Université Henri Poincaré – Nancy I

Département de Formation Doctorale en Informatique **École Doctorale IAE + M**

Perception de signaux complexes et interaction homme-machine

MÉMOIRE

présenté et soutenu publiquement le Lundi 28 Octobre 1996

pour l'obtention de l'

Habilitation à Diriger des Recherches
(Spécialité Informatique)

par

Jean-François Mari

Composition du jury

Président :	Laurent Miclet
Rapporteurs :	Régine Andre-Obrecht Marc El-Beze Noelle Carbonell
Examinateurs :	Jean-Paul Haton Amedeo Napoli
Invité :	Guy Pérennou

Centre de Recherche en Informatique de Nancy

Résumé

Ce mémoire présente une synthèse de mes travaux de recherche en reconnaissance de la parole et plus généralement en reconnaissance des formes. Je décris une nouvelle classe de modèles stochastiques, appelés modèles de Markov du deuxième ordre. Ces modèles généralisent les modèles de Markov du premier ordre et permettent des performances supérieures. Différentes comparaisons sont faites sur des corpus de référence dans la cadre de la reconnaissance de la parole.

En plus de la description de l'originalité scientifique du travail de recherche, je décris différentes actions d'encadrement d'étudiants en thèses ainsi que différentes actions de valorisation : projets européens et contrats avec des partenaires nationaux.

Mots-clés: modèles stochastiques, HMM1, HMM2

Abstract

We propose an extension of the Viterbi algorithm that makes second-order hidden Markov models computationally efficient. A comparative study between first-order (HMM1) and second-order Markov models (HMM2) is carried out. Experimental results show that HMM2 provide a better state occupancy modeling and have alone performances comparable to HMM1 plus post-processing.

Remerciements

J'ai mis un long moment avant de me décider à écrire ce document en vue d'obtenir une habilitation. Je remercie en premier Jean-Paul Haton qui m'a convaincu de l'intérêt de cette démarche. Je tiens à le remercier particulièrement pour sa patience quand je n'étais pas un chercheur assidu.

Je remercie ensuite mes trois rapporteurs, Régine, Noelle et Marc pour avoir lu ce document pendant le mois août et pour l'attention qu'ils y ont portée.

Je remercie aussi mes examinateurs, Laurent et Guy pour avoir été toujours présents aux instants importants de ma carrière et Amedeo pour être resté tard le soir au bistrot tout en parlant de perspectives de recherche ; ce qui prouve qu'il est possible de faire de la recherche avec plein de gens différents et dans des endroits insolites. Ainsi, je remercie tous ceux qui ont été complices avec moi de prés ou de loin, sur la terre ou dans les airs, dans les vignes ou dans les bois.

Il y a beaucoup de figures dans ce mémoire. Elles sont presque toutes dues à Dominique, grand magicien de X11, qui a été la personne avec laquelle j'ai toujours travaillé avec plaisir.

Enfin, je tiens à remercier les institutions qui m'ont abrité pendant ce travail : l'école publique, l'université et le CNRS qui reste un bel endroit pour faire de la recherche.

Je dédie ce mémoire à tous ceux qui vont le lire.

Table des matières

Table des figures

Liste des tableaux

Chapitre 1

Introduction

1 Itinéraire

Après une maîtrise de mathématiques pures, j'ai commencé mes recherches dans l'équipe RFIA du CRIN sur la reconnaissance de la parole. Ce domaine, intrinsèquement pluri-disciplinaire me permit d'adapter des techniques d'analyse syntaxique issues du domaine de la compilation à la spécificité du signal vocal. J'obtins en 1979 une thèse de troisième cycle [Mari 79] dans laquelle je décrivais et rapportais les résultats d'une nouvelle version du système MYRTILLE 1 capable de reconnaître des langages artificiels syntaxiquement très contraints. La particularité de ce système était d'être conduit par un analyseur syntaxique fonctionnant par îlots de confiance. Chaque îlot se composait initialement d'un mot trouvé par un algorithme de localisation puis s'agrandissait au fur et à mesure des hypothèses émises par l'analyseur syntaxique. Je n'étais pas à l'époque impliqué dans la reconnaissance acoustico-phonétique et utilisais des suites simulées de phonèmes entachés d'erreurs[1]. Cette solution n'était pas satisfaisante et il nous fallait concentrer nos efforts sur l'obtention de treillis de phonèmes directement à partir du signal vocal. Cette constatation était d'actualité puisque à la même époque s'achevait aux USA le premier projet ARPA [Reddy 76] qui avait fait dire que l'effort principal devait être porté sur le décodage acoustico-phonétique.

J'abandonnai les hauts niveaux de la reconnaissance de la parole pour rechercher des traits acoustiques capables de localiser et d'identifier des mots dans un système mono-locuteur. Ces travaux aboutirent à un système expérimental [Mari 84] où la reconnaissance se faisait en plusieurs temps :

- localisation de traits acoustiques ;
- accès à la cohorte de mots possédant la même suite de traits.

Ce système resta sans lendemain faute de corpus pour effectuer un apprentissage et une évaluation. Pourtant, l'idée d'accéder à un sous-ensemble du vocabulaire (c.-à-d. la cohorte) à l'aide d'un patron phonétique avait donné des résultats prometteurs et était utilisée par d'autres équipes [Huttenlocher 84].

