l'arbre de fusion (cf. figure 5.2)

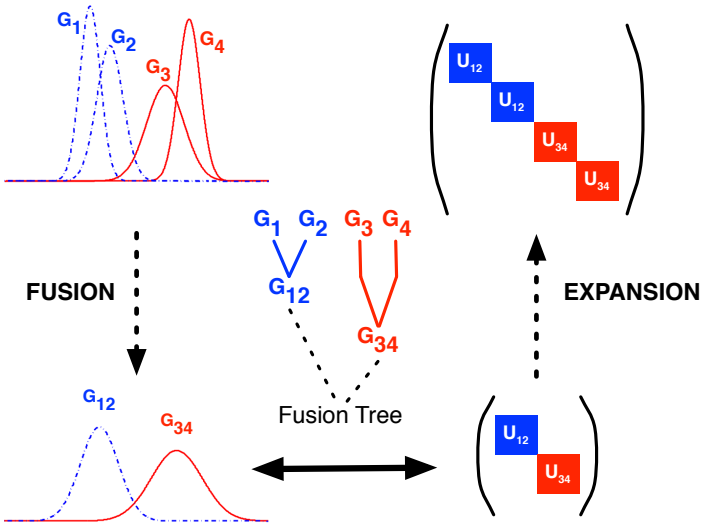

Figure 5.2 – Schéma de principe de l'apprentissage décorrélé de la matrice.

Après la phase d'apprentissage, seuls le modèle du monde UBM^{grand} et la matrice d'*Eigen Channels*, U^{grand} sont utilisés de façon standard.

5.2.2 Performances et discussion

Les performances de l'approche proposée sont évaluées selon le protocole de la condition 7 de l'évaluation NIST-SRE08 short2-short3 homme [Martin et Greenberg, 2009]. Les résultats sont présentés sous forme de courbe DET dans les figures 5.3, 5.4 et 5.5.

La figure 5.3 présente les résultats de trois systèmes utilisant une matrice d'*Eigen Channels* apprise grâce à un modèle du monde à 32 distributions.

- le premier système utilise un modèle du monde à 32 distributions, obtenu par fusion des distributions d'un modèle à 512 distributions en suivant la procédure de fusion décrite ci-dessus. La matrice d'*Eigen Channels* est apprise à partie de ce modèle à 32 distributions ;
- le second système utilise la même matrice d'*Eigen Channels* étendue pour correspondre au modèle à 512 distributions, comme décrit par la figure 5.2 ;
- le troisième système est un système *Eigen Channel* classique pour lequel un nouveau modèle du monde à 32 distributions est appris par EM. Une nouvelle matrice d'*Eigen Channels* correspondante est apprise de façon classique.

L'expansion de la matrice d'*Eigen Channels* du deuxième système apporte un léger gain avec un taux d'égales erreurs d'environ 7% alors que le système réduit à 32 distributions obtient un EER de 8,6%. La courbe DET permet de voir que le gain n'est pas aussi important pour les taux de fausse alarme plus bas (partie supérieure de la courbe). Ce premier

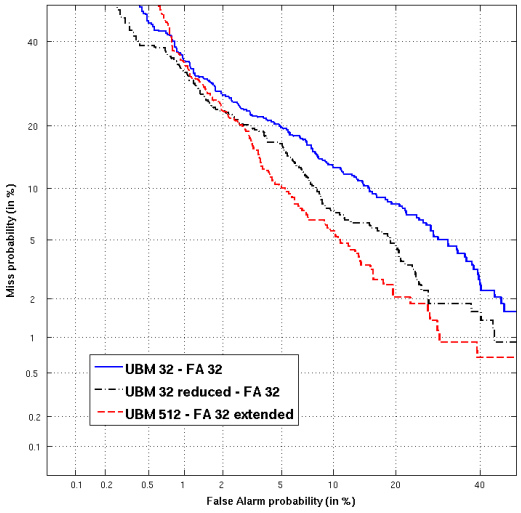

Figure 5.3 – Courbes DET obtenues pour un système étendu de 32 à 512 distributions sans aucune normalisation de score.

résultat confirme qu'il est possible de décorréler les dimensions du modèle du monde et de la matrice d'*Eigen Channels* et que l'utilisation d'un modèle du monde avec plus de distributions améliore les performances. Le troisième système obtient un EER supérieur aux deux autres, avec 11%. Ce résultat laisse penser qu'il est préférable d'apprendre un modèle du monde avec un nombre important de distributions et de les fusionner ensuite. Ce résultat devrait être vérifié, mais il indique au moins que la réduction du nombre de distributions par fusion ne dégrade pas systématiquement les performances du système. Ce résultat confirme l'hypothèse déjà discutée que le critère d'apprentissage du modèle GMM par maximum de vraisemblance n'est pas idéal.

La figure 5.4 présente le résultat de systèmes identiques aux précédents, mais de dimension plus importante. Dans ce nouvel exemple, le système de départ possède toujours 512 distributions, mais le modèle réduit possède maintenant 128 distributions. Dans ce cas, les deux modèles à 128 distributions, c.-à-d. celui dont le modèle du monde est obtenu par fusion du modèle à 512 distributions et celui pour lequel le modèle du monde est appris par EM, obtiennent des performances très proches. Ce résultat confirme le fait que la fusion des distributions produit un GMM de bonne qualité. Le système pour lequel la matrice est étendue à 512 distributions obtient un taux d'égales erreurs comparable aux deux autres (environ 6%), mais il est visible sur la courbe DET que ce système se comporte moins bien pour tous les autres points de fonctionnement.

L'application d'une normalisation des scores (comme décrit dans [Bimbot et al., 2004]), z-normalisation appliquée après une t-normalisation, confirme que l'expansion de la matrice d'*Eigen Channels* dégrade les performances puisque l'EER de 6% est supérieur à ceux des autres systèmes qui est de 4,6%. Cette différence est observable sur l'ensemble de la

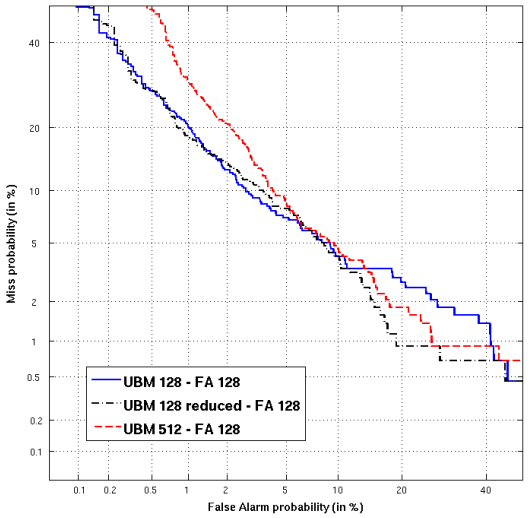

Figure 5.4 – Courbes DET obtenues pour un système étendu de 128 à 512 distributions sans aucune normalisation de score.

courbe DET de la figure 5.5. Il est probable que ce résultat soit dû à un processus de fusion et d'expansion trop simpliste.

D'autres approches de fusion ou d'expansion (comme des approches pondérées) pourraient être envisagées. Ces travaux n'ont pas pu être approfondis, mais on peut noter dans des travaux récents que l'utilisation de modèles dont le nombre de distributions augmente apporte un gain notable [Snyder et al., 2015] et il pourrait être intéressant d'exploiter les réseaux de neurones pour guider le processus de fusion et d'expansion. On pourrait envisager par exemple d'utiliser la méthode de calcul des statistiques proposée par [Lei et al., 2014] et présentée dans la section 4.1 en rajoutant une couche intermédiaire de dimension plus importante que la sortie dans le réseau afin de simuler un modèle du monde de dimension plus importante.

Ces travaux sont à mettre en relation avec ceux présentés sur l'estimation de la variabilité session dans l'espace des *i*-vecteurs dans la section, 5.3.2 car ils supportent des hypothèses similaires sur le sous-espace de variabilité canal, à savoir que l'estimation de cette variabilité peut être faite localement, mais que la variabilité canal dépend de la position dans l'espace des locuteurs.

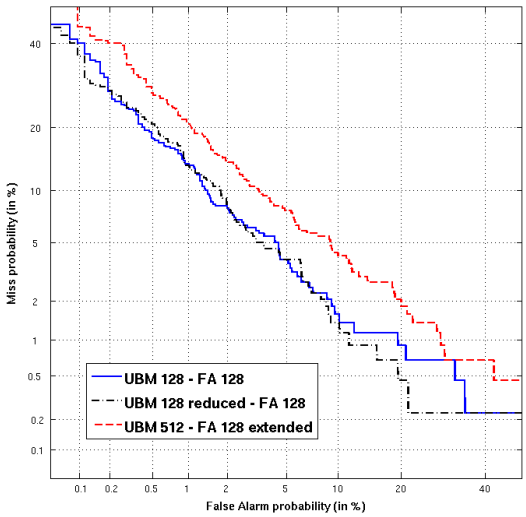

Figure 5.5 – Courbes DET obtenues pour un système étendu de 128 à 512 distributions avec normalisation de score ZT.

5.3. Normalisation des vecteurs

Les travaux présentés dans cette section ont été initiés par Pierre-Michel Bousquet afin de développer un processus de normalisation des *i*-vecteurs adapté aux classifieurs utilisés communément.

5.3.1 Analyse de la length normalization des i-vecteurs

Le modèle du *Factor Analyser* prévoit que les variables latentes de l'équation 3.3.1 suivent une distribution normale, $\mathcal{N}_x(\mathbf{x}|0,\mathbf{I})$. Dans [Garcia-Romero et Espy-Wilson, 2011] et [Bousquet et al., 2011], les auteurs montrent que ce n'est pas le cas. La distribution statistique de la norme euclidienne des *i*-vecteurs en particulier permet de montrer que leur distribution n'est pas Gaussienne alors que les modèles et traitements utilisés pour comparer les *i*-vecteurs (analyse linéaire discriminante, PLDA, modèle à deux covariances, distance de Mahalanobis) reposent sur cette hypothèse.

Aussi les auteurs de ces deux papiers proposent de corriger la distribution des *i*-vecteurs en la centrant (vecteur moyen nul), la réduisant et en divisant chaque *i*-vecteur par sa norme euclidienne. De ce fait, les *i*-vecteurs se retrouvent projetés sur une sphère unité centrée, ce qui améliore la distribution des *i*-vecteurs en la rapprochant d'une Gaussienne. Cette normalisation est communément appelée *length normalization* dans la littérature.

Soit $\mathcal{T}$ un ensemble de *i*-vecteurs disponible pour l'entraînement du système et p la dimension de l'espace des *i*-vecteurs . La normali-

sation appliquée aux *i*-vecteurs est décrite par l'algorithme suivant :

Pour $i = 1$ à $nb\_iterations$
  Calcule la moyenne $\overline{\mathbf{w}_i}$ et la matrice de covariance Σ_i de $\mathcal{T}$.
  Pour chaque $\mathbf{w}$ de $\mathcal{T}$:
    étape 1 : $\mathbf{w} \leftarrow \Sigma_i^{-\frac{1}{2}} (\mathbf{w} - \overline{\mathbf{w}_i})$

    étape 2 : $\mathbf{w} \leftarrow \frac{w}{\|w\|}$

La division des *i*-vecteurs par leur norme euclidienne améliore les performances des systèmes PLDA ou des systèmes reposant sur la distance de Mahalanobis, mais Bousquet et al. [2011] observent qu'après cette division, la distribution des *i*-vecteurs est moins Gaussienne. C'est pourquoi ils proposent d'appliquer cette normalisation de façon itérative afin de corriger la distribution statistique des *i*-vecteurs petit à petit, ce traitement est appelé *Eigen Factor Radial*.

Afin de visualiser l'effet de la division des *i*-vecteurs par leur norme euclidienne, nous présentons les graphes spectraux des distributions de *i*-vecteurs dans différentes bases. Le graphe spectral est constitué des représentations des données $diag\{\Sigma\}$, $diag\{\mathbf{B}\}$ et $diag\{\mathbf{W}\}$ dans une base choisie, où Σ, $\mathbf{B}$ et $\mathbf{W}$ sont respectivement les matrices de covariance totale, inter-et intra-classes de l'ensemble $\mathcal{T}$ de *i*-vecteurs d'entraînement.

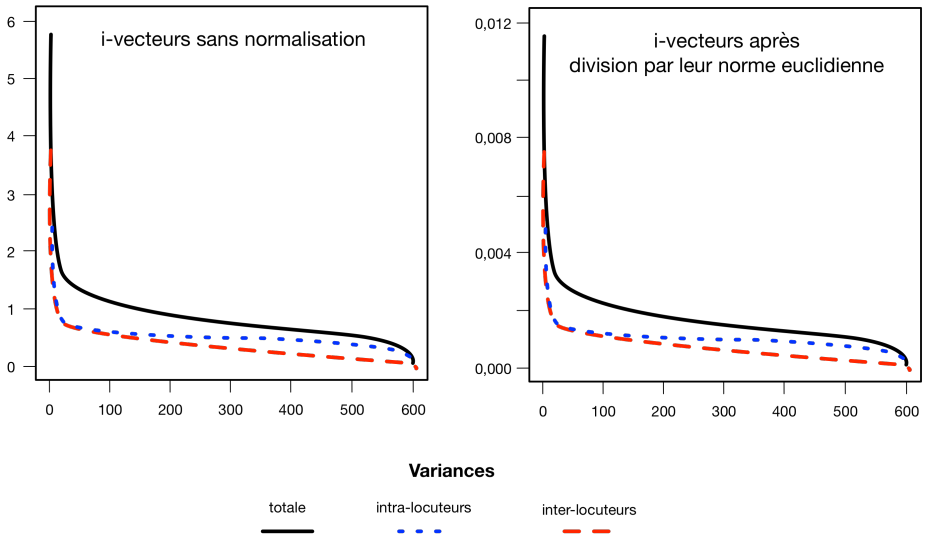

Figure 5.6 – Graphe spectral avant et après division des *i*-vecteurs par leur norme euclidienne.

La figure 5.6 représente les graphes spectraux avant et après division des *i*-vecteurs par leur norme euclidienne dans la base des vecteurs propres de la matrice de covariance totale, Σ. On observe que le graphe spectral n'est absolument pas modifié par ce traitement. Les

i-vecteurs ayant été projetés sur les vecteurs propres de Σ, les courbes pleines, noires de ces deux graphes représentent les valeurs propres de Σ, tandis que les courbes bleu et rouge ne correspondent pas aux valeurs propres de $\mathbf{B}$ et $\mathbf{W}$.

5.3.2 Normalisations spectrales

Cas de la LDA

L'analyse linéaire discriminante (LDA) est une technique de réduction de dimension qui tend à maximiser la variance inter-classes et minimiser la variance intra-classes afin de faciliter la séparation des classes. Le problème de la LDA consiste à maximiser le quotient de Rayleigh donné par :

$$\mathbf{J}(v) = \frac{v^t \mathbf{B} v}{v^t \mathbf{W} v} \tag{5.8}$$

En pratique, en reconnaissance du locuteur, les matrices de covariance inter- et intra-classes, $\mathbf{B}$ et $\mathbf{W}$, sont remplacées par les matrices de dispersion (*scattering matrices*) qui ne prennent pas en compte le nombre d'observations (*i*-vecteurs) de chaque locuteur.

L'objectif de la normalisation spectrale proposée dans [Bousquet et al., 2011] et analysée dans [Bousquet et al., 2012] est de projeter les *i*-vecteurs dans un espace qui maximise le quotient de Rayleigh. Après trois itérations de l'algorithme *Eigen Factor Radial*, le *i*-vecteur moyen est très proche du vecteur nul et la matrice, Σ, de covariance totale de l'ensemble d'apprentissage $\mathcal{T}$ est très proche de la matrice identité à un facteur près. Si on observe le graphe spectral des *i*-vecteurs après cette normalisation (dans la base de vecteurs propres de $\mathbf{B}$, cf. figure 5.7) on peut constater que la matrice Σ est très proche de $p^{-1}\mathbf{I}$ où p est la dimension de l'espace des *i*-vecteurs. Comme la variance totale est la somme des variances inter- et intra-classes , $\Sigma = \mathbf{B} + \mathbf{W}$, chaque vecteur propre de $\mathbf{B}$ est également vecteur propre de $\mathbf{W}$ et la somme de leur valeur propre correspondante est égale à p^{-1}. Plus de détails sont fournis dans [Bousquet et al., 2012]. Ainsi, les dimensions qui maximisent la variabilité inter-locuteurs minimisent la variabilité intra-locuteurs (comme on peut l'observer sur la figure 5.7) et la base, solution du problème de LDA après application de l'EFR, est formée par les premiers vecteurs propres de $\mathbf{B}$.

Cas de la PLDA

Le modèle PLDA cherche à estimer, entre autres, la matrice de covariance intra-locuteurs. Cette matrice est ensuite utilisée pour tous les locuteurs. Considérant que les *i*-vecteurs sont projetés sur une sphère, l'hypothèse selon laquelle tous les locuteurs, répartis

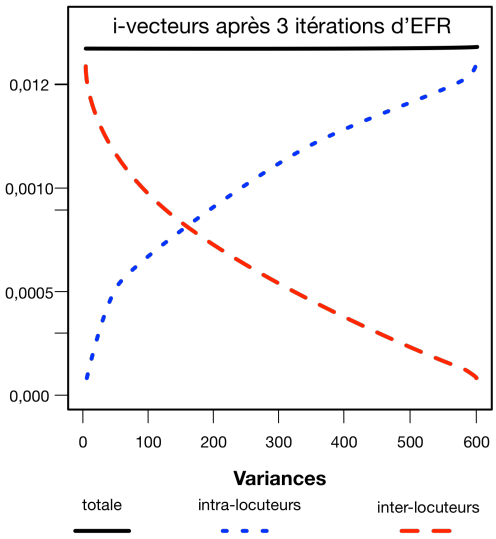

Figure 5.7 – Graphe spectral avant et après 3 itérations de la normalisation *Eigen Factor Radial*.

de part et d'autre de la sphère unité, partagent la même matrice de covariance intra-classe semble invraisemblable.

Il paraît donc opportun de trouver un espace dans lequel la variabilité intra-classe n'est portée par aucun axe de façon préférentielle. Pour obtenir cet espace, les *i*-vecteurs sont normalisés en remplaçant la matrice Σ dans l'algorithme EFR (cf. algorithme 5.3.1) par la matrice de covariance intra-classe $\mathbf{W}$. Après quelques itérations, la matrice $\mathbf{W}$ est presque égale à la matrice identité à un facteur près et le graphe spectral dans la base des vecteurs propres de $\mathbf{B}$ est donné sur la figure 5.8. Sur cette figure, on observe que la variabilité intra-classe est portée par tous les axes de façon presque égale tandis que les premiers vecteurs propres de $\mathbf{B}$ portent la partie la plus importante de la variabilité inter-locuteurs, produisant ainsi une base optimale pour la PLDA. Ce processus de normalisation est appelé *Spherical Nuisance Normalization*.

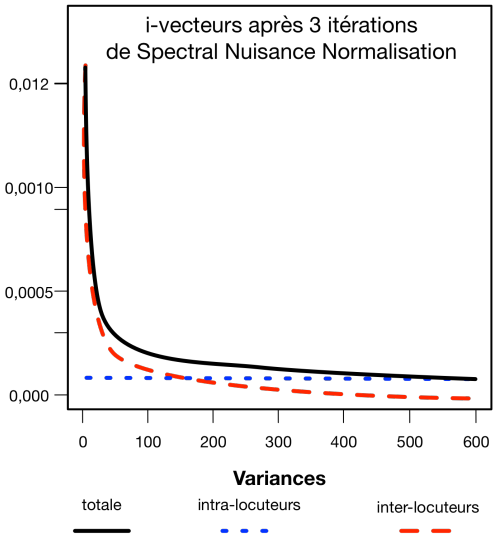

Figure 5.8 – Graphe spectral après 3 itérations de la *Spherical Nuisance Normalisation*.

Table 5.3 – Comparaison de cinq systèmes utilisant une LDA pour la condition 5 de la tâche principale de l'évaluation NIST-SRE 2010 donnés en termes d'EER et minDCF. Tous les systèmes utilisent comme classifieur un modèle à 2-covariances.

Normalisation	Type de LDA	hommes		femmes	
		EER	DCF	EER	DCF
length norm	LDA_{S_B,S_W}	**1.27**	**0.31**	2.27	0.38
EFR 1 iter.	LDA_{S_B,S_W}	1.36	0.33	2.29	0.39
EFR 1 iter.	$\text{LDA}_{B,W}$	1.36	0.30	**1.89**	**0.35**
EFR 1 iter.	LDA_{S_B,S_W}	1.30	0.32	2.30	0.39
EFRnorm 2 iter.	$\text{LDA}_{B,W}$	**1.27**	**0.31**	**1.89**	**0.35**

5.3.3 Performances et discussion

Les performances des deux normalisations proposées sont évaluées sur la condition 5 (téléphone-téléphone) de NIST-SRE 2010 en termes de taux d'égales erreurs (EER) et de fonction de coût de décision (DCF).

Conditionnement pour la LDA

Le tableau 5.3 présente les performances obtenues par un système *i*-vecteurs utilisant la LDA pour conditionner les vecteurs et un modèle 2-covariance comme décrit dans [Bousquet et al., 2012]. Après normalisation par *Eigen Factor Radial*, la LDA est réalisée en utilisant les matrices de covariance ($\text{LDA}_{B,W}$) ou avec les matrices de dispersion (LDA_{S_B,S_W}). Les meilleures performances sont obtenues pour une dimension de LDA de 80 pour l'ensemble des systèmes. Les résultats obtenus après une et deux itérations d'EFR sont comparés à une normalisation standard par *length normalization*. L'utilisation des matrices de covariance fonctionne mieux que les matrices de dispersion et 2 itérations d'EFR fournissent les meilleures performances ; augmenter le nombre d'itérations ne modifie pas ce résultat. La normalisation EFR améliore les performances pour les tests femme et maintient les performances de la *length normalization* pour les hommes, sans les améliorer. Ces résultats indiquent que la normalisation EFR suivie d'une LDA utilisant les matrices de covariance fournit les résultats optimaux pour un système à 2-covariances.

Conditionnement pour la PLDA

Le tableau 5.4 contient les résultats obtenus par deux systèmes PLDA précédés d'une *length normalization* ou de 2 itérations de *Spherical Nuisance Normalization*. Le rang de la matrice d'*Eigen Voices* de la PLDA est 80 et la matrice d'*Eigen Channels* est de rang

Table 5.4 – Comparaison de deux systèmes PLDA pour la condition 5 de la tâche principale de NIST SRE 2010 en termes d'EER et de minDCF.

Normalisation	hommes		femmes	
	EER	minDCF	EER	minDCF
Lenght normalization	1.22	0.32	1.81	0.34
Spherical Nuisance Normalization	**1.08**	**0.31**	**1.77**	**0.34**

plein. Pour palier à l'effet de l'initialisation aléatoire de la PLDA, l'expérience a été répétée 10 fois et les résultats présentés ici sont la moyenne de ces 10 répétitions.

L'utilisation de la *Spherical Nuisance Normalization* améliore les performances moyennes du système PLDA pour tous les indices. Cette normalisation permet au modèle PLDA de converger plus rapidement vers une solution qui fournit de meilleures performances en moyenne.

L'étude théorique des classifieurs a permis de modifier légèrement une approche déjà très répendue dans la littérature (*length normalization*) et d'en optimiser les performances pour la LDA et la PLDA. À l'avenir, ce type étude devrait être reconduit dans le contexte des *x*-vecteurs qui ne proviennent plus d'un modèle génératif Gaussien mais de réseaux de neurones et pour lesquels les hypothèses sous-jacentes diffèrent.

CHAPITRE 6

Approches probabilistes

Les travaux présentés dans cette section ont été réalisés à l'Institute for Infocomm Research de Singapour avec Kong Aik Lee, Chang Huai You, Haizhou Li, Ma Bin et Jiang Ye. Les deux sections ci-dessous présentent des améliorations apportées au modèle PLDA en considérant l'ensemble de la chaîne de traitement : extracteur de *i*-vecteurs et classification par PLDA. Il s'agit principalement de simplifications calculatoires et d'adaptation à des tâches spécifiques issues du protocole de l'évaluation NIST-SRE 2012.

6.1. Optimisation en grandes dimensions

Un système *i*-vecteurs PLDA opère deux réductions de dimension consécutives, chacune au moyen d'un *Factor Analyser* ; la première lors de l'extraction des *i*-vecteurs avec le modèle de variabilité totale et la seconde de manière implicite au sein du modèle PLDA lors du calcul du rapport de vraisemblances. Il est probable que cette double compression ne soit pas optimale et que l'extraction des *i*-vecteurs supprime une information utile à la PLDA qui est apprise de façon discriminante quand l'apprentissage du modèle de variabilité totale est non supervisé. Les travaux menés en 2012 avec Jiang Ye visent à supprimer une étape de compression de la chaîne de traitement [Jiang et al., 2012].

6.1.1 Difficultés et solutions

Des deux modèles qui opèrent une compression dans un système *i*-vecteurs PLDA, seule la PLDA est apprise de manière discriminante et apporte une valeur ajoutée là où l'extraction des *i*-vecteurs tente juste de perdre le moins d'information possible sans avoir aucune notion de classes. Il paraît donc opportun de supprimer la compression des super-vecteurs en *i*-vecteur (cf. section 3.3.1) et de directement présenter ces super-vecteurs à la PLDA. Deux difficultés se présentent alors.

— l'apprentissage du modèle de PLDA nécessite l'inversion de matrices de grande dimension ;

— en pratique, la distribution des *i*-vecteurs n'est pas exactement Gaussienne et pour obtenir des performances optimales, la PLDA Gaussienne requiert une normalisation des *i*-vecteurs . Cette normalisation peut varier, mais il est communément admis que la distribution des *i*-vecteurs doit être centrée et réduite [Bousquet et al., 2012; Garcia-Romero et Espy-Wilson, 2011]. En grande dimension, et du fait d'un nombre limité de sessions par locuteur, il est impossible de normaliser les super-vecteurs comme peuvent l'être des *i*-vecteurs de dimension plus faible.

Ces deux difficultés peuvent être résolues en calculant le modèle PLDA de manière efficace et en introduisant une normalisation appropriée pour les super-vecteurs, la *rank-norm* Gaussienne.

Utilisation efficace du modèle PLDA

Pour rappel, le modèle PLDA est régi par l'équation générative suivante :

$$\mathbf{x}_{i,s} = \boldsymbol{\mu} + \mathbf{F}\mathbf{h}_i + \mathbf{G}\mathbf{w}_{i,s} + \boldsymbol{\epsilon}_{i,s} \tag{6.1}$$

où un locuteur i est enregistré lors d'une session s. Ce modèle suppose que les distributions des variables $\mathbf{h}_i$, $\mathbf{w}_{i,s}$ et $\mathbf{x}_{i,s}$ respectent :

$$p(\mathbf{x}_{i,s}|\mathbf{h}_i, \mathbf{w}_{i,s}) = \mathcal{N}(\mathbf{x}_{i,s}|\boldsymbol{\mu} + \mathbf{F}\mathbf{h}_i + \mathbf{G}\mathbf{w}_{i,s}) \tag{6.2}$$

$$p(\mathbf{h}_i) = \mathcal{N}(O, \mathbf{I}) \tag{6.3}$$

$$p(\mathbf{w}_{i,s}) = \mathcal{N}(O, \mathbf{I}) \tag{6.4}$$

L'estimation du modèle PLDA nécessite un nombre important, I, de locuteurs disposant chacun d'un nombre J de sessions d'enregistrement. Le nombre de sessions J peut varier d'un locuteur à l'autre. Lors de l'étape E de l'algorithme EM, l'estimation de la moyenne

a posteriori nécessite l'inversion d'une matrice $\mathbf{L}$ de la forme :

$$\mathbf{L}^{-1} = \left[\begin{array}{c|cccc} J\mathbf{F}^T\Sigma^{-1}\mathbf{F}+\mathbf{I} & \mathbf{F}^T\Sigma^{-1}\mathbf{G} & \mathbf{F}^T\Sigma^{-1}\mathbf{G} & \cdots & \mathbf{F}^T\Sigma^{-1}\mathbf{G} \\ \hline \mathbf{G}^T\Sigma^{-1}\mathbf{F} & \mathbf{G}^T\Sigma^{-1}\mathbf{G}+\mathbf{I} & 0 & \cdots & 0 \\ \mathbf{G}^T\Sigma^{-1}\mathbf{F} & 0 & \mathbf{G}^T\Sigma^{-1}\mathbf{G}+\mathbf{I} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{G}^T\Sigma^{-1}\mathbf{F} & 0 & 0 & \cdots & \mathbf{G}^T\Sigma^{-1}\mathbf{G}+\mathbf{I} \end{array}\right]^{-1} \tag{6.5}$$

Cette matrice $\mathbf{L}$ est de très grandes dimensions puisque sa taille dépend du nombre total de sessions de chaque locuteur utilisé pour l'apprentissage de la PLDA. Alors qu'il est important de maximiser le nombre de sessions par locuteur pour estimer la variabilité due aux sessions de manière robuste, l'augmentation de cette dimension rend difficile l'inversion de $\mathbf{L}$. Cette matrice peut être inversée grâce au complément de Schur qui fournit une solution efficace [Prince, 2012]. Ainsi une matrice L de la forme :

$$\mathbf{L}^{-1} = \left[\begin{array}{cc} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{array}\right] \tag{6.6}$$

peut être inversée si la matrice $\mathbf{D}$ est inversible. Dans notre cas, l'inversibilité de la matrice $\mathbf{D}$ est garantie, car cette matrice est diagonale par bloc et que chacun de ses blocs est lui même inversible.

Ainsi nous avons :

$$\mathbf{L}^{-1} = \left[\begin{array}{cc} \mathbf{M} & -\mathbf{M}\mathbf{B}\mathbf{D}^{-1} \\ -\mathbf{D}^{-1}\mathbf{C}\mathbf{M} & \mathbf{D}^{-1}+\mathbf{D}^{-1}\mathbf{C}\mathbf{M}\mathbf{B}\mathbf{D}^{-1} \end{array}\right] \tag{6.7}$$

La matrice $\mathbf{M}$ étant donnée par :

$$\mathbf{M} = \left[J\mathbf{F}^T\Sigma^{-1}(\mathbf{F}-\mathbf{G}\Lambda)+\mathbf{I}\right]^{-1} \tag{6.8}$$

$$\Lambda = \mathbf{Q}\mathbf{G}^T\Sigma^{-1}\mathbf{F} \tag{6.9}$$

$$\mathbf{Q} = (\mathbf{G}^T\Sigma^{-1}\mathbf{G}+\mathbf{I}) \tag{6.10}$$

Grâce à cette formulation, les équations des étapes E et M du processus d'apprentissage du modèle PLDA peuvent être exprimées en fonction des sous-matrices de $\mathbf{L}$ comme suit :

$$E[h_i] = \mathbf{M}\left[\sum_{j=1} J\mathbf{F}^T\Sigma^{-1}\mathbf{x'}_{i,j}\right] - \mathbf{M}\Lambda^T\left[\sum_{j=1} J\mathbf{G}^T\Sigma^{-1}\mathbf{x'}_{i,j}\right] \tag{6.11}$$

$$E[w_{i,j}] = \mathbf{Q}\left(\mathbf{G}^T\Sigma^{-1}\mathbf{x'}_{i,j}\right) - \Lambda E[h_i] \tag{6.12}$$

où $\mathbf{x}'_{i,j} = \mathbf{x}_{i,j} - \boldsymbol{\mu}$ est le vecteur (i-vecteur ou super-vecteur) centré grâce à la moyenne globale $\boldsymbol{\mu} = \frac{1}{NJ}\sum_{i,j}\mathbf{x}_{i,j}$. Durant l'étape M, les paramètres du modèle PLDA (les matrices $\mathbf{F}$ et $\mathbf{G}$) sont ré estimés comme suit :

$$[\mathbf{FG}] = \left\{\sum_{i,j}\mathbf{x}'_{i,j}E[\widetilde{\mathbf{w}}_{i,j}]^T\right\}\left\{\sum_{i,j}E[\widetilde{\mathbf{w}}_{i,j}\widetilde{\mathbf{w}}_{i,j}^T]\right\}^{-1} \tag{6.13}$$

$$\Sigma = \frac{1}{NJ}\sum_{i,j} diag\left[\mathbf{x}'_{i,j}{\mathbf{x}'_{i,j}}^T\right] - [\mathbf{FG}]\,E[\widetilde{\mathbf{w}}_{i,j}]{\mathbf{x}'_{i,j}}^T \tag{6.14}$$

où $\widetilde{\mathbf{w}}_{i,j}^T = \left[\mathbf{h}_i^T\mathbf{w}_{i,j}^T\right]$ et $\mathbf{h}_i$, variable cachée d'un locuteur, est concaténée à la variable cachée $\mathbf{w}_{i,j}$ de chacune des sessions de ce locuteur. Le moment du second ordre est calculé via :

$$E[\widetilde{\mathbf{w}}_{i,j}\widetilde{\mathbf{w}}_{i,j}^T] = \begin{bmatrix} \mathbf{M} & -\mathbf{M}\Lambda^T \\ -\Lambda\mathbf{M} & \mathbf{Q}+\Lambda\mathbf{M}\Lambda^T \end{bmatrix} + E[\widetilde{\mathbf{w}}_{i,j}]E[\widetilde{\mathbf{w}}_{i,j}]^T \tag{6.15}$$

Normalisation dans un espace de grande dimension

Une condition nécessaire aux bonnes performances de la PLDA réside dans la Gaussianité de la distribution des observations ; c'est pourquoi la distribution des vecteurs est généralement centrée et réduite afin de satisfaire aux hypothèses du modèle Gaussien. Dans le cas de vecteurs de grande dimension (ici les super-vecteurs), il n'est pas possible de réduire la distribution par cause de manque de données. Une solution consiste à utiliser une *rank norm* Gaussienne [Stolcke et al., 2008] qui transforme les super-vecteurs de la façon suivante :

$$r^{(m)} = \frac{|\left\{b \in B^{(m)} : b < \Phi_l^{(m)}\right\}|}{|B^{(m)}|} \tag{6.16}$$

où $\Phi^{(m)}$ est l'élément m du super-vecteur Φ et $|\dot{|}$ est l'opérateur qui renvoie le cardinal d'un ensemble. Chaque élément du super-vecteur est transformé pour respecter une distribution Gaussienne et non pas uniforme comme dans la *rank norm* originale. La valeur normalisée est ensuite obtenue en utilisant l'inverse de la fonction de répartition Gaussienne (la fonction $probit$) de la façon suivante :

$$\Phi_l^{(m)} \leftarrow \sqrt{2}erf^{-1}(2r^{(m)} - 1) \tag{6.17}$$

Après cette normalisation, les super-vecteurs sont divisés par leur norme euclidienne comme dans le cas de la *length norm* telle qu'appliquée aux i-vecteurs .