Cette nécessité de posséder un corpus suffisant pour créer un système de reconnaissance est apparue clairement à cette époque. Au même moment, la communauté française de recherche en reconnaissance de la parole misait sur les systèmes experts à base de règles pour

1. Nous gardions à chaque instant les trois meilleurs candidats ; ces suites étaient improprement appelées treillis de phonèmes.

décoder en phonèmes le signal acoustique. Les informaticiens qui mettaient au point ces systèmes constituaient des corpus ... de règles. Influencée par la réussite du système Harpy de CMU [Lowerre 76], la communauté anglo-saxonne choisissait une approche différente. La variabilité du signal de parole était modélisée par un processus de Markov dont il fallait estimer les paramètres sur de gros corpus ... de parole.

J'ai eu la chance d'être initié à cette dernière technique de modélisation entre 1983 et 1984 pendant une période sabbatique passée dans les laboratoires BBN (Bolt Beranek et Newman, USA) dans l'équipe de John Makhoul. Je découvrais les méthodes stochastiques de modélisation d'un mot ou d'un phonème. J'étais au plus bas dans la hiérarchie des traitements qui permettent de passer du signal de parole issu d'un microphone jusqu'à sa compréhension par un auditeur. Je côtoyais des filtres, des cepstres, j'estimais et lissais des densités de probabilités sur de gros corpus multi-locuteurs [Mari 85b, Mari 85a] et mesurais combien je possédais mal ces outils de reconnaissance de formes que sont les modèles stochastiques et l'analyse de données dont l'importance allait grandissant en reconnaissance de la parole. Rentré en France après cette débauche d'expériences, d'apprentissages et de tests comparatifs, je me retrouvais au sein de l'équipe RFIA du Crin comme responsable de la modélisation stochastique en reconnaissance des formes, plus particulièrement en reconnaissance de la parole.

2 Définition du thème actuel

Ce thème pose le problème de l'utilisation de modèles stochastiques pour la perception de signaux complexes. Il dépasse le champs de la reconnaissance des formes pour apparaître dans des domaines de l'intelligence artificielle [Berndt 96, Hughey 96]. Il possède aussi de fortes implications en traitement du signal, en classification, en estimation statistique et en théorie de l'apprentissage.

Initialement issue des mathématiques, la modélisation stochastique pose des problèmes fondamentaux et appliqués. L'aspect fondamental se justifie puisqu'il faut spécifier des modèles originaux de perception et des nouveaux algorihmes d'apprentissage et de reconnaissance. L'aspect appliqué reste notre souci car il faut évaluer l'influence des hypothèses simplificatrices posées et ramener à une mesure raisonnable la complexité d'algorithmes d'apprentissage et de reconnaissance afin d'implémenter des modèles opérationnels.

2.1 Importance scientifique

La modélisation stochastique est une méthode qui cherche à représenter la variabilité d'un signal temporel complexe à l'aide d'un ensemble de variables aléatoires dont il s'agit d'estimer efficacement les paramètres [Blaydon 70]. En considérant l'émission d'un signal comme le résultat d'un processus aléatoire, nous cherchons à modéliser ainsi notre ignorance. De ce fait, ce thème de recherche peut être vu comme étant complémentaire d'autres thèmes de l'intelligence artificielle car l'utilisation simultanée de connaissances et de modèles stochastiques permet une perception plus efficace de signaux complexes. Ce thème de recherche conduit tout naturellement à la spécification de systèmes qualifiés de *mixtes* ou d'*hybrides* qui font l'objet de recherches dans un de nos domaines privilégié d'application : la reconnaissance de la parole [Haton 95]. Dans ce domaine d'application et en visant une plus longue échéance, nous cherchons à faire progresser l'interaction homme-machine en réalisant des systèmes de reconnaissance de la parole pour une interaction multimodale avec un système multimédia.

2.2 Impact potentiel

Les impacts potentiels des recherches menées sur la modélisation stochastique se situent à différents niveaux suivant l'aspect fondamental ou appliqué envisagé.

A un niveau fondamental, la modélisation stochastique pose essentiellement des problèmes de classification [Fu 70, CLA 96] puisque les modèles stochastiques peuvent être vus comme des outils de classification temporelle qui permettent à la fois une segmentation temporelle du signal et une modélisation spectrale. Les classes[2] mises à jour par les HMM[3] sont de différentes natures suivant qu'elles concernent le spectre, la segmentation ou l'évolution du signal. On distingue :

les segments temporels qui possèdent une cohérence spectrale, soit stationnaire, soit transitoire ;

les spectres à court terme – ou trames – qui sont regroupés en *codebook* ;

les trajectoires des trames ou des états du processus stochastique qui modélisent une évolution temporelle du signal ;

les modèles stochastiques qui représentent des unités homogènes ayant le même comportement en contexte.

L'objectif est d'aboutir à une description donc à une interprétation des classes obtenues après l'apprentissage. Ce problème est un problème de catégorisation, d'extraction et de représentation de connaissances tel qu'il est abordé en fouille de données [Fayyad 96]. Nous envisageons ces problèmes conjointement avec les chercheurs qui travaillent sur l'organisation hiérarchique des connaissances, la fouille des données et le raisonnement par classification. Ceci s'est traduit récemment par l'organisation d'une journée d'étude sur les problèmes de classification [CLA 96].