Table 6.1 – Comparaison de la PLDA appliquée aux *i*-vecteurs et super-vecteurs pour différentes normalisations sur la condition 6 de la tâche principale de NIST-SRE 2008 en termes d'EER et minDCF.

i-vecteurs	hommes		femmes	
	EER	minDCF	EER	minDCF
Aucune normalisation	6,1785	3,1206	8,1486	3,7028
Division par la norme euclidienne	4,9411	2,6286	6,4409	3,0581
Réduction et division par la norme euclidienne	4,5458	2,4546	6,3193	3,0065
super-vecteurs	hommes		femmes	
	EER	minDCF	EER	minDCF
Aucune normalisation	5,2632	2,6605	6,7976	3,3368
Division par la norme euclidienne	4,9199	2,6271	6,3667	3,3624
rank norm Gaussienne et division par la norme euclidienne	4,8982	2,6676	6,0976	3,2588

6.1.2 Résultats et discussion

Les performances de la PLDA appliquée aux super-vecteurs sont évaluées sur la condition 6 de la tâche principale de NIST-SRE08 et rapportées en termes d'EER et de minDCF pour les hommes et les femmes séparément. Un système *i*-vecteurs est appris pour chaque genre comprenant un modèle du monde à 512 distributions, et une matrice de variabilité totale de rang 500. Les paramètres acoustiques sont constitués de 18 MFCC et de leurs dérivées premières et secondes. La dimension des super-vecteurs est de 27 648 alors que les *i*-vecteurs utilisés pour comparaison sont de dimension 500. Les rangs des matrices **F** et **G** du modèle PLDA sont respectivement 300 et 200 dans le cas des super-vecteurs alors que pour les *i*-vecteurs le rang de **F** est fixé à 300, la matrice **G** est supprimée et la matrice Σ utilisée est pleine.

Le tableau 6.1 présente les performances des systèmes utilisant des *i*-vecteurs ou des super-vecteurs pour différentes normalisations appliquées avant la PLDA. Pour chaque type de vecteur, la première ligne présente le cas des vecteurs bruts (non normalisés), la deuxième ligne le cas des vecteurs divisés par leur norme euclidienne, la troisième ligne le cas où la distribution des *i*-vecteurs est réduite tandis que pour les super-vecteurs on applique la *rank norm* Gaussienne puis une division par la norme euclidienne.

Quel que soit le type de vecteur, on constate tout d'abord que la division par la norme euclidienne améliore toujours les performances de la PLDA et que la réduction de la distribution ou l'application de la *rank norm* améliore encore celles-ci.

Lorsque la PLDA est appliquée sur les vecteurs bruts, les super-vecteurs obtiennent des EER et minDCF inférieurs à ceux des *i*-vecteurs pour les hommes comme pour les femmes. Ce résultat pourrait s'expliquer par le fait que les super-vecteurs contiennent plus

d'information que les *i*-vecteurs qui ont déjà subi une compression, mais également par le fait que la distribution des super-vecteurs pourrait être plus Gaussienne que celle des *i*-vecteurs .

Lorsque les vecteurs sont divisés par leur norme euclidienne, l'écart observé entre *i*-vecteurs et super-vecteurs se réduit considérablement ce qui laisse supposer que c'est la Gaussianité de la distribution des super-vecteurs qui leur permet d'obtenir de meilleures performances lorsqu'aucune normalisation n'est appliquée.

Enfin lorsque les *i*-vecteurs sont blanchis et divisés par leur norme euclidienne, ils obtiennent de meilleures performances que les super-vecteurs auxquels on applique une *rank norm* Gaussienne et une division par leur norme euclidienne. Ce comportement est observable similairement pour les hommes et les femmes. La normalisation appliquée aux super-vecteurs n'apporte pas autant de gain que le blanchissement appliqué aux *i*-vecteurs et ce travail pourrait être étendu afin de trouver une normalisation des super-vecteurs plus adaptée à la PLDA. On peut noter toutefois que les performances offertes par la PLDA appliquée aux super-vecteurs restent relativement proches de celle d'un système *i*-vecteur standard en faisant l'économie d'une première compression. Cette réduction du coût de calcul pourrait être utile dans le cas d'une application avec des ressources limitées.

6.2. Enrôlement multi-session et identification en milieu partiellement ouvert

En 2012, le NIST organise une évaluation dans laquelle chaque locuteur dispose d'un nombre variable de sessions d'enrôlement et où le segment de test peut provenir d'un des locuteurs enrolé dans le système ou d'un inconnu. Ce scénario dépasse donc le cadre de la vérification pure, car l'information des locuteurs connus peut être utilisée et l'a priori sur la connaissance du locuteur peut être fixé dans le système afin de satisfaire au cadre opérationnel. Le travail réalisé avec Kong Aik Lee, Chang Huai You, Ma Bin et Haizhou Li [Lee et al., 2013] vise dans un premier lieu l'exploitation optimale des multiples sessions d'enrôlement disponibles par locuteur et vise ensuite à généraliser le calcul du score PLDA pour permettre l'utilisation de celle-ci en milieu partiellement ouvert.

6.2.1 Gestion de multiples sessions d'enrôlement

Les scénarios considérés pour la reconnaissance du locuteur étant fortement influencés par les données disponibles et donc par les directions choisies par les sponsors institutionnels ou commerciaux du domaine, il est fréquent de considérer qu'une seule session

d'enrôlement est disponible par locuteur. Dans le cas où plusieurs sessions sont disponibles, les systèmes PLDA classiques nécessitent de se rapporter au cas d'un seul *i*-vecteur par locuteur. On peut alors accumuler les statistiques de toutes les sessions disponibles pour extraire un *i*-vecteur unique pour ce locuteur ou moyenner les *i*-vecteurs de toutes les sessions disponibles pour obtenir un vecteur unique. Dans le cadre de l'évaluation NIST-SRE 2012 nous avons souhaité obtenir une solution théorique permettant de répondre à ce problème en utilisant plusieurs *i*-vecteurs d'enrôlement pour un même locuteur.

Calcul d'un score utilisant plusieurs sessions d'enrôlement

Dans le paradigme de la PLDA, considérer que plusieurs *i*-vecteurs proviennent d'un même locuteur revient à considérer que ces *i*-vecteurs sont générés à partir de la même valeur de variable $\mathbf{h}_i$ dans l'équation générative rappelée ci-dessous :

$$\mathbf{x}_{i,s} = \boldsymbol{\mu} + \mathbf{F}\mathbf{h}_i + \mathbf{G}\mathbf{w}_{i,s} + \boldsymbol{\epsilon}_{i,s} \tag{6.18}$$

Nous faisons l'hypothèse que les *i*-vecteurs sont tous statistiquement indépendants, ce qui nous permet d'exprimer, pour un modèle PLDA donné, la vraisemblance d'un ensemble de *i*-vecteurs provenant d'un même locuteur de façon simple comme décrit par l'équation ci-dessous :

$$p(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_R|\Theta_{PLDA}) = \mathcal{N}\left(\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\\ \vdots\\ \mathbf{x}_R\end{bmatrix} = \begin{bmatrix}\boldsymbol{\mu}_1\\ \boldsymbol{\mu}_2\\ \vdots\\ \boldsymbol{\mu}_R\end{bmatrix}, \Omega\Omega_R^T + \mathbf{S}_R\right) \tag{6.19}$$

où

$$\Omega_R = \begin{bmatrix}\mathbf{F} & \mathbf{G} & \cdots & 0\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{F} & 0 & \cdots & \mathbf{G}\end{bmatrix} \tag{6.20}$$

$$\mathbf{S}_R = \begin{bmatrix}\Sigma & \cdots & 0\\ \vdots & \ddots & \vdots\\ 0 & \cdots & \Sigma\end{bmatrix} \tag{6.21}$$

Le logarithme de la vraisemblance des *i*-vecteurs sur le modèle PLDA est donné par :

$$\log(p(\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_R|\Theta_{PLDA})) = \frac{1}{2}\left[\sum_{r=1}^{R}\mathbf{F}^T\mathbf{J}\mathbf{y}_r\right]^T\mathbf{K}_R\times\left[\sum_{r=1}^{R}\mathbf{F}^T\mathbf{J}\mathbf{y}_r\right] + \frac{1}{2}\alpha(R) - \frac{1}{2}\sum_{r=1}^{R}\mathbf{y}_r^T\mathbf{J}\mathbf{y}_r - Rc \tag{6.22}$$

où R est le nombre de sessions du locuteur, $c = 0,5 \times (D \log(2\pi) - \log|\mathbf{J}|)$ est un scalaire constant pour un modèle PLDA donné, $\mathbf{y}_r$ est le i-vecteur centré $\mathbf{y}_r = \mathbf{x}_r - \boldsymbol{\mu}$ et $\alpha(R) = \log|\mathbf{K}_R|$ est le logarithme du déterminant de la matrice dépendant du nombre de sessions du locuteur courant.

Les deux matrices de précision : $\mathbf{J}$ et $\mathbf{K}_R$ sont données par :

$$\mathbf{J} = \left[\mathbf{G}\mathbf{G}^T + \Sigma\right]^{-1} \tag{6.23}$$

$$\mathbf{K}_R = \left[R\mathbf{F}^T\mathbf{J}\mathbf{F} + \mathbf{I}\right]^{-1} \tag{6.24}$$

L'équation 6.22 est utilisée pour calculer le rapport de vraisemblances pour la tâche de reconnaissance du locuteur. Il apparaît que seuls les deux premiers termes de cette expression sont utiles alors que les termes suivants s'annuleront dans le rapport. Le calcul du rapport de vraisemblances est détaillé dans la section suivante.

6.2.2 Généralisation au cas d'un milieu partiellement ouvert

Soit un scénario dans lequel N locuteurs sont connus du système. Nous considérons ici le cas d'une authentification en milieu ouvert. Selon le paradigme de la PLDA, chaque locuteur est représenté par un vecteur $\mathbf{h}_l$. Dans le cas du milieu ouvert, l'hypothèse selon laquelle le i-vecteur de test est généré par un locuteur inconnu (c.-à-d. n'appartenant pas à l'ensemble des N locuteurs connus) signifie qu'un locuteur inconnu, représenté par un vecteur $\mathbf{h}_{N+1}$ a généré ce i-vecteur .

Ces hypothèses sont représentées sous forme de modèles graphiques dans la figure 6.1 pour $N = 3$. Chacun des modèles $\mathcal{M}_l$ avec $l = 1, 2, ..., N$ considère que le i-vecteur de test $\mathbf{x}_t$ appartient au locuteur l modélisé par la variable $\mathbf{h}_l$. Les flèches allant de la variable $\mathbf{h}_l$ aux observations $\mathbf{x}_t$ et $\mathbf{x}_{l,r}$ avec $\{\mathbf{x}_{l,r}\}_{r=1}^{R}$ indiquent l'appartenance de ces observations au locuteur l. Le modèle $\mathcal{M}_{N+1}$ ($\mathcal{M}_4$ sur la figure 6.1) représente l'hypothèse du milieu ouvert selon laquelle le i-vecteur de test est généré par un locuteur inconnu.

On note ici L_l la vraisemblance du modèle $\mathcal{M}_l$ dont l'expression sera donnée par la suite. Pour la vérification du locuteur, on calcule le logarithme du rapport de vraisemblances entre l'hypothèse nulle et les hypothèses alternatives. Ce rapport est décrit par :

$$s_l(\mathbf{x}_t) = \log\left[\frac{L_l(\mathbf{x}_t)}{(P_{connu})\frac{1}{N-1}\sum_{k\neq l} L_k(\mathbf{x}_t) + (1 - P_{connu})L_{N+1}(\mathbf{x}_t)}\right] \tag{6.25}$$

La vraisemblance de l'hypothèse nulle est donnée par $L_l(\mathbf{x}_t)$ au numérateur. Le dénominateur est composé de deux termes. Le premier terme dépend de l'ensemble de

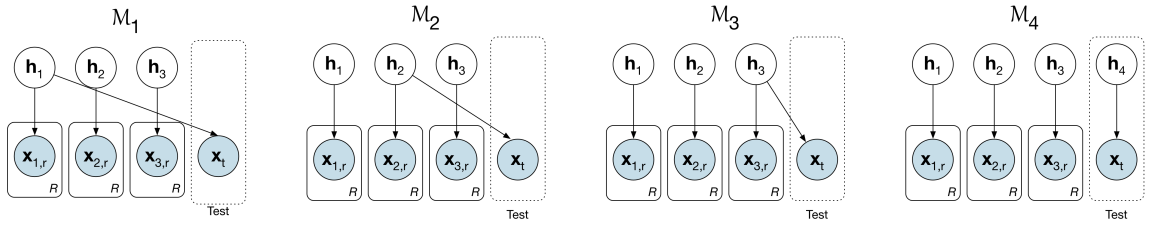

Figure 6.1 – Modèle graphique de l'identification en milieu ouvert pour le cas de 3 locuteurs connus. Pour simplifier ce schéma, les variables dépendantes de la session **w** ne sont pas représentées et chaque locuteur dispose de R sessions d'enrôlement représentées par les i-vecteurs $\{\mathbf{x}_l, r\}_{r=1}^{R}$. Chacun des modèles représente une hypothèse sur l'identité du locuteur générant le i-vecteur de test $\mathbf{x}_t$.

locuteurs connus tandis que le second terme correspond au cas où le i-vecteur de test appartient à un locuteur inconnu (cas du milieu ouvert). P_{connu} est la probabilité que le i-vecteur de test appartienne à un des N locuteurs connus. Cette probabilité détermine le poids de chacun des termes dans l'hypothèse alternative et il est clair que le cas où $P_{connu} = 1$ correspond à une identification en milieu fermé tandis que le cas où $P_{connu} = 0$ correspond au cas de la vérification.

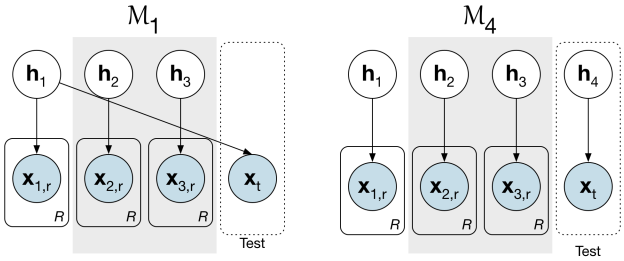

Figure 6.2 – Lors du calcul du rapport de vraisemblances, un certain nombre de termes s'annulent et permettent une implémentation simplifiée du calcul du score. Dans cet exemple, les parties grisées sont communes aux deux termes et s'annuleront lors du calcul du rapport de vraisemblances.

On note $\Lambda_l = \frac{L_l}{L_{N+1}}$ le rapport de vraisemblances correspondant au rapport du modèle $\mathcal{M}_l$ avec le modèle correspondant au cas de l'identification en milieu ouvert $\mathcal{M}_{N+1}$. On observe sur la figure 6.2 qu'une grande partie des termes des vraisemblances sont communs entre les deux quantités (numérateur et dénominateur) ; plus précisément les parties correspondant aux locuteurs connus autres que le locuteur l. On exprime alors le score de l'équation 6.25 en fonction de Λ_l

$$s_l(\mathbf{x}_t) = \log\left[\frac{\Lambda_l(\mathbf{x}_t)}{(P_{connu})\frac{1}{N-1}\sum_{k\neq l}\Lambda_k(\mathbf{x}_t) + (1 - P_{connu})}\right] \tag{6.26}$$

On observe que Λ_{N+1} est égal à 1. L'expression du logarithme de Λ_l se simplifie par

annulation des termes communs et devient :

$$\log \Lambda_l(\mathbf{x}_t) = \log p(\mathbf{x}_t, \mathbf{x}_{l,1}, ..., \mathbf{x}_{l,R}|\Theta_{PLDA}) - \log p(\mathbf{x}_t|\Theta_{PLDA}) - \log p(\mathbf{x}_{l,1}, ..., \mathbf{x}_{l,R}|\Theta_{PLDA}) \tag{6.27}$$

Le troisième terme du logarithme du rapport de vraisemblances peut être calculé grâce à l'équation 6.22 et les premier et deuxième termes peuvent être calculés grâce à la formule suivante, car ils correspondent au cas où le locuteur possède $R+1$ et 1 sessions respectivement.

$$\begin{aligned}\log \Lambda_l(\mathbf{x}_t) = &\frac{1}{2}\left[\sum_{r=1}^{R}\mathbf{y}_r + \mathbf{y}_t\right]^T \mathbf{K}_{R+1}\left[\sum_{r=1}^{R}\mathbf{y}_r + \mathbf{y}_t\right] + \frac{1}{2}\alpha(R+1) \\ &- \frac{1}{2}\left[\sum_{r=1}^{R}\mathbf{y}_r\right]^T \mathbf{K}_R\left[\sum_{r=1}^{R}\mathbf{y}_r\right] - \frac{1}{2}\alpha(R) \\ &- \frac{1}{2}\mathbf{y}_t^T\mathbf{K}_1\mathbf{y}_t - \frac{1}{2}\alpha(1)\end{aligned}$$

Afin de simplifier les notations, les *i*-vecteurs ont été centrés réduits comme suit :

$$\mathbf{y} \leftarrow \mathbf{F}^T\mathbf{J}\mathbf{y} \tag{6.28}$$

6.2.3 Résultats et discussion

Un système *i*-vecteurs est appris par genre. Ces systèmes disposent de modèles du monde à 512 distributions et d'un extracteur d'*i*-vecteurs de dimension 600. Les matrices de variabilité totales sont constituées de deux matrices concaténées : l'une apprise sur des données téléphoniques et l'autre apprise avec des données provenant de microphones divers. Les paramètres cepstraux utilisés sont 19 MFCC avec leurs dérivées premières et secondes. Une LDA est appliquée pour réduire la dimension des *i*-vecteurs à 400 et les rangs des matrices **F** et **G** des modèles PLDA sont respectivement 250 et 50.

Afin d'évaluer l'effet du nombre de sessions d'enrôlement sur les performances du système PLDA, nous sélectionnons les locuteurs cibles de l'évaluation NIST-SRE 2012 disposant d'au moins 10 sessions d'enrôlement et faisons varier le nombre de sessions utilisées de 1 à 10. Les taux d'égales erreurs obtenus sont présentés sur la graphique 6.3. Ces résultats confirment le fonctionnement du score multi-sessions puisque le taux d'erreurs diminue lorsque le nombre de sessions utilisées augmente.

La figure 6.4 présente les distributions de scores client et imposteur pour les systèmes PLDA utilisant un score multi-sessions avec une et quatre sessions d'enrôlement. Malgré

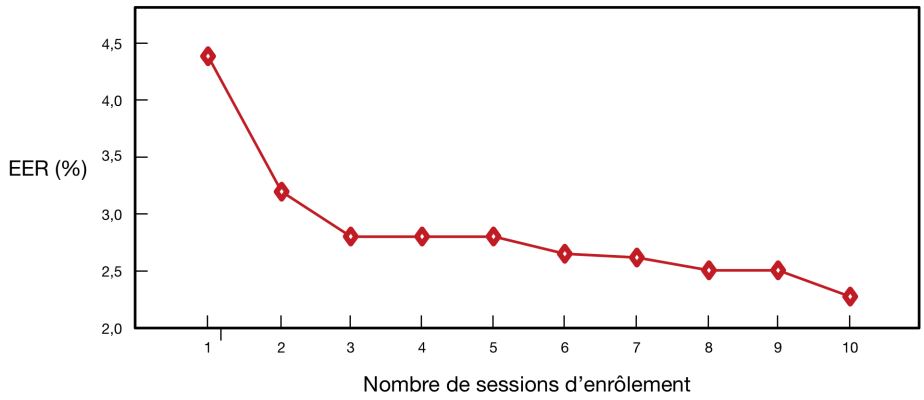

Figure 6.3 – Taux d'égales erreurs (EER) en fonction du nombre de sessions d'enrôlement utilisées par locuteur. Performances évaluées sur la condition 2 de la tâche principale de NIST-SRE 2012.

l'amélioration (réduction) du taux d'égales erreurs lorsque le nombre de sessions augmente, on observe une dilatation des distributions de scores lorsque le nombre de sessions augmente. Cet effet est gênant lorsque le nombre de sessions d'enrôlement varie entre locuteurs en créant un défaut de calibration des scores. Cet effet est probablement dû à l'hypothèse d'indépendance des *i*-vecteurs qui n'est pas vérifiée. L'erreur de calibration

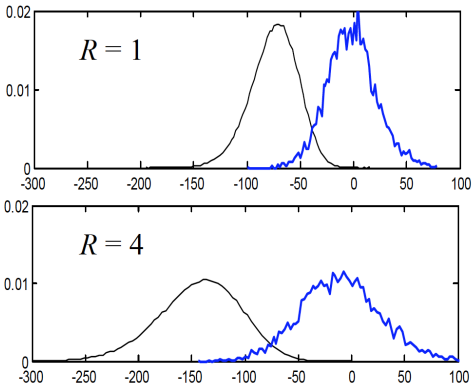

Figure 6.4 – Distributions des scores clients et imposteurs pour une ($R = 1$)) ou quatre ($R = 4$) sessions d'enrôlement.

créée par le nombre variable de sessions d'enrôlement pourrait être compensée par une normalisation des scores (*s-norm* [Kenny, 2010]). Une normalisation des scores dépendante du nombre de sessions d'enrôlement pourrait également être envisagée et laisse des pistes de recherche à explorer.

Dans un deuxième temps, nous évaluons l'efficacité du score PLDA dérivé pour le cas du milieu partiellement ouvert. En utilisant toujours la condition principale de NIST-SRE 2012 qui pondère les distributions de scores des locuteurs connus et inconnus en utilisant P_{connu}. Le tableau 6.2

Lorsque le poids des scores des locuteurs connus augmente, les performances du système de vérification simple se dégradent, ce qui nous informe que le jeu de locuteurs

Table 6.2 – Performances du système *i*-vecteurs PLDA sur la tâche principale de NIST-SRE 2012 selon le poids donné aux scores des locuteurs connus et inconnus avec et sans utilisation du score proposé pour la prise en compte du milieu partiellement ouvert.

P_{connu}	Taux d'égales erreur (EER %)	
	Score de vérification	Score proposé
0	2,5704	2,5704
1/4	2,5972	2,4340
1/2	2,6244	2,2981
3/4	2,6910	2,1757
1	2,7468	1,5774

connu est plus *difficile* que le jeu de locuteurs inconnus. Cependant, lorsqu'on utilise le score prenant en compte un a priori sur le ratio de tests de locuteurs connus ou non, on observe (dans la dernière colonne du tableau) que les performances s'améliorent lorsque l'a priori est utilisé. La prise en compte de l'a priori amène au maximum à une diminution de 43% du taux d'égales erreurs dans le cas ou $P_{connu} = 1$.

Les résultats présentés dans cette partie montrent que le modèle PLDA peut être optimisé pour travailler en grandes dimensions, qu'il a pu être modifié pour tirer parti de plusieurs sessions d'enrôlement disponible par locuteur en dérivant le modèle théorique au lieu de recourir à des raccourcis empiriques. L'ensemble des méthodes et améliorations proposées dans cette partie pour les systèmes *i*-vecteurs PLDA ont été intégrées et rendues publiques dans les plateformes libres ALIZE et SIDEKIT.

CHAPITRE 7

Approches neuronales

Les travaux présentés ici ont été réalisés durant la thèse de Gaël Le Lan [Le Lan, 2017] en collaboration avec Delphine Charlet d'Orange Labs. et Sylvain Meignier. La thèse de Gaël avait pour cadre la segmentation et le regroupement en locuteurs, mais je ne présenterai ici que ce qui a trait à la reconnaissance du locuteur (regroupement). Nous reviendrons sur le thème principal de cette thèse pour évoquer les perspectives dans lesquelles elle s'inscrit, notamment dans le thème de l'apprentissage non supervisé.

La tâche de segmentation et regroupement en locuteur [Meignier, 2015] consiste à découper un ou plusieurs documents audio en segments ne contenant que du signal de parole d'un unique locuteur. Ces segments sont ensuite regroupés selon l'identité du locuteur afin d'indiquer dans ces documents « qui parle ? et « à quels moments ?» Cette tâche complexe nécessite plusieurs passes de traitement du signal audio. Dans le cadre de ce document, je ne m'intéresserai qu'à la sous-tâche de regroupement en locuteur qui consiste à regrouper les segments précédemment détectés afin de regrouper ceux qui ont été produits par un même locuteur. Le nombre de locuteurs présents dans les documents n'est généralement pas connu et cette tâche s'apparente donc à de l'identification en milieu ouvert qui peut être décomposée comme une succession de tâches de vérification.

L'approche proposée dans le cadre de la thèse de Gaël et publiée dans [Le Lan et al., 2017] repose sur un réseau de neurones associé à une simple similarité Cosine pour remplacer la PLDA.

7.1. Description du réseau et de la Triplet-Loss

Comme je l'ai expliqué dans le chapitre 4, l'utilisation de réseaux de neurones pour la tâche de vérification du locuteur n'est pas intuitive. La tâche de vérification consiste à classer des couples de segments audio selon qu'ils ont été prononcés par un même locuteur ou par deux locuteurs différents (cf. figure 3.3 pour le cas de la PLDA). Les principales difficultés liées à cette tâche sont :

- la représentation acoustique des segments de parole est complexe et de dimension importante ;
- il est impossible de présenter au réseau tous les locuteurs existant lors de l'apprentissage, le réseau doit donc être capable de généraliser ;
- étant donné un ensemble de n segments audio, les combinaisons possibles de *couples* pour la vérification sont au nombre de :

$$C_n^2 = \frac{n!}{2 \times (n-2)!} \tag{7.1}$$

  et l'apprentissage d'un réseau de neurones utilisant l'ensemble des exemples possible nécessite une puissance de calcul importante.

Afin de présenter au réseau de neurones une représentation des segments audio de dimension réduite et fixe (afin de simplifier l'architecture), nous avons choisi d'utiliser la représentation en i-vecteurs qui fait référence dans le domaine à l'époque de ces travaux.

L'architecture Triplet-Ranking [Wang et al., 2014] vise à projeter les i-vecteurs de façon non linéaire sur une sphère unité de façon à maximiser la distance inter-classes (entre locuteurs) et minimiser la distance intra-classes (intra-locuteurs). La projection est réalisée au moyen d'un réseau de neurones entièrement connecté à une couche suivie d'une fonction d'activation : tanh dans notre cas.

Soit un triplet de i-vecteurs (Φ_a, Φ_p, Φ_n) choisi tel que Φ_a soit le i-vecteur de référence (*anchor*), Φ_p un exemple de la même classe que la référence (*positif*) et Φ_n un exemple *négatif*, n'appartenant pas à la classe de la référence. La fonction de coût (Triplet-Loss) à optimiser est donnée par :

$$\mathcal{L}(\mathcal{T}) = \sum_i^N \max(0, \Delta_i + \alpha) \tag{7.2}$$

où N est le nombre de triplets possibles, α est une marge fixée pour forcer la séparation des classes dans le nouvel espace et :

$$\Delta_i = -\frac{f(\Phi_a^i) f(\Phi_p^i)^T}{\|f(\Phi_a^i)\| \|f(\Phi_p^i)\|} + \frac{f(\Phi_a^i) f(\Phi_n^i)^T}{\|f(\Phi_a^i)\| \|f(\Phi_n^i)\|} \tag{7.3}$$

Idéalement, nous espérons obtenir $\Delta_i + \alpha < 0$ pour tous les triplets i. Ain d'optimiser l'apprentissage du réseau, seuls les triplets vérifiant $0 < \Delta_i + \alpha < \alpha$ sont sélectionnés. Ainsi pour chaque époque d'apprentissage, pour chaque locuteur possédant au moins 3 i-vecteurs , pour les pairs (Φ_a^i, Φ_p^i), un exemple négatif est sélectionné parmi les i-vecteurs générant un des k plus proches voisins ($k - NN$) dans l'espace des *embeddings* (c.-à-d. après projection) en s'assurant que pour le triplet formé respecte : $0 < \Delta_i + \alpha < \alpha$. Tous les triplets ainsi formés sont utilisés pour mettre à jour le réseau.

7.2. Évaluation du Triplet-Ranking

Les travaux de Gaël Le Lan se situent dans le cadre de la segmentation et du regroupement en locuteur. Aussi les performances de l'approche par *Triplet-Ranking*ont été évaluées sur deux corpus de radio et télévision françaises issus des évaluations REPERE [Galibert et Kahn, 2013] et ETAPE [Galibert et al., 2014]. La description complète des données et du protocole utilisé est donnée dans [Le Lan et al., 2017]. Une description succincte des deux corpus utilisés est donnée dans le tableau 7.1

La figure 7.1 présente l'évolution des taux d'égales erreurs (EER) et de la fonction de coût minimum (minDCF) telle que définie pour l'évaluation NIST-SRE 2008 en fonction du nombre d'époques d'apprentissage du réseau de neurones *Triplet-Ranking*. La dimension de la couche cachée du réseau est fixée à 200, qui est également la dimension des i-vecteurs extraits pour cette tâche. Les résultats sont présentés pour plusieurs valeurs de la marge α et comparés à deux systèmes de référence : un système i-vecteurs PLDA et un système utilisant une normalisation WCCN et une similarité cosine.

Les expériences ont été effectuées pour des valeurs de α variant de 0,4 à 1,0, mais une courbe sur deux est produite ici pour plus de lisibilité. Pour chaque valeur de α, le résultat présenté est une moyenne de 20 expériences. Pour les deux métriques utilisées, le réseau de neurones est, en moyenne, meilleur que la PLDA.

Dans la littérature, les réseaux utilisant une *Triplet loss* sont entraînés avec un grand nombre d'exemples par classe. Or, les données disponibles varient entre 3 et 59 sessions par

Table 7.1 – Composition des corpus utilisés pour l'évaluation du *Triplet-Ranking*.

Corpus	LCP	BFM
Nombre d'épisodes	45	42
Durée de parole annotée	10 h 8 min	19 h 57 min
# Locuteurs apparaissant une seule fois	127	345
# locuteurs apparaissant au moins 2 fois	93	77
# locuteurs apparaissant au moins 3 fois	48	35

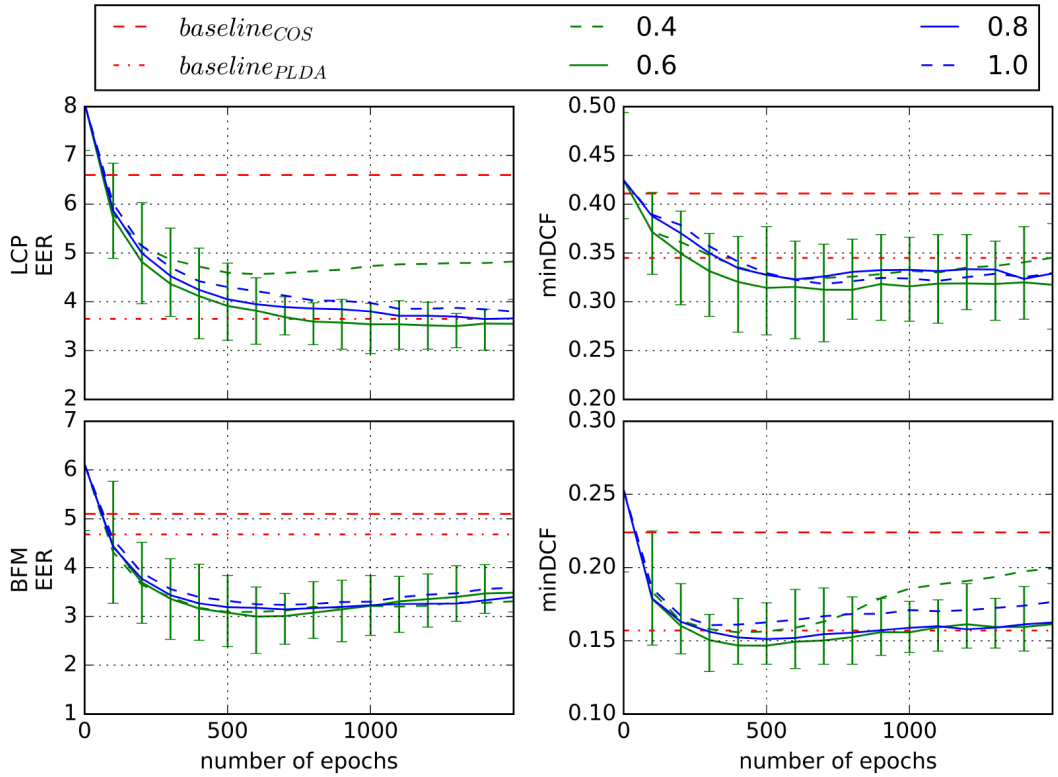

Figure 7.1 – Évolution de l'EER et de la minDCF pour différentes valeurs de la marge, α, lors de l'apprentissage du *Triplet-Ranking* comparé à deux systèmes de référence : un système *i*-vecteurs PLDA et un système utilisant une normalisation WCCN et une similarité Cosine. Les résultats présentés pour le réseau de neurones sont une moyenne de 20 répétitions de cette expérience.

locuteur, ce qui limite le nombre de paires positives par locuteur. Utiliser toutes les paires possibles entraînerait un déséquilibre entre classes qui pourrait être préjudiciable pour le système. Nous étudions ici l'influence du nombre d'exemples par classe pour l'apprentissage du modèle. Un autre inconvénient est dû au fait qu'avec un faible nombre d'exemples par classe, certaines classes ne contribuent plus à la *loss* après quelques époques. Les résultats pour 1, 3 et 5 exemples par classe sont présentés par la figure 7.2. Chaque expérience est moyennée après 20 tests.

Il apparaît qu'utiliser un seul triplet par locuteur n'est pas suffisant, car le système a besoin d'apprendre la variabilité due au locuteur. L'utilisation de trois triplets par classe fournit les meilleurs résultats en termes d'EER et de minDCF.