A un niveau appliqué, nos travaux en reconnaissance de la parole à l'aide de ces méthodes sont autant d'exemples différents :

- réalisation de systèmes de reconnaissance de mots ;
- réalisation de systèmes de décodage acoustico-phonétique du français et de l'anglais ;
- adaptation des algorithmes à de la parole téléphonique permettant l'accès à distance à des centres de renseignements.

Nous avons aussi envisagé d'autres types de signaux :

- reconnaissance de l'écriture latine ou cursive, manuscrite ou imprimée ;
- localisation et identification de signatures sonores en milieux sous-marins à des fins de défense ;
- identification de cris de détresse dans des lieux publics (les couloirs du métro) ;
- planification de trajectoires pour un robot mobile par modélisation de son environnement et de ses différents états.

2. Il s'agit en fait de catégories.
3. Hidden Markov Model ou Modèles de Markov Cachés. Ils seront définis dans le chapitre suivant.

2.3 Défi en recherche

Nous tenons à spécifier des modèles originaux afin de les intégrer dans des systèmes hybrides de perception. Dans le domaine de la reconnaissance de la parole, nous travaillons sur un prototype de machine à dicter qui sera évalué dans le cadre du projet francophone AUPELF-UREF rassemblant les meilleures équipes du domaine travaillant sur le français. Parallèlement à cette activité de recherche, nous tenons à poursuivre notre effort de création d'une boîte à outils de modélisation stochastique constitués de progiciels d'apprentissage et de reconnaissance à l'aide de modèles stochastiques de type *modèles de Markov cachés* (*HMM* pour reprendre la terminologie anglo-saxonne). La constitution de cette boîte à outils représente un effort pour garder la maîtrise de ce thème ainsi qu'un savoir-faire suffisant pour continuer à réaliser des systèmes dans de nouveaux domaines. Parmi ceux ci, nous espérons de bons résultats en planification pour la robotique. Dans un futur proche, nous comptons diffuser cette boîte à outils.

Un autre défi important consiste à spécifier des modes d'interaction entre modèles numériques et modèles symboliques[4] pour réaliser un système *mixte* ou *hybride*. Notre expérience dans ce domaine s'est matérialisée par la réalisation de systèmes hiérarchiques de reconnaissance [Mari 94a, Pican 96] dans lesquels interviennent des connaissances, des réseaux de neurones et des modèles markoviens mais il est important d'envisager d'autres types d'interaction entre ces modèles.

3 Objectifs et moyens

3.1 Implantation d'algorithmes parallèles

La mise en œuvre d'algorithmes d'apprentissage et de reconnaissance en modélisation stochastique se fait avantageusement sur des machines parallèles.

Les algorithmes utilisés pour l'apprentissage de modèles stochastiques de mots ou de phonèmes sont parallélisables à différents niveaux de granularité. Le processus d'estimation statistique sur un échantillon de phrases revient à appliquer le même traitement sur toutes les phrases d'un corpus et ceci, sans communication avec le traitement relatif à une autre phrase puisque toutes ces phrases sont supposées indépendantes. Il n'existe qu'un rendez-vous, en fin de traitement, des processus pour permettre de transformer sous des formes plus ou moins compliquées des fréquences d'observation en probabilités. A l'heure actuelle nous distribuons les traitements sur différentes machines de notre réseau local.

A un niveau de granularité plus fin, on remarque que les algorithmes d'apprentissage et de reconnaissance sont des algorithmes où, à chaque instant, on prolonge N chemins dans un graphe en appliquant à chaque sommet de ce graphe le même traitement. La fusion ou l'élimination de différents chemins intervient dans un deuxième temps. A ce niveau, nous utilisons des outils pour paralléliser et implanter ces algorithmes sur les machines parallèles que le Centre Charles Hermite, projet de recherche fédérateur régional sur *la modélisation et le calcul à haute performances*, met à notre disposition. Nous comptons poursuivre notre effort dans ce centre au sein du projet *Algorithmes pour le Calcul à Haute Performance*.

L'hypothèse sous-jacente à la modélisation stochastique de la parole est que les segments qui composent le signal de parole sont les réalisations de variables aléatoires dont on peut estimer les paramètres. La méthode la plus connue consiste à utiliser des modèles de Markov cachés [Baker 74, Jelinek 76] pour représenter les propriétés spectrales des mots ou des phonèmes.

4. John Makhoul disait : « how to compare apples and oranges ? »

Ainsi, ces modèles ont permis la réalisation de systèmes de reconnaissance indépendants du locuteur capables de travailler sur des vocabulaires de plusieurs milliers de mots. Pourtant les limitations intrinsèques d'une telle modélisation sont assez vite apparues et différentes équipes dans le monde ont essayé d'adapter ces modèles afin qu'ils collent mieux à la réalité. Quelques solutions ont été proposées tout particulièrement dans le domaine de la modélisation de la durée et de la corrélation entre événements acoustiques. En plus de ces limitations dues au choix de la nature des modèles, la philosophie de l'apprentissage a aussi été remise en cause. L'apprentissage le plus couramment utilisé suit le principe du maximum de vraisemblance qui consiste à estimer un modèle à l'aide d'un corpus d'apprentissage indépendamment des autres modèles. On présume de la ressemblance entre l'espace sur lequel les modèles ont été appris et l'espace de test sur lequel on veut obtenir les meilleures performances. Quand ces deux espaces ne se ressemblent pas, le système se révèle être un très mauvais reconnaisseur. C'est pour cela qu'ont vu le jour des nouvelles techniques d'apprentissage dans lesquelles on essaie de mettre en compétition les modèles pour qu'ils se différencient le plus les uns des autres tout en restant proches du corpus d'apprentissage. Dans de telles techniques, on utilise fréquemment des réseaux de neurones pour estimer les probabilités *a posteriori* d'observer un modèle à partir de l'évidence acoustique [Bourlard 94]. D'autres techniques d'apprentissage cherchent à maximiser une mesure d'information mutuelle sur l'ensemble du vocabulaire. Toutes ces méthodes sont à l'ordre du jour et font l'objet de recherches intensives. Mais les temps de calcul demandés sont devenus prohibitifs, sauf pour ceux qui peuvent utiliser une machine parallèle ... C'est dans ces directions que nous désirons aller.