Les travaux menés dans le cadre de la segmentation et du regroupement en locuteur ont permis de montrer qu'un réseau de neurones appris avec une *Triplet Loss* pouvait améliorer les performances d'une PLDA classique. Le classifieur ainsi obtenu a pu être utilisé lors de l'évaluation NIST-SRE 2016 au sein du consortium *I4U* [Lee et al., 2017] et a montré les mêmes performances. Cependant, il faudrait prolonger ces travaux pour évaluer l'impact des conditions acoustiques sur les performances du classifieur. On peut également envisager d'utiliser une augmentation de données en ajoutant du bruit et de la

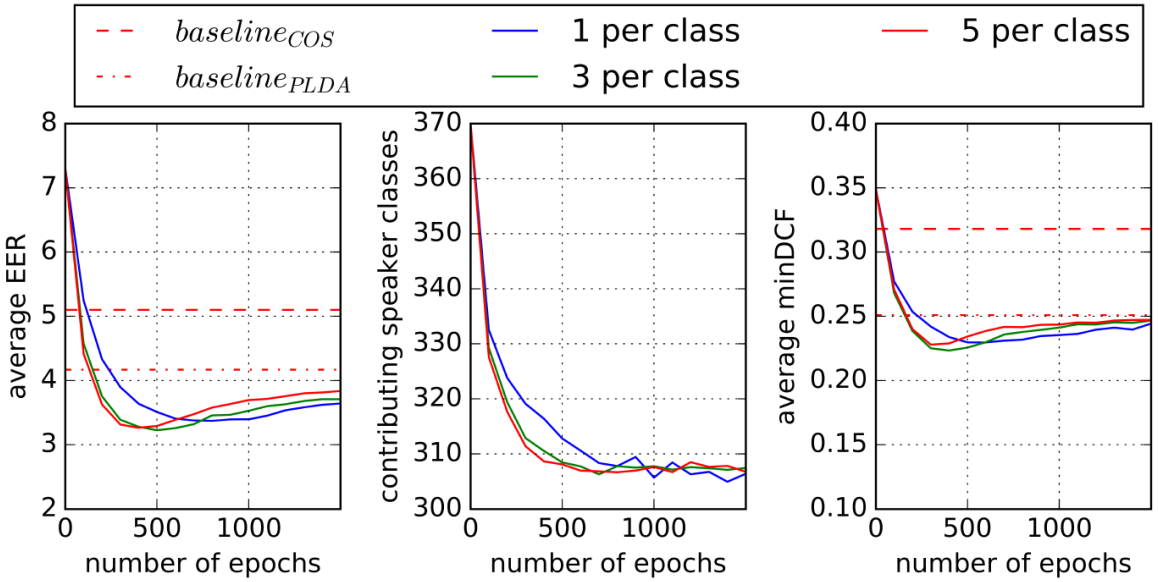

Figure 7.2 – Évolution de l'EER et de la minDCF moyens lors de l'apprentissage du *Triplet-Ranking* en fonction du nombre d'exemples choisis dans chaque classe. La moyenne est obtenue à partir de 20 répétitions de l'expérience. Le nombre de classes (locuteurs) contribuant à l'apprentissage est également représenté. Les résultats sont comparés à deux systèmes de référence : un système *i*-vecteurs PLDA et un système utilisant une normalisation WCCN et une similarité Cosine. Les résultats présentés pour le réseau de neurones sont une moyenne de 20 répétitions de cette expérience.

réverbération il est devenu fréquent pour d'autres approches mettant en jeu des réseaux de neurones.

Discussions

Les travaux que j'ai menés en reconnaissance du locuteur indépendante du texte sont en grande partie motivés par des contraintes techniques en vue d'optimiser l'aspect calculatoire ou de lever des verrous technologiques ralentissant le déploiement de la technologie. Ainsi, la plupart de mes travaux ont donné lieu à la publication de codes source dans des logiciels libres : ALIZE et SIDEKIT.

ALIZE

Certainement la plateforme libre la plus utilisée actuellement pour la reconnaissance du locuteur dans le monde de l'industrie. ALIZE intègre toutes les approches Gaussiennes évoquées dans la partie 4.2.1 et son implémentation en C++ facilite le déploiement, notamment dans un environnement mobile sur lequel l'accent a été mis ces dernières années. J'ai personnellement intégré une implémentation optimisée du *Factor Analyser* dans ALIZE afin d'offrir un outil « clefs en main » pour déployer des systèmes standards performants.

SIDEKIT

L'écriture de ce logiciel, dont je suis le principal auteur, est motivée par une volonté de fournir un code simple, lisible et très efficace aux acteurs académiques et industriels. Développée en Python, cette plateforme intègre une modélisation simple qui facilite les modifications et le développement tout en tirant parti des outils récents de parallélisation du calcul, d'exploitation de clusters de grande dimension et en offrant une grande compatibilité avec les outils disponibles en apprentissage automatique qui sont pour la plupart développés dans le même langage (Python).

Les efforts fournis pour développer et maintenir ces outils s'intègrent dans ma volonté de faciliter le transfert technologique, mais également de catalyser le développement de nouvelles approches en permettant au plus grand nombre de chercheurs d'implémenter des approches standards auxquels ils peuvent se comparer.

Troisième partie

Modélisation du locuteur pour les courtes durées

En reconnaissance du locuteur, on peut distinguer deux principaux cadres applicatifs.

1. La surveillance pour laquelle il est possible de recueillir des enregistrements de parole de plusieurs dizaines de secondes ou minutes. C'est le cas traité dans les campagnes d'évaluations du NIST qui supposent que des conversations téléphoniques de plusieurs minutes sont accessibles pour l'enrôlement et le test.
2. L'authentification utilisée pour contrôler l'accès à des lieux, des applications, des services qui nécessite une procédure très courte pour ne pas rebuter les utilisateurs. C'est le cadre des applications commerciales ou des contrôles d'accès. Dans ce cadre applicatif, l'acceptabilité de la reconnaissance vocale par les utilisateurs est fortement liée au temps de parole nécessaire au bon fonctionnement du système d'authentification. Le système doit répondre en 2 ou 3 secondes.

Bien sûr, d'autres applications de la technologie existent, par exemple le contrôle permanent de l'identité qui vise à vérifier en temps réel et de façon continue que l'utilisateur d'un système est autorisé et qu'il ne change pas au cours du temps, ou encore la segmentation et le regroupement en locuteur pour l'analyse de documents multimédia ou la transcription enrichie de réunions qui ont été abordés dans le chapitre 7. Cependant, les deux cadres applicatifs considérés ici mettent en lumière une des problématiques principales en reconnaissance du locuteur : la durée des sessions d'enrôlement et de test. De nombreuses études montrent que les taux d'erreurs des systèmes de vérification du locuteur sont inversement proportionnels à la durée des enregistrements disponibles [Fauve, 2009; Fauve et al., 2007a; Ferrer et al., 2003; Hasan et al., 2013; Kenny et al., 2013; Mandasari et al., 2013; Martinez et al., 2014; McCree et al., 2008; Vogt et Sridharan, 2009; Vogt et al., 2008, 2009].

Dans cette partie je m'appliquerai à mettre en lumière un des facteurs qui affecte la reconnaissance du locuteur avec des segments de courte durée, à savoir la variabilité phonétique. L'impact de ce facteur sera d'abord illustré pour un système GMM-UBM [Reynolds et al., 2000] et plus en détail pour un système *i*-vecteurs . Je présenterai ensuite des utilisations possibles de la normalisation des *i*-vecteurs et du modèle PLDA dans ce cadre particulier des courtes durées avant d'étudier le lien entre variabilité locuteur et variabilité phonétique dans l'espace des *i*-vecteurs . Les travaux présentés ici sont issus de [Larcher et al., 2012a, 2013] mais également d'études non publiées à ce jour.

Dans un deuxième chapitre, je traiterai de la reconnaissance du locuteur dépendante du texte qui est un cas particulier de la reconnaissance du locuteur pour l'instant exclusivement dédié aux courtes durées, mais qui pourrait s'étendre aux durées plus longues dans l'avenir. Dans ce chapitre, je reviendrai sur les travaux initiés avec Jean-François Bonastre et Corinne Fredouille [Larcher et al., 2013] qui ont été étendus avec Kong Aik Lee, Ma Bin et Haizhou Li pour permettre la publication de deux brevets [Larcher et al., 2013a,b] ainsi que l'intégration du système développé dans le téléphone LENOVO A586 commercialisé

en 2012 [Larcher et al., 2014a,b,c]. Je décrirai ensuite comment l'architecture du système développé peut être utilisée pour classer les tests en différentes catégories propres à la reconnaissance du locuteur dépendante du texte.

CHAPITRE 8

Étude des modèles existant dans le contexte des courtes durées

Les systèmes de reconnaissance du locuteur, comme décrits dans la partie 4.2.1 obtiennent des représentations des locuteurs en procédant à une somme (ou moyenne) au cours du temps (cf. figures 3.8 et 4.2) et ne tiennent donc pas compte de la structure temporelle de la parole. Lorsque les segments de parole considérés sont de durée assez longue (de l'ordre de la minute), la distribution des phonèmes qu'ils contiennent se rapproche de la distribution des phonèmes dans la langue considérée. En revanche, deux segments très courts (quelques secondes) peuvent contenir des phonèmes différents ; cette disparité crée une variabilité phonétique qui affecte sévèrement les systèmes de reconnaissance du locuteur.
Afin d'étudier l'effet du contenu phonétique sur les performances des systèmes de reconnaissance du locuteur, nous utilisons dans cette partie des corpus, protocoles et une terminologie provenant de la vérification du locuteur dépendante du texte.

La reconnaissance du locuteur dépendante du texte vise à améliorer les performances en intégrant un a priori sur le contenu phonétique prononcé. Cet a priori peut provenir d'une contrainte imposée au locuteur par l'utilisation d'une phrase commune, le choix d'un mot de passe personnalisé ou la lecture d'un prompteur. On pourrait aussi considérer le cas où le texte prononcé, reconnu a posteriori par un système de reconnaissance de la parole, est utilisé pour améliorer les performances de la reconnaissance du locuteur. Dans ce document, je distinguerai ce cas de la reconnaissance du locuteur dépendante du texte qui impose au locuteur, au moins lors de la phase de test, le texte à prononcer. Le cas

où l'information phonétique est obtenue à partir du signal de parole sera discuté dans les perspectives.

Selon cette définition de la reconnaissance du locuteur dépendante du texte, un système est confronté à quatre types de tests selon l'identité du locuteur : locuteur cible ou imposteur et selon le texte prononcé : correct ou faux :

Client-correct le locuteur cible prononce le texte attendu par le système ;

Imposteur-correct un imposteur prononce le texte attendu par le système ;

Client-faux le locuteur cible prononce un texte différent de celui attendu par le système

Imposteur-faux un imposteur prononce un texte différent de celui attendu par le système

Parmi ces quatre types de tests, un seul doit être classé positif par le système : *Client-correct* tandis que les trois autres types doivent être rejetés. Le type *Imposteur-correct* correspond au cas où un imposteur a connaissance de la phrase à prononcer, typiquement le cas où elle est commune à tous les locuteurs, où un imposteur lit le prompteur ou le cas où un imposteur a usurpé le mot de passe du client. Le cas des *Client-faux* peut correspondre à un play-back pour lequel un malfaiteur aurait enregistré le client, mais sans réussir à voler son mot de passe. C'est aussi le cas d'un play-back joué alors qu'un prompteur aléatoire affiche un texte différent de celui enregistré. Le cas des tests *Imposteur-faux* correspond à une imposture naïve sans connaissance du mot de passe personnalisé. Ces quatre types de tests correspondent à des tests « naturels » qui ne font intervenir aucune technologie comme la synthèse ou la transformation vocale [Lorenzo-Trueba et al., 2018].

Les résultats présentés dans cette partie utilisent pour les parties indépendantes du texte, les protocoles et bases de données des évaluations NIST-SRE, [Martin et Greenberg, 2010] mais également, pour le cas des tests dépendant du texte, sur la partie 1 de la base de données *RSR2015* [Larcher et al., 2014c]. Ce corpus regroupe des enregistrements de 300 locuteurs enregistrés lors de 9 sessions. Dans la partie 1 de *RSR2015*, les locuteurs prononcent 30 phrases courtes, dans la partie 2 ils prononcent 30 commandes vocales et dans la partie 3 ils prononcent trois séquences aléatoires de 10 chiffres et 10 séquences aléatoires de 5 chiffres.

8.1. Effet du contenu phonétique en courtes durées

Dans le cas de segments acoustiques de courtes durées, le contenu phonétique prononcé lors de l'enrôlement et du test affecte les performances des systèmes de reconnaissance du locuteur.

8.1.1 Cas d'un système GMM-UBM

Afin de mettre en évidence l'effet du contenu phonétique sur les performances des systèmes de vérification du locuteur, une première expérience est réalisée avec un système UBM-GMM. La figure 8.1 représente les distributions de scores hommes d'un système GMM-UBM à 128 distributions pour la partie 1 de la base de données *RSR2015*. Les modèles de locuteurs sont appris par adaptation MAP d'un modèle du monde dépendant du genre.

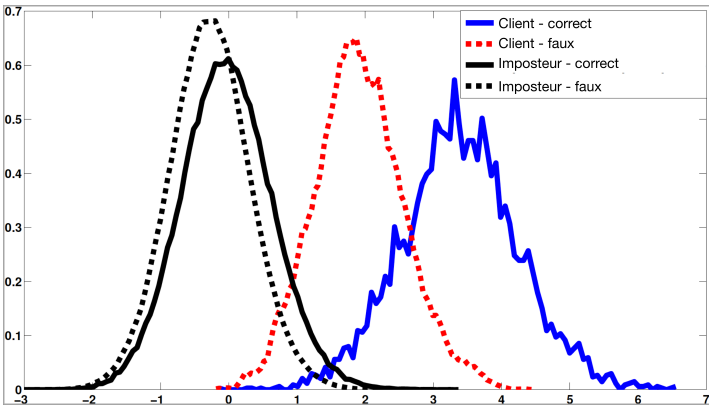

Figure 8.1 – Distributions des scores *Client-correct*, *Client-faux*, *Imposteur-correct*, *Imposteur-faux* pour un système GMM-UBM à 128 distributions pour la partie 1 de la base de données *RSR2015*.

En comparant les distributions de score 2 à 2 ; *Client-correct*et *Client-faux*d'une part, et *Imposteur-correct*et *Imposteur-faux*d'autre part, on observe l'influence du contenu phonétique sur les scores d'un système de reconnaissance du locuteur. Les scores obtenus par les clients lorsque les contenus phonétiques sont similaires sont nettement plus élevés que si les contenus phonétiques diffèrent. Cet effet est moins important pour les imposteurs où l'effet observé se mélange à l'effet de la variabilité locuteur. Ceci explique la suprématie de la reconnaissance du locuteur dépendante du texte sur la reconnaissance indépendante du texte pour des durées de quelques secondes.

8.1.2 Cas d'un système i-vecteurs

Les systèmes *i*-vecteurs souffrent également des différences de contenu phonétique entre segments d'enrôlement et segments de test [Larcher et al., 2012a]. Le gain en performance obtenu en contraignant le contenu phonétique des segments d'enrôlement et de test peut être observé sur la figure 8.2

Dans le cas où le texte prononcé lors du test par les clients et les imposteurs diffère du texte prononcé par le locuteur cible durant l'enrôlement, le taux d'égales erreurs

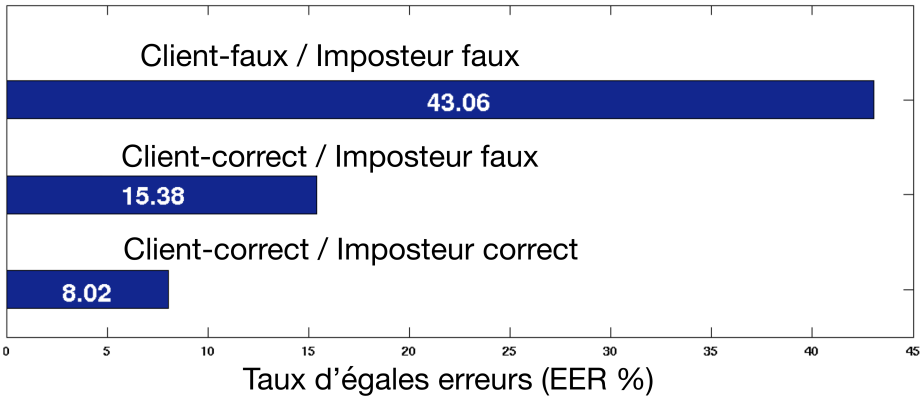

Figure 8.2 – Taux d'égales erreurs (%) obtenus sur le jeu d'évaluation de la partie 1 de la base de données *RSR2015*pour différentes définitions des contenus phonétiques de test.

(EER) est de 43, 06%. Ce cas correspond à un cas extrême de reconnaissance du locuteur indépendante du texte. Rappelons que les sessions d'enrôlement et de test sont très courtes (environ 2 secondes de parole), et que les phrases sélectionnées pour la partie 1 de *RSR2015*ont un recouvrement phonétique limité. L'EER diminue de 74% relatif (à 15, 36%) si les clients et les imposteurs prononcent le même texte durant le test que celui prononcé par les clients pendant l'enrôlement. Enfin l'EER diminue jusqu'à 8, 02% si seuls les clients prononcent le mot de passe correct durant le test (même contenu phonétique : test Client-correct) alors que les imposteurs prononcent un texte différent (contenu phonétique différent : test Imposteur-faux). Dans ce cas, le système tire avantage de la connaissance du mot de passe par les clients alors que les *i*-vecteurs des imposteurs diffèrent des données d'enrôlement en termes de locuteur et de contenu phonétique.

Cette expérience montre que comme pour les systèmes GMM-UBM, l'information phonétique portée par les *i*-vecteurs peut être utilisée pour améliorer les performances des systèmes de reconnaissance du locuteur pour les courtes durées. Par la suite, je propose une analyse qui vise à déterminer si il est possible de séparer l'information locuteur de l'information phonétique.

8.2. Analyse fréquentielle pour la reconnaissance du locuteur dépendante du texte

Cette section contient des analyses qui n'ont jamais été publiées. L'objectif est ici de mettre en évidence les différentes informations : phonétique, et locuteur, disponibles dans le signal de parole dans le cadre de segments de courte durée, d'analyser leur répartition fréquentielle et d'exploiter les résultats de cette étude dans le cadre de la reconnaissance du locuteur dépendante du texte. Ces analyses sont à mettre en relation avec les travaux

de Laurent Besacier qui a analysé la distribution de l'information caractéristique du locuteur par bandes de fréquences [Besacier et al., 2000] sans toutefois analyser le ratio d'information locuteur et phonétique.

Les résultats présentés dans cette section sont obtenus sur la base de données *RSR2015* qui est enregistrée avec un échantillonnage de 16kHz. Nous comparons les résultats obtenus par deux systèmes en ne faisant varier que la bande de fréquence considérée pour l'extraction des paramètres acoustiques. Ainsi, tous les systèmes présentés utilisent 20 MFCC extraits toutes les 10ms sur une fenêtre glissante de 25ms grâce au logiciel SPRO 5 [1].

Les dérivées premières et secondes des MFCC leur sont adjointes avant l'étape de détection de la parole qui est effectuée en classifiant les vecteurs acoustiques en trois classes selon leur niveau d'énergie. Il faut noter que la détection parole/ non-parole est effectuée une seule fois en utilisant la bande de fréquences complète [0-8000Hz] et que ce résultat est utilisé pour toutes les configurations. Le nombre de vecteurs acoustiques utilisés par tous les systèmes est donc identique.

8.2.1 Analyse d'un système GMM-UBM

Un système GMM-UBM est utilisé pour modéliser chaque locuteur en utilisant 3 répétitions des 30 phrases de la partie 1, développement, de *RSR2015* et une adaptation MAP. Lors des tests, les quatre types de tests décrits dans le chapitre précédent sont considérés (*Client-correct*, *Client-faux*, *Imposteur-correct* et *Imposteur-faux*).

Six expériences sont réalisées au cours desquelles les MFCC sont extraits sur la bande de fréquence complète [0-8000], et sur cinq bandes de fréquences réduites. Les bandes de fréquences (en Hz) sont réduites à [0-6000], [0-5000], [0-4000], [0-3000], [0-2000] et [0-1000].

Les performances obtenues par le système GMM-UBM sont présentées dans le tableau 8.1 pour les différentes bandes de fréquences considérées et pour différentes définitions des tests imposteurs.

Dans la première ligne du tableau 8.1, les deux types de scores : positifs et négatifs font intervenir le locuteur cible. Dans les tests positifs, il prononce la phrase correcte et dans le cas négatif une phrase différente parmi les 29 autres phrases de la partie 1. La tâche assignée au système est donc une tâche de reconnaissance du texte prononcé en considérant un locuteur constant. Les performances du système sont très mauvaises lorsqu'on limite la bande de fréquences à [0-1000Hz] et le taux d'égales erreurs est de

1. https://gforge.inria.fr/projects/spro, vu le 6 septembre 2018

Table 8.1 – Taux d'égales erreurs (EER %) pour différentes définitions des tests positifs et négatifs d'un système GMM/UBM et différentes bandes de fréquences. Les fréquences indiquées dans la deuxième ligne du tableau sont les fréquences maximales de la bande utilisée pour l'extraction des coefficients MFCC.

Locuteur	Cible		Imposteur		GMM-UBM (% EER)							
Texte	Correct	Faux	Correct	Faux	1kHz	2kHz	3kHz	4kHz	5kHz	6kHz	7kHz	8kHz
Tests	positif	négatif	-	-	19.35	3.66	2.22	2.26	2,58	3,08	3.34	3,39
	positif	-	négatif	-	15,45	8,28	4,87	3,47	2,96	2.69	2,47	2,38
	positif	-	-	négatif	9,12	1,44	0,56	0,40	0,40	0.43	0,45	0,49
	-	positif	-	négatif	31,73	35,43	30,36	25,84	22,97	21.18	19,25	20,38
	-	-	positif	négatif	38,14	21,98	20,37	21,17	21.49	22,67	24,18	25,48

19,35%. Comme on peut s'y attendre, l'élargissement de la bande de fréquences améliore nettement les performances jusqu'à [0-3000Hz]. Par la suite, les performances se dégradent pour obtenir 3,39% d'EER lorsque la bande complète [0-8000Hz] est utilisée. Il semble que l'information sur le contenu phonétique présente dans les segments audio soit dégrégadée lorsqu'on utilisé les hautes fréquences. La cinquième ligne du tableau correspond à la même tâche : reconnaissance du texte prononcé, mais on considère cette fois-ci des locuteurs différents entre l'enrôlement et le test. La tâche consiste ici à reconnaître le texte prononcé, mais le locuteur n'est plus le même pour les deux segments comparés. Les taux d'erreurs obtenus sont beaucoup plus importants, +612% en moyenne que dans le cas précédent, ce qui s'explique par la variabilité locuteur qui brouille la reconnaissance du texte. On note cependant que l'évolution des taux d'erreurs en fonction de la bande de fréquence utilisée suit la même tendance. Un minimum est obtenu pour une bande de [0-3000Hz].

La deuxième ligne du tableau présente les taux d'égales erreurs obtenus pour une tâche de reconnaissance du locuteur dépendante du texte. On considère que le texte prononcé est identique au sein d'un couple de segments d'enrôlement et de test. Le taux d'égales erreurs décroît lorsque la bande de fréquences d'extraction des MFCCs augmente, et ce, jusqu'à la bande maximale [0-8000Hz]. La quatrième ligne du tableau correspond au cas de la reconnaissance du locuteur indépendante du texte : les tests positifs corerspondent au *Client-correct*et les tests négatifs aux *Imposteur-faux*. Dans ce cas, le texte prononcé lors du test est toujours différent du test d'enrôlement. Les performances sont nettement moins bonnes que pour le cas précédent, car la variabilité due au texte affecte la reconnaissance du locuteur. On observe cependant la même évolution : plus la bande de fréquence est large et plus les taux d'erreurs diminuent.

La troisième ligne du tableau montre les performances obtenues pour une tâche de reconnaissance du locuteur où seuls les locuteurs ciblent prononcent la phrase d'enrôlement. C'est le cas idéal de la reconnaissance du locuteur dépendante du texte : le système discrimine à la fois le locuteur et le texte. En augmentant la bande de fréquence, l'EER décroît jusqu'à [0-4000Hz] et augmente par la suite. On peut y voir un mélange entre le cas de la

reconnaissance du locuteur et la reconnaissance du texte, mais les différences d'EER ne sont pas significatives au-delà d'une bande de fréquences de [0-4000Hz]

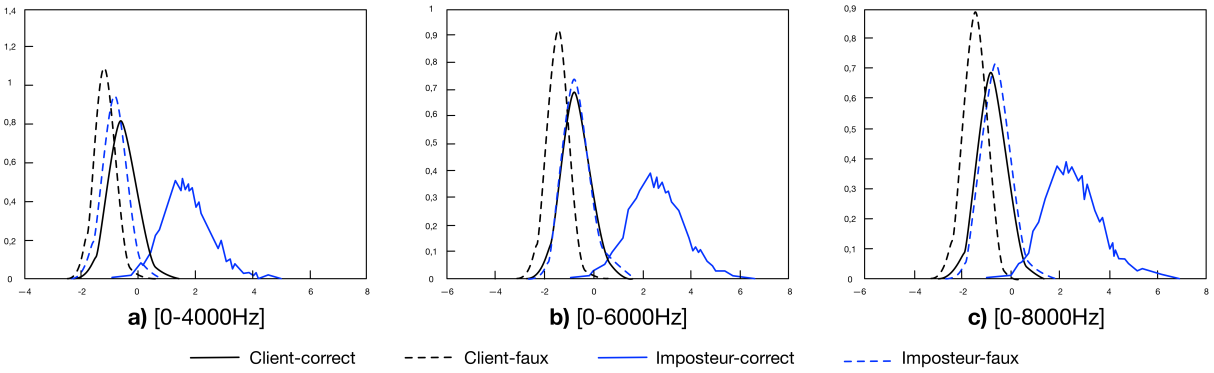

Figure 8.3 – Distributions des scores d'un système GMM-UBM obtenus sur le jeu d'évaluation de la partie 1 de la base de données *RSR2015* pour les 4 types de scores d'un système de reconnaissance de locuteur dépendant du texte.

La figure 8.3 représente les distributions de 4 types de scores pour 3 bandes de fréquences : [0-4000Hz], [0-6000Hz], [0-8000Hz]. Les scores obtenus pour lorsque le locuteur et le texte sont identiques (*Client-correct*) entre enrôlement et tests sont toujours les plus élevés. Les scores obtenus lorsque le locuteur et le texte diffèrent (*Imposteur-faux*) sont toujours les plus faibles. Cependant, on observe que les distributions des scores *Client-faux* et *Imposteur-correct* effectuent une permutation lorsque la bande de fréquence varie. Pour une bande de basses fréquences [0-4000Hz], les scores *Imposteur-correct* sont plus élevés que les scores *Client-faux*, il semble que le système distingue mieux le texte que le locuteur. Pour une bande de fréquence plus large [0-6000Hz], le système distingue le locuteur et le texte avec autant de précision puisque les deux distributions de scores *Client-faux* et *Imposteur-correct* sont superposées. Enfin lorsque la bande de fréquence est maximale [0-8000Hz], le système discrimine légèrement plus facilement le locuteur que le texte.

Dans cette expérience il faut considérer que les deux informations : phonétique et locuteur représentent chacune un bruit pour la reconnaissance de l'autre, aussi il n'est pas facile de quantifier la quantité d'information locuteur ou phonétique disponible, mais on peut plutôt juger du « ratio » d'information locuteur et phonétique présent dans la bande de fréquence considérée. En formulant l'hypothèse selon laquelle d'autres informations ou bruits présents dans le signal de parole peuvent être considérés comme du bruit pour les deux tâches, on peut formuler les conclusions suivantes :

— plus la bande de fréquence est étendue vers les hautes fréquences et plus le ratio $\frac{locuteur}{texte+bruits\ autres}$ augmente.
— le ratio $\frac{texte}{locuteur+bruits\ autres}$ décroît lorsque la bande de fréquences augmente de [0-1000Hz] à [0-3000Hz] et augmente par la suite.

8.2.2 Analyse de l'information portée par les i-vecteurs

Les *i*-vecteurs offrent une représentation intéressante pour analyser la répartition des informations locuteur et texte. La partie 1 de la base de données *RSR2015* étant enregistrée dans des conditions studio nous faisons l'hypothèse que la variabilité entre segments est principalement due aux locuteur et au texte prononcé. Cette hypothèse considère donc que la variabilité intra-locuteur est principalement due au texte.

Des MFCCs sont extraits sur différentes bandes de fréquences variant de [0-1000Hz] à [0-8000Hz] et deux extracteurs de *i*-vecteurs sont appris sur l'ensemble des 157 locuteurs hommes de *RSR2015* et des 143 locuteurs femmes. Deux modèles du monde sont appris avec 512 distributions et les matrices de variabilité totale sont de rang 150 pour les hommes et 140 pour les femmes. Un *i*-vecteur est extrait pour chaque phrase la partie de 1 de *RSR2015* et les matrices de covariance intra-classes (**W**) et inter-classes (**B**) sont calculées pour chaque genre. Pour chacune des matrices de covariance, les valeurs propres sont calculées : $\{\lambda_W^n\}$, $\{\lambda_B^n\}$. On calcule ensuite le rapport entre les traces de **B** et **W** comme suit :

$$R = \frac{\text{trace}(\mathbf{B})}{\text{trace}(\mathbf{W})} = \frac{\sum_{150}^{i=1} \lambda_B^i}{\sum_{150}^{i=1} \lambda_W^i} \tag{8.1}$$

Ce ratio est tracé pour chaque genre sur la figure 8.4 pour des MFCCs extraits sur différentes bandes de fréquences.

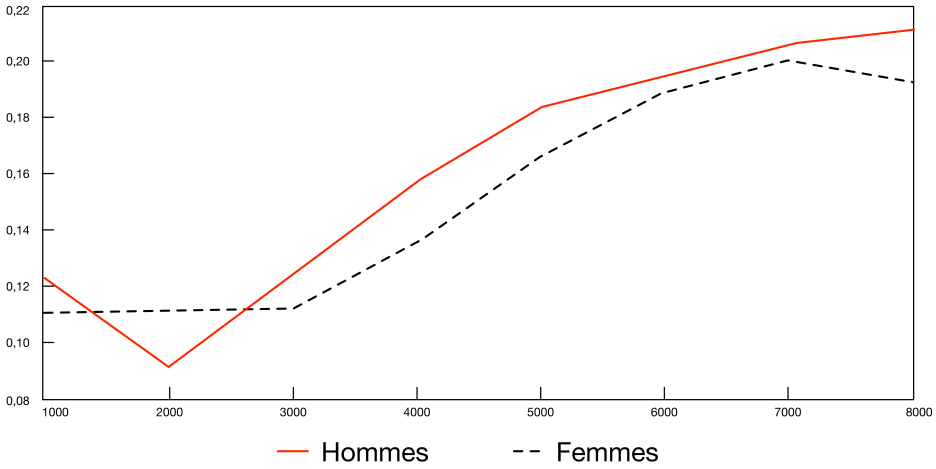

Figure 8.4 – Rapport des traces des matrices de covariance intra-locuteurs et inter-locuteurs pour des MFCCs extraits sur différentes bandes de fréquences. sur la partie 1 hommes et femmes de la base de données *RSR2015*.

Le rapport des traces des matrices inter- et intra-locuteurs donne une estimation du ratio d'information locuteur et texte contenus dans les *i*-vecteurs pour différentes bandes de fréquences. On observe ici que ce ratio augmente avec la limite supérieure de la bande de fréquences d'extraction des MFCCs. Ce résultat confirme l'information obtenue pour

le système GMM-UBM. Le facteur de plus grande variabilité dans la base de données *RSR2015* est le locuteur et le ratio $\frac{locuteur}{texte+autre\ bruit}$ augmente avec la bande de fréquence.

8.3. Prise en compte du contenu phonétique pour un système i-vecteurs

Cette section décrit les travaux publiés dans [Larcher et al., 2012a] et [Larcher et al., 2013]. Les modèles acoustiques développés pour la reconnaissance du locuteur indépendante du texte sont utilisés de façon à tirer parti des contraintes imposées au locuteur dans le cas de la reconnaissance dépendante du texte.

8.3.1 Normalisation des i-vecteurs

Nous avons illustré précédemment les gains qui pouvaient être obtenus par des systèmes indépendants du texte comme un système *i*-vecteurs dans le cas où les contenus phonétiques prononcés lors de l'enrôlement et du test sont identiques. Nous avons également montré que lorsqu'on utilise la bande de fréquence complète ([0-8000kHz]), l'information phonétique dégrade considérablement les performances de la reconnaissance du locuteur en renforçant la variabilité intra-locuteur. Dans le paradigme *i*-vecteurs PLDA, le processus d'extraction des *i*-vecteurs est supervisé, une normalisation est appliquée aux *i*-vecteurs afin de réduire la variabilité intra-locuteur ou d'augmenter la variabilité inter-locuteur (cf. section 5.3). Dans le cas où il est possible d'imposer un contenu phonétique au locuteur lors des phases d'enrôlement et de test, nous faisons dans [Larcher et al., 2012a] l'hypothèse qu'une nouvelle normalisation des *i*-vecteurs peut être plus adaptée à la tâche de reconnaissance du locuteur dépendante du texte. En effet, la contrainte de texte imposé aux locuteurs consiste à redéfinir les classes à séparer : une classe est maintenant définie par un couple locuteur + texte.

Dans [Larcher et al., 2012a], la redéfinition des classes est appliquée à deux normalisations :

— EFR : Eigen Factor Radial

— WCCN : Within Class Covariance Noramlization

La première a été détaillée dans 5.3 et est appliquée ici avant de calculer une distance de Mahalanobis et la deuxième est utilisée pour calculer une similarité cosine entre deux *i*-vecteurs , $\mathbf{w}_1$ et $\mathbf{w}_2$ de la façon suivante :

$$CS(\mathbf{w}_1, \mathbf{w}_2) = \frac{< \mathbf{B}^t\mathbf{w}_1 | \mathbf{B}^t\mathbf{w}_2 >}{||\mathbf{B}^t\mathbf{w}_1||\ ||\mathbf{B}^t\mathbf{w}_2||} \tag{8.2}$$

Table 8.2 – Performance de deux systèmes *i*-vecteurs utilisant une normalisation Eigen Factor Radial suivie d'une distance de Mahalanobis (EFR) et d'une similarité cosine (CS) avec et sans normalisation WCCN sur la partie 1 de la base de données *RSR2015*. Les performances sont données en taux d'égales erreurs (EER, %) pour différentes définitions des classes à discriminer.

Définition des classes	Système		
	EFR	CS + WCCN	CS
Locuteur	9.37	10.01	15,38
Locuteur + texte	7.88	9.67	

où $\mathbf{B}$ est la décomposition de Cholesky de la matrice de covariance intra-classes $\mathbf{W}_{wccn}$ telle que : $\mathbf{W}_{wccn}^{-1} = \mathbf{B}\mathbf{B}^t$ avec :

$$\mathbf{W}_{wccn} = \frac{1}{S}\sum_{s=1}^{S}\frac{1}{n_s}\sum_{i=1}^{n_s}\mathbf{w}_i^s \tag{8.3}$$

Le tableau 8.2 contient les taux d'égales erreurs obtenus pour deux systèmes *i*-vecteurs utilisant les normalisations EFR et WCCN dans le cadre de la reconnaissance du locuteur dépendante du texte. Les tests positifs correspondent aux tests *Client-correct* et les tests négatifs aux tests *Imposteur-correct*. La première ligne du tableau présente l'EER des deux systèmes lorsque les classes considérées ne dépendent que du locuteur et la deuxième ligne utilise une définition de classe correspondant à des couples locuteur + texte.

La redéfinition des classes à discriminer pour normaliser les *i*-vecteurs dans le cadre de la reconnaissance dépendante du texte améliore sensiblement les performances des deux systèmes : Similarité Cosine avec WCCN et distance de Mahalanobis avec EFR. Les taux d'égales erreurs diminuent respectivement de 15,9% et 2,4% relatif.

8.3.2 Adaptation de la PLDA

La redéfinition des classes utilisant l'information phonétique dans le cadre de la reconnaissance du locuteur dépendante du texte telle qu'elle a été formulée dans la section précédente peut être appliquée pour la normalisation des *i*-vecteurs mais également dans l'apprentissage discriminant de la PLDA.

Dans Larcher et al. [2013] nous proposons d'appliquer cette définition des classes acoustiques à un système *i*-vecteurs PLDA ainsi qu'à la Spherical Nuisance Normalization proposée dans [Bousquet et al., 2012]. Les performances d'un système *i*-vecteurs PLDA sont présentées dans le tableau 8.3 pour deux définitions des classes acoustiques.

Table 8.3 – Performance d'un système *i*-vecteurs PLDA pour différentes définitions des classes acoustiques avec et sans Spherical Nuisance Normalization. Les résultats sont donnés en taux d'égales erreurs (EER %) sur la partie 1 hommes de la base de données *RSR2015*. Les tests positifs correspondent aux tests *Client-correct* et les tests négatifs aux tests *Imposteur-correct*.

Définition des classes	Normalisation	Imp-same
Locuteur	-	10,35
	SphNorm	9.06
Locuteur + Texte	-	7,20
	SphNorm	6.96

L'observation des lignes 1 et 3 du tableau 8.3 montre que l'apprentissage de la PLDA bénéficie d'une définition des classes acoustiques prenant en compte le texte prononcé. En effet, les taux d'erreurs chutent de 30% de 10,35% à 7,20% par rapport au cas où les classes de locuteurs contiennent différents textes. Les lignes 2 et 4 révèlent de plus que comme dans le cas indépendant du texte (cf. section 5.3) l'utilisation de la Spherical Nuisance Normalization apporte un gain supplémentaire dans tous les cas.

CHAPITRE 9

Modélisation dépendante du texte pour la caractérisation d'impostures

Nous avons vu dans le chapitre précédent que dans le cas de segments audio de courtes durées (quelques secondes), les systèmes de reconnaissance du locuteur qui utilisent une modélisation du locuteur indépendante du texte : GMM ou *i*-vecteur , obtiennent de meilleures performances lorsque les textes prononcés pour l'enrôlement et les tests sont identiques. C'est ce qu'exploitent les systèmes de reconnaissance du locuteur dépendants du texte. La plupart des systèmes dépendant du texte existant exploitent également la structure temporelle du signal de parole afin de comparer des segments encore plus courts [Hébert, 2008; Larcher et al., 2014c]. Les informations acoustiques et phonétiques peuvent être utilisées pour deux finalités :

— segmenter les données audio pour réduire la variabilité intra-locuteur sur des segments très courts ;

— reconnaître le locuteur et le texte prononcé pour une double authentification du locuteur utilisant son modèle acoustique et une connaissance personnelle (mot de passe).

9.1. Un modèle acoustique à architecture hiérarchique

Les travaux menés au laboratoire informatique d'Avignon entre 2006 et 2010 puis à l'Institute for Infocomm Research de Singapour entre 2010 et 2014 ont conduit au

développement d'un système de reconnaissance du locuteur dépendant du texte disposant d'un modèle à architecture hiérarchique simple et robuste. Ce système a fait l'objet de deux brevets [Larcher et al., 2013a,b] et a été intégré dans le système d'exploitation du téléphone LENOVO A586, commercialisé en 2012. L'architecture proposée permet dans un processus simple de tirer parti du texte prononcé par les locuteurs pour améliorer les performances de reconnaissance du locuteur et pour caractériser le type d'imposture rencontrée.

9.1.1 Description du système

Le système de reconnaissance du locuteur dépendant du texte, HiLAM, [Larcher et al., 2012, 2014c; Lee et al., 2011] est une extension du système GMM-UBM développé pour la reconnaissance du locuteur dépendante du texte décrit dans la section 2.2.2. Son architecture est représentée sur la figure 9.1. Tous les états de ce modèle sont des mixtures

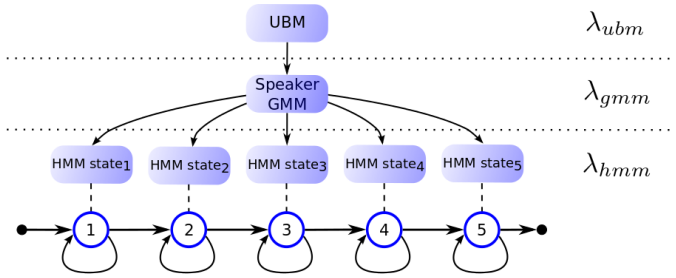

Figure 9.1 – L'architecture hiérarchique du modèle HiLAM.

de Gaussiennes (GMMs) qui partagent les même paramètres de variance et les mêmes vecteurs de poids. Les deux couches supérieures de cette architecture correspondent à un système UBM-GMM standard. Le modèle du monde (couche supérieure) modélise l'espace acoustique de la parole. La couche intermédiaire modélise les spécificités d'un locuteur qui sont obtenues par une adaptation MAP du modèle du monde. Enfin, la couche inférieure exploite la capacité des modèles de Markov cachés (HMM) à modéliser la structure temporelle des mots de passe. Les états acoustiques du HMM sont obtenus par adaptation MAP du modèle de locuteur. Pour le modèle de locuteur, comme pour les états du HMM, seuls les paramètres de moyenne sont adaptés, ce qui diffère de l'approche initiale proposée dans [Larcher et al., 2008c].

L'apprentissage du modèle HiLAM est identique à celui décrit dans [Larcher et al., 2008c] ; un modèle du monde UBM est appris sur un nombre important de locuteurs. Un modèle de locuteur indépendant du texte est ensuite adapté en utilisant seulement trois répétitions d'un même mot de passe prononcé lors de l'enrôlement. La durée d'enrôlement est donc inférieure à 10 secondes. Finalement, un processus itératif permet d'apprendre le modèle HMM de la troisième couche. Les états du HMM sont initialisés avec le modèle GMM de la couche intermédiaire. Une première adaptation est effectuée en utilisant

une segmentation uniforme du segment audio. Ce segment est découpé en S segments $\{seg_i\}_{i\in[1,S]}$ de même longueur. Une nouvelle segmentation est obtenue grâce à un alignement de Viterbi. Les états sont alors adaptés en respectant cette nouvelle segmentation. Le processus d'adaptation est répété jusqu'à convergence. Notons que le nombre d'états est fixé empiriquement. les probabilités de transitions entre états sont choisies équiprobables pour un HMM gauche-droite.

Étant donnée une séquence acoustique de test, $\mathcal{O}$, il est possible de calculer la vraisemblance de la séquence de données pour les trois couches du modèle acoustique. La vraisemblance du modèle du monde : $\Lambda(\mathcal{O}|\lambda_{UBM})$, la vraisemblance du modèle de locuteur indépendant du texte : $\Lambda(\mathcal{O}|\lambda_{GMM})$ et la vraisemblance du modèle de locuteur dépendant du texte : $\Lambda(\mathcal{O}|\lambda_{HMM})$.

Un score dépendant du texte est calculé comme suit :

$$\mathcal{S}_{HMM}(\mathcal{X}) = \log \frac{\Lambda(\mathcal{O}|\lambda_{HMM})}{\Lambda(\mathcal{O}|\lambda_{UBM})} \tag{9.1}$$

où $\mathcal{S}_{HMM}(X)$ est le log-rapport de vraisemblances de la séquence de test sur le modèle de locuteur dépendant du texte aligné par décodage de Viterbi, $\Lambda(\mathcal{O}|\lambda_{HMM})$, avec la vraisemblance du modèle du modèle UBM, $\Lambda(\mathcal{O}|\lambda_{UBM})$. Dans la suite de ce document, le nombre d'états des modèles HMM est fixé à 5.

Un système *i*-vecteurs PLDA est entraîné sur des données issues des bases NIST-SRE de 2004 à 2008 ainsi que SwitchBoard. Pour cette raison, toutes les données, incluant *RSR2015* sont échantillonnées à 8kHz. Le système *i*-vecteurs PLDA est entraîné comme décrit dans le chapitre précédent. Des trois phrases d'enrôlement sont extraits trois *i*-vecteurs et le score de vérification est calculé comme proposé dans la section 6.2. La description complète du protocole est donnée dans [Larcher et al., 2014c].

9.1.2 Performances

Les résultats présentés ici sont extraits de [Larcher et al., 2014c]. Le tableau 9.1 présente les performances du système HiLAM comparé à un système *i*-vecteurs PLDA indépendant du texte en termes d'EER et de minDCF. Le système *i*-vecteurs PLDA est similaire à celui présenté dans le chapitre précédent et exploite l'information du texte prononcé, pour normaliser les *i*-vecteurs et apprendre le modèle PLDA [Larcher et al., 2013].

Dans toutes les conditions et pour les hommes comme pour les femmes, le système exploitant la structure temporelle grâce aux HMM obtient de meilleures performances que le système *i*-vecteurs . Dans le meilleur des cas, le taux d'erreurs est réduit de 66% lorsque

les imposteurs hommes prononcent un texte différent de celui utilisé pour l'enrôlement. Le résultat n'est pas surprenant étant donné la très courte durée des segments et la variabilité acoustique très réduite de la base de données *RSR2015*[Stafylakis et al., 2013]. On observe de plus sur la figure 9.2 que cet avantage du système HiLAM persiste, quel que soit le point de fonctionnement considéré. Comme dans le cadre de la reconnaissance du locuteur indépendante du texte, les taux d'erreurs observées pour les femmes sont plus élevés que pour les hommes.

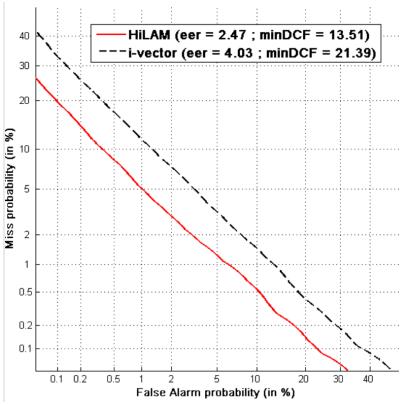

Figure 9.2 – Courbe DET obtenue pour les hommes sur le sous-ensemble d'évaluation de la partie 1 de la base de données *RSR2015* dans le cas où les locuteurs cibles et les imposteurs prononcent lors du test, le contenu phonétique utilisé pour l'enrôlement.

Une explication de ce phénomène pourrait provenir de la répartition de l'information locuteur et phonétique au gré des bandes de fréquences. Des études préliminaires montrent qu'une part importante de l'information spécifique au locuteur se trouve dans les plus hautes fréquences pour les femmes et est supprimée lorsque le signal est sous-échantillonné à 8kHz comme c'est le cas pour les résultats présentés afin de comparer à un système *i*-vecteurs optimal. De ce fait, pour un signal échantillonné à 8kHz, la variabilité phonétique pourrait affecter plus les femmes que les hommes. Des études complémentaires sont nécessaires pour confirmer cette hypothèse et nous discuterons du rôle que les modèles neuronaux pourraient avoir pour compenser cet effet.

Les résultats présentés dans le tableau 9.1 montrent que le système HiLAM qui modélise chaque phrase par un HMM à 5 états est plus à même de rejeter le locuteur cible lorsqu'il prononce une mauvaise phrase qu'un imposteur qui prononcerait la phrase correcte. Ceci démontre la capacité du système à prendre en compte la structure temporelle du mot de passe prononcé. La même tendance est observée pour le système *i*-vecteurs PLDA ; ceci est dû à la très courte durée des segments audio et au fait que l'information du texte est fortement présente dans les *i*-vecteurs , comme nous l'avons montré dans la section 8.1.2.

Table 9.1 – Performances du système *HiLAM* et d'un système *i*-vecteurs sur la partie développement 1 de la base de données *RSR2015* en termes d'EER et de minimum DCF (EER % / minDCF×100) pour plusieurs définitions des scores positifs et négatifs.

Texte	Locuteur cible		Imposteur		Homme		Femme	
	correct	faux	correct	faux	HiLAM	*i*-vector	HiLAM	*i*-vector
Tests	positifs	négatifs	-	-	1,66 / 7,40	2,87 / 13,56	1,77 / 7,42	3,05 / 17,26
	positifs	-	négatifs	-	3,69 / 16,78	5,95 / 26,74	3,24 / 15,39	7,87 / 40,45
	positifs	-	-	négatifs	0,49 / 1,65	0,74 / 3,43	0,45 / 1,81	0,94 / 4,65

La nomenclature est la suivante : un contenu lexical *correct* signifie que le texte prononcé durant le test est le même que celui prononcé lors de l'enrôlement ; un contenu *faux* signifie que les contenus lexicaux d'enrôlement et de test sont différents.

9.2. Redéfinir le problème de la reconnaissance du locuteur dépendante du texte

Comme évoqué précédemment, la tâche de reconnaissance du locuteur dépendante du texte est moins étudiée que la tâche indépendante du texte, du fait des intérêts liés à la défense et la sécurité des états qui sponsorisent la collection de données. Durant ces dix dernières années, j'ai travaillé à la collection de données pour la reconnaissance du locuteur dépendante du texte et à la mise en place d'une définition précise de cette tâche. Les travaux présentés ici ont été publiés dans [Larcher et al., 2014b] et [Larcher et al., 2014a] et participent à ce travail de *taxonomie*. Il existe différentes façons de contraindre le contenu lexical prononcé par un locuteur [Aronowitz, 2012; Hébert, 2008; Larcher et al., 2014c] ; la partie qui suit se concentre sur le cas où chaque locuteur cible est libre de choisir un mot de passe personnalisé lors de son enrôlement dans le système. C'est ce protocole qui a été utilisé pour l'intégration du système HiLAM sur le téléphone LENOVO A586. Chaque utilisateur était libre de choisir son mot de passe avec pour seule recommandation d'utiliser un mot de passe pouvant s'écrire avec entre 3 et 5 caractères chinois.

9.2.1 Redéfinition théorique de l'hypothèse alternative

La tâche de vérification du locuteur [Bimbot et al., 2004; Kinnunen et Li, 2010] est une tâche de classification binaire. Étant donné un segment audio $\mathcal{O}$ prononcé par un locuteur $\mathcal{X}$, un système automatique doit décider si l'hypothèse selon laquelle ce segment a été prononcé par le locuteur cible est vraie ou fausse. Idéalement, un score de vérification, traduisant la confiance du système dans l'hypothèse $H_{\mathcal{X}}$ est calculée comme un rapport de vraisemblances entre $H_{\mathcal{X}}$ et son hypothèse alternative : $H_{\overline{\mathcal{X}}}$ qui suppose que le segment $\mathcal{O}$ a été prononcé par un imposteur.

L'introduction d'une contrainte temporelle transforme cette tâche binaire en une tâche

de classification à deux dimensions. Le système dépendant du texte doit déterminer si le segment de parole fourni au système a été prononcé par le locuteur cible, mais également si ce segment de parole correspond au contenu phonétique attendu. Les systèmes de reconnaissance du locuteur dépendant du texte sont donc exposés à quatre types de tests décrits dans le tableau 9.2.

Une nouvelle hypothèse de vérification doit alors être considérée : $H_{(\mathcal{X},\mathcal{P})}$ dans laquelle le segment de parole contient le contenu lexical correct prononcé par le locuteur cible. C'est le cas où le test appartient à la catégorie $(\mathcal{X},\mathcal{P})$. En conséquence, la nouvelle hypothèse alternative : $H_{\overline{(\mathcal{X},\mathcal{P})}}$ peut être définie de la même façon en considérant l'union des trois autres hypothèses :

— $H_{(\mathcal{X},\overline{\mathcal{P}})}$, dans laquelle le locuteur cible prononce un mauvais contenu lexical ;
— $H_{(\overline{\mathcal{X}},\mathcal{P})}$, dans laquelle un imposteur prononce le contenu lexical correct ;
— $H_{(\overline{\mathcal{X}},\overline{\mathcal{P}})}$, dans laquelle un imposteur prononce un mauvais contenu lexical.

Table 9.2 – Quatre types de tests auxquels est exposé un système de reconnaissance du locuteur dépendant du texte.

	Contenu phonétique correct $\mathcal{P}$	Faux contenu phonétique $\overline{\mathcal{P}}$
Locuteur cible $\mathcal{X}$	$(\mathcal{X},\mathcal{P})$	$(\mathcal{X},\overline{\mathcal{P}})$
Imposteur $\overline{\mathcal{X}}$	$(\overline{\mathcal{X}},\mathcal{P})$	$(\overline{\mathcal{X}},\overline{\mathcal{P}})$

Les approches existant dans la littérature traitent généralement ce problème en deux étapes [Heck et Genoud, 2001; Reynolds et Heck, 1991]. Un système de reconnaissance de la parole transcrit d'abord le texte prononcé et un système de reconnaissance du locuteur indépendant du texte est ensuite utilisé. Si le contenu lexical prononcé et l'identité détectée sont corrects, alors l'utilisateur est accepté. Ces approches fournissent de bons résultats, mais nécessitent l'utilisation de deux systèmes complexes. Dans [Larcher et al., 2014b] nous avons proposé d'exploiter l'architecture unique du système HiLAM présenté dans la section précédente afin de calculer un unique rapport de vraisemblances qui fourni un score combinant l'information locuteur et lexicale. Ce système unique permet d'estimer simplement et efficacement l'hypothèse de vérification : $H_{(\mathcal{X},\mathcal{P})}$, ainsi que son hypothèse alternative : $H_{\overline{(\mathcal{X},\mathcal{P})}}$.

D'autres travaux [Sarkar et Umesh, 2010; Zhang et al., 2010] ont également montré qu'une définition appropriée de l'hypothèse alternative du rapport de vraisemblances améliore les performances d'un système de reconnaissance du locuteur dépendant du texte.

Pour rappel, le rapport de vraisemblances calculé pour la tâche de vérification du locu-

teur indépendante du texte est utilisé pour prendre une décision comme suit :

$$\log p(\mathcal{O}|H_{\mathcal{X}}) - \log p(\mathcal{O}|H_{\overline{\mathcal{X}}}) \lessgtr \Theta \begin{cases} H_{\mathcal{X}} \text{ rejeté} \\ H_{\mathcal{X}} \text{ accepté} \end{cases} \tag{9.2}$$

où Θ est le seuil de décision fixé a priori.

Dans le nouveau paradigme proposé dans [Larcher et al., 2014b], l'hypothèse nulle considère que le segment $\mathcal{O}$ appartient à la classe $(\mathcal{X}, \mathcal{P})$, c'est-à-dire que $\mathcal{O}$ contient le texte correct prononcé par le locuteur cible. Une nouvelle hypothèse alternative est définie en considérant l'union des trois autres classes de tests définies dans le tableau 9.2. On cherchera alors à estimer la probabilité de cette hypothèse comme :

$$P(\mathcal{O}|H_{\overline{(\mathcal{X},\mathcal{P})}}) = P(\mathcal{O}|H_{(\mathcal{X},\overline{\mathcal{P}})}) + P(\mathcal{O}|H_{(\overline{\mathcal{X}},\mathcal{P})}) + P(\mathcal{O}|H_{(\overline{\mathcal{X}},\overline{\mathcal{P}})}) \tag{9.3}$$

9.2.2 Modélisation acoustique et estimation des nouvelles hypothèses

Modélisation acoustique par l'architecte HiLAM

Le modèle acoustique hiérarchique, HiLAM, présenté dans la section précédente est utilisé ici pour estimer les modèles acoustiques correspondant aux différentes classes de tests. Pour rappel, lorsqu'un segment de test est comparé au modèle hiérarchique, il est possible de calculer la vraisemblance du segment acoustique pour les trois couches :

- $\Lambda(\mathcal{O}|\lambda_{ubm})$ est la vraisemblance du segment $\mathcal{O}$ sur le modèle du monde : indépendant du locuteur et du texte prononcé (première couche du modèle HiLAM) ;
- $\Lambda(\mathcal{O}|\lambda_{gmm})$ est la vraisemblance du segment $\mathcal{O}$ sur le modèle GMM du locuteur appris en utilisant l'ensemble des échantillons audio disponible pour le locuteur cible. On considère donc que ce modèle est dépendant du locuteur, mais indépendant du texte ;
- $\Lambda(\mathcal{O}|\lambda_{hmm})$ est la vraisemblance du segment $\mathcal{O}$ sur le modèle HMM de la troisième couche du modèle HiLAM : un modèle dépendant du locuteur et du texte.

Dans le cas où chaque locuteur peut choisir lui-même son mot de passe, il est impossible d'apprendre un modèle acoustique des imposteurs prononçant le mot de passe correct en recourant à des enregistrements réels. Il serait envisageable d'utiliser des technologies de transformation de voix ou de synthèse vocale pour apprendre ce modèle, mais nous ne traitons pas ce cas de figure et considérons que le modèle du monde, indépendant du locuteur et du texte, servira à modéliser deux hypothèses. On fait l'hypothèse que λ_{ubm} modélise $H_{(\overline{\mathcal{X}},\mathcal{P})} \cup H_{(\overline{\mathcal{X}},\overline{\mathcal{P}})}$. Le nombre d'hypothèses alternatives est réduit à 2 : $\{(\mathcal{X}, \overline{\mathcal{P}}); \overline{\mathcal{X}}\}$

Formation du score dépendant du texte

Exploitant l'architecture d'HiLAM, les vraisemblances introduites ci-dessus peuvent être combinées de différentes façons pour former un score de vérification. Deux options sont proposées dans [Larcher et al., 2014a].

La première option utilise une approche commune en reconnaissance de la parole Katagiri et al. [1998] et des langues [Lee, 2008; Li et al., 2006]. Il s'agit d'une moyenne pondérée des vraisemblances des sous-hypothèses. L'expression de cette combinaison est :

$$p(\mathcal{O}|H_{\overline{(\mathcal{X},\mathcal{P})}}) = \left(\frac{1}{N}\sum_{c\,\in\,\Omega} p(\mathcal{O}|H_c)^{\eta}\right)^{\frac{1}{\eta}} \tag{9.4}$$

où Ω est l'ensemble des classes de tests correspondant à l'hypothèse $H_{\overline{(\mathcal{X},\mathcal{P})}}$et η est une constante positive. Dans notre cas, $\Omega = \{(\mathcal{X},\overline{\mathcal{P}});(\overline{\mathcal{X}},\mathcal{P});(\overline{\mathcal{X}},\overline{\mathcal{P}})\}$ mais comme nous l'avons vu précédemment, Ω est réduit à $\Omega = \{(\mathcal{X},\overline{\mathcal{P}});\overline{\mathcal{X}}\}$

La seconde option proposée consiste à réaliser une fusion de scores, comme c'est souvent le cas pour combiner les sorties de plusieurs systèmes de reconnaissance du locuteur [Brümmer et al., 2007; Hautamaki et al., 2012, 2013]. Le logarithme de $p(\mathcal{O}|H_{\overline{(\mathcal{X},\mathcal{P})}})$ est calculé comme la moyenne des log-vraisemblances des sous-hypothèses. Ainsi :

$$\log p(\mathcal{O}|H_{\overline{(\mathcal{X},\mathcal{P})}}) = \frac{1}{N}\sum_{c\,\in\,\Omega} \log p(\mathcal{O}|H_c) \tag{9.5}$$

En pratique, à cause de l'impossibilité de modéliser l'hypothèse des imposteurs prononçant le contenu lexical correct, les scores proposés dans les équations 9.4 et 9.5 doivent être approximés par les expressions ci-dessous. La première approximation du score donné dans l'équation 9.4, est :

$$\mathcal{S}_1^{\eta}(\mathcal{O}) = \log\Lambda(\mathcal{O}|\lambda_{hmm}) - \log\left[\left(\frac{\Lambda(\mathcal{O}|\lambda_{gmm})^{\eta}}{2} + \frac{\Lambda(\mathcal{O}|\lambda_{ubm})^{\eta}}{2}\right)^{\frac{1}{\eta}}\right] \tag{9.6}$$

Cependant, lorsque η tends vers l'infini, $\mathcal{S}_1^{\eta}(\mathcal{O})$ tends vers $\mathcal{S}_1^{max}(\mathcal{O})$ dont l'expression est :

$$\mathcal{S}_1^{max}(\mathcal{O}) = \log\Lambda(\mathcal{O}|\lambda_{hmm}) - \log\max\left\{\Lambda(\mathcal{O}|\lambda_{gmm}),\Lambda(\mathcal{O}|\lambda_{ubm})\right\} \tag{9.7}$$

En ce qui concerne le score décrit par l'équation 9.5, l'approximation conduit au score :

$$\mathcal{S}_2(\mathcal{O}) = \log\Lambda(\mathcal{O}|\lambda_{hmm}) - \left[\frac{\log\Lambda(\mathcal{O}|\lambda_{gmm})}{2} + \frac{\log\Lambda(\mathcal{O}|\lambda_{ubm})}{2}\right] \tag{9.8}$$

Afin d'apprécier les performances des scores proposés, ils sont comparés dans la sec-

tion suivante aux scores proposés initialement avec le système HiLAM, à savoir un score dépendant du locuteur et du texte : $\mathcal{S}_{HMM}(\mathcal{O})$ [Larcher et al., 2012, 2014c].

$$\mathcal{S}_{HMM}(\mathcal{O}) = \log \Lambda(\mathcal{O}|\lambda_{hmm}) - \log \Lambda(\mathcal{O}|\lambda_{ubm}) \tag{9.9}$$

Dans ce cas, l'hypothèse alternative est modélisée par le modèle du monde (première couche du modèle HiLAM) et ne prend pas en compte le cas des tests où le locuteur cible prononce un mauvais contenu lexical. Enfin, le score classique d'un système GMM-UBM [Reynolds et al., 2000] est utilisé pour comparaison.

$$\mathcal{S}_{GMM}(\mathcal{O}) = \log \Lambda(\mathcal{O}|\lambda_{gmm}) - \log \Lambda(\mathcal{O}|\lambda_{ubm}) \tag{9.10}$$

9.2.3 Évaluation de l'approche proposée

Les trois scores de vérification proposés ci-dessus sont comparés pour la tâche de reconnaissance du locuteur dépendant du texte sur la partie 1, hommes, de la base de données *RSR2015*. Le modèle du monde de la première couche de l'architecture HiLAM est estimé en utilisant les parties 2 et 3 de la même base de données, ce qui garantit que le système n'a pas connaissance des 30 phrases de la partie 1.

Indicateur de performances

Les différents types de tests : imposteurs prononçant le contenu lexical correct ou non, locuteur cible prononçant un mauvais contenu, n'apparaissent pas avec la même probabilité. De plus, il est plus facile pour les systèmes automatiques de rejeter un test dans le cas où ni le locuteur ni le texte ne correspondent. Aussi, nous choisissons de définir une fonction de coût qui exclut le cas le plus facile où un imposteur prononce un mauvais contenu lexical. La fonction de coût proposé est similaire à celle proposée pour l'évaluation NIST-SRE 2012 [1]. Il s'agit d'un indicateur unique qui prend en compte deux types d'impostures $(\mathcal{X}, \overline{\mathcal{P}})$ et $(\overline{\mathcal{X}}, \mathcal{P})$. La fonction de coût est donnée par :

$$C_{Norm} = P_{\text{Miss}|\mathcal{X},\mathcal{P}} + \frac{\beta}{2} \times (P_{\text{FA}|\mathcal{X},\overline{\mathcal{P}}} + P_{FA|\overline{\mathcal{X}},\mathcal{P}}) \tag{9.11}$$

où $\beta = \frac{C_{FA}}{C_{Miss}} \times \frac{(1-P_{\mathcal{X},\mathcal{P}})}{P_{\mathcal{X},\mathcal{P}}}$ Les paramètres choisis par la suite sont :

- $P_{\mathcal{X},\mathcal{P}}$, la probabilité a priori que le locuteur de test soit le locuteur cible prononçant le contenu lexical correct ;
- $P_{Miss|\mathcal{X},\mathcal{P}}$, la probabilité de faux rejet ;

1. https://www.nist.gov/multimodal-information-group/speaker-recognition-evaluation-2012 vu le 20/09/2018

— $P_{FA|\mathcal{X},\overline{\mathcal{P}}}$, la probabilité de fausse acceptation d'un locuteur prononçant un mauvais contenu lexical ;

— $P_{FA|\overline{\mathcal{X}},\mathcal{P}}$, la probabilité de fausse acceptation pour un imposteur prononçant le contenu lexical corect ;

— C_{FA}, le coût d'une fausse acceptation ;

— C_{Miss}, le coût d'un faux rejet.

Les probabilités qu'un test imposteur appartienne aux classes $(\mathcal{X}, \overline{\mathcal{P}})$ ou $(\overline{\mathcal{X}}, \mathcal{P})$ sont considérées égales et les coûts C_{FA} et C_{Miss} sont fixés à 1.

Enfin, deux valeurs de Cnorm sont utilisées, correspondant à :

$$\begin{cases} \text{Cnorm}_A & \text{pour } P_{\mathcal{X},\mathcal{P}} = 0.01 \\ \text{Cnorm}_B & \text{pour } P_{\mathcal{X},\mathcal{P}} = 0.001 \end{cases} \tag{9.12}$$

Résultats

Une première série d'expériences [Larcher et al., 2014b] nous a permis de déterminer la valeur optimale du paramètre η dans l'équation 9.6. L'évolution des fonctions de coût Cnorm_A et Cnorm_B est présentée sur la figure 9.3. Les meilleures performances sont obtenues pour $\eta = 0,1$ et c'est cette valeur qui sera utilisée par la suite.

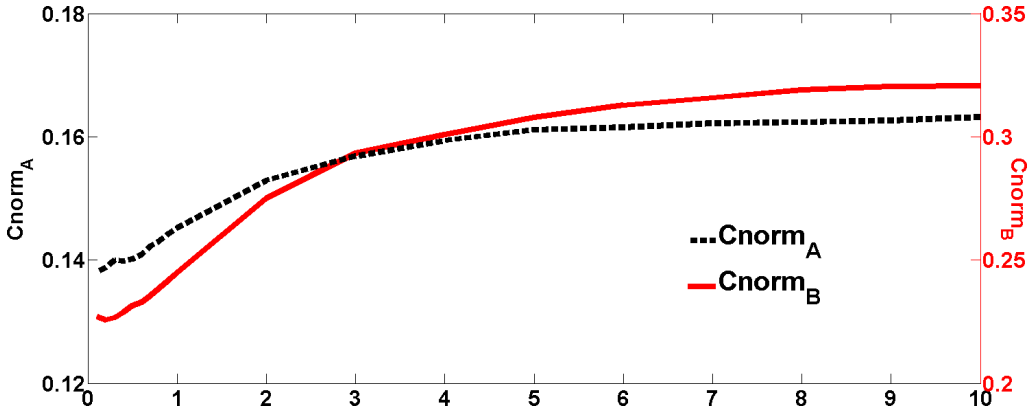

Figure 9.3 – Évolution des fonctions de coûts pour différentes valeurs du paramètre η utilisé pour calculer le score $\mathcal{S}_1^{\eta}$.

Les trois scores proposés, S_1^{η}, S_1^{max} et S_2, sont maintenant comparés aux deux scores de référence, $\mathcal{S}_{GMM}(\mathcal{O})$ et $\mathcal{S}_{HMM}(\mathcal{O})$ et les résultats sont présentés pour les deux fonctions de coût proposées : Cnorm_A et Cnorm_B dans le tableau 9.3.

La méthode de combinaison des scores semble ne pas influer fortement sur les performances puisque S_1^{η} et S_2 obtiennent des résultats comparables. Il est évident, lorsqu'on compare ces résultats aux deux scores standard, que la modélisation de l'hypothèse alternative apporte un gain conséquent. L'utilisation du score S_1^{η} réduit les coûts minimum

Table 9.3 – Performances obtenues pour différents scores en termes de *minimum detection cost* pour les deux valeurs de probabilité a priori $P_{\mathcal{X},\mathcal{P}}$.

Function de coût	S_1^η	S_1^{max}	S_2	S_{HMM}	S_{GMM}
Cnorm_A	**0,130**	0,171	0,132	0,336	1
Cnorm_B	**0,245**	0,313	**0.245**	0,474	1

Cnorm_A et Cnorm_B de 61% et 48% respectivement par rapport au score dépendant du texte S_{HMM}.

Comme on pouvait s'y attendre à la lecture de la figure 9.3, le score S_1^{max} fonctionne moins bien que S_1^η. En effet, nous avons observé que l'augmentation de η dégradait les performances et S_1^{max} en est le cas limite.

Le tableau 9.4 détaille les performances des cinq scores en termes de taux d'égales erreurs (EER) pour différentes définitions des tests négatifs. L'étude de ces résultats nous renseigne sur l'effet de la modélisation de l'hypothèse alternative proposée.

Table 9.4 – Performances de différents scores en termes d'EER (%) pour différents types d'impostures correspondant aux différents tests auquel un système de reconnaissance du locuteur dépendant du texte est confronté.

Type de test négatif	S_1^η	S_1^{max}	S_2	S_{HMM}	S_{GMM}
$(\mathcal{X}, \overline{\mathcal{P}})$	1,51	**0.46**	1,68	4,57	50
$(\overline{\mathcal{X}}, \mathcal{P})$	1,75	2,22	1,75	**1.60**	4,92
$(\overline{\mathcal{X}}, \overline{\mathcal{P}})$	0,24	**0.20**	0,25	0,37	5,04

La dernière colonne du tableau 9.4 illustre le caractère indépendant du texte du score S_{GMM}. Il est impossible de rejeter avec certitude les locuteurs cible prononçant un mauvais contenu lexical. Ce phénomène explique le coût de 1 obtenu par ce score dans le tableau 9.3.

Les scores S_1^η et S_2, qui minimisent les fonctions de coût ne minimisent l'EER dans aucune condition. Ils présentent cependant un bon compromis lorsqu'on considère l'ensemble des définitions de tests négatifs. C'est ce bon compromis que reflète le minimum des fonctions de coût.

Il est intéressant de noter que le score S_1^{max} obtient les EER les plus bas pour deux types de test négatif. Ce score sélectionne en effet la classe de test la plus probable pour modéliser l'hypothèse alternative.

Une analyse complémentaire montre que pour 99,07% des tests *Client-faux*, le score S_1^{max} utilise pour modélisation l'hypothèse alternative le seul score dépendant du locuteur et indépendant du texte λ_{gmm}. Cette sélection permet au score S_1^{max} de réduire l'EER de

90% relatif par rapport au score dépendant du locuteur et du texte S_{HMM} pour ce type particulier de test négatif.

La sélection de l'hypothèse alternative en fonction du type de test ouvre des perspectives intéressantes puisqu'elle reproduit le fonctionnement d'un système combinant reconnaissance du texte et reconnaissance du locuteur, mais en n'utilisant qu'un unique système. Dans la section suivante, nous proposons d'exploiter ce résultat afin de caractériser le type d'imposture rencontré par le système.