L'émergence des techniques stochastiques en reconnaissance de la parole est principalement due à l'apparition de nouveaux serveurs de calcul puissants. Beaucoup d'hypothèses simplicatrices ont été posées dans les années 1980 pour implanter des algorithmes d'apprentissage et de reconnaissance : l'indépendance des segments de parole entre-eux et l'utilisation de chaînes de Markov du premier ordre en sont les plus connues. Nous nous intéressons à l'utilisation de modèles stochastiques d'ordre supérieur comme les modèles de Markov d'ordre 2 et les modèles neuromimétiques. Ainsi, l'utilisation de machines puissantes nous permet d'explorer des domaines de la modélisation stochastique encore nouveaux et de pouvoir répondre aux appels d'offres internationaux. A ce titre, notre groupe est gros consommateur de temps CPU.

Très tôt, les chercheurs ont noté que les systèmes de reconnaissance de la parole pouvaient être réalisés grâce à des tâches parallèles mettant en œuvre des sources de connaissances différentes et complémentaires. Aucun travail concret n'a été réalisé dans ce sens, principalement à cause du manque de matériel adéquat. Une machine parallèle nous permettra de tester différentes architectures qui n'existent actuellement que sur papier ou tableaux noirs. Dans ce domaine, notre groupe a développé des outils multi-agents pour fédérer des méthodes numériques et symboliques de perception et d'interprétation que nous aimerions tester en vraie grandeur.

3.2 Évaluation sur de gros corpus

La mise en œuvre d'algorithmes d'apprentissage de modèles stochastiques implique la manipulation de gros corpus. En reconnaissance de la parole nous nous sommes intéressés à plusieurs corpus de parole français et anglais et à plusieurs type de parole : parole lue de bonne qualité et parole téléphonique plus ou moins spontanée qui représente le cas connu le plus difficile. Décrivons ces différents corpus.

TI-NIST [Leonard 84] est un corpus de suites de chiffres enregistré par Texas-Instruments. Il s'agit de parole de bonne qualité enregistrée à 20 kHz. Ce corpus contient 11 chiffres : *zero, one, . . . , nine, oh* enregistrés par 225 adultes (111 hommes et 114 femmes). Il existe un sous-corpus constitué de phrases prononcées par des enfants mais nous ne l'utilisons pas. Chaque locuteur a prononcé 77 suites de chiffres d'une longueur comprise entre 1 et 7 chiffres. Ce corpus est subdivisé en deux sous-ensembles de taille égale et ne contenant aucun locuteur commun : le corpus d'apprentissage et le corpus de test. Ce corpus nous sert à évaluer des systèmes de reconnaissance de suites de mots.

TIMIT [Garofolo 93] est un acronyme formé des initiales de *Texas Instruments* et *Massachusetts Institute of Technology*. Les premiers ont enregistré le corpus tandis que les seconds étiquetaient le signal en phonèmes. Conçu initialement pour permettre l'acquisition de connaissances en acoustique et phonétique ainsi que pour évaluer les systèmes de décodage acoustico-phonétique, ce corpus, constitué d'un seul CDROM s'avéra vite insuffisant pour évaluer les systèmes de reconnaissance.

TIMIT est un corpus de parole lue. Les textes et les locuteurs appartiennent à deux partitions du corpus : l'une pour le test et l'autre pour l'apprentissage. Cette dichotomie s'est faite sur les critères suivants :

- approximativement, 20 % des textes et des locuteurs servent pour le test et 80 % pour l'apprentissage ;
- aucun locuteur n'apparaît à la fois dans l'espace de test et d'apprentissage ;
- un nombre minimun de mots apparaît simultanément dans le sous-ensemble d'apprentissage et dans le sous-ensemble de test ;
- tous les phonèmes sont représentés dans un maximum de contextes différents.

Le sous-ensemble de test contient un total de 1344 phrases prononcées par 168 personnes (112 hommes et 56 femmes). Quant au sous-ensemble d'apprentissage, il contient 3696 phrases prononcées par 462 personnes (308 hommes et 154 femmes). Du sous-ensemble de test a été extrait un ensemble de 192 phrases destinées à tester les systèmes de reconnaissance en minimisant le biais introduit à l'apprentissage par des contextes phonétiques communs avec le test. Ce sous-ensemble est construit en limitant le corpus de test aux seules phrases ne comportant pas de mots qui apparaissent simultanément à l'apprentissage et au test. Ce sous ensemble, connu sous le nom *core test*, est trop petit pour avoir une signification statistique. Le corpus de test de *TIMIT* contient 50754 phonèmes ce qui fait que le rayon de l'intervalle de confiance au risque de 5 % est de l'ordre de 0.4 % au voisinage des taux de reconnaissance couramment publiés (de l'ordre de 70 à 75 %).

BREF80 correspond à l'effort d'un laboratoire français, *le LIMSI*, pour maîtriser cet outil d'évaluation que constitue le gros corpus de parole lue. Au fur et à mesure que les performances des systèmes de reconnaissance de la parole augmentent, il est nécessaire de les tester sur des ensembles de plus en plus volumineux. Dans le cadre de la reconnaissance du discours continu en français, le corpus *BREF80* [Gauvain 90, Gauvain 91] constitué de 5330 phrases lues par 80 locuteurs est un outil puissant d'évaluation et de comparaison de systèmes de reconnaissance de la parole. Contrairement à *TIMIT* qui avait été étiqueté, *BREF80* ne l'est pas. Ce travail reste à la charge de l'équipe qui désire l'utiliser. Nous avons rapidement effectué ce travail qui a été publié récemment [Fohr 96] afin de tester un système de décodage acoustico-phonétique utilisant nos modèles stochastiques sur de la parole continue. Dans le cadre du projet francophone AUPELF-UREF portant sur l'évaluation de systèmes de dictée automatique, le *LIMSI* prévoit de mettre à disposition de nouveaux CDROM constituant une extension de *BREF80*. Pendant la campagne officielle de tests du projet AUPELF-UREF, le comité d'organisation mettra des corpus de développement et de test à la disposition des équipes participantes.

OGI achèvera notre description des corpus utilisés. Il s'agit d'un corpus de parole téléphonique distribué par le centre *Spoken Language Understanding* du *Oregon graduate Institute* [Cole 91, Cole 92]. Plus de 4000 personnes appelèrent au téléphone pour donner leur voix. A chaque fois, un répondeur leur demandait d'épeler leurs noms et prénoms avec ou sans pause. Afin d'obtenir suffisamment de prononciations de lettres, chaque personne devait réciter l'alphabet. Tous n'y arrivèrent pas ! Chaque appel est transcrit sous forme de la liste des lettres prononcées et de quelques annotations représentant des évènements comme les toux, rires, hésitations, ... Nous décidâmes de retirer tous les appels qui contenaient autre chose que de la parole ainsi que tous les appels dans lesquels l'ensemble des consignes n'était pas respecté. Finalement, nous avons partagé le corpus restant en trois parties : les ensemble d'apprentissage, de test et de validation. La description de chacun d'eux se trouve dans la table 1.1. Chaque locuteur n'appartient qu'à un seul ensemble. Le signal est enregistré à 8 kHz. Ce corpus est

	Apprentissage	Validation	Test
nbr. d'alphabets	240	40	0
nbr. de prénoms et de noms épelés avec pauses	1060	1017	0
nbr. de noms épelés normalement	1222	186	491

TAB. 1.1 – *Taille des ensembles d'apprentissage, de validation et de test pour nos expériences sur le corpus OGI*

particulièrement intéressant pour l'étude de la parole téléphonique. Il s'agit d'appels de provenance quelconque : appels locaux ou lointains. Sur ce corpus, nous avons pu évaluer différentes méthodes d'analyse acoustique cherchant à diminuer l'influence de la qualité de la ligne qui variait énormément d'un appel à l'autre. Une autre caractéristique de ce corpus n'a pas encore été mise à profit ; ce corpus contient de la parole spontanée émise par des personnes qui ne sont pas habituées à parler à un ordinateur. Nous comptons utiliser ce corpus pour mener des recherches en localisation de mots pour traiter de la parole spontanée.

4 Compétences et forces

Après 10 années de travail sur la modélisation stochastique, nous pouvons faire un premier bilan en matière de publications relatant des résultats scientifiques majeurs et en termes d'encadrement d'étudiants.

4.1 Encadrement de DEA ou thèses

J'ai co-encadré deux thèses d'université. La première soutenue par A. Kriouile traitait de la reconnaissance de la parole par des modèles markoviens du deuxième ordre qui généralisent les modèles du premier ordre [Kriouile 90a] et l'autre soutenue par J. Anigbogu qui concernait la reconnaissance de l'écriture à l'aide des mêmes méthodes [Anigbogu 92].

Plusieurs étudiants de DEA ont choisi les sujets proposés sur ce thème. En plus de A. Kriouile en 1986 et J. Anigbogu en 1988, il faut citer les noms de tous ceux qui n'ont pas soutenu de thèse :

- H. Binda, s'est intéressé en 1994 à l'utilisation de réseaux de neurones conjointement avec des modèles de Markov ; ce travail fait l'objet d'un paragraphe dans le chapitre de ce document consacré à l'utilisation des modèles stochastiques ;
- en 1995, M. Parot a utilisé des HMM pour segmenter le signal en grandes classes de phonèmes avant de reconnaître chaque phonème par extraction d'indices et utilisation de réseaux de neurones ;
- en 1995, P. Laroche a débuté une étude où les modèles de Markov sont utilisés en planification de déplacement d'un robot. Ce travail doit se poursuivre par une thèse, une fois l'étudiant dégagé des obligations militaires ;
- actuellement I. Zitouni étudie les algorithmes de reconnaissance de mots dans un système de dictée automatique afin de réaliser à terme le niveau de reconnaissance lexical qui nous manque.