9.2.4 Caractérisation des impostures

Comme nous l'avons souligné précédemment, les systèmes de vérification du locuteur sont confrontés à quatre types de tests : le test positif et trois types de test négatif. (cf. tableau 9.2). Les probabilités et la gravité des différents types de tests négatifs diffèrent grandement. Une imposture par *play-back* lors de laquelle un imposteur rejoue un enregistrement du locuteur cible ne prononçant pas le bon contenu lexical démontre une grande volonté de nuire. Il en est de même de l'usurpation du mot de passe du client par un imposteur. Le cas de ces impostures peut nécessiter une intervention du gestionnaire du système automatique pour demander au client de modifier son mot de passe ou pour conserver une trace des enregistrements qui ont été rejoués au système en *play-back*. Pour ces raisons et en utilisant les résultats de la section précédente [Larcher et al., 2014b]. Nous proposons dans cette section de classifier les scores des tests d'un système de reconnaissance du locuteur dépendant du texte en quatre région dans un espace à deux dimensions afin de classifier les quatre types de tests décrits dans le tableau 9.2.

Un score de reconnaissance du locuteur en deux dimensions

Nous avons introduit précédemment un système de reconnaissance du locuteur dépendant du texte utilisant un modèle acoustique à structure hiérarchique : HiLAM (cf. figure 9.1). Ce modèle est utilisé pour calculer deux scores décrits par les équations 9.10 et 9.9. Dans ces deux expressions, le dénominateur du rapport de vraisemblances est calculé grâce au modèle du monde (UBM) qui modélise l'ensemble des locuteurs sauf le locuteur cible. Ainsi, aucun de ces deux scores n'est destiné à rejeter les tests impliquant le locuteur cible correspondant aux *play-backs* de la classe *Client-faux*. Afin de rejeter précisément les tests impliquant le locuteur cible prononçant un mauvais contenu lexical nous avons proposé dans [Larcher et al., 2014a] un score calculé comme un rapport de vraisemblances entre deux hypothèses liées au locuteur cible.

Ce score est défini par :

$$\mathcal{S}_{sn}(\mathcal{O}) = \log \Lambda(\mathcal{O}|\lambda_{hmm}) - \log \Lambda(\mathcal{O}|\lambda_{gmm}) \tag{9.13}$$

Ce score compare explicitement les hypothèses du locuteur prononçant le contenu lexical correct avec le même locuteur prononçant un mauvais contenu lexical. Ce score peut être exprimé comme la différence entre les scores dépendant du texte et indépendant du texte : $\mathcal{S}_{sn} = \mathcal{S}_{hmm} - \mathcal{S}_{gmm}$ La figure 9.4 permet de comparer les distributions de

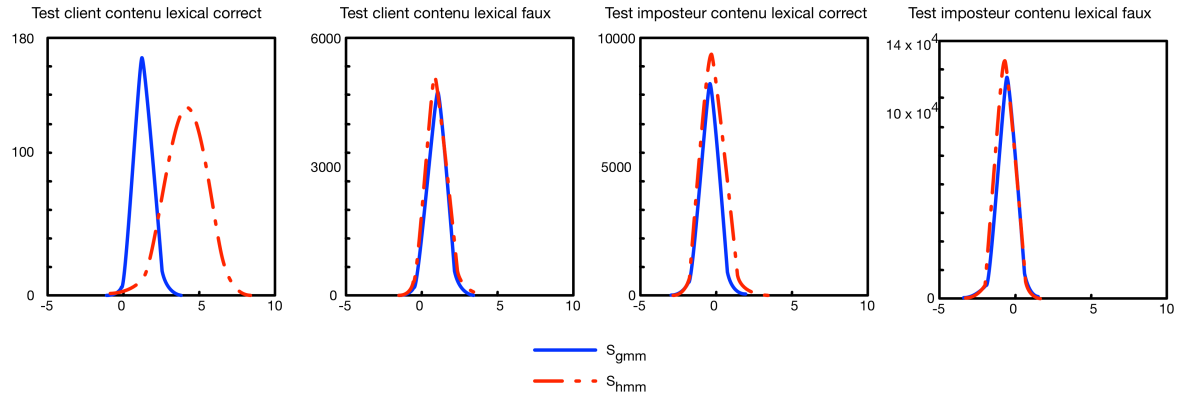

Figure 9.4 – Distributions des scores du système HiLAM :$\mathcal{S}_{hmm}$ et $\mathcal{S}_{gmm}$, pour les quatre classes de tests rencontrés par un système de reconnaissance du locuteur dépendant du texte.

scores dépendant et indépendant du texte fournis par le système HiLAM. On observe que l'utilisation de l'information lexicale (par le HMM) n'affecte pas les distributions des scores des clients prononçant un mauvais contenu lexical ni aucune des distributions de scores impliquant un imposteur. En revanche, la distribution des scores du locuteur cible prononçant le contenu lexical correct se décale vers la droite (les scores augmentent) lorsque le contenu lexical est pris en compte.

Le score $\mathcal{S}_{sn}$ est proposé pour exploiter les comportements différents de ces deux scores, et permettre de séparer les quatre types de scores rencontrés par les systèmes de reconnaissance du locuteur dépendant du texte dans un espace à deux dimensions. Le résultat est illustré sur la figure 9.5 où les tests sont représentés dans un espace à deux dimensions en conjuguant les scores dépendant du locuteur et du texte $\mathcal{S}_{hmm}$ calculés avec le système HiLAM et le score proposé pour rejeter le locuteur cible prononçant un mauvais contenu lexical $\mathcal{S}_{sn}$. On peut noter que cette représentation présente une similitude avec le tableau 9.2 puisqu'on distingue les quatre zones de l'espace correspondant aux différents types de tests.

La classification des impostures dans un espace en deux dimensions peut être effectuée par divers moyens : régression logistique multi-classe [Leeuwen et Brümmer, 2006], machines à vecteurs supports [Hsu et Lin, 2002]. Les expériences présentées dans [Larcher

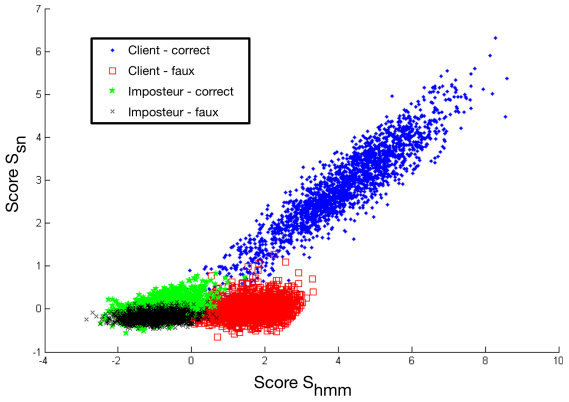

Figure 9.5 – Représentation des quatre classes de tests rencontrées par un système de reconnaissance du locuteur dépendant du texte dans un espace de scores à deux dimensions où $\mathcal{S}_{hmm}$ est le score dépendant du texte du système HiLAM et $\mathcal{S}_{sn}$ est le score proposé pour détecter les impostures par *play-back*. Tests obtenus sur la partie 1 développement de la base de données *RSR2015*.

et al., 2014a] utilisent un classifieur Gaussien hétéroscédastique (hetero-scedastic Gaussian back-end [2]) entraîné sur les scores de la partie développement de la base de données *RSR2015* et évalué sur la partie évaluation du même corpus. Étant donné un ensemble de scores en deux dimensions obtenus sur l'ensemble de développement, une distribution Gaussienne multivariée est apprise pour chaque classe de scores selon le critère du maximum de vraisemblance comme décrit dans [Li et al., 2013]. Lors de la phase de test, le score de classification est calculé comme un rapport de vraisemblance utilisant l'ensemble des distributions de scores apprises.

Performances

Les performances de l'approche proposée sont évaluées sur la base de données *RSR2015*. Pour chaque genre, un modèle du monde est appris en utilisant les parties 2 et 3 *background* du corpus afin d'éviter que les contenus lexicaux utilisés par les locuteurs cibles soient inclus dans le modèle du monde. Les performances sont évaluées sur la partie développement et validées sur la partie évaluation du corpus. Parmi les 9 sessions disponibles par locuteur, 3 sont utilisées en enrôlement et 6 en test. Pour chaque locuteur cible, un modèle dépendant du texte est appris pour chacune des 15 premières phrases de la partie 1 de *RSR2015*. Le modèle indépendant du texte de chaque locuteur est appris en utilisant ces 15 mêmes phrases.

Lors des tests, les 6 occurrences des 30 phrases sont utilisées pour générer les 4 types de tests. il faut noter que les tests sont dépendants du genre et que parmi les 30 contenus

2. https://sites.google.com/site/nikobrummer/focalmulticlass vu le 20/09/2018

lexicaux utilisés en tests, 15 ont été utilisées pour l'enrôlement alors que 15 n'ont pas été utilisés.

Une première expérience est effectuée sur la partie développement de *RSR2015* afin de comparer les performances des différents scores proposés pour une tâche de vérification classique, c'est-à-dire ne faisant intervenir qu'un unique score.
Le système obtient de plus faibles taux d'erreurs pour les femmes, mais les conclusions restent comparables entre genres. Le score indépendant du texte, $\mathcal{S}_{gmm}$ présenté pour référence obtient les plus mauvais résultats pour tous les types d'impostures (Tableau 9.5). En particulier, ce score est incapable de rejeter les locuteurs cibles prononçant un mauvais contenu lexical. Cependant, ce score permet de mieux rejeter les imposteurs prononçant un mauvais contenu lexical que ceux prononçant le contenu lexical correct. Ceci est probablement dû au fait que la moitié des contenus lexicaux corrects sont utilisés pour entraîner les GMMs qui ne sont donc pas complètement indépendants du texte étant donné la faible quantité de données utilisée pour l'apprentissage.

Table 9.5 – Performances des trois scores calculés à partir de l'architecture HiLAM sur la partie développement de la base de données *RSR2015*. Les performances sont indiquées en taux d'égales erreurs (% EER) pour les trois types d'impostures et les trois scores comparés.

Imposture	Hommes			Femmes		
	$\mathcal{S}_{gmm}$	$\mathcal{S}_{hmm}$	$\mathcal{S}_{sn}$	$\mathcal{S}_{gmm}$	$\mathcal{S}_{hmm}$	$\mathcal{S}_{sn}$
Client - faux	43,48	6,23	0,59	42,99	2,50	0,22
Imposteur - correct	6,14	1,82	1,90	5,29	0,93	0,88
Imposteur - faux	5,53	0,59	0,20	4,63	0,12	0,07

L'utilisation du contenu lexical par le score $\mathcal{S}_{hmm}$ permet de réduire les taux d'erreurs pour tous les types d'impostures (colonnes 3 et 6 du tableau 9.5). Cependant, pour les impostures de type *play-back* (locuteur cible prononçant un mauvais contenu lexical) l'EER reste à 6,23% pour les hommes et 2,50% pour les femmes. C'est l'imposture la plus difficile à rejeter. Si l'on différencie les tests impliquant les 15 phrases utilisées lors de l'enrôlement des 15 phrases inutilisées on observe que les tests *play-back* qui utilisent des contenus lexicaux jamais vus par le système sont plus difficiles à rejeter et les taux d'EER varient de 7,21% à 5,27% pour les hommes et 2,95% à 1,91% pour les femmes.

Le score proposé, $\mathcal{S}_{sn}$, est supposé mieux rejeter les impostures de type *play-back*. Les scores des colonnes 4 et 7 du tableau 9.5 confirment l'effet désiré. Le taux d'EER est réduit de 90% relatif pour les hommes et les femmes pour atteindre respectivement 0,59% et 0,22%. Les impostures de type *play-back* générées pour les 15 phrases utilisées

Table 9.6 – Performance du score deux dimensions comparé au score dépendant du texte du système HiLAM pour les parties de développement et d'évaluation de la base de données *RSR2015*. Les Résultats sont indiqués en termes de C_{llr} et C_{llr}^{min}. La loss de référence est indiqué pour comparaison.

		Hommes		Femmes	
		Dev	*Eval*	*Dev*	*Eval*
Score dépendant du texte ($\mathcal{S}_{hmm}$)	C_{llr}	0.9071	0.9429	0.8297	0.8860
	C_{llr}^{min}	0,9069	0,9410	0,8271	0,8774
Score 2-dimensions ($\mathcal{S}_{sn}$)	C_{llr}	0.5941	0.6110	0.6055	06325
	C_{llr}^{min}	0,5896	0,6075	0,5857	0,6061
Loss de référence		2			

en enrôlement sont mieux rejetées que celles générées avec les 15 phrases inutilisées. Ceci est dû au fait que les modèles GMM supposés indépendants du texte ont été entraînés avec ces 15 phrases. Les taux d'EER obtenus pour les phrases utilisées ou inutilisées varient de 0,38% à 0,67% pour les hommes et de 0,14% à 0,24% pour les femmes. On observe aussi de meilleures performances pour les scores $\mathcal{S}_{sn}$ dans le cas des imposteurs prononçant un mauvais contenu lexical.

Une deuxième expérience est conduite pour évaluer la capacité du système a classifier les quatre types de tests (*Client-correct*, *Client-faux*, *Imposteur-correct*et *Imposteur-faux*) en termes de C_{llr} et de C_{llr}^{min}. Ces performances sont comparées à celles d'un système utilisant un classifieur Gaussien (*Gaussian Back-End*). Les résultats sont présentés dans le tableau 9.6.

Le score en deux dimensions réduit considérablement le C_{llr} pour les femmes comme pour les hommes ; environ 29% et 35% relatifs respectivement. Les performances sur la partie développement sont données en référence puisque le classifieur Gaussien a été entraîné sur cette partie. On observe toutefois un gain similaire sur la partie évaluation du corpus.

Le score proposé, $\mathcal{S}_{sn}$ améliore sensiblement les performances du système de référence lorsqu'il s'agit de rejeter une imposture par *play-back* dans laquelle le locuteur cible prononce un mauvais contenu lexical. L'EER est réduit de 90% relatif dans cette condition pour les hommes et les femmes par rapport au système dépendant du texte. Ce même score $\mathcal{S}_{sn}$ améliore également les résultats dans le cas d'un imposteur prononçant un mauvais contenu lexical.

La deuxième contribution de ce travail consiste à combiner le score dépendant du texte avec le nouveau score $\mathcal{S}_{sn}$ afin de catégoriser les types de tests rencontrés. L'intégration de ce score dual dans un classifieur Gaussien réduit d'au moins 29% pour les hommes comme pour les femmes. Cette amélioration est obtenue pour un coût de calcul négligeable puisque les vraisemblances utilisées pour le calcul du nouveau rapport de vraisemblance sont déjà calculées au sein de l'architecture HiLAM.

Discussions

Bilan

Les travaux présentés dans cette partie ont donné lieu à deux brevets et à l'intégration d'un système incluant le score en deux dimensions dans le système d'exploitation du téléphone LENOVO A586 qui a été commercialisé en 2012.

En plus de ce transfert technologique, les projets que j'ai menés dans le cadre de la reconnaissance du locuteur dépendante du texte m'ont permis d'initier la collecte de plusieurs corpus largement utilisés dans la communauté depuis. Ce point me paraît essentiel pour supporter la recherche dans un domaine où l'évaluation est primordiale et où le développement de nouvelles technologies requiert des quantités toujours plus importantes de données.

RSR2015

La base de données *RSR2015*[Larcher et al., 2012, 2014c] a été collectée dans le cadre du projet *Home2015* financé par l'Institute for Infocomm Research de Singapour. 300 locuteurs (hommes et femmes) ont enregistré 9 sessions chacun en utilisant plusieurs téléphones portables et tablettes afin de collecter 30 phrases courtes et 30 commandes vocales ainsi que 13 séries de chiffres. Cette base de données est utilisée pour reproduire différents scénarios de reconnaissance du locuteur dépendante du texte et est à l'heure actuelle la base de données la plus utilisée dans le domaine. Une version rejouée et enregistrée à travers un canal VHF marine a également été produite pour les besoins d'un autre projet avec les autorités portuaires de Singapour [Larcher et al., 2014d]. Cette seconde version qui permet d'étudier les dégradations du signal de parole à travers un canal largement utilisé dans le monde est également disponible publiquement.

RedDots

La base de données *RSR2015* offre différents scénarios de vérification du locuteur et un grand nombre de locuteurs (comparé aux autres corpus disponibles). Ce corpus souffre cependant du manque de bruit et de variabilité dû au canal d'enregistrement. Afin de pallier ce manque, nous avons, avec LEE Kong Aik, proposé l'enregistrement d'un nouveau corpus : *RedDots* collecté de façon volontaire et collaborative à travers une plateforme mise en place à I^2R (Singapour). Ce corpus consiste en une centaine de locuteurs bénévoles qui ont enregistré une session sur leur propre téléphone portable toutes les semaines au cours d'une année dans des conditions diverses et réalistes. Ce corpus est beaucoup plus difficile que, *RSR2015* mais souffre d'un nombre trop faible de locuteurs. Il permet néanmoins de comparer réellement le cas d'usage de la reconnaissance du locuteur dépendante et indépendante du texte de par la conception des textes prononcés lors des sessions d'enregistrement. Plus de détails sont disponibles dans [Lee et al., 2015].

Perspectives

Les travaux que j'ai réalisés en reconnaissance du locuteur dépendante du texte au LIA puis à I^2R ont été suivit d'autres travaux dans la communauté visant à exploiter la structure temporelle du signal de parole en dérivant un modèle de locuteur indépendant du texte où en contraignant le calcul des statistiques nécessaires à l'estimation des modèles. Ainsi, les travaux de [Zeinali et al., 2016] montrent qu'exploiter l'information phonétique lors du processus d'extraction des *i*-vecteurs apporte un gain substantiel en termes de taux d'erreurs. Dans ces travaux, les auteurs proposent d'extraire les statistiques en utilisant une structuration temporelle de l'espace acoustique par un modèle HMM remplaçant le modèle du monde GMM.

Certaines publications [Variani et al., 2014] supportent l'idée que l'utilisation actuelle des réseaux de neurones pour la reconnaissance du locuteur dépendante du texte nécessite une quantité très importante de données d'apprentissage, plusieurs dizaines de milliers de locuteurs. Dans ce contexte, tous les scénarios ne sont pas envisageables et les seuls résultats positifs dans ce domaine concernent le cas où tous les locuteurs prononcent un mot de passe unique et commun, limitant fortement le potentiel commercial de ces systèmes. Dans un avenir proche, il me semble important de trouver un compromis entre ces approches neuronales développées pour la reconnaissance dépendante du texte et celles proposées pour la reconnaissance indépendante du texte. Dans ce cadre, les travaux présentés dans [Liu et al., 2018] laissent envisager un fort potentiel pour prendre en compte le contenu phonétique et apprendre des représentations de locuteurs robustes à cette variabilité.

De plus, l'observation de la figure 8.1 laisse entrevoir un fort potentiel d'amélioration dans la normalisation phonétique des modèles de locuteurs. La représentation locale (c.-à-d. des courtes durées) du locuteur me semble une voix très prometteuse pour la représentation des locuteurs dans le cadre de la reconnaissance indépendante du texte. En effet, pour les durées inférieures à 10 secondes, la variabilité phonétique reste un des problèmes majeures. Il me paraît également essentiel que les systèmes de reconnaissance intègrent de façon plus importante l'information intra-locuteur en déterminant, si possible de façon automatique, les facteurs caractéristiques d'un individu. Ces éléments caractéristiques doivent être estimés avec un indicateur de leur stabilité [Moez et al., 2016]. Les réseaux de neurones paraissent un outils intéressant pour déterminer de façon non supervisée les descripteurs pertinents des locuteurs car ils sont actuellement les seuls modèles capables d'assimiler la quantité très importante de données nécessaire à cette analyse.

Mon objectif à moyen terme sera donc d'intégrer entre autres l'information phonétique au sein du processus de modélisation du locuteur afin d'apprendre des représentations qui permettront de supprimer ce contexte.

Quatrième partie

Perspectives de recherche

Au cours de ce document, j'ai présenté les avancées qui me paraissent marquantes dans le domaine de la reconnaissance du locuteur pour ces vingt-cinq dernières années. J'ai présenté les travaux que j'ai menés dans ce domaine en précisant que tous ces travaux ont été réalisés dans l'objectif d'être diffusés le plus largement possible à travers des logiciels libres ou des bases de données publiques. Dans les années à venir, les travaux que j'envisage de mener se situent principalement dans trois axes de recherche que je présente ci-après, avant de discuter de perspectives à plus long terme.

Perspectives en reconnaissance du locuteur

Dans la prolongation des travaux que j'ai menés depuis 10 ans, je souhaite continuer ma réflexion sur les modèles acoustiques pour la reconnaissance du locuteur en travaillant principalement sur l'intégration d'informations auxiliaires pour les courtes durées.

L'évolution que je constate dans les domaines de la reconnaissance du locuteur indépendante et dépendante du texte me conforte dans l'idée que le verrou technologique actuel principale consiste à faire converger ces deux technologies.

Les contraintes imposées en reconnaissance dépendante du texte permettent de réduire les taux d'erreurs, mais soulèvent deux problèmes principaux :

— pour parvenir à des taux d'authentification assez bas pour être exploités commercialement, la contrainte sur le texte doit être très forte et la convivialité des systèmes s'en trouve affectée ;

— aux vues des analyses rapportées dans la partie III, la nature de l'information qui est reconnue lors du processus d'authentification dépendante du texte n'est pas certaine, et la part de l'information phonétique présente dans les représentations reste très importante.

Du point de vue de la reconnaissance du locuteur indépendante du texte, les méthodes embarquant une information phonétique pour l'extraction des *i*-vecteurs [Lei et al., 2014] ou des *x*-vectors [Liu et al., 2018] amènent des gains substantiels qui laissent entrevoir ce potentiel.

Mon objectif est donc de travailler à la jonction des technologies dépendantes et indépendantes du texte pour relâcher la contrainte sur les utilisateurs tout en exploitant l'information phonétique afin de contextualiser les représentations de locuteur dans un espace acoustique en contexte. Les travaux récents sur les réseaux de neurones et la plasticité des architectures laissent entrevoir un fort potentiel d'intégration de différents flux d'information dans les systèmes de reconnaissance du locuteur.

Mes travaux s'inscriront d'abord dans le cadre du projet Deep Privacy[3]. Dans ce pro-

3. financé par l'ANR sur la période 2019-2021

jet, je souhaite exploiter l'information phonétique et lexicale afin d'entraîner un système neuronal à modéliser les locuteurs en contexte ou plutôt en séparant le contexte lexical de l'identité du locuteur. L'objectif est de projeter les informations liées au locuteur dans un espace indépendant de l'information lexicale tout en extrayant en parallèle l'information lexicale et en réduisant au maximum les composantes propres à l'identité du locuteur au cours de ce processus. Nous explorerons pour commencer les processus d'apprentissage collaboratif ou conjoint pour évaluer la part d'information lexicale qui se retrouve dans les représentations des locuteurs (et vice-versa). Notre objectif ultérieur étant de définir de nouvelles architectures de systèmes améliorant la séparation des informations pour renforcer l'indépendance lexicale des représentations de locuteurs.

À plus long terme, il me paraît souhaitable d'exploiter d'autres flux d'informations au cours de la modélisation acoustique des locuteurs, par exemple le genre ou la langue qui permettraient d'exploiter des quantités de données plus importantes tout en produisant des systèmes plus génériques, ou bien des informations relatives à l'environnement sonore pour rendre les systèmes plus robustes au bruit. L'intégration de ces informations pourrait permettre de mieux comprendre le fonctionnement des systèmes automatiques et d'analyser le rôle des différentes informations dans le processus de reconnaissance du locuteur.

Rapprochement avec les systèmes de captation

Le deuxième axe de recherche que je privilégierai dans les années à venir s'établit dans une coopération avec le Laboratoire d'Acoustique de l'Université du Mans (LAUM) à travers deux projets financés.

Cet axe a pour ambition de supprimer la barrière existant entre les domaines de l'acoustique et du traitement de la parole. Mon but est de développer un système unique exploitant les sorties d'une antenne de microphones pour détecter, localiser, identifier et suivre plusieurs locuteurs dans un même environnement afin d'améliorer les performances des systèmes de transcription enrichies. Les avancées récentes dans les domaines du *beamforming*, de la séparation de source, du débruitage, de la détection de parole, de la transcription et de la reconnaissance du locuteur utilisent majoritairement des réseaux de neurones profonds. Je propose dans ce projet de fusionner les approches acoustiques et de traitement de la parole au sein d'un réseau de neurones unique. L'enjeu de ces travaux peut être résumé par la figure 9.6. La chaîne actuelle (Figure 9.6) pour le traitement de la parole regroupe la captation ainsi qu'un premier traitement des signaux obtenus, effectué pour produire un signal mono (ou stéréo). Dans un deuxième temps, et de façon distincte, le signal obtenu est fourni à un système de traitement de la parole qui ignore tout du système de captation et des prétraitements effectués. La partie inférieure de la figure décrit la chaîne de traitement proposée pour ce projet : les multiples flux captés subissent

un prétraitement (facultatif) et sont fournis à un système de traitement de la parole multi-flux qui exploite la configuration du système de captation afin d'améliorer les performances pour les différentes tâches.

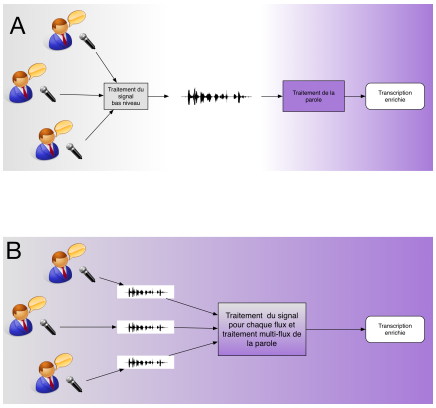

Figure 9.6 – A. Chaîne de traitement actuelle : deux parties distinctes.
B. chaîne de traitement proposée : un système unique où l'information sur le système de captation est fournie au module de traitement de la parole.

Les questions fondamentales liées à cette problématique sont les suivantes :

— quelle partie des prétraitements peut être intégrée dans les réseaux de neurones ?
— quelle partie des prétraitements est-il souhaitable d'effectuer dans les réseaux de neurones, étant donné les ressources et le temps de calcul nécessaires à leur utilisation ?
— quelles sont les informations disponibles sur l'antenne de capteurs qui peuvent être exploitées par les modèles neuronaux ?

Dans une première étape je souhaite utiliser un réseau de neurones pour faire de la détection de parole et localiser la source dans l'espace (estimer les coordonnées de la source) afin de constituer une « carte » de l'environnement. Ici, il s'agit de fusionner la localisation de source faite au LAUM avec la caractérisation faite au LIUM (classification du signal en parole, bruit, musique, silence), tout en utilisant des techniques neuronales. Par la suite, nous étudierons la possibilité, avec une même antenne de capteurs, de localiser et caractériser une source dans différents environnements acoustiques (extérieur, pièce avec différents degrés de réverbération) comme le ferait un humain.

Dans une deuxième étape je souhaite abandonner les antennes à géométrie fixe pour exploiter des réseaux de microphones disparates, par exemple l'ensemble des téléphones portables présents dans une salle de réunion ou, plus généralement, une collection d'objets connectés disposant des capteurs appropriés.

Systèmes autonomes

En plus de la reconnaissance du locuteur, j'ai pu, depuis mon intégration au LIUM, explorer des thèmes de recherche qui n'ont pas été traités dans ce document comme l'adaptation au locuteur des systèmes de reconnaissance de la parole (lors de la thèse de Natalia Tomaschenko [Tomashenko, 2017; Tomashenko et al., 2016a,b]). D'autres travaux m'ont permis d'aborder la détection de parole avec Florent Desnous que je co-encadre en thèse [Desnous et al., 2018] ou la détection de genre [Doukhan et al., 2018]. Dans la prolongation des travaux menés lors de la thèse de Gaël Le Lan pour un apprentissage continu non supervisé [Le Lan et al., 2018], je coordonne le projet ALLIES pour l'extension au life-long learning, dont le but est de favoriser le développement de systèmes autonomes en fournissant un cadre d'évaluation objectif. Aujourd'hui, les systèmes d'apprentissage automatique tirent leur valeur de l'exploitation de quantité de données importantes. La sélection et la préparation de ces données ainsi que le réglage des paramètres des systèmes nécessitent des connaissances expertes en apprentissage automatique qui ne sont accessibles qu'à des entreprises disposant de services R&D importants. Malheureusement, les performances des systèmes actuels dépendent de la proximité entre les données d'apprentissage et les données rencontrées lors de leur utilisation. Lorsque l'environnement du système change, il est nécessaire de ré apprendre les modèles et des entreprises de taille modeste ne peuvent financer ce processus. Dans le but de permettre l'utilisation de ces systèmes par le plus grand nombre d'acteurs, il est nécessaire de rendre ces systèmes autonomes afin de s'affranchir des experts en apprentissage automatique.

Le projet ALLIES, que je coordonne, vise à développer des systèmes non supervisés qui soient capables de s'adapter à leur environnement (collecter des données, adapter leurs modèles), mais également de s'auto-évaluer afin de maintenir leur niveau de performance au cours du temps. L'autonomie recherchée n'interdit cependant pas le recours à des compétences humaines, à travers de l'*active-* ou *interactive-learning*. Dans ce cas, ce n'est pas un expert en apprentissage automatique qui interagira avec le système, mais bien un expert «métier» qui sera en mesure d'évaluer les sorties du système pour la tâche considérée.

Si chaque tâche requiert le développement de systèmes adaptés, l'ensemble des tâches doit partager un processus d'évaluation des systèmes autonomes. Le projet ALLIES tend à développer un cadre d'évaluation incluant des métriques, des protocoles et des normes, propres à favoriser le développement et l'évaluation de systèmes autonomes par différentes communautés scientifiques. Les tâches considérées dans le projet sont la segmentation et regroupement en locuteur ainsi que la traduction automatique.

Perspectives à long terme

Les perspectives présentées ci-dessus traitent de problématiques technologiques et d'évaluation ; deux domaines qui ont vu des avancées marquantes durant les dernières décennies. Il est plus que probable que les performances des systèmes de reconnaissance du locuteur poursuivent leur progression dans les années qui viennent, en offrant davantage de robustesse à l'environnement et de flexibilité pour l'utilisateur. De même, la communauté scientifique poursuivra certainement le renforcement des méthodes d'évaluations, des protocoles et la tendance actuelle à permettre la reproductibilité des expériences démontrent une volonté générale d'objectivation.

Lorsqu'on dépasse cette perspective technologique, il est essentiel de réaliser que cette décennie a été marquée par le déploiement massif des technologies vocales, qu'il s'agisse des terminaux mobiles ou de la domotique, des traitements de la parole pour les centres d'appels ou des progrès dans le domaine médical.

Les systèmes automatiques ont atteint des performances suffisantes et le grand public est prêt à utiliser ces technologies : un cap a été franchi.

À l'heure où les grands groupes industriels engagent des moyens colossaux dans le développement des algorithmes il est du devoir de la recherche académique de diffuser nos connaissances afin d'offrir un support citoyen aux législateurs, aux juristes, aux États et aux acteurs commerciaux, afin d'encadrer l'utilisation de ces technologies. Alors qu'un premier règlement européen sur la protection des données[4] vient d'être mis en place, il est essentiel de participer à la définition des données personnelles, de caractériser l'aspect «biométrique» ou «personnel» des données utilisées en reconnaissance du locuteur et de permettre une meilleure compréhension des informations et des algorithmes utilisés. Permettre la rencontre des communautés juridiques, institutionnelles, scientifiques et civiles fera partie des actions que je souhaite mener sur la durée.

4. https://www.dpms.eu/rgpd/ vu le 18/10/2018

Annexes

ANNEXE A

Théorie du Factor Analyser

Cette annexe est fournie pour une meilleure compréhension d'un modèle largement utilisé en reconnaissance du locuteur. De nombreux ouvrages donnent une preuve partielle des résultats fondamentaux et il est souvent compliqué pour les lecteurs d'assembler toutes les informations parcellaires publiées dans différentes publications du domaine. Je tenais à présenter cette annexe afin de rassembler une fois pour toute ("once for all") cette démonstration qui m'a souvent fait défaut.

A.1. Définition et théorie

Dans le paradigme du *Factor Analyser* , on suppose que la plus grande partie de la variabilité d'une souce suivant une distributin normale réside dans un sous-espace linéaire de dimension réduite. Dans ce sous-espace, la covariance de la distribution de probabiltié est modélisée par une matrice pleine tandis que la covariance résiduelle est modélisée par une matrice diagonale de rang plein. La densité de probabilité d'un *Factor Analyser* est ainsi décrite par l'équation suivante :

$$Pr(\mathbf{x}) = \mathcal{N}_{\mathrm{x}}(\boldsymbol{\mu}, \Phi\Phi^T + \Sigma) \tag{A.1}$$

où Φ est une matrice portrait rectangulaire de rang$r$ (donc une matrice de dimension $d \times r$) dont les colonnes définissent une base du sous-espace linéaire de rang r et Σ est

une matrice diagonale de rang plein. La matrice Φ est appelée matrice de facteurs. Une représentation graphique du modèle de *Factor Analyser* est donnée par la figure A.1.

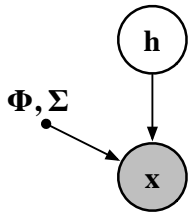

Figure A.1 – Modèle graphique du *Factor Analyser* .