4.2 Publications de travaux de recherche

Pour avoir dit que la classification est un thème de recherche qui nous intéresse, il nous reste à définir les axes factoriels sur lesquels projeter nos publications afin de mieux représenter notre espace de recherche. A quelques exceptions près [Amin 89, CLA 96], tous ces travaux concernent la reconnaissance de la parole. Nous avons choisi trois axes principaux :

- évaluation de nos méthodes sur des corpus de référence suivant qu'elles s'appliquent à la reconnaissance de suites de mots ou de parole continue ;
- description d'algorithmes originaux ;
- utilisation de parole de bonne qualité ou téléphonique.

Publications de résultats sur des corpus de référence

Nous considérons que l'évaluation de nos méthodes sur des corpus de référence est une nécessité pour garder une audience nationale ou internationale et intéresser des contractants. C'est une notion que j'ai admise en travaillant chez BBN qui luttait pour obtenir des fonds de l'agence ARPA.

Pour évaluer nos méthodes, nous avons utilisé les corpus décrits précédemment et publié nos résultats dans les congrés suivants :

- ICSLP'94 [Mari 94b] pour la reconnaissance de suites de chiffres sur *TI-NIST* ;
- ICASSP'96 [Mari 96] pour le décodage acoustico-phonétique de la parole continue en anglais sur *TIMIT* ;
- JEP'96 [Fohr 96] pour le décodage acoustico-phonétique de la parole continue en français sur *BREF80* ;
- ICSLP'94 [Mari 94a], ICASSP'95 [Junqua 95b] et EUROSPEECH'95 [Junqua 95a] pour l'évaluation de systèmes de reconnaissance de suites de mots à base de HMM d'ordre un et deux sur la parole téléphonique du corpus *OGI*.

Publications d'algorithmes originaux

Les algorithmes originaux publiés concernent :

les modèles de Markov du deuxième ordre pour lesquels A. Kriouile a spécifié en 1990 l'apprentissage selon le principe du maximum de vraisemblance [Kriouile 90a]. Ils ont été publiés en 1990 à ICASSP [Kriouile 90c] puis évalués sur *TI-NIST* en 1994 [Mari 94b] et enfin sur *TIMIT* en 1996 [Mari 96] ;

les systèmes hybrides ont fait l'objet de trois publications : en 1987 [Haton 87], en 1994 [Mari 94a] où sont reportés des résultats sur *OGI* et en 1996 [Pican 96] sur la classification des occlusives de *TIMIT* à l'aide de réseaux de neurones OWE dépendants du contexte ;

l'organisation d'un lexique et son accès à des cohortes de mots possédant le même patron phonétique a été publié en 1984 [Mari 84] ;

l'algorithme de Viterbi-bloc inspiré de l'algorithme de Viterbi [Forney 73] utilise un principe d'optimalité locale et permet un décodage plus efficace en termes d'occupation mémoire. Il fait l'objet d'un paragraphe dans le chapitre consacré à la modélisation stochastique.

l'analyse syntaxique par îlots de confiance fut mon premier souci et donna naissance à une publication dans une revue [Mari 81]. Elle reviendra au goût du jour quand nous progresserons dans l'implémentation des niveaux lexical et syntaxique de la machine à dicter : sujet de thèse de I. Zitouni.

Je ne parlerai pas, volontairement, des systèmes hybrides car ils font l'objet d'une préoccupation commune entre D. Fohr avec qui je co-signe la presque totalité de mes travaux depuis 1994. Dans ce domaine, D. Fohr a amené son expertise sur les réseaux de neurones et les systèmes experts [Fohr 89]. La mise en commun de nos savoir-faire a permis de développer des systèmes mixtes utilisant des HMM pour segmenter et des réseaux de neurones pour identifier des noms de personnes par épellation de lettres au téléphone. Ce travail sera détaillé dans l'habilitation de Dominique. Dernièrement, et dans le cadre du décodage acoustico-phonétique de la parole continue, nous avons expérimenté un nouveau type de réseau neuromimétique de type OWE pour classer un phonème de type occlusif en fonction de son contexte [Pican 96].

La comparaison entre la modélisation à l'aide de modèles de Markov du premier ordre (HMM1) et du deuxième ordre (HMM2) a été longtemps notre souci. Le chapitre consacré aux modèles stochastiques retrace les péripéties sur différents corpus de parole de bonne qualité. Pour évaluer ces modèles sur de la parole téléphonique, nous avons publié en collaboration avec les laboratoires STL de Panasonic (Santa Barbara, Californie) différents éléments de comparaison : comparaison avec différentes analyses acoustiques telles que analyse par calcul de coefficients cepstraux ou coefficients RASTA-PLP ; utilisation de coefficients de régression calculés sur différentes fenêtres temporelles et utilisation de soustraction cepstrale. Ces travaux ont été publiés aux congrès internationaux du domaine : ICASSP'95 [Junqua 95c] et EUROSPEECH'95 [Junqua 95b]. N'étant pas moi-même à l'initiative de ces travaux, ceux-ci doivent être revendiqués par Dominique.

Nous conclurons ce paragraphe consacré aux objectifs et moyens en disant que le travail en équipe est fondamental dans un domaine aussi pluri-disciplinaire que la reconnaissance de la parole. Nous avons la chance et le désir d'y arriver, ce qui constitue un atout majeur pour continuer à travailler dans ce domaine et fournir des résultats de qualité.