A.2. Le Factor Analyser vu comme une marginalisation

Le *Factor Analyser* peut être vu comme la marginalisation d'une distribution conjointe entre une variable observée, $\mathbf{x}$, et une variable cachée, $\mathbf{h}$, de dimension r telles que :

$$\begin{cases} \mathbf{h} \sim \mathcal{N}(0, \mathbf{I}) & \text{(A.2)} \\ \boldsymbol{\epsilon} \sim \mathcal{N}(0, \Sigma) & \text{(A.3)} \\ \mathbf{x} = \boldsymbol{\mu} + \Phi\mathbf{h} + \boldsymbol{\epsilon} & \text{(A.4)} \end{cases}$$

Pour démontrer ce point nous commençons par exprimer une distribution Gaussienne multivariée avec des variables $\mathbf{h}$ et $\mathbf{x}$:

$$\begin{bmatrix} \mathbf{h} \\ \mathbf{x} \end{bmatrix} \sim \mathcal{N}(\boldsymbol{\mu}^{\text{hx}}, \Sigma^{\text{hx}}) \tag{A.5}$$

pour calculer ensuite $\boldsymbol{\mu}^{\text{hx}}$ et Σ^{hx}. L'équation A.2 nous indique que :

$$E[h] = 0 \tag{A.6}$$

et l'équation A.4 nous donne :

$$\begin{aligned} E[\mathbf{x}] &= E[\boldsymbol{\mu} + \Phi\mathbf{h} + \boldsymbol{\epsilon}] \\ &= \boldsymbol{\mu} + \Phi E[\mathbf{h}] + E[\boldsymbol{\epsilon}] \\ &= \boldsymbol{\mu}. \end{aligned}$$

Ainsi nous obtenons :

$$\boldsymbol{\mu}^{\text{hx}} = \begin{bmatrix} 0 \\ \boldsymbol{\mu} \end{bmatrix}. \tag{A.7}$$

La matrice Σ^{hx} peut être calculée en trois parties car :

$$\Sigma^{hx} = \begin{bmatrix} \Sigma_{hh} & \Sigma_{hx} \\ \Sigma_{hx}{}^{T} & \Sigma_{xx} \end{bmatrix} \tag{A.8}$$

L'équation A.2 nous donne : $\Sigma_{hh} = Cov(\mathbf{h}) = \mathbf{I}$.
Et nous avons aussi :

$$\begin{aligned} \Sigma_{hx} &= E\left[(\mathbf{h} - E[\mathbf{h}])(\mathbf{x} - E[\mathbf{x}])^T\right] \\ &= E\left[\mathbf{h}(\boldsymbol{\mu} + \Phi\mathbf{h} + \boldsymbol{\epsilon} - \boldsymbol{\mu})^T\right] \\ &= E\left[\mathbf{h}\mathbf{h}^T\right]\Phi^T + E\left[\mathbf{h}\boldsymbol{\epsilon}^T\right] \\ &= \Phi^T \end{aligned}$$

Le troisième terme nous donne :

$$\begin{aligned} \Sigma_{xx} &= E\left[(\mathbf{x} - E[\mathbf{x}])(\mathbf{x} - E[\mathbf{x}])^T\right] \\ &= E\left[(\boldsymbol{\mu} + \Phi\mathbf{h} + \boldsymbol{\epsilon} - \boldsymbol{\mu})(\boldsymbol{\mu} + \Phi\mathbf{h} + \boldsymbol{\epsilon} - \boldsymbol{\mu})^T\right] \\ &= E\left[\Phi\mathbf{h}\mathbf{h}^T\Phi^T + \boldsymbol{\epsilon}\mathbf{h}\Phi^T + \Phi\mathbf{h}\boldsymbol{\epsilon}^T + \boldsymbol{\epsilon}\boldsymbol{\epsilon}^T\right] \\ &= \Phi E\left[\mathbf{h}\mathbf{h}^T\right]\Phi^T + E\left[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T\right] \\ &= \Phi\Phi^T + \Sigma \end{aligned}$$

et donc :

$$\begin{bmatrix} \mathbf{h} \\ \mathbf{x} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} 0 \\ \boldsymbol{\mu} \end{bmatrix}, \begin{bmatrix} \mathbf{I} & ,\Phi^T \\ \Phi & ,\Phi\Phi^T + \Sigma \end{bmatrix} \right) \tag{A.9}$$

De là nous pouvons voir que la distribution marginale de $\mathbf{x}$ est donnée par $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \Phi\Phi^T + \Sigma)$ comme dans l'équation A.1.

Malheureusement, aucune solution ne peut être trouvée directement pour estimer les paramètrs $\boldsymbol{\theta} = (\boldsymbol{\mu}, \Phi, \Sigma)$ du *Factor Analyser* .

Nous utilisons le fait que le *Factor Analyser* peut être représenté comme une marginalisation afin d'estimer ses paramètres à l'aide d'un algorithme EM. Le processus d'estimation est décrit dans la section A.4 mais il est nécessaire de démontrer quelques propriétés des distributions Gaussiennes multi-variées avant d'aller plus loin.

A.3. Distribution Gaussienne multi-variée

A.3.1 Probabilité conditionelle

Considérons deux variables aléatoires multi-variées $\mathbf{x}_1 \in \mathbb{R}^d$ et $\mathbf{x}_2 \in \mathbb{R}^r$ telles que $\mathbf{x} \in \mathbb{R}^{d+r}$ soit défini par :

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} \tag{A.10}$$

Supposons que $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, où

$$\boldsymbol{\mu} = \begin{bmatrix} \boldsymbol{\mu}_1 \\ \boldsymbol{\mu}_2 \end{bmatrix}, \Sigma = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \tag{A.11}$$

Nous nous proposons de montrer que la distribution conditionnelle de $\mathbf{x}_i$ étant donné $\mathbf{x}_j$ est normale et que sopn vecteur moyen et sa matrice de covariance sont donnés par :

$$\begin{cases} \boldsymbol{\mu}_{i|j} = \boldsymbol{\mu}_i + \Sigma_{ij}\Sigma_{jj}^{-1}(x_j - \boldsymbol{\mu}_j) & \text{(A.12)} \\ \Sigma_{i|j} = \Sigma_{ii} - \Sigma_{ij}\Sigma_{jj}^{-1}\Sigma_{ji} & \text{(A.13)} \end{cases}$$

Contrairement à ce qu'annonce de nombreux ouvrages, la preuve de ces formules n'est pas triviale et nécessite quelques résultats préliminaires qui sont fournis par les théorèmes 1, 2, 3 et 4. Étant donné ces résultats, les expressions du vecteur moyen et de la matrice de covariance peuvent être prouvées comme suit.

A.3.2 Preuve

La densité de probabilité de $\mathbf{x}$ est donnée par :

$$Pr(\mathbf{x}) = Pr(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{(2\Pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^T\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu})\right]$$

Nous noterons $\Upsilon(\mathbf{x}_1, \mathbf{x}_2)$ le terme quadratique de l'exponentielle tel que :

$$Pr(\mathbf{x}) = Pr(\mathbf{x}_1, \mathbf{x}_2) = \frac{1}{(2\Pi)^{\frac{n}{2}}|\Sigma|^{\frac{1}{2}}} \exp\left[-\frac{1}{2}\Upsilon(\mathbf{x}_1, \mathbf{x}_2)\right] \tag{A.14}$$

Focalisons nous d'abord sur le terme quadratque

En utilisant le résultat du théorème 2 on note :

$$\begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{bmatrix}$$

et

$$\begin{aligned} \Upsilon(\mathbf{x}_1, \mathbf{x}_2) &= (\mathbf{x} - \boldsymbol{\mu})^T \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu}) \\ &= \left[(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T, (\mathbf{x}_2 - \boldsymbol{\mu}_2)^T \right] \begin{bmatrix} \mathbf{S}_{11} & \mathbf{S}_{12} \\ \mathbf{S}_{21} & \mathbf{S}_{22} \end{bmatrix} \begin{bmatrix} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \\ (\mathbf{x}_2 - \boldsymbol{\mu}_2) \end{bmatrix} \\ &= (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \mathbf{S}_{11} (\mathbf{x}_1 - \boldsymbol{\mu}_1) + 2(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \mathbf{S}_{12} (\mathbf{x}_2 - \boldsymbol{\mu}_2) + (\mathbf{x}_2 - \boldsymbol{\mu}_2)^T \mathbf{S}_{22} (\mathbf{x}_2 - \boldsymbol{\mu}_2) \end{aligned}$$

En utilisant le résultat du théorème 2 et en en utilisant les équations B.3, B.4 et B.5 dans cette expression, on obtient :

$$\begin{aligned} \Upsilon(\mathbf{x}_1, \mathbf{x}_2) &= (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \left[\Sigma_{11}^{-1} + \Sigma_{11}^{-1} \Sigma_{12} (\Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12})^{-1} \Sigma_{12}^T \Sigma_{11}^{-1} \right] (\mathbf{x}_1 - \boldsymbol{\mu}_1) \\ &\quad - 2(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \left[\Sigma_{11}^{-1} \Sigma_{12} (\Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12})^{-1} \right] (\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &\quad + (\mathbf{x}_2 - \boldsymbol{\mu}_2)^T \left[\Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12} \right]^{-1} (\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &= (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \Sigma_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \\ &\quad + (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \left[\Sigma_{11}^{-1} \Sigma_{12} (\Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12})^{-1} \Sigma_{12}^T \Sigma_{11}^{-1} \right] (\mathbf{x}_1 - \boldsymbol{\mu}_1) \\ &\quad - 2(\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \left[\Sigma_{11}^{-1} \Sigma_{12} (\Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12})^{-1} \right] (\mathbf{x}_2 - \boldsymbol{\mu}_2) \\ &\quad + (\mathbf{x}_2 - \boldsymbol{\mu}_2)^T \left[\Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12} \right]^{-1} (\mathbf{x}_2 - \boldsymbol{\mu}_2) \end{aligned}$$

En utilisant maintenant le théorème 4 on obtient :

$$\begin{aligned} \Upsilon(\mathbf{x}_1, \mathbf{x}_2) &= (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \Sigma_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \\ &+ \left[(\mathbf{x}_2 - \boldsymbol{\mu}_2) - \Sigma_{12}^T \Sigma_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \right]^T \left(\Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12} \right)^{-1} \left[(\mathbf{x}_2 - \boldsymbol{\mu}_2) - \Sigma_{12}^T \Sigma_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \right] \end{aligned}$$

Pour plus de lisibilité on note par la suite :

$$\begin{cases} \mathbf{b} = \boldsymbol{\mu}_2 + \Sigma_{12}^T \Sigma_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) & \text{(A.15)} \\ \mathbf{A} = \Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12} & \text{(A.16)} \end{cases}$$

et :

$$\begin{cases} \Upsilon_1(\mathbf{x}_1) = (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \Sigma_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) & \text{(A.17)} \\ \Upsilon_2(\mathbf{x}_1, \mathbf{x}_2) = (\mathbf{x}_2 - \mathbf{b})^T \mathbf{A}^{-1} (\mathbf{x}_2 - \mathbf{b}) & \text{(A.18)} \end{cases}$$

Le terme quadratique peut alors être écrit :

$$\Upsilon(\mathbf{x}_1, \mathbf{x}_2) = \Upsilon_1(\mathbf{x}_1) + \Upsilon_2(\mathbf{x}_1, \mathbf{x}_2) \tag{A.19}$$

Focalisons maintenant le premier terme

Le théorème 3 nous permet d'écrire le dénominateur du premier terme ainsi :

$$\begin{aligned} (2\Pi)^{\frac{n}{2}} |\Sigma|^{\frac{1}{2}} &= (2\Pi)^{\frac{d+r}{2}} |\Sigma_{11}|^{\frac{1}{2}} |\Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12}|^{\frac{1}{2}} \\ &= (2\Pi)^{\frac{d}{2}} |\Sigma_{11}|^{\frac{1}{2}} (2\Pi)^{\frac{r}{2}} |\mathbf{A}|^{\frac{1}{2}} \end{aligned}$$

Conclusion

La densité de probabilité peut maintenant être formulée comme suit :

$$Pr(\mathbf{x}) = \frac{1}{(2\Pi)^{\frac{d}{2}} |\Sigma_{11}|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \Sigma_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \right] \frac{1}{(2\Pi)^{\frac{r}{2}} |\mathbf{A}|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} (\mathbf{x}_2 - \mathbf{b})^T \mathbf{A}^{-1} (\mathbf{x}_2 - \mathbf{b}) \right] \tag{A.20}$$

$$= \mathcal{N}(\mathbf{x}_1, \boldsymbol{\mu}_1, \Sigma_{11}) \mathcal{N}(\mathbf{x}_2, \mathbf{b}, \mathbf{A}) \tag{A.21}$$

Ainsi, la distribution marginale de $\mathbf{x}_1$ est :

$$\begin{aligned} pr_1(\mathbf{x}_1) &= \int Pr(\mathbf{x}_1, \mathbf{x}_2) d\mathbf{x}_2 \\ &= \frac{1}{(2\Pi)^{\frac{d}{2}} |\Sigma_{11}|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} (\mathbf{x}_1 - \boldsymbol{\mu}_1)^T \Sigma_{11}^{-1} (\mathbf{x}_1 - \boldsymbol{\mu}_1) \right] \end{aligned}$$

et la distribution conditionnelle de $\mathbf{x}_2 | \mathbf{x}_1$ est normale :

$$\begin{aligned} Pr_{2|1}(\mathbf{x}_2 | \mathbf{x}_1) &= \frac{Pr(\mathbf{x}_1, \mathbf{x}_2)}{Pr_1(\mathbf{x}1)} \\ &= \frac{1}{(2\Pi)^{\frac{r}{2}} |A|^{\frac{1}{2}}} \exp\left[-\frac{1}{2} (\mathbf{x}_2 - \mathbf{b})^T A^{-1} (\mathbf{x}_1 - \mathbf{b}) \right] \end{aligned}$$

et sont vecteur moyen et sa matrice de covariance sont :

$$\begin{cases} \mathbf{b} = \boldsymbol{\mu}_2 + \Sigma_{12}^T \Sigma_{11}^{-1}(\mathbf{x}_1 - \boldsymbol{\mu}_1) & \text{(A.22)} \\ \mathbf{A} = \Sigma_{22} - \Sigma_{12}^T \Sigma_{11}^{-1} \Sigma_{12} & \text{(A.23)} \end{cases}$$

Ce qui est ce que nous voulions montrer.

A.4. Algorithme EM pour le Factor Analyser

A.4.1 Expectation (estimation de l'espérance)

Pendant la phase d'expectation, on utilise la distribution décrite par $q_n(\mathbf{h}_n)$ tel que :

$$q_n(\mathbf{h}_n) = Pr(\mathbf{h}_n|\mathbf{x}, \boldsymbol{\mu}, \Phi, \Sigma)$$

En remplaçant la distribution donnée par l'équation A.9 dans les formules A.12 et A.13 on voit que la distribution conditionelle de $\mathbf{h}$ étant donné $\mathbf{x}$ est une distribution normale telle que :

$$\mathbf{h}|\mathbf{x}, \boldsymbol{\mu}, \Phi, \Sigma \sim \mathcal{N}(\boldsymbol{\mu}_{\mathrm{h|x}}, \Sigma_{\mathrm{h|x}}) \quad \text{(A.24)}$$

avec :

$$\begin{cases} \boldsymbol{\mu}_{\mathrm{h|x}} = \Phi^T(\Phi\Phi^T + \Sigma)^{-1}(\mathbf{x} - \boldsymbol{\mu}) \\ \Sigma_{\mathrm{h|x}} = I - \Phi^T(\Phi\Phi^T + \Sigma)^{-1}\Phi \end{cases} \quad \text{(A.25)}$$

Ce qui est équivalent à :

$$\begin{cases} \boldsymbol{\mu}_{\mathrm{h|x}} = (\Phi\Sigma^{-1}\Phi + \mathbf{I})^{-1}\Phi^T\Sigma^{-1}(\mathbf{x} - \boldsymbol{\mu}) & \text{(A.26)} \\ \Sigma_{\mathrm{h|x}} = (\Phi\Sigma^{-1}\Phi + \mathbf{I})^{-1} & \text{(A.27)} \end{cases}$$

L'équivalence pour $\boldsymbol{\mu}_{h|x}$ est obtenue en appliquant le théorème : 1 dans :

$$\begin{aligned}(\Phi\Sigma^{-1}\Phi+\mathbf{I})^{-1}\Phi^T\Sigma^1 &= (\mathbf{I}-\Phi^T(\Sigma+\Phi\Phi^T)^{-1}\Phi)\Phi^T\Sigma^{-1}\\ &= \Phi^T\Sigma^{-1}-\Phi^T(\Sigma+\Phi\Phi^T)^{-1}\Phi\Phi\Sigma^{-1}\\ &= \Phi^T(\Sigma^{-1}-(\Sigma+\Phi\Phi^T)^{-1}\Phi\Phi\Sigma^{-1})\\ &= \Phi^T(\Sigma+\Phi\Phi^T)^{-1}\Big((\Sigma+\Phi\Phi^T)\Sigma^{-1}-\Phi\Phi\Sigma^{-1}\Big)\\ &= \Phi^T(\Sigma+\Phi\Phi^T)^{-1}\Big(\mathbf{I}+\Phi\Phi^T\Sigma^{-1}-\Phi\Phi^T\Sigma^{-1}\Big)\\ &= \Phi^T(\Sigma+\Phi\Phi^T)^{-1}\end{aligned}$$

Et l'équivalence pour $\Sigma_{h|x}$ est obtenue en appliquant le théorème 1 à l'équation A.25.

Maintenant on peut calculer la distribution $Pr(\mathbf{h}_i|\mathbf{x}_i)$ pour chaque $\mathbf{h}_i$ étant donnée l'observation associée $\mathbf{x}_i$ et la valeur courante des paramètres du *Factor Analyser* . L'espérance qui en résulte est donnée par :

$$E[\mathbf{h}_i] = (\Phi\Sigma^{-1}\Phi+\mathbf{I})^{-1}\Phi^T\Sigma^1(\mathbf{x}_i-\boldsymbol{\mu}_i) \tag{A.28}$$

$$\begin{aligned}E\left[\mathbf{h}_i\mathbf{h}_i^T\right] &= E\left[(\mathbf{h}_i-E[\mathbf{h}_i])(\mathbf{h}_i-E[\mathbf{h}_i])^T\right]+E[\mathbf{h}_i]\,E[\mathbf{h}_i]^T\\ &= (\Phi\Sigma^{-1}\Phi+\mathbf{I})^{-1}+E[\mathbf{h}_i]\,E[\mathbf{h}_i]^T\end{aligned} \tag{A.29}$$

A.4.2 Maximization

Pendant l'étape M, on met à jour les paramètres θ pour maximiser la borne inférieure et

$$\begin{aligned}\hat{\theta} &= \underset{\theta}{\operatorname{argmax}}\left[\sum_{n=1}^{N}\int q_n(\mathbf{h}_n)\log\left[\frac{Pr(\mathbf{x}_n,\mathbf{h}_n|\theta)}{q_n(\mathbf{h}_n)}\right]d\mathbf{h}_n\right]\\ &= \underset{\theta}{\operatorname{argmax}}\left[\sum_{n=1}^{N}\int q_n(\mathbf{h}_n)\log\left[Pr(\mathbf{x}_n,\mathbf{h}_n|\theta)\right]d\mathbf{h}_n\right]\\ &= \underset{\theta}{\operatorname{argmax}}\Big[E\left[\log(Pr(\mathbf{x}_n|\mathbf{h}_n,\theta))\right]\Big]\end{aligned}$$

On note que :

1. le dénominateur peut être ignoré car il ne dépend par du paramètre θ ;
2. l'espérance est calculée e, considérant la distribution $q_n(\mathbf{h}_n)$

Afin de maximiser la borne inférieure par rapport aux autres paramètres, l'expression

de :

$$E\left[\log(Pr(\mathbf{x}_n|\mathbf{h}_n,\theta)\right] = E\left[-\frac{d\log(2\Pi)+\log(|\Sigma|)+(\mathbf{x}_n-\boldsymbol{\mu}-\Phi\mathbf{h}_n)^T(\mathbf{x}_n-\boldsymbol{\mu}-\Phi\mathbf{h}_n)}{2}\right] \tag{A.30}$$

est dérivée et on estime les paramètres lorsque cette dérivée est nulle : On obtient ainsi :

$$\begin{cases} \hat{\mu} = \dfrac{\sum_{i=1}^{N}\mathbf{x}_n}{N} & \text{(A.31)} \\ \hat{\Phi} = \left(\displaystyle\sum_{i=1}^{N}(\mathbf{x}_n-\hat{\boldsymbol{\mu}})E\left[\mathbf{h}_n\right]^T\right)\left(\sum_{i=1}^{N}E\left[\mathbf{h}_n\mathbf{h}_n^T\right]\right)^{-1} & \text{(A.32)} \\ \hat{\Sigma} = \dfrac{1}{N}\displaystyle\sum_{n=1}^{N}\text{diag}\left[(\mathbf{x}_n-\boldsymbol{\mu})(\mathbf{x}_n-\boldsymbol{\mu})^T-\hat{\Phi}E\left[\mathbf{h}_n\right](\mathbf{x}_n-\hat{\boldsymbol{\mu}})^T\right] & \text{(A.33)} \end{cases}$$

ANNEXE B

Théorèmes utiles

B.1. Théorème 1 : Inverse d'une somme

Théorème 1: Inverse d'une somme

Étant donnée une matrice carré pouvant s'écrire : $A+CBD$, avec A et B inversibles. On propose de montrer que :

$$(A+CBD)^{-1} = A^{-1} - A^{-1}C(B^{-1}+DA^{-1}C)^{-1}DA^{-1} \tag{B.1}$$

Preuve

Ce théorème est démontrée en calculant le produit de la mtrice d'origine par son inverse.

$$\begin{aligned}(A+CBD)\left[A^{-1}-\right. & \left. A^{-1}C(B^{-1}+DA^{-1}C)^{-1}DA^{-1}\right] \\ &= (A+CBD)A^{-1} - (A+CBD)A^{-1}C(B^{-1}+DA^{-1}C)^{-1}DA^{-1} \\ &= I + CBDA^{-1} - (C+CBDA^{-1}C)(B^{-1}+DA^{-1}C)^{-1}DA^{-1} \\ &= I + CBDA^{-1} - CB(B^{-1}+DA^{-1}C)(B^{-1}+DA^{-1}C)^{-1}DA^{-1} \\ &= I + CBDA^{-1} - CBDA^{-1} \\ &= I\end{aligned}$$

B.2. Théorème 2 : Complément de Schur

Théorème 2: L'inverse d'une matrice par bloc

Pour A une matrice symétrique $n \times n$ définie telle que :

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} \\ A_{12}^T & A_{22} \end{bmatrix}$$

On note que la seconde égalité est due au fait que A est symétrique.
L'inverse, B, de la matrice A est une matrice $n \times n$ symétrique qui vérifie :

$$A^{-1} = B = \begin{bmatrix} B_{11} & B_{12} \\ B_{12}^T & B_{22} \end{bmatrix}$$

où A_{11} et B_{11} sont $d \times d$; A_{22} et B_{22} sont $r \times r$; A_{12} et B_{12} sont $d \times r$; $d + r = n$; avec

$$\begin{aligned} B_{11} &= (A_{11} - A_{12}A_{22}^{-1}A_{12}^T)^{-1} && \text{(B.2)} \\ &= A_{11}^{-1} + A_{11}^{-1}A_{12}(A_{22} - A_{12}^T A_{11}^{-1}A_{12})^{-1}A_{12}^T A_{11}^{-1} && \text{(B.3)} \\ B_{22} &= (A_{22} - A_{12}^T A_{11}^{-1}A_{12})^{-1} && \text{(B.4)} \\ &= A_{22}^{-1} + A_{22}^{-1}A_{12}^T(A_{11} - A_{12}A_{22}^{-1}A_{12}^T)^{-1}A_{12}A_{22}^{-1} \\ B_{12}^T &= -A_{22}^{-1}A_{12}^T(A_{11} - A_{12}A_{22}^{-1}A_{12}^T)^{-1} && \text{(B.5)} \\ B_{12} &= -A_{11}^{-1}A_{12}(A_{22} - A_{12}^T A_{11}^{-1}A_{12})^{-1} \end{aligned}$$

On note que les secondes égalités pour B_{11} et B_{22} sont obtenues en appliquant le théorème 1.

Preuve

On montre que le produit de A et de son inverse est l'identité. C'est à dire que :

$$AA^{-1} = AB = \begin{bmatrix} A_{11} & A_{12} \\ A_{12}^T & A_{22} \end{bmatrix} \begin{bmatrix} B_{11} & B_{12} \\ B_{12}^T & B_{22} \end{bmatrix} = I_n$$

Ce qui est équivalent à :

$$\begin{bmatrix} A_{11}B_{11} + A_{12}B_{12}^T & A_{11}B_{12} + A_{12}B_{22} \\ A_{12}^T B_{11} + A_{22}B_{12}^T & A_{12}^T B_{12} + A_{22}B_{22} \end{bmatrix} = \begin{bmatrix} I_d & 0 \\ 0 & I_r \end{bmatrix}$$

La preuve pour le premier bloc est obtenue en remplaçant B_{12}^T par sa première expression dans :

$$\begin{aligned} A_{11}B_{11} + A_{12}B_{12}^T &= A_{11}B_{11} - A_{12}A_{22}^{-1}A_{12}^T(A_{11} - A_{12}A_{22}^{-1}A_{12}^T)^{-1} \\ &= A_{11}B_{11} - A_{12}A_{22}^{-1}A_{12}^T B_{11} \\ &= (A_{11} - A_{12}A_{22}^{-1}A_{12}^T)B_{11} \\ &= I_d \end{aligned}$$

La dernière égalité provient du fait que $B_{11} = (A_{11} - A_{12}A_{22}^{-1}A_{12}^T)^{-1}$. Le résultat pour le dernier bloc (en bas à droite) peut être obtenu de manière symétrique.
Le résultat du second bloc (en haut à droite) est obtenu en utilisant B_{22} dans l'expression de B_{12} :

$$\begin{aligned} B_{12} &= -A_{11}^{-1}A_{12}(A_{22} - A_{12}^T A_{11}^{-1}A_{12})^{-1} \\ &= -A_{11}^{-1}A_{12}B_{22} \\ A_{11}B_{12} &= -A_{12}B_{22} \\ 0 &= A_{11}B_{12} + A_{12}B_{22} \end{aligned}$$

Le résultat pour le troisième bloc (en bas à gauche) est obtenu en utilisant B_{11} dans l'expression de B_{12}^T :

$$\begin{aligned} B_{12}^T &= -A_{22}^{-1}A_{12}^T(A_{11} - A_{12}A_{22}^{-1}A_{12}^T)^{-1} \\ &= -A_{22}^{-1}A_{12}^T B_{11} \\ A_{22}B_{12}^T &= -A_{12}^T B_{11} \\ 0 &= A_{12}^T B_{11} + A_{22}B_{12}^T \end{aligned}$$

B.3. Thèorème 3 : Déterminant d'une matrice symétrique par bloc

Theorem 3: Déterminant d'une matrice symétrique par bloc

Étant donnée une matrice symétrique A telle que :

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \tag{B.6}$$

et

$$A_{21} = A_{12}^T \tag{B.7}$$

Le déterminant de la matrice A peut être exprimé en utilisant les différents blocs de A ainsi :

$$|A| = \begin{vmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{vmatrix} = |A_{22}||A_{11} - A_{12}A_{22}^{-1}A_{12}^T| = |A_{11}||A_{22} - A_{12}^T A_{11} - 1A_{12}| \tag{B.8}$$

Preuve

La matrice A peut être décomposée comme suit :

$$\begin{aligned} A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} &= \begin{bmatrix} A_{11} & 0 \\ A_{12}^T & I \end{bmatrix} \begin{bmatrix} I & A_{11}^{-1}A_{12} \\ 0 & A_{22} - A_{12}^T A_{11}^{-1} A_{12} \end{bmatrix} \\ &= \begin{bmatrix} I & A_{12} \\ 0 & A_{22} \end{bmatrix} \begin{bmatrix} A_{11} - A_{12}A_{22}^{-1}A_{12}^T & 0 \\ A_{22}^{-1}A_{12}^T & I \end{bmatrix} \end{aligned}$$

Alors la preuve du théorème est obtenue en appliquant les propriétés du déterminant qui sont :

$$|AB| = |A||B| \tag{B.9}$$

et

$$\begin{vmatrix} B & 0 \\ C & D \end{vmatrix} = \begin{vmatrix} B & C \\ 0 & D \end{vmatrix} = |B||D| \tag{B.10}$$

B.4. Théorème 4 :

Théorème 4

Pour tous vecteurs u, v et toute matrice symétrique A on a :

$$u^T Au - 2u^T Av + v^T Av = (v-u)^T A(v-u) \tag{B.11}$$

Preuve

$$\begin{aligned} u^T Au - 2u^T Av + v^T Av &= u^T Au - u^T Av - u^T Av + v^T Av \\ &= u^T A(u-v) - (u-v)^T Av \\ &= u^T A(u-v) - v^T A(u-v) \\ &= (u-v)^T A(u-v) \\ &= (v-u)^T A(v-u) \end{aligned}$$

ANNEXE C

L'espace de total variabilité

Cette annexe présente quelques éléments du modèle de variabilité totale sans prétendre décrire de façon complète le processus d'apprentissage du modèle par algorithme EM. Le but est ici de présenter les équations du modèle complet correspondant aux figures 3.6 et 3.7 de la section 3.3.1. Nous nous contenteons ici de présenter la forme Gaussienne et les formules des paramètres du modèle.

Motivations du modèle :

— représenter la distribution d'une séquence de vecteurs dans un espace de dimension réduite → **partager la variable cachée pour toute la séquence d'observations** ;

— représenter le super-vecteur d'un GMM par un unique vecteur de dimension réduite → **partager la variable entre les distributions du GMM** ;

— utiliser la corélation qui existe entre les distributions d'un GMM afin d'obtenir une estimation robuste de ce GMM.

C.1. Un nouveau Factor Analyser

Soit $\mathcal{X}_\rho = \{\mathbf{x}_t\}_{t\in[1,N_\rho]}$ une séquence d'observations multi-dimensionnelles N_ρ extraites d'un enregistrement ρ. Notre première hypothèse consiste à supposer que toutes ces observations ont été générées par un mélange de Gaussiennes à C distributions dont le super-vecteur de moyennes, c.-à-d. le vecteur obtenu en concaténant les vecteurs moyens de chaque disribution du GMM est noté $\mathbf{m}_\rho$. Nous supposons que chaque observation utilisé pour apprendre le modèle peut être alignée avec une distribution du modèle GMM.

Notre seconde hypothèse consiste à penser que $\mathbf{m}_\rho$ appartient à un sous-espace linéaire de dimension réduite et d'origine $\boldsymbol{\mu}$ tel que :

$$\mathbf{m}_\rho = \boldsymbol{\mu} + \Phi \mathbf{h}_\rho \tag{C.1}$$

$\mathbf{h}_\rho$ est une variable latente représentant la session ρ dans le sous-espace considéré. $\mathbf{h}_\rho$ est supposé suivre une distribution de probabilité normale telle que : $\mathbf{h} \sim \mathcal{N}(0, \mathbf{I})$. La matrice portrait Φ de dimensios $dC \times r$, a un rang plein r et est appelée matrice de variabilité totale.

Il est évident à la lecture de l'équation C.1 que le paradigme de variabilité totale est lié au *Factor Analyser* . Cependant, ce modèle présent un certain nombre de différences avec le *Factor Analyser* tel qu'il a été décrit dans les chapitre précédant. Les différences et les hypothèses principales sont décrites ci-dessous.

C.2. Spécificités du modèle de variabilité totale

C.2.1 Expressions de la moyenne et de la variance dans le cas d'un Factor Analyser multi-modal

On note que l'équation C.1 diffère de l'équation *A*.1 pour trois raisons.

— Le super-vecteur $\mathbf{m}_\rho$ n'est jamais observé directement. Les observations ne sont que les vecteurs acoustiques , $\mathbf{x}_t$, que l'on suppose générés par le modèle GMM dont $\mathbf{m}_\rho$ est le super-vecteur.

— La variable cachée $\mathbf{h}_\rho$ dépend de la session ρ et non pas d'une observation unique $\mathbf{x}_t$ comme c'est le cas dans le *Factor Analyser* . Ceci peut être reflété par le fait que la variable cachée $\mathbf{h}_\rho$ est seulement indexée par le numéro de la session ρ dans C.1.

— Il existe une unique variable cachées par session qui est partagée par toutes les distributions du mélange de Gaussiennes.

On note ainsi que le modèle de variabilité totale est différent d'un mélange de *Factor Analyser* car il ne considère aucun poinds entre les distributions qui génèrent le observations. Les poids sont implicitement pris en compte à travers les statistiques d'ordre 0 et 1 qui sont eux calculés en utilisant un modèle du monde GMM.

Tous ces éléments constituent les différences entre le modèle de variabilité totale et une mixture de *Factor Analyser* **?** où avec le modèle proposé par **?**. En effet, ce modèle considère une variable cachée par observation alors que le modèle de variabilité totale considère une variable cachée partagée entr etoutes les distributions et les obervations

d'une même session.

Nous introduisons un certain nombre de notations :

$\boldsymbol{\mu}_c$ vecteur moyen de la c^{ieme} distribution du modèle UBM

Σ_c matrice de covariance de la c^{ieme} distribution du modèle UBM

Σ la matrice diagonale par bloc $dC \times dC$ dont les blocs sont les Σ_c

Φ_c c^{th} bloc de la matrice de variabilité totale qui vérifie $\Phi = \left[\Phi_1^T, \Phi_2^T, ..., \Phi_C^T\right]^T$

$\mathbf{m}_c$ vecteur des moyennes de la c^{ieme} distribution du GMM dépendant de la session.

$N_{\rho,c}$ nombre d'observations composant la session à modéliser et provenant (générées) par la c^{ieme} distribution du modèle GMM pour la session ρ

$\mathbf{N}$ la matrice diagonale par bloc $dC \times dC$ dont les blocs sont $N_{c,\rho}\mathbf{I}$

$S_{X,c,\rho} = \sum_{c=1}^{C} \sum_{t=1}^{N_{c,\rho}} (\mathbf{X}_t - \boldsymbol{\mu}_c)$

$S_X = \left[S_{X,1,\rho}^T, S_{X,2,\rho}^T, ..., S_{X,C,\rho}^T\right]^T$

En considérant l'hypothèse d'indépendance des observations acoustiques, la vraisemblance de R sessions d'enregistrement calculée sur le modèle de variabilité totale est donnée par :

$$\Lambda(\theta) = \prod_{\rho=1}^{R} \int \left(\sum_{c=1}^{C} \sum_{t=1}^{N_{\rho,c}} \mathcal{N}(\mathbf{x}_t | \boldsymbol{\mu}_c + \Phi_c \mathbf{h}_\rho, \Sigma_c) \right) \mathcal{N}(\mathbf{h}_\rho | 0, \mathbf{I}) d\mathbf{h}_\rho \tag{C.2}$$

avec $\theta = (\{\boldsymbol{\mu}_c, \Phi_c, \Sigma_c, \}_{c \in [1,C]})$ les métaparamètres du modèle. On retrouve dans cette formule, les sommes et intégrales qui résultent du partage de la variable cachée pour toutes les distributions et toutes les observations. Ce point a été discuté dans la section 3.3.1 et décrit par les figures 3.6 et 3.7. Il est visible sur l'équation C.2 que la variable cachée, $\mathbf{h}_\rho$, ne dépend ni de la distribution ni des observations mais est unique pour une session donnée. On note que le vecteur moyen $\boldsymbol{\mu}_c$ et la covariance Σ_c sont généralement fixés et pris directement du modèle du monde (UBM).

C.2.2 Algorithme EM pour le modèle de variabilité totale

L'algorithme EM utilisé pour estimer les paramètres du modèle de variabilité totale vise à optimiser les paramètres de la distribution $q(\mathbf{h})$ choisie comme probabilité conditionnée par la variable cachée. Cette fonction est choisie comme suit :

$$q(\mathbf{h}) = Pr(\mathbf{h} | \mathcal{X}, \theta)$$

où $\mathcal{X}$ est l'union de tous les vecteurs acoustiques de toutes les sessions utilisées pour l'apprentissage du modèle. Comme dans le cas du *Factor Analyser* simple, on a besoin de l'expression de $Pr(\mathbf{h}|\mathcal{X})$ pour déterminer son vecteur moyen et sa matrice de covariance. Alors que pour le *Factor Analyser* , il est possible de donner une expression exacte de $Pr(\mathcal{X}, \mathbf{h})$ (cf. équation A.21), c'est plus difficile dans le cas de la variabilité totale.

On propose alors de tirer parti des propriétés des distributions Gaussiennes afin d'identifier la moyenne et la covariance de la distribution a posteriori $Pr(\mathbf{h}|\mathcal{X})$. Ce sera fait en deux étapes.

Produit de deux distributions Gaussiennes

Montrons d'abord que le produit de $P(\mathcal{X}|\mathbf{h})$ et $Pr(\mathbf{h})$ est proportionnel à une troisième distribution Gaussienne en $\mathbf{h}$ et que le coefficient de proportionnalité s'annule exactement avec $Pr(\mathcal{X})$.

Preuve

De manière générale, $Pr(\mathcal{X}, \mathbf{h})$ peut être formulé sous le forme suivante :

$$Pr(\mathcal{X}, \mathbf{h}) = Pr(\mathcal{X}|\mathbf{h})Pr(\mathbf{h}) = Kf(\mathcal{X})\mathcal{N}(\mathbf{h}|\boldsymbol{\mu}(\mathcal{X}), \Sigma(\mathcal{X})) \tag{C.3}$$

où K ne dépend que des métaparamètres, f est une fonction de $\mathcal{X}$ seulement, et $\boldsymbol{\mu}(\mathcal{X})$ et $\Sigma(\mathcal{X})$ sont les moyennes et covariance ne dépendant que de $\mathcal{X}$. Une preuve de ceci est donnée dans le cas du *Factor Analyser* dans le chapitre A.

On écrit alors :

$$\begin{aligned}
Pr(\mathbf{h}|\mathcal{X}) &= \frac{Pr(\mathcal{X}|\mathbf{h})Pr(\mathbf{h})}{Pr(\mathcal{X})} \\
&= \frac{Pr(\mathcal{X}|\mathbf{h})Pr(\mathbf{h})}{\int Pr(\mathcal{X}|\mathbf{h}')Pr(\mathbf{h}')d\mathbf{h}'} \\
&= \frac{Kf(\mathcal{X})\mathcal{N}(\mathbf{h}|\boldsymbol{\mu}(\mathcal{X}), \Sigma(\mathcal{X}))}{\int Kf(\mathcal{X})\mathcal{N}(\mathbf{h}'|\boldsymbol{\mu}(\mathcal{X}), \Sigma(\mathcal{X}))d\mathbf{h}'} \\
&= \frac{\mathcal{N}(\mathbf{h}|\boldsymbol{\mu}(\mathcal{X}), \Sigma(\mathcal{X}))}{\int \mathcal{N}(\mathbf{h}'|\boldsymbol{\mu}(\mathcal{X}), \Sigma(\mathcal{X}))d\mathbf{h}'} \\
&= \frac{\mathcal{N}(\mathbf{h}|\boldsymbol{\mu}(\mathcal{X}), \Sigma(\mathcal{X}))}{1} \\
&= \mathcal{N}(\mathbf{h}|\boldsymbol{\mu}(\mathcal{X}), \Sigma(\mathcal{X}))
\end{aligned} \tag{C.4}$$

La première et la dernière ligne nous donne le résultat attendu.

Ré écire $Pr(\mathcal{X}, \mathbf{h})$

Nous ré écrivons maintenant $Pr(\mathcal{X}, \mathbf{h})$ afin d'identifier la moyenne, $\boldsymbol{\mu}(\mathcal{X})$, et la variacne, $\Sigma(\mathcal{X})$ de $Pr(\mathbf{h}|\mathcal{X})$.

On commence par quelques manipulations de $Pr(\mathcal{X}, \mathbf{h})$ pour une unique session.

$$Pr(\mathcal{X}, \mathbf{h}) = \prod_{c=1}^{C} \left(\frac{1}{(2\Pi)^{\frac{d}{2}} |\Sigma_c|^{\frac{1}{2}}} \right)^{N_c} \exp\left[-\frac{1}{2} \sum_{c=1}^{C} \sum_{t=1}^{N_c} (X_t - \mathbf{m}_c)^T \Sigma^{-1} (X_t - \mathbf{m}_c) \right] \quad \text{(C.5)}$$

On note alors :

$$\begin{cases} A = \prod_{c=1}^{C} \left(\frac{1}{(2\Pi)^{\frac{d}{2}} |\Sigma_c|^{\frac{1}{2}}} \right)^{N_c} & \text{(C.6)} \\ B = \sum_{c=1}^{C} \sum_{t=1}^{N_c} (X_t - \mathbf{m}_c)^T \Sigma^{-1} (X_t - \mathbf{m}_c) & \text{(C.7)} \end{cases}$$

où A ne dépend que des métaparamètres du modèle. On peut alors écrire :

$$\begin{aligned} B &= \sum_{c=1}^{C} \sum_{t=1}^{N_c} (\mathbf{X}_t - \mathbf{m}_c)^T \Sigma^{-1} (\mathbf{X}_t - \mathbf{m}_c) \\ &= \sum_{c=1}^{C} \sum_{t=1}^{N_c} (\mathbf{X}_t - \boldsymbol{\mu}_c - \Phi_c \mathbf{h})^T \Sigma^{-1} (\mathbf{X}_t - \boldsymbol{\mu}_c - \Phi_c \mathbf{h}) \\ &= \sum_{c=1}^{C} \sum_{t=1}^{N_c} (\mathbf{X}_t - \boldsymbol{\mu}_c)^T \Sigma^{-1} (\mathbf{X}_t - \boldsymbol{\mu}_c) - 2 \sum_{c=1}^{C} \sum_{t=1}^{N_c} (\Phi_c \mathbf{h})^T \Sigma^{-1} (\mathbf{X}_t - \boldsymbol{\mu}_c) + \sum_{c=1}^{C} \sum_{t=1}^{N_c} (\Phi_c \mathbf{h})^T \Sigma^{-1} (\Phi_c \mathbf{h}) \end{aligned}$$

Et pour simplifier on notera :

$$F(\mathcal{X}) = \exp(-\frac{1}{2} \sum_{c=1}^{C} \sum_{t=1}^{N_c} (\mathbf{X}_t - \boldsymbol{\mu}_c)^T \Sigma^{-1} (\mathbf{X}_t - \boldsymbol{\mu}_c)) \quad \text{(C.8)}$$

Ainsi on obtient :

$$\begin{aligned} Pr(\mathcal{X}, \mathbf{h}) Pr(\mathbf{h}) &= A F(\mathcal{X}) \exp\left[\mathbf{h}^T \Phi^T \Sigma^{-1} S_{\mathrm{X}} - \frac{1}{2} \mathbf{h}^T \Phi^T \mathbf{N} \Sigma^{-1} \Phi \mathbf{h} \right] \frac{1}{(2\Pi)^{\frac{r}{2}}} \exp\left[-\frac{1}{2} \mathbf{h}^T \mathbf{h} \right] \\ &= \frac{1}{(2\Pi)^{\frac{r}{2}}} A F(\mathcal{X}) \exp\left[\mathbf{h}^T \Phi^T \Sigma^{-1} S_{\mathrm{X}} - \frac{1}{2} \mathbf{h}^T \Phi^T \mathbf{N} \Sigma^{-1} \Phi \mathbf{h} - \frac{1}{2} \mathbf{h}^T \mathbf{h} \right] \end{aligned}$$

Et en considérant désormais :

$$\Lambda = (\mathbf{I} + \Phi \mathbf{N} \Sigma^{-1} \Phi) \quad \text{(C.9)}$$

$$\mathbf{a} = (\mathbf{I} + \Phi \mathbf{N} \Sigma^{-1} \Phi)^{-1} \Phi^T \Sigma^{-1} S_X \quad \text{(C.10)}$$

On peut alors écrire :

$$Pr(\mathcal{X}, \mathbf{h})Pr(\mathbf{h}) = \frac{1}{(2\Pi)^{\frac{r}{2}}} AF(\mathcal{X}) \exp\left[-\frac{1}{2}(\mathbf{h}-\mathbf{a})^T \Lambda (\mathbf{h}-\mathbf{a}) \right] \tag{C.11}$$

L'expression de $Pr(\mathcal{X}, \mathbf{h})Pr(\mathbf{h})$ et le résultat précédent (cf. équation C.4) nous permettent d'identifier $\boldsymbol{\mu}_{\mathrm{h}|\mathcal{X}}$ et $\Sigma_{\mathrm{h}|\mathcal{X}}$ dans l'équationC.11 et d'écrire que la distribution de probabilité conditionelle de $\mathbf{h}$ étant donné $\mathcal{X}$ est une Gaussienne de moyenne et de covariance données par :

$$\begin{cases} \boldsymbol{\mu}_{\mathrm{h}|\mathcal{X}} = \mathbf{a} = (\mathbf{I} + \Phi \mathbf{N} \Sigma^{-1} \Phi)^{-1} \Phi^T \Sigma^{-1} S_X & \text{(C.12)} \\ \Sigma_{\mathrm{h}|\mathcal{X}} = \Lambda^{-1} = (\mathbf{I} + \Phi \mathbf{N} \Sigma^{-1} \Phi)^{-1} & \text{(C.13)} \end{cases}$$

Ces expressions permettront par la suite de dérouler un algorithme EM permettant d'estimer ces paramètres.

Table des figures

Liste des tableaux

Bibliographie

(Alam et al., 2016) M. J. Alam, P. Kenny, et V. Gupta, 2016. Tandem features for text-dependent speaker verification on the reddots corpus. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 420–424.

(Aronowitz, 2012) H. Aronowitz, 2012. Text-Dependent Speaker Verification Using a Small Development Set. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 312–316.

(Besacier et al., 2000) L. Besacier, J.-F. Bonastre, et C. Fredouille, 2000. Localization and selection of speaker-specific information with statistical modeling. *Speech Communication 31*(2-3), 89–106.

(Bimbot et al., 2004) F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meigner, T. Merlin, J. Ortega-Garcia, D. Petrovska-Delacretaz, et D. A. Reynolds, 2004. A tutorial on text-independent speaker verification. *EURASIP Journal on Applied Signal Processing 4*, 430–451.

(Bimbot et al., 1995) F. Bimbot, I. Magrin-Chagnolleau, et L. Mathan, 1995. Second-order statistical measures for text-independent speaker identification. *Speech Communication 17*(1-2), 177–192.

(Bonastre et al., 2008) J.-F. Bonastre, N. Scheffer, D. Matrouf, C. Fredouille, A. Larcher, A. Preti, G. Pouchoulin, N. Evans, B. Fauve, et J. S. Mason, 2008. ALIZE/SpkDet : a state-of-the-art open source software for speaker recognition. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 20–27. http ://mistral.univ-avignon.fr/.

(Bousquet et al., 2012) P.-M. Bousquet, A. Larcher, D. Matrouf, J.-F. Bonastre, et O. Plchot, 2012. Variance-Spectra based Normalization for I-vector Standard and Probabilistic Linear Discriminant Analysis. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 1–8.

(Bousquet et al., 2011) P.-M. Bousquet, D. Matrouf, et J.-F. Bonastre, 2011. Intersession compensation and scoring methods in the i-vectors space for speaker recognition. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 485–488.

(Brümmer et al., 2007) N. Brümmer, L. Burget, J. H. Cernocky, O. Glembek, F. Grezl, M. Karafiat, D. A. V. Leeuwen, P. Matejka, P. Schwarz, et A. Strasheim, 2007. Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006. *IEEE Transactions on Audio, Speech, and Language Processing* *15*(7), 2072–2084.

(Campbell et al., 2006) W. Campbell, D. Sturim, et D. Reynolds, 2006. Support Vector Machines Using GMM Supervectors for Speaker Verification. *IEEE Signal Processing Letters* *13*(5), 308–311.

(Campbell et al., 2007) W. M. Campbell, F. Richardson, et D. A. Reynolds, 2007. Language recognition with word lattices and support vector machines. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, Volume 4, 989–929. IEEE.

(Campbell et al., 2008) W. M. Campbell, D. E. Sturim, P. A. Torres-Carrasquillo, et D. A. Reynolds, 2008. A comparison of subspace feature-domain methods for language recognition. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 309–312.

(Cumani et Laface, 2013a) S. Cumani et P. Laface, 2013a. Memory and Computation Trade-Offs for Efficient I-Vector Extraction . *IEEE Transactions on Audio, Speech, and Language Processing* *21*(5), 934–944.

(Cumani et Laface, 2013b) S. Cumani et P. Laface, 2013b. Factorized Sub-space Estimation for Fast and Memory Effective I-vector Extraction. *IEEE Transactions on Audio, Speech, and Language Processing*, 248–259.

(Dehak et al., 2009) N. Dehak, R. Dehak, P. Kenny, N. Brummer, P. Ouellet, et P. Dumouchel, 2009. Support Vector Machines versus Fast Scoring in the Low-Dimensional Total Variability Space for Speaker Verification. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 1559–1562.

(Dehak et al., 2011a) N. Dehak, P. Kenny, R. Dehak, P. Dumouchel, et P. Ouellet, 2011a. Front-End Factor Analysis for Speaker Verification. *IEEE Transactions on Audio, Speech, and Language Processing* *19*(4), 788–798.

(Dehak et al., 2011b) N. Dehak, P. A. Torres-Carrasquillo, D. A. Reynolds, et R. Dehak, 2011b. Language Recognition via i-vectors and Dimensionality Reduction. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 857–860.

(Delacourt, 2000) P. Delacourt, 2000. La segmentation et le regroupement par locuteurs pour l'indexation de documents audio.

(Desnous et al., 2018) F. Desnous, A. Larcher, et S. Meignier, 2018. Étude de l'impact de différents systèmes de détection de la parole sur des tâches traitement automatique de la parole. Dans les actes de *Journees d'etudes sur le parole (JEP)*, 550–558.

(Doukhan et al., 2018) D. Doukhan, J. Carrive, F. Vallet, A. Larcher, S. Meignier, et F. Le Mans, 2018. An open-source speaker gender detection framework for monitoring gender equality. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 5214–5218.

(Fauve, 2009) B. Fauve, 2009. *Tackling Variabilities in Speaker Verification with a Focus on Short Durations*. Thèse de Doctorat, School of Engineering Swansea University.

(Fauve et al., 2007a) B. Fauve, N. Evans, N. Pearson, J.-F. Bonastre, et J. S. Mason, 2007a. Influence of task duration in text-independent speaker verification. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 794–797.

(Fauve et al., 2007b) B. Fauve, D. Matrouf, N. Scheffer, J. Bonastre, et J. Mason, 2007b. State-of-the-art performance in text-independent speaker verification through open-source software. *IEEE Transactions on Audio, Speech, and Language Processing* *15*(7), 1960–1968.

(Ferrer et al., 2003) L. Ferrer, H. B. andVenkata R.R. Gadde, S. Kajarekar, E. Shriberg, K. Sonmez, A. Stolcke, et A. Venkataraman, 2003. Modeling duration patterns for speaker recognition. Dans les actes de *European Conference on Speech Communication and Technology (Eurospeech)*, 2017–2020.

(Fredouille, 2000) C. Fredouille, 2000. *Approche Statistique pour la Reconnaissance Automatique du Locuteur : Informations Dynamiques et Normalisation Bayesienne des Vraisemblances*. Thèse de Doctorat.

(Furui, 1981) S. Furui, 1981. Cepstral analysis technique for automatic speaker verification. *IEEE Transactions on Acoustics, Speech, and Signal Processing [see also IEEE Transactions on Signal Processing]* *29*(2), 254–272.

(Galibert et Kahn, 2013) O. Galibert et J. Kahn, 2013. The first official repere evaluation. Dans les actes de *First Workshop on Speech, Language and Audio in Multimedia*.

(Galibert et al., 2014) O. Galibert, J. Leixa, G. Adda, K. Choukri, et G. Gravier, 2014. The etape speech processing evaluation. Dans les actes de *Language Resources and Evaluation Conference (LREC)*, 3995–3999.

(Garcia-Romero et Espy-Wilson, 2011) D. Garcia-Romero et C. Y. Espy-Wilson, 2011. Analysis of i-vector length normalization in speaker recognition systems. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 249–252.

(Glembeck et al., 2011) O. Glembeck, L. Burget, P. Matejka, M. Karafiat, et P. Kenny, 2011. Simplification and optimization of I-Vector extraction. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 4516–4519.

(Hasan et al., 2013) T. Hasan, R. Saeidi, J. H. L. Hansen, et D. A. van Leeuwen, 2013. Duration Mismatch Compensation for I-Vector Based Speaker Recognition Systems. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 7663–7667.

(Hautamaki et al., 2012) V. Hautamaki, K. A. Lee, A. Larcher, T. Kinnunen, B. Ma, , et H. Li, 2012. Variational Bayes logistic regression as regularized fusion for NIST SRE 2010. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 268–274.

(Hautamaki et al., 2013) V. Hautamaki, K. A. Lee, D. van Leeuwen, R. Saeidi, A. Larcher, T. Kinnunen, T. Hasan, S. O. Sadjadi, G. Liu, H. Boril, J. H. L. Hansen, et B. Fauve, 2013. Automatic Regularization of Cross-Entropy Cost for Speaker Recognition. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 1609–1613.

(Hautomaki et al., 2011) V. Hautomaki, K. A. Lee, A. Larcher, T. Kinnunen, B. Ma, et H. Li, 2011. Experiments with large scale regularized fusion on NIST SRE 2010. Dans les actes de *NIST-SRE Analysis Workshop*.

(Hébert, 2008) M. Hébert, 2008. *Text-dependent speaker recognition*. Springer-Verlag, Heidelberg.

(Heck et Genoud, 2001) L. Heck et D. Genoud, 2001. Integrating Speaker and Speech Recognizers : Automatic Identity Claim Capture for Speaker Verification. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 249–254.

(Heigold et al., 2016) G. Heigold, I. Moreno, S. Bengio, et N. Shazeer, 2016. End-to-end text-dependent speaker verification. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 5115–5119. IEEE.

(Hermansky et Sharma, 1999) H. Hermansky et S. Sharma, 1999. Temporal patterns (TRAPs) in ASR of noisy speech. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, Volume 1, 289–292.

(Hernández-Lobato et Adams, 2015) J. M. Hernández-Lobato et R. Adams, 2015. Probabilistic backpropagation for scalable learning of bayesian neural networks. Dans les actes de *International Conference on Machine Learning*, 1861–1869.

(Hinton et al., 2012) G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al., 2012. Deep neural networks for acoustic modeling in speech recognition : The shared views of four research groups. *IEEE Signal processing magazine* *29*(6), 82–97.

(Hinton et al., 2006) G. E. Hinton, S. Osindero, et Y.-W. Teh, 2006. A fast learning algorithm for deep belief nets. *Neural computation* *18*(7), 1527–1554.

(Hsu et Lin, 2002) C.-W. Hsu et C.-J. Lin, 2002. A comparison of methods for multiclass support vector machines. *IEEE Transactions on Neural Networks,* *13*(2), 415–425.

(Huang et al., 1990) X. D. Huang, Y. Ariki, et M. A. Jack, 1990. *Hidden Markov models for speech recognition*. Edinburgh university press Edinburgh.

(Jiang et al., 2012) Y. Jiang, K. A. Lee, Z. Tang, B. Ma, A. Larcher, et H. Li, 2012. PLDA Modeling in I-vector and Supervector Space for Speaker Verification. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 1680–1683.

(Jolliffe, 2002) I. Jolliffe, 2002. Principal component analysis. *Encyclopedia of Statistics in Behavioral Science*.

(Katagiri et al., 1998) S. Katagiri, B.-H. Juang, et C.-H. Lee, 1998. Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method. *Proceedings of the IEEE* *86*(11), 2345–2373.

(Kenny, 2010) P. Kenny, 2010. Bayesian speaker verification with heavy-tailed priors. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*.

(Kenny et al., 2005a) P. Kenny, G. Boulianne, et P. Dumouchel, 2005a. Eigenvoice modeling with sparse training data. *IEEE Transactions on Speech and Audio Processing* *13*(3), 345–354.

(Kenny et al., 2005b) P. Kenny, G. Boulianne, P. Ouellet, et P. Dumouchel, 2005b. Factor analysis simplified. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, Volume 1, 637–640.

(Kenny et al., 2007a) P. Kenny, G. Boulianne, P. Ouellet, et P. Dumouchel, 2007a. Joint factor analysis versus eigenchannels in speaker recognition. *IEEE Transactions on Audio, Speech, and Language Processing 15*(4), 1435–1447.

(Kenny et al., 2007b) P. Kenny, G. Boulianne, P. Ouellet, et P. Dumouchel, 2007b. Speaker and session variability in GMM-based speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing 15*, 1448–1460.

(Kenny et Dumouchel, 2004) P. Kenny et P. Dumouchel, 2004. Disentangling speaker and channel effects in speaker verification. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 37–40.

(Kenny et al., 2013) P. Kenny, T. Stafylakis, P. Ouellet, J. Alam, et P. Dumouchel, 2013. PLDA for Speaker Verification with Utterances of Arbitrary Duration. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 7649–7653.

(Kinnunen et Li, 2010) T. Kinnunen et H. Li, 2010. An overview of text-independent speaker recognition : From features to supervectors. *Speech Communication 52*(1), 12–40.

(Kuhn et al., 1998) R. Kuhn, P. Nguyen, J.-C. Junqua, L. Goldwasser, N. Niedzielski, S. Fincke, K. Field, et M. Contolini, 1998. Eigenvoices for speaker adaptation. Dans les actes de *Proceedings International Conference on Spoken Language Processing, ICSLP*, Sydney (Australia), 1771–1774.

(Larcher, 2009) A. Larcher, 2009. *Modèles acoustiques à structure temporelle renforcée pour la vérification du locuteur embarquée.* Thèse de Doctorat.

(Larcher et al., 2008a) A. Larcher, J.-F. Bonastre, et J. S. Mason, 2008a. Utilisation de la structure de mots de passe personnalises pour la reconnaissance de locuteurs embarquee. Dans les actes de *Journees d'etudes sur le parole (JEP)*, Avignon (France), 1–8.

(Larcher et al., 2010) A. Larcher, J.-F. Bonastre, et J. S. Mason, 2010. Constrained Viterbi decoding for embedded user-customised password speaker recognition. Dans les actes de *ACM Symposium On Applied Computing*, 1501–1502.

(Larcher et al., 2013) A. Larcher, J.-F. Bonastre, et J. S. Mason, 2013. Reinforced temporal structure of acoustic models for speaker recognition. *Digital Signal Processing 23*(6), 1910–1917.

(Larcher et al., 2008b) A. Larcher, J.-F. Bonastre, et J. S. D. Mason, 2008b. From gmm to hmm for embedded password-based speaker recognition. Dans les actes de *European Signal and Image Processing Conference (EUSIPCO)*, Lausanne (Switzerland).

(Larcher et al., 2008c) A. Larcher, J.-F. Bonastre, et J. S. D. Mason, 2008c. Reinforced temporal structure information for embedded utterance-based speaker recognition. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 371–374.

(Larcher et al., 2008d) A. Larcher, J.-F. Bonastre, et J. S. D. Mason, 2008d. Short utterance-based video aided speaker recognition. Dans les actes de *IEEE International workshop on Multimedia Signal Processing*, 897–901.

(Larcher et al., 2012a) A. Larcher, P.-M. Bousquet, K. A. Lee, D. Matrouf, H. Li, et J.-F. Bonastre, 2012a. I-vectors in the context of phonetically-constrained short utterances for speaker verification. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 4773–4776.

(Larcher et al., 2012b) A. Larcher, P.-M. Bousquet, D. Matrouf, et J.-F. Bonastre, 2012b. Analyse en Composante Principale pour l'extraction des i-vecteurs en vérification du locuteur. Dans les actes de *Journees d'etudes sur le parole (JEP)*.

(Larcher et al., 2013a) A. Larcher, K. A. Lee, M. Bin, et T. N. T. H. Helen, 2013a. Dual scoring for text-dependent speaker verification.
Chinese

(Larcher et al., 2013b) A. Larcher, K. A. Lee, B. Ma, et T. N. T. H. Helen, 2013b. Method and System for Dual Scoring for Text-dependent Speaker Verification.

(Larcher et al., 2012) A. Larcher, K. A. Lee, B. Ma, et H. Li, 2012. The RSR2015 : Database for Text-Dependent Speaker Verification using Multiple Pass-Phrases. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 1580–1583.

(Larcher et al., 2013) A. Larcher, K. A. Lee, B. Ma, et H. Li, 2013. Phonetically-Constrained PLDA Modeling for Text-Dependent Speaker Verification with Multiple Short Utterances. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 7673–7677.

(Larcher et al., 2014a) A. Larcher, K. A. Lee, B. Ma, et H. Li, 2014a. Imposture classification for text-dependent speaker verification. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 739–743.

(Larcher et al., 2014b) A. Larcher, K.-A. Lee, B. Ma, et H. Li, 2014b. Modelling the Alternative Hypothesis for Text-Dependent Speaker Verification. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 734–738.

(Larcher et al., 2014c) A. Larcher, K. A. Lee, B. Ma, et H. Li, 2014c. Text-dependent Speaker Verification : Classifiers, Databases and RSR2015. *Speech Communication 60*, 56–77.

(Larcher et al., 2014d) A. Larcher, K. A. Lee, P. L. S. Martínez, T. H. Nguyen, B. Ma, et H. Li, 2014d. Extended RSR2015 for text-dependent speaker verification over VHF channel. Dans les actes de *Fifteenth Annual Conference of the International Speech Communication Association*, 1322–1326.

(Larcher et al., 2010a) A. Larcher, C. Lévy, D. Matrouf, et J.-F. Bonastre, 2010a. Decoupling session variability modelling and speaker characterisation. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 2314–2317.

(Larcher et al., 2010b) A. Larcher, C. Lévy, D. Matrouf, et J.-F. Bonastre, 2010b. Reconnaissance automatique du locuteur embarqu'ee dans un téléphone portable. Dans les actes de *Proceedings of journees d'études sur le parole, JEP*, Mons (Belgium), 1–8.

(Lawson et al., 2011) A. Lawson, P. Vabishchevich, M. Huggins, Ardis, B. Battles, et A. Stauffer, 2011. Survey and evaluation of acoustic features for speaker recognition. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 5444–5447.

(Le Lan, 2017) G. Le Lan, 2017. *Analyse en locuteurs de collections de documents multimedia*. Phd thesis.

(Le Lan et al., 2016) G. Le Lan, D. Charlet, A. Larcher, et S. Meignier, 2016. Iterative plda adaptation for speaker diarization. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, Volume 2016, 2175–2179.

(Le Lan et al., 2017) G. Le Lan, D. Charlet, A. Larcher, et S. Meignier, 2017. A triplet ranking-based neural network for speaker diarization and linking. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 3572–3576.

(Le Lan et al., 2018) G. Le Lan, D. Charlet, A. Larcher, et S. Meignier, 2018. An adaptive method for cross-recording speaker diarization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 1821–1832.

(Le Lan et al., 2016a) G. Le Lan, S. Meignier, D. Charlet, et A. Larcher, 2016a. Autoapprentissage pour le regroupement en locuteurs : premières investigations. Dans les actes de *Journees d'etudes sur le parole (JEP)*, 80–88. AFCP.

(Le Lan et al., 2016b) G. Le Lan, S. Meignier, D. Charlet, et A. Larcher, 2016b. First investigations on self trained speaker diarization. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 152–157.

(Lee, 2008) C.-H. Lee, 2008. *Principles of Spoken Language Recognition*, Chapter 39, 785–796. Springer.

(Lee et al., 2016) K. Lee, H. Li, L. Deng, V. Hautamäki, W. Rao, X. Xiao, A. Larcher, H. Sun, T. Nguyen, G. Wang, et al., 2016. The 2015 nist language recognition evaluation : the shared view of i2r, fantastic4 and singams. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, Volume 2016, 3211–3215.

(Lee et al., 2017) K.-A. Lee, V. Hautamäki, T. Kinnunen, A. Larcher, C. Zhang, A. Nautsch, T. Stafylakis, G. Liu, M. Rouvier, W. Rao, et al., 2017. The i4u mega fusion and collaboration for NIST speaker recognition evaluation 2016. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 1328–1332.

(Lee et al., 2011) K. A. Lee, A. Larcher, H. Thai, B. Ma, et H. Li, 2011. Joint Application of Speech and Speaker Recognition for Automation and Security in Smart Home. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 3317–3318.

(Lee et al., 2015) K. A. Lee, A. Larcher, G. Wang, P. Kenny, N. Brümmer, D. v. Leeuwen, H. Aronowitz, M. Kockmann, C. Vaquero, B. Ma, et al., 2015. The reddots data collection for speaker recognition. Dans les actes de *Sixteenth Annual Conference of the International Speech Communication Association*, 2996–3000.

(Lee et al., 2013) K. A. Lee, A. Larcher, C. H. You, B. Ma, et H. Li, 2013. Multi-session PLDA Scoring of I-vector for Partially Open-Set Speaker Detection. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 3651–3655.

(Lee et al., 2011) K. A. Lee, C. H. You, V. Hautomaki, A. Larcher, et H. Li, 2011. Spoken Language Recognition in the Latent Topic Simplex. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 2933–2936.

(Leeuwen et Brümmer, 2006) D. A. V. Leeuwen et N. Brümmer, 2006. Channel-dependent GMM and Multi-class Logistic Regression models for language recognition. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 1–8.

(Lei et al., 2014) Y. Lei, N. Scheffer, L. Ferrer, et M. McLaren, 2014. A novel scheme for speaker recognition using a phonetically-aware deep neural network. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 1695–1699.

(Li et al., 2012) H. Li, B. Ma, K. Lee, C. You, H. Sun, et A. Larcher, 2012. Iir system description for the nist 2012 speaker recognition evaluation. Dans les actes de *NIST SRE'12 Workshop*.

(Li et al., 2013) H. Li, B. Ma, et K. A. Lee, 2013. Spoken language recognition : from fundamentals to practice. *Proceedings of the IEEE*.

(Li et al., 2006) J. Li, S. Yaman, C.-H. Lee, B. ma, R. Tong, D. Zhu, et H. Li, 2006. Language Recognition Based on Score Discrimination Feature Vectors and Discriminative Classifier Fusion. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 1–5.

(Liu et al., 2018) Y. Liu, L. He, J. Liu, et M. Johnson, 2018. Speaker Embedding Extraction with Phonetic Information. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 2247–2251.

(Lorenzo-Trueba et al., 2018) J. Lorenzo-Trueba, J. Yamagishi, T. Toda, D. Saito, F. Villavicencio, T. Kinnunen, et Z. Ling, 2018. The voice conversion challenge 2018 : Promoting development of parallel and nonparallel methods. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 195–202.

(Lozano-Diez et al., 2016) A. Lozano-Diez, A. Silnova, P. Matejka, O. Glembek, O. Plchot, J. Pešán, L. Burget, et J. Gonzalez-Rodriguez, 2016. Analysis and optimization of bottleneck features for speaker recognition. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, Volume 2016, 352–357.

(Ma et al., 2018) J. Ma, V. Sethu, E. Ambikairajah, et K. A. Lee, 2018. Speaker-phonetic vector estimation for short duration speaker verification. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 5264–5268. IEEE.

(Mandasari et al., 2013) M. I. Mandasari, R. Saeidi, M. McLaren, et D. A. van Leeuwen, 2013. Quality Measure Functions for Calibration of Speaker Recognition Systems in Various Duration Conditions. *IEEE Transactions on Audio, Speech, and Language Processing*, 2425–2438.

(Martin et Greenberg, 2009) A. F. Martin et C. S. Greenberg, 2009. NIST 2008 speaker recognition evaluation : performance across telephone and room microphone channels. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 2579–2582.

(Martin et Greenberg, 2010) A. F. Martin et C. S. Greenberg, 2010. The NIST 2010 speaker recognition evaluation. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 2726–2729.

(Martinez et al., 2014) P. L. S. Martinez, B. Fauve, A. Larcher, et J. S. Mason, 2014. Speaker verification performance with constrained durations. Dans les actes de *International Workshop on Biometrics and Forensics*, 1–6.

(Matejka et al., 2011) P. Matejka, O. Glembeck, F. Castaldo, M. Alam, O. Plchot, P. Kenny, L. Burget, et J. Cernocky, 2011. Full-covariance UBM and heavy-tailed PLDA in I-Vector speaker verification. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 4828–4831.

(Matrouf et al., 2007) D. Matrouf, N. Scheffer, B. Fauve, et J.-F. Bonastre, 2007. A straightforward and efficient implementation of the factor analysis model for speaker verification. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 1242–1245.

(McCree et al., 2008) A. McCree, F. Richardson, E. Singer, et D. Reynolds, 2008. Beyond frame independence : Parametric modelling of time duration in speaker and language recognition. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 767–770.

(McLaren et al., 2018) M. McLaren, D. Castan, M. K. Nandwana, L. Ferrer, et E. Yılmaz, 2018. How to train your speaker embeddings extractor. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 327–334.

(McLaren et al., 2016) M. McLaren, L. Ferrer, et A. Lawson, 2016. Exploring the role of phonetic bottleneck features for speaker and language recognition. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 5575–5579. IEEE.

(McLaren et al., 2015) M. McLaren, Y. Lei, et L. Ferrer, 2015. Advances in deep neural network approaches to speaker recognition. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 4814–4818. IEEE.

(McLaren et al., 2014) M. McLaren, Y. Lei, N. Scheffer, et L. Ferrer, 2014. Application of Convolutional Neural Networks to Speaker Recognition in Noisy Conditions. Dans les

actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 686–690.

(Meignier, 2015) S. Meignier, 2015. Détection et identification des locuteurs des émissions radiophoniques et télévisées.

(Moez et al., 2016) A. Moez, B. Jean-François, B. K. Waad, R. Solange, et K. Juliette, 2016. Phonetic content impact on forensic voice comparison. Dans les actes de *Spoken Language Technology Workshop (SLT), 2016 IEEE*, 210–217. IEEE.

(Prince, 2012) S. J. Prince, 2012. *Computer vision : models, learning, and inference*. Cambridge University Press.

(Prince et Elder, 2007) S. J. Prince et J. H. Elder, 2007. Probabilistic linear discriminant analysis for inferences about identity. Dans les actes de *International Conference on Computer Vision*, 1–8. IEEE.

(Rabiner, 1989) L. R. Rabiner, 1989. A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE 77*(2), 257–286.

(Rabiner et Juang, 1993) L. R. Rabiner et B.-H. Juang, 1993. *Fundamentals of speech recognition*, Volume 14. PTR Prentice Hall Englewood Cliffs.

(Reynolds et Heck, 1991) D. A. Reynolds et L. Heck, 1991. Integration of Speaker and Speech Recognition Systems. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 869–872.

(Reynolds et al., 2000) D. A. Reynolds, T. F. Quatieri, et R. B. Dunn, 2000. Speaker Verification Using Adapted Gaussian Mixture Models. *Digital Signal Processing 10*, 19–41.

(Reynolds et Rose, 1995) D. A. Reynolds et R. C. Rose, 1995. Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. *IEEE Transactions on Acoustics, Speech and Signal Processing 3*(1), 72–83.

(Saeidi et al., 2013) R. Saeidi, K. A. Lee, T. Kinnunen, T. Hasan, B. Fauve, P.-M. Bousquet, E. Khoury, P. L. S. Martinez, J. M. K. Kua, C. H. You, H. Sun, A. Larcher, P. Rajan, V. Hautamaki, C. Hanilci, B. Braithwaite, R. Gonzales-Hautamaki, S. O. Sadjadi, G. Liu, H. Boril, N. Shokouhi, D. Matrouf, L. E. Shafey, P. Mowlaee, J. Epps, T. Thiruvaran, D. A. van Leeuwen, B. Ma, H. Li, J. H. L. Hansen, J.-F. Bonastre, S. Marcel, J. S. Mason, et E. Ambikairajah, 2013. I4U submission to NIST SRE 2012 : A large-scale collaborative effort for noise-robust speaker verification. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 1986–1990.