5 Positionnement dans le contexte français et étranger

5.1 Notre originalité

Le nombre d'articles consacrés à la modélisation stochastique dans le congrès phare qu'est ICASSP est un moyen de suivre la popularité grandissante de cette méthode au cours des années. Depuis 1985, il n'est pas une équipe qui n'ait essayé d'utiliser des modèles markoviens pour réaliser un système de reconnaissance de la parole. Les modèles de Markov du premier ordre généralisaient la programmation dynamique et permettaient un apprentissage automatique. Initialement [Baker 74, Jelinek 76], les HMM étaient discrets ; ils utilisaient des lois non-paramétriques et discrètes pour modéliser les densités associées aux observations (*cf.* chapitre 2 sur la modélisation stochastique) mais rapidement ces lois devinrent continues [Rabiner 85]. Les HMM semi-continus furent introduits par Huang [Huang 90]. En diminuant le nombre de densités, ils permettent une estimation plus robuste de leurs paramètres. En 1993, S.J. Young [Young 93] commençait à diffuser un progiciel à base de HMM continus : HTK. Cette boîte à outils est couramment utilisée pour construire des systèmes à base de HMM par beaucoup d'équipes qui n'ont pas voulu (ou pu) affronter la complexité d'une telle réalisation. Quant à nous, nous avons choisi de développer notre propre progiciel à base de modèles de Markov du second ordre.

Le chapitre 2 consacré à la modélisation stochastique retrace les différentes étapes de la comparaison sur différents corpus entre HMM1 et HMM2. Il nous permet d'évaluer cette nouvelle hypothèse : la reconnaissance de la parole peut être avantageusement modélisée par un processus markovien d'ordre supérieur.

Le chapitre 3 décrit la mise en œuvre des modèles de Markov d'ordre deux sur différents corpus.

Afin de diminuer la complexité des algorithmes d'apprentissage et de test, une autre hypothèse sur la nature des lois de probabilités est faite fréquemment ; celles ci sont supposées être des combinaison linéaires de lois normales à matrice de covariances diagonale. Nous avons exploré l'influence de cette hypothèse. Le chapitre 4 présente une synthèse des résultats sur différents corpus.

Le chapitre 5 décrit différents systèmes de reconnaissance de la parole que nous avons réalisés dans le but d'étudier les applications potentielles de la reconnaissance de la parole en interaction homme-machine.

5.2 Nos concurrents

Dans le domaine de la réalisation de systèmes de reconnaissance de la parole et de leur évaluation sur des corpus de référence, de nombreuses équipes ont fourni des très bons résultats aux niveaux phonétique et lexical. En Europe, nous citerons le *LIMSI* [Lamel 95] et les laboratoires *Philips* [Dugast 95, Aubert 95]. Il y a trop de corpus comme ceux déjà mentionnnés et trop d'équipes aux USA qui travaillent sur ces corpus et qui revendiquent des résultats issus de systèmes parfaitement réglés pour les citer toutes. Nous avons pris dans ce domaine un retard certain que nous comblons progressivement. Actuellement notre équipe a accueilli un étudiant de DEA qui travaille sur la réalisation du niveau lexical d'un système de dictée automatique qu'il reste à évaluer sur *BREF80* dans le cadre du projet AUPELF-UREF.

Dans le domaine de la parole téléphonique, nous ne connaissons pas d'équipe qui travaille sur le corpus *OGI*. En France, il ne faut pas oublier le *CNET* à Lannion qui travaille sur des corpus de parole téléphonique qui ne sont pas diffusés et donc qui ne permettent pas de comparaison avec d'autres systèmes.

Dans le domaine de l'intégration de réseaux de neurones (*Artificial Neural Networks* ou *ANN*) dans des systèmes à base de HMM, il faut citer les travaux de H. Bourlard et N. Morgan [Bourlard 94] et Y. Bengio [Bengio 96] qui utilisent des *ANN* pour approcher des densités et/ou transformer l'espace des vecteurs d'entrée.

5.3 Participation à des projets européens ou francophones

Sur le plan international, nous participons ou avons participé aux projets suivants :

SAM – *Speech Assessment Methods* – fut un projet européen ESPRIT 1 dans lequel nous avons développé une méthodologie d'évaluation de systèmes de reconnaissance de grands vocabulaires. Ce fut une expérience intéressante car les relations de travail avec la communauté anglo-saxonne y furent difficiles. Celle-ci s'est beaucoup mieux adaptée que nous aux lourdeurs administratives imposées par la CEE et a très bien su tirer partie de ce projet qui n'aboutit qu'à la spécification d'une station d'évaluation de systèmes de reconnaissance de mots isolés. Une publication [Bourjot 89] résume notre contribution ;

COST 249 – 250 sont deux projets européens qui s'intéressent au traitement de la parole téléphonique. Ces projets, faiblement subventionnés, sont des groupes de réflexion et d'échanges concernant l'implémentation de systèmes de reconnaissance de la parole téléphonique. Nous y avons exposé nos travaux [Fohr 95, Junqua 95b, Mari 96] sur l'influence de l'analyse acoustique dans les performances de systèmes de reconnaissance à base de HMM1 et HMM2 évalués sur le corpus *OGI* ;

AUPELF-UREF est une action francophone qui a débuté en 1995. Nous participons au projet *B1* destiné à évaluer notre système de dictée vocale intégrant nos HMM2. Nous possédons un bon décodage en phonèmes indépendant du locuteur qui donne plus de 72 % de taux de reconnaissance à l'aide de HMM2 indépendants du contexte. Actuellement, nous portons notre effort sur la reconnaissance lexicale et la spécification de phonèmes en contexte : les allophones.