(Sarkar et Umesh, 2010) A. K. Sarkar et S. Umesh, 2010. Investigation of Speaker-Clustered UBMs based on Vocal Tract Lengths and MLLR matrices for Speaker Verification. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 13–20.

(Scheffer, 2006) N. Scheffer, 2006. *Structuration de l'espace acoustique par le modèles générique pour la vérification du locuteurs*. Thèse de Doctorat, Université d'Avignon et des Pays de Vaucluse.

(Senior et Lopez-Moreno, 2014) A. Senior et I. Lopez-Moreno, 2014. Improving dnn speaker independence with i-vector inputs. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 225–229. IEEE.

(Snyder et al., 2015) D. Snyder, D. Garcia-Romero, et D. Povey, 2015. Time delay deep neural network-based universal background models for speaker recognition. Dans les actes de *IEEE Workshop on Automatic Speech Recognition & Understanding*, 92–97. IEEE.

(Snyder et al., 2017) D. Snyder, D. Garcia-Romero, D. Povey, et S. Khudanpur, 2017. Deep neural network embeddings for text-independent speaker verification. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 999–1003.

(Snyder et al., 2016) D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, et S. Khudanpur, 2016. Deep neural network-based speaker embeddings for end-to-end speaker verification. Dans les actes de *Spoken Language Technology Workshop (SLT), 2016 IEEE*, 165–170. IEEE.

(Stafylakis et al., 2013) T. Stafylakis, P. Kenny, P. Ouellet, J. Perez, M. Kockmann, et P. Dumouchel, 2013. Text-dependent speaker recognition using PLDA with uncertainty propagation. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 3684–3688.

(Stolcke et al., 2008) A. Stolcke, S. Kajarekar, et L. Ferrer, 2008. Nonparametric feature normalization for svm-based speaker verification. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, 1577–1580. IEEE.

(Tipping et Bishop, 1999) M. E. Tipping et C. M. Bishop, 1999. Probabilistic principal component analysis. *Journal of the Royal Statistical Society : Series B (Statistical Methodology) 61*(3), 611–622.

(Tomashenko, 2017) N. Tomashenko, 2017. *Speaker adaptation of deep neural networkacoustic models using Gaussian mixturemodel framework in automatic speechrecognition systems*. Thèse de Doctorat.

(Tomashenko et al., 2016a) N. Tomashenko, Y. Khokhlov, A. Larcher, et Y. Estève, 2016a. Exploration de paramètres acoustiques dérivés de gmms pour l'adaptation non supervisée de modèles acoustiques à base de réseaux de neurones profonds. Dans les actes de *Journees d'etudes sur le parole (JEP)*, 337–345. AFCP.

(Tomashenko et al., 2016b) N. Tomashenko, Y. Khokhlov, A. Larcher, et Y. Estève, 2016b. Exploring gmm-derived features for unsupervised adaptation of deep neural network acoustic models. Dans les actes de *International Conference on Speech and Computer.*, 304–311. Springer.

(Variani et al., 2014) E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, et J. Gonzalez-Dominguez, 2014. Deep neural networks for small footprint text-dependent speaker verification. Dans les actes de *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP*, Volume 14, 4052–4056. Citeseer.

(Vogt et Sridharan, 2009) R. Vogt et S. Sridharan, 2009. Minimising speaker verification utterance length through confidence based early verification decisions. Dans les actes de *Proceedings of the Third International Conference on Advances in Biometrics*, 463–472. Springer.

(Vogt et al., 2008) R. J. Vogt, C. J. Lustri, et S. Sridharan, 2008. Factor analysis modelling for speaker verification with short utterances. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 1–4. IEEE.

(Vogt et al., 2009) R. J. Vogt, J. Pelecanos, N. Scheffer, S. Kajarekar, et S. Sridharan, 2009. Within-session variability modelling for factor analysis speaker verification. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 1563–1566.

(Wang et al., 2014) J. Wang, Y. Song, T. Leung, C. Rosenberg, J. Wang, J. Philbin, B. Chen, et Y. Wu, 2014. Learning fine-grained image similarity with deep ranking. Dans les actes de *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 1386–1393.

(Zeinali et al., 2016) H. Zeinali, H. Sameti, L. Burget, J. Cernockỳ, N. Maghsoodi, et P. Matejka, 2016. i-vector/hmm based text-dependent speaker verification system for reddots challenge. Dans les actes de *Annual Conference of the International Speech Communication Association (Interspeech)*, 440–444.

(Zhang et al., 2010) W.-Q. Zhang, Y. Shan, et J. Liu, 2010. Multiple Background Models for Speaker Verification. Dans les actes de *Odyssey Speaker and Language Recognition Workshop*, 47–51.

Curriculum vitæ - Anthony Larcher

1. Identification et données administratives

— Né le 28 janvier 1982 à Besançon (25).
— Maître de conférences échelon 5 à l'Université du Mans en section CNU 27.
— Membre du Laboratoire d'Informatique de l'Université du Mans (LIUM) - EA 4023.

1.1 Parcours professionnel

2016 à aujourd'hui Co-responsable de la filière informatique à l'École Nationale Supérieure d'Ingénieurs du Mans (ENSIM).

2014 à aujourd'hui Maître de conférences à l'Université du Mans, LIUM, équipe LST. Thèmes de recherche : caractérisation du locuteur et de la langue, modélisation acoustique de la parole.

2010-2014 Chercheur à L'Institute for Infocomm Research (I^2R), Singapour

2009-2010 Post-doctorant au Laboratoire Informatique d'Avignon (LIA), Université d'Avignon et des pays de Vaucluse.

2006-2009 Doctorant au Laboratoire Informatique d'Avignon (LIA), Université d'Avignon et des pays de Vaucluse.

1.2 Diplômes universitaires

2009 Doctorat en informatique (félicitations du jury), Université d'Avignon et des Pays de Vaucluse. Directeur de thèse : Professeur Jean-François Bonastre, co-directeur : Professeur John S.D. Mason (Swansea University, Pays de Galles).

2005 Master recherche (mention assez bien), Institut National Polytechnique de Grenoble

2005 Ingénieur électricien (mention assez bien), Institut National Polytechnique de Grenoble

2. Encadrement doctoral et scientifique

2.1 Doctorats

Natalia Tomashenko (défendue le 01/12/2017) : *On the Use of Gaussian Mixture Model Framework to Improve Speaker Adaptation of Deep Neural Network Acoustic*

Models. Implication dans l'encadrement : 40%. Directeur : Yannick Estève. Co-directeur : Yuri Matveev (PR, ITMO University, Saint-Petersbourg, Russie), co-encadrement Anthony Larcher.

Gaël Le Lan (defendue le 06/10/2017) *Analyse en locuteurs d'une collection de documents multimédia*. Implication dans l'encadrement : 30%, Directeur : Sylvain Meignier, co-encadrement : Delphine Charlet (Orange Labs) et Anthony Larcher.

Kévin Vythelingum (débutée en mars 2016) *Construction rapide, performante et mutualisée de systèmes multilingues pour la reconnaissance et la synthèse automatique de la parole*. Implication dans l'encadrement :40%, Directeur : Yannick Estève, co-encadrement Anthony Larcher.

Florent Desnous (débutée en octobre 2016) *Modélisation à contexte variable pour le reconnaissance du locuteur*. Implication dans l'encadrement : 60%, Directeur : Sylvain Meignier.

Yevhenii Prokopalo (débutée en septembre 2018) *Apprentissage au lng de la vie pour des systèmes autonomes intelligents*. Implication dans l'encadrement : 50%, Directeur Anthony Larcher, co-encadrement : Loïc Barrault.

Stéphane Kombo (débutée en octobre 2018) *Traitement automatique de la parole en réunion par dissémination de capteurs* Implication dans l'encadrement : 30%, Directeur : Jean-Hugh Thomas, co-encadrement : Silvio Montresor et Anthony Larcher

Pablo Sordo Matrinez (défendue en 2016) *Assessment of GMM-based Speaker Verification Systems under Variable Duration Conditions*. Implication dans l'encadrement : encadrement d'un stage de six mois entre février et juillet 2014, Directeur John S.D. Mason (Pr. Swansea University, Pays de Galles)

2.2 Master / Ingénieurs

Yevhenii Prokopalo (2018) Ajout d'information phonétique dans un système neuronal de vérification du locuteur

Florent Desnous (2016) Détection de la parole à partir de réseaux de neurones.

Adélice Lévy (2016) Réseau de neurones pour la reconnaissance du locuteur.

Adrien Brunet (2014) Compensation de la durée des segments acoustiques dans l'espace des scores pour la reconnaissance de la langue.

2.3 Rapporteur de thèse

Nguyen Duc Hoang Ha (2016) School of Computer Science and Engineering of the Nanyang Technological University, *Feature-based Robust Techniques For Speech Recognition*, rapporteur, 2016, Singapour

David James Vandyke (2014) University of Canberra : Faculty of Education, Science, Technology & Maths, *Glottal Waveforms for Speaker Inference & A Regression Score Post-Processing Method Applicable to General Classification Problems*, rapporteur, Canberra (Australie)

3. Rayonnement

3.1 Récompense

— Lauréat du prix **IES Prestigious Engineering Achievement Award 2013, Singapore** qui récompense le développement et la commercialisation à grande échelle d'un système de reconnaissance du locuteur dépendant du texte (dont je suis le principal contributeur). Prix remis par le ministre de l'environnement singapourien.

3.2 Sociétés savantes

— Élu au bureau de l'Association Francophone de la Communication Parlée (AFCP) suppléant de 2014 à 2016, titulaire depuis 2016 jusqu'en 2020.
— Membre de l'ISCA (International Speech Communication Association).

3.3 Relecteur

Journaux

— IEEE/ACM Transactions on Audio Speech and Language Processing ;
— Computer, Speech, and Language ;
— Speech Communication ;
— Eurasip Journal on Audio, Speech and Music Processing ;
— IEEE Transactions on Information Forensics & Security
— IET Information Security

Conférences

— Annual conference of the International Speech Communication Association (Interspeech) ;
— IEEE Internation Conference on Audio, Speech and Signal Processing (ICASSP) ;
— Speaker Odyssey ;
— European Signal Processing Conference (EUSIPCO) ;
— International Conference on Pattern Recognition (ICPR) ;
— Journées d'Études sur la Parole (JEP).

3.4 Commissions d'évaluation et expertises

— Expertises régulières pour l'Agence Nationale de la Recherche (ANR).
— Expertises internationales pour la Research Grants Council (RGC) of Hong Kong (2016).
— Membre de comités de sélection en 2016 et 2017 pour l'Université du Mans (postes MCF 27).

3.5 Campagnes d'évaluation

J'ai participé à plusieurs campagnes d'évaluation au sein de consortiums internationaux. Mon rôle dans ces campagnes a été très important (développement d'un ou plusieurs classifieurs, extraction de représentations qui ont été utilisées par plusieurs équipes du consortium).

NIST Speaker Recognition Evaluation NIST-SRE est l'évaluation majeure dans le domaine de la reconnaissance du locuteur. Cette évaluation est organisée depuis la fin des années 1990 tous les ans ou tous les deux ans plus récemment. J'ai participé à cette évaluation en 2008 et 2010 au sein de l'équipe du Laboratoire Informatique d'Avignon, en 2012 avec l'Institute for Infocomm Research et en 2016 au LIUM. Depuis 2012, ces participations s'inscrivent au sein d'un consortium qui a regroupé jusqu'à 16 équipes en 2016.

NIST Language Recognition Evaluation NIST-LRE est la principale campagne d'évaluation internationale dans le domaine de la reconnaissance de la langue. J'ai participé à cette campagne avec l'Institute for Infocomm Research en 2011 et au LIUM en 2015. Ces deux participations ont eu lieu dans le cadre d'un consortium de plusieurs équipes.

Speaker in the Wild est une compagne de vérification du locuteur organisée en 2015 par le SRI. J'y ai participé en 2015 afin de démontrer les performances de la plateforme libre SIDEKIT.

J'ai coorganisé, avec Kong Aik LEE, le RedDots Challenge dont les résultats ont été présentés lors d'une session spéciale à Interspeech 2016. Ce challenge portait sur plusieurs tâches de vérification du locuteur dépendante et indépendante du texte.

4. Valorisation

4.1 Développement de logiciels

Avec mon collègue Sylvain Meignier et mon ancien collègue Kong Aik Lee, j'ai développé la plateforme open-source SIDEKIT http://lium.univ-lemans.fr/sidekit qui est à la disposition de la communauté

En collaboration avec le LIA (Université d'Avignon) j'ai été le contributeur principal de la version 3.0 de la plateforme ALIZE (http://alize.univ-avignon.fr) distribué sous licence LGPL qui est une des plateforme les plus utilisées dans le domaine de la reconnaissance du locuteur au niveau international

4.2 Collections de corpus

Membre du projet HOME 2015 avec Kong Aik Lee j'ai contribué à la collection et à la distribution du corpus RSR2015 qui est l'une des ressources les plus importantes en termes de nombre de locuteurs dans le domaine de la reconnaissance du locuteur dépendante du texte.

Dans le cadre d'un projet avec les autorités portuaires de Singapour, j'ai dirigé la retransmission de la base de données RSR2015 à travers un canal VHF marine. La collection de données est disponible publiquement.

Instigateur du projet RedDots avec Kong Aik Lee j'ai contribué à la collection et à la distribution du corpus RedDots qui permet la comparaison et l'évaluation de systèmes de reconnaissance du locuteur dépendante et indépendante du texte en conditions réelles https://sites.google.com/site/thereddotsproject/. La conception et l'exploitation de ce corpus ont fait l'objet de sessions spéciales dans les conférences Interspeech 2014 et 2015.

5. Responsabilités scientifiques

— membre du conseil de laboratoire (LIUM) depuis 2016

5.1 Colloques

2018 Co-organisateur de Speaker Odyssey 2018 (Sables d'Olonne, France)

2017 Membre du comité d'organisation de SLSP 2017 (Le Mans, France)

2014 Membre du comité d'organisation d'Interspeech 2014 (Singapour, Singapour)

2012 Membre du comité d'organisation de Speaker Odyssey 2012 (Singapour, Singapour)

2012 Membre du comité d'organisation de IEEE Student 2012 (Kuala Lumpur, Malaisie)

2010 Membre du commité d'organisation de JEP/TALN 2010 (Avignon, France)

Organisateur de sessions spéciales :

— **Text-Dependent Speaker Verification with Short Utterances** à Interspeech 2014.

— **The RedDots Challenge : TowardsS Characterizing Speakers from Short Utterances** à Interspeech 2016.

5.2 Engagements contractuels

Coordinateur du projet ALLIES, 2018-2021 Le but du projet ALLIES est de développer un cadre d'évaluation pour des systèmes autonomes intelligents. Ce cadre inclut des métriques, protocoles et une plateforme d'évaluation reproductible. Les processus d'évaluation sont validés sur deux modalités : la segmentation regroupement en locuteurs et la traduction automatique. Des systèmes autonomes (capables de s'adapter et de s'évaluer de façon non supervisée) sont également développés pour les deux modalités visées.

Responsable scientifique pour le LIUM du projet Deep Privacy, 2019-2021 Les applications de reconnaissance de la parole nécessitent de centraliser de grandes quantités de données pour l'apprentissage ou l'adaptation de modèles génériques pouvant être utilisés à grande échelle. D'autre part, la reconnaissance de la parole tire parti d'informations spécifiques au locuteur (accent, expressions régionales ou professionnelles) qui nécessitent le partage d'information caractéristique du

locuteur. Afin de préserver la confidentialité des données, le projet Deep Privacy tend à dépersonnaliser les données (acoustiques et lexicales) fournies par les différents utilisateurs d'un système pour permettre l'échange de données utiles à la reconnaissance sans mettre en danger les données personnelles des différents locuteurs.

Responsable scientifique pour le LIUM du projet CapDiff, 2019-2020 Le projet CapDif permettra le transfert technologique des technologies de transcription enrichie, plus particulièrement l'aspect segmentation et regroupement en locuteur en exploitant des architectures matérielles incluant plusieurs microphones distants. Les applications visées incluent la transcription enrichie de réunions et les interactions avec des robots conversationnels.

Participation au projet LMAC est un projet mené en collaboration avec le Laboratoire d'Acoustique de l'Université du Mans (LAUM). La finalité de ce projet réside dans l'intégration des composantes de traitement du signal (débruitage, *beamforming*, localisation spatiale) développés par les acousticiens et des systèmes statistiques ou neuronaux développés au LIUM pour la reconnaissance du locuteur et de la parole afin d'optimiser l'ensemble de la chaîne de traitement pour la tâche visée. Ce projet tend à dépasser les performances des systèmes actuels qui sont développés de façon disjointe dans les deux domaines.

Participation au projet MPA, 2013-2014 Projet d'authentification vocale de capitaines de bateaux dans le port de Singapour à travers le canal VHF. Les technologies vocales devaient composer avec les fortes dégradations dues au canal très bruité et de très courtes durées d'authentification liées au contexte d'utilisation.

Membre du Joint Lab I^2R-DBS, 2013-2014 Le projet mené dans le cadre d'un laboratoire conjoint avec la banque singapourienne DBS a permis d'adapter un système de reconnaissance du locuteur dépendant du texte aux besoins d'authentification pour des opérations bancaires dans un cadre d'authentification multi-facteurs.

Membre du Joint Lab I^2R-BAIDU, 2011-2013 Mes travaux au sein du laboratoire conjoint I^2R / BAIDU ont permi d'implémenter un système de reconnaissance du locuteur en mandarin, intégré dans le système d'exploitation des téléphone LENOVO A586 qui ont été commercialisés en Chine.

Participation au projet HOME2015, 2010-2012 Développement d'un système de commande vocale personnalisé et sécurisé pour les besoins d'une *smart home*. La base de données collectée pour les besoins de ce projet est disponible publiquement (RSR2015).

Participation au projet européen FP7 MOBIO, 2009-2010 Le projet MOBIO avait pour but de développer un système d'authentification biométrique bi-modal (audio-vidéo) sur téléphone portable. Le projet a également permis de collecter et de

distribuer une base de données audio-vidéo et d'organiser une évaluation internationale.

Participation au projet ANR Biobimo, 2006-2009 Le projet Biobimo visait le développement de deux systèmes d'authentification bi-modales embarqués : un système de reconnaissance du locuteur renforcé par une information temporelle issue de la vidéo et un système de reconnaissance de visage combiné à une authentification vocale.

6. Enseignement

Mes activités d'enseignement se déroulent principalement à l'École Nationale Supérieure d'Ingénieurs du Mans (ENSIM) mais également dans différentes composantes de l'UFR sciences de l'Université du Mans comme le Master Analyse et Fouille de Données (AFD), le master Analyse et Traitement du Langage (ATAL) en collaboration avec l'université de Nantes. J'ai également enseigné dans le cadre du master international IMDEA en collaboration avec le département d'acoustique de l'Université du Mans. Je bénéficie d'un CRCT pour le deuxième semestre de l'année universitaire 2018-2020.

6.1 Responsabilités pédagogiques

— Responsable de l'option Interfaces Personnes Systèmes (IPS) pour l'École Nationale Supérieure d'Ingénieurs du Mans (ENSIM), depuis 2016.
— Co-responsable de la filière informatique de l'École Nationale Supérieure d'Ingénieurs du Mans (ENSIM), depuis 2016.
— Membre du pôle recrutement de l'École Nationale Supérieure d'Ingénieurs du Mans (ENSIM) depuis 2014.
— Responsable des relations avec le concours Archimède-Polytech et avec le concours ATS pour l'École Nationale Supérieure d'Ingénieurs du Mans (ENSIM) depuis 2014 et 2018.

6.2 Liste de mes enseignements

6.3 Encadrement de projets ingénieurs (ENSIM)

— 2018-2019 : Projet Jetlag, développement d'une application pour un réseau social de voyageurs (2 étudiants en troisième année de cycle ingénieur)

Algorithmique - structures de données	2014-2018	(35h/an)	Listes chaînées, piles, files, arbres binaires, arbres rouge/noir
C# et Windows Presentation Foundation	2015-2019	(30h/an)	Patron de conception MVVM, base du C# et XAML, conception d'applications
Architecture Big Data	2016-2019	(14h/an)	Programmation Hadoop, HDFS, MapReduce
Système de contrôle de versions	2017-2018	(17,5h/an)	Systèmes centralisés (Subversion) et distribués (GIT)
Traitement de la parole	2017-2019	(14,5h/an)	Reconnaissance du locuteur, paradigme de la modélisation Gaussienne, modèles neuronaux
Analyse et Conception d'IHM	2014-2016	(16h/an)	Techniques et méthodes d'analyse et de conception d'Interface Homme Machine.
Design d'Interaction	2014-2016	(14h/an)	Storyboards, prototype d'interfaces, de services ou de produits
Programmation en C	2015-2016	(12h/an)	Base de la programmation en C, pointeurs , fonctions
Développement d'Interfaces Homme/ Machines	2014	(6h/an)	Base de la programmation d'interfaces homme/machine en Java
Contrôle de versions	2017-2019	(25,5h/an)	Introduction au logiciels de contrôle de versions

— 2018-2019 : Système de reconnaissance du locuteur autonome, projet effectué dans le cadre du projet de recherche ALLIES (5 étudiants en deuxième année de cycle ingénieur)
— 2018-2019 : Service client/serveur pour un démonstrateur de reconnaissance du locuteur (2 étudiants en deuxième année de cycle ingénieur)
— 2017-2018 : Jeu vidéo sur table surface (6 étudiants en deuxième année de cycle ingénieur)
— 2017-2018 : Traitement en flux pour la reconnaissance du locuteur (2 étudiants en troisième année de cycle ingénieur)
— 2016-2017 : SpeakerDemo, conception d'une interface graphique pour un démonstrateur de reconnaissance du locuteur (1 étudiant en troisième année de cycle ingénieur)
— 2016-2017 : EasyVideo, assistant d'édition vidéo pour mise en ligne de MOOCs (5 étudiants en deuxième année de cycle ingénieur)
— 2016-2017 : RichMeeting, traitement en flux d'un signal de parole pour la création d'un démonstrateur (2 étudiants en troisième année de cycle ingénieur)
— 2016-2017 : Projet avec Valéo Étude des procédés de mesure indirecte de vitesse

de rotation pour des systèmes de ventilation automobile. (2 étudiants en troisième année de cycle ingénieur)

7. Publications et productions scientifiques

7.1 ACLi : Revues internationales avec comité de lecture

[Le Lan 2018] Gaël Le Lan, Delphine Charlet, Anthony Larcher et Sylvain Meignier, **An Adaptive Method for Cross-Recording Speaker Diarization** dans *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, volume 26, 2018, pp. 1821–1832

[Larcher 2014-c] Anthony Larcher, Kong Aik Lee, Bin Ma et Haizhou Li, **Text-dependent Speaker Verification : Classifiers, Databases and RSR2015** dans *Speech Communication*, volume 60, 2014, pp. 56–77

[Larcher 2013-d] Anthony Larcher, Jean-Francois Bonastre et John S.D. Mason, **Reinforced temporal structure of acoustic models for speaker recognition** dans *Digital Signal Processing*, volume 6, 2013, pp. 1910–1917

7.2 Brevets

[Larcher 2013-d] A. Larcher, K.A. Lee, B. Ma, H. Thai Ngoc Thuy Huong, **Method and system for Dual Scoring for Text-Dependent Speaker Verification** *U.S.*, P2012016/US, 2013

[Larcher 2013-c] A. Larcher, K.A. Lee, B. Ma, H. Thai Ngoc Thuy Huong, **Method and System for Dual Scoring for Text-dependent Speaker Verification** *China*, CN 103456304A, 2013

7.3 ACTI : Communications avec actes dans un congrès international

[Doukhan 2018] D. Doukhan, J. Carrive, F. Vallet, A. Larcher et S. Meignier, **An open-source speaker gender detection framework for monitoring gender equality** dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5214–5218

[Broux 2018] P.-A. Broux, F. Desnous, A. Larcher, S. Petitrenaud, J. Carrive, et S. Meignier, **S4D : Speaker Diarization Toolkit in Python** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2018, pp. 1368–1372

[Lelan 2017] G. Le Lan, D. Charlet, A. Larcher et S. Meignier, **A triplet ranking-based neural network for speaker diarization and linking** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 3572–3576

[Lee 2017] K.A. Lee, V. Hautamäki, T. Kinnunen, A. Larcher, C. Zhang, A. Nautsch, T. Stafylakis, G. Liu, M. Rouvier, W. Rao et al., **The i4u mega fusion and collaboration for NIST speaker recognition evaluation 2016** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2017, pp. 1328–1332

[Tomashenko 2016] N. Tomashenko, Y. Khokhlov, A. Larcher et Y. Estève, **Exploring GMM-derived features for unsupervised adaptation of deep neural network acoustic models** dans *International Conference on Speech and Computer*, 2016, pp. 304–311

[Le Lan 2016-b] G. Le Lan, S. Meignier, D. Charlet et A. Larcher, **First investigations on self trained speaker diarization** dans *Odyssey Speaker and Language Recognition Workshop*, 2016, pp. 152–157

[Le Lan 2016-a] G. Le Lan, D. Charlet, A. Larcher et S. Meignier, **Iterative plda adaptation for speaker diarization** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2016, pp. 2175–2179

[Lee 2016] K.A. Lee, H. Li, L. Deng, V. Hautamäki, W. Rao, X. Xiao, A. Larcher, H. Sun, T. Nguyen, G. Wang, et al., **The 2015 NIST language recognition evaluation : the shared view of I2R, Fantastic4 and SingaMS** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2016, pp. 3211–3215

[Larcher 2016] A. Larcher, K.A. Lee et Sylvain Meignier, **An extensible speaker identification sidekit in python** dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 5095–5099

[Lee 2015] K.A. Lee, A. Larcher, G. Wang, P. Kenny, N. Brümmer, D. van Leeuwen, H. Aronowitz, M. Kockmann, C. Vaquero, B. Ma, et al., **The RedDots data collection for speaker recognition** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2015, pp. 2996–3000

[Sordo 2014] P. L. Sordo Martínez, B. Fauve, A. Larcher et J.S.D. Mason, **Speaker verification performance with constrained durations** dans *International Workshop on Biometrics and Forensics*, 2014, pp. 1–6

[Larcher 2014-c] A. Larcher, K.A. Lee, P. L. Sordo Martínez, T. H. Nguyen, B. Ma, H. Li, **Extended RSR2015 for text-dependent speaker verification over VHF channel** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2014, pp. 1322–1326

[Larcher 2014-b] A. Larcher, K.A. Lee, B. Ma et H. Li, **Imposture classification for text-dependent speaker verification** dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 739–743

[Larcher 2014-a] A. Larcher, K.A. Lee, B. Ma et H. Li, **Modelling the Alternative Hypothesis for Text-Dependent Speaker Verification** dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 734–738

[Wu 2013] Z. Wu and Anthony Larcher and Kong Aik Lee and Eng Siong Chng and Tomi Kinnunen and Haizhou Li, **Vulnerability evaluation of speaker verification under voice conversion spoofing : the effect of text constraints** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2013, pp. 950–954

[Saeidi 2013] R. Saeidi, K.A. Lee, A. Larcher et al., **I4U submission to NIST SRE 2012 : A large-scale collaborative effort for noise-robust speaker verification** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2013, pp. 1986–1990

[Lee 2013] K.A. Lee, A. Larcher, C.-H. You, B. Ma et H. Li, **Multi-session PLDA Scoring of I-vector for Partially Open-Set Speaker Detection** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2013, pp. 3651–3655

[Larcher 2013-b] A. Larcher, J.-F. Bonastre, B. Fauve, K.A. Lee, C. Lévy, H. Li, J.S.D. Mason et J.-Y. Parfait, **ALIZE 3.0 - Open Source Toolkit for State-of-the-Art Speaker Recognition** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2013, pp. 2768–2773

[Larcher 2013-a] Anthony Larcher and Kong Aik Lee and Bin Ma and Haizhou Li, **Phonetically-Constrained PLDA Modeling for Text-Dependent Speaker Verification with Multiple Short Utterances** dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7673–7677

[Hautamäki 2013] V. Hautamäki, K.A. Lee, D. van Leeuwen, R. Saeidi, A. Larcher, T. Kinnunen, T. Hasan, S. Omid Sadjadi, G. Liu, H. Boril, J.H.L. Hansen, B. Fauve, **Automatic Regularization of Cross-Entropy Cost for Speaker Recognition** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2013, pp. 1609–1613

[Hautamäki 2012] V. Hautamäki, K.A. Lee, A. Larcher, T. Kinnunen, B. Ma et H. Li, **Variational Bayes logistic regression as regularized fusion for NIST SRE 2010** dans *Odyssey Speaker and Language Recognition Workshop*, 2012, pp. 268–274

[Larcher 2012-b] A. Larcher, K.A. Lee, B. Ma et H. Li, **The RSR2015 : Database for Text-Dependent Speaker Verification using Multiple Pass-Phrases** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2012, pp. 1580–1583

[Larcher 2012-a] A. Larcher, P.-M. Bousquet, K.A. Lee, D. Matrouf, H. Li at J.-F. Bonastre, **I-vectors in the context of phonetically-constrained short utterances for speaker verification** dans *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2012, pp. 4773-4776

[Jiang 2012] Y. Jiang, K.A. Lee, Z. Tang, B. Ma, A. Larcher, H. Li, **PLDA Modeling in I-vector and Supervector Space for Speaker Verification** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2012, pp. 1680-1683

[Bousquet 2012] Pierre-Michel Bousquet and Anthony Larcher and Driss Matrouf and Jean-Francois Bonastre and Oldrich Plchot, **Variance-Spectra based Normalization for I-vector Standard and Probabilistic Linear Discriminant Analysis** dans *Odyssey Speaker and Language Recognition Workshop*, 2012, pp. 1–8

[Lee 2011-b] K.A. Lee, A. Larcher, H. Thai, B. Ma et H. Li, **Joint Application of Speech and Speaker Recognition for Automation and Security in Smart Home** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2011, pp. 3317–3318

[Lee 2011-a] K.A. Lee, C.-H. You, V. Hautomaki, A. Larcher et H. Li, **Spoken Language Recognition in the Latent Topic Simplex** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2011, pp. 2933-2936

[Marcel 2010] S. Marcel, C. McCool, P. Matejka, J. Cernocky, J. Kittler, O. Glembek, O. Plchot, Z. Jancik, A. Larcher et C. Lévy, **On the Results of the First Mobile Biometry (MOBIO) Face and Speaker Verification Evaluation** dans *Lecture Notes in Computer Science*, 2010, pp. 210–225

[McCool 2010] C. McCool, S. Marcel, A. Hadid, M. Pietikainen, P. Matejka, J. Cernocky, J. Kittler, A. Larcher, C. Levy, D. Matrouf, J.-F. Bonastre, N. Poh, P. Tresadern et T. Cootes, **Bi-Modal Person Recognition on a Mobile Phone : using mobile phone data** dans *IEEE International Conference on Multimedia & Expo*, 2012, pp. 635–640

[Larcher 2010-b] A. Larcher, C. Lévy, D. Matrouf et J.-F. Bonastre, **Decoupling session variability modelling and speaker characterisation** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2010, pp. 2314–2317

[Larcher 2010-a] A. Larcher, J.-F. Bonastre et J.S.D. Mason, **Constrained Viterbi decoding for embedded user-customised password speaker recognition** dans *ACM Symposium on Applied Computing*, 2010, pp. 1501–1502

[Charton 2010] E. Charton, A. Larcher, C. Lévy, J.-F. Bonastre, **Mistral : open source biometric platform** dans *ACM Symposium on Applied Computing*, 2010, pp. 1503–1504

[Larcher 2008-c] A. Larcher, J.-F. Bonastre et J.S.D. Mason, **Short utterance-based video aided speaker recognition** dans *IEEE International workshop on Multimedia Signal Processing (MMSP)*, 2008, pp. 897–901

[Larcher 2008-b] A.Larcher, Jean-François Bonastre et John S.D. Mason, **From GMM to HMM for Embedded Password-Based Speaker Recognition** dans *European Signal and Image Processing Conference (EUSIPCO)*, 2008, pp. 1–5

[Larcher 2008-a] A.Larcher, Jean-François Bonastre et John S.D. Mason, **Reinforced temporal structure information for embedded utterance-based speaker recognition** dans *Annual conference of the International Speech Communication Association (INTERSPEECH)*, 2008, pp. 371–374

[Bonastre 2008] J.-F. Bonastre, N. Scheffer, D. Matrouf, C. Fredouille, A. Larcher, A. Preti, G. Pouchoulin, N. Evans, B. Fauve et J.S.D. Mason, **ALIZE/SpkDet : a state-of-the-art open source software for speaker recognition** dans *Odyssey Speaker and Language Recognition Workshop*, 2008, pp. 1–8

7.4 ACTN : Communications avec actes dans un congrès national

[Desnous 2018] F. Desnous, A. Larcher et S. Meignier, **Étude de l'impact de différents systèmes de détection de la parole sur des tâches traitement automatique de la parole** dans *Journées d'Études sur la Parole*, 2018, pp. 550–558

[Tomashenko 2016] N. Tomashenko, Y. Khokhlov, A. Larcher, Y. Estève, **Exploration de paramètres acoustiques dérivés de GMMs pour l'adaptation non supervisée de modèles acoustiques à base de réseaux de neurones profonds** dans *Journées d'Études sur la Parole*, 2016, pp. 1–8

[Larcher 2012] A. Larcher, P.-M. Bousquet, D. Matrouf et J.-F. Bonastre, **Analyse en Composante Principale pour l'extraction des i-vecteurs en vérification du locuteur** dans *Journées d'Études sur la Parole*, 2012, pp. 1–8

[Larcher 2008-d] A. Larcher, J.-F. Bonastre et J.S.D. Mason, **Reconnaissance automatique du locuteur embarqu'ee dans un téléphone portable** dans *Journées d'Études sur la Parole*, 2010, pp. 1–8

[Meigner 2008] S. Meigner, T. Merlin, C. Lévy, A. Larcher, E. Charton, J.-F. Bonastre, L. Besacier, J. Farinas et B. Ravera, **Mistral : Plate-forme open source d'authentification biométrique** dans *Journées d'Études sur la Parole*, 2008, pp. 1–8

7.5 Thèses

[Larcher 2009] Anthony Larcher, **Modèles acoustiques à structure temporelle renforcée pour la vérification du locuteur embarquée.**, *Université d'Avignon et des Pays de Vaucluse*, 2009, Présenté pour l'obtention de grade de Docteur

[Larcher 2005] Anthony Larcher, **Approche multi-énergies avec un détecteur spectrométrique en radiographie appliquée à la reconnaissance de matériaux.**, *Institut National Polytechnique de Grenoble*, 2005, Master Recherche