VODIS – *Voice Operated Driver Information Systems* – est le nom du dernier projet européen dans lequel nous sommes impliqués depuis juin 1996. Ce projet regroupe essentiellement des constructeurs automobiles désirant équiper leurs modèles d'un auto-radio contrôlable par la voix. Notre activité première, qui a permis de nous faire accepter dans ce projet, est de collecter puis d'étiqueter un corpus de phrases prononcées par 200 personnes ; une partie de ces phrases représente les commandes du conducteur pour l'auto-radio, le téléphone mobile et le système de navigation routière. A plus long terme, cette implication nous permettra de posséder un gros corpus de parole bruitée et de tester, nous aussi, en parallèle avec d'autres contractants, nos systèmes de reconnaissance de la parole en ambiance bruitée.

Les appels d'offres du CNET termineront notre liste. Notre projet de recherche en collaboration avec France Telecom et le Centre National d'études des Télécommunications sur les interactions multimodales pour les sytèmes multimédia a été récemment retenu. Il permettra de financer une bourse de thèse pendant trois ans. Les objectifs du contrat, les méthodes utilisées ainsi que les résultats attendus sont décrits plus précisemment au chapitre 5 traitant de l'interaction homme-machine.

5.4 Notre apport à la recherche

Notre apport à la recherche se situe principalement dans la définition et la diffusion d'algorithmes originaux : modèles stochastiques du deuxième ordre et algorithmes de reconnaissance à optimalité locale. Le chapitre suivant est consacré à ces méthodes.

Nous nous sommes aussi intéressés aux architectures hybrides à base de HMM et de modèles connexionnistes neuromimétiques. Nous expérimentons des architectures hiérarchiques à base de HMM2 et perceptrons [Mari 94a] ou des modèles originaux de type OWE [Pican 96]. Ce domaine n'est pas développé dans ce document mais apparaîtra dans le document de Dominique Fohr.

Les modèles de Markov cachés sont des outils de classification de signaux temporels. Nos travaux peuvent vus comme des exemples d'extraction de connaissances à partir de gros corpus. Nous maintenons une interaction forte avec les chercheurs de notre équipe qui travaillent sur la fouille de données et l'apprentissage symbolique.

Enfin, nous tenons à ce que nos travaux de recherche débouchent sur des réalisations pratiques intéressant le monde industriel comme le témoignent les deux derniers contrats passés, l'un avec *VODIS* et l'autre avec le *CNET*. Ainsi, nous avons consacré un chapitre de ce document à la description de systèmes de reconnaissance opérationnels réalisés ces deux dernières années. Dans cette description, nous mettons plus particulièrement l'accent sur l'interaction homme-machine.

Chapitre 2

Outils pour la modélisation stochastique

1 Introduction

Une façon intuitive de justifier l'utilisation des modèles de Markov en reconnaissance de la parole est d'approcher l'évolution du conduit vocal pendant l'élocution par un processus stochastique.

Imaginons un instant que nous soyons en présence d'un phénomène difficilement explicable comme par exemple la variabilité du signal de parole quand une personne dit plusieurs fois la même chose. Il y a plusieurs façons d'aborder le problème :

- soit on se penche sur le phénomène pour en comprendre le fonctionnement grâce à un ensemble de règles, de causes possibles et d'effets. On peut alors découvrir que l'émission des sons correspond à une articulation complexe des organes vocaux. On se rend vite compte que la modélisation physique de ces phénomènes est difficile ou grossière par des méthodes analytiques ;

- soit on refuse de comprendre l'ensemble de ces phénomènes et on fait des statistiques en considérant que le geste intelligent qu'est la production d'un mot dans un contexte phonétique particulier n'est qu'une suite temporelle d'événements aléatoires.

Ces deux approches s'opposent l'une à l'autre. Dans la première, il s'agit de représenter des connaissances alors que dans la deuxième, on cherche à modéliser le hasard ce qui, en quelque sorte, revient à modéliser l'ignorance.

Dans ce chapitre, nous présenterons une classe particulière de modèles statistiques : *les modèles de Markov cachés* ou *MMC* (en anglais *Hidden Markov Models* ou *HMM*). Ils ont été introduits dans les années soixante dix par des chercheurs américains d'IBM [Baker 74, Jelinek 76] et de *Dragon system* [Baker 75] pour la reconnaissance de la parole. Ces modèles présentent l'avantage de permettre un apprentissage automatique à partir de gros corpus de parole. À l'heure actuelle, après un effort important pour collecter ces corpus, effort essentiellement anglo-saxon soit dit en passant, les taux de reconnaissance de phonèmes approchent les quatre vingts pour-cent dans le cas de la parole continue non bruitée. Cette valeur reste encore insuffisante pour permettre la réalisation d'applications de type grand public d'autant plus que l'utilisation du téléphone pour véhiculer la voix depuis l'utilisateur jusqu'au système de reconnaissance pose des problèmes supplémentaires qui sont loin d'être résolus.

À mes débuts en recherche sur les HMM, j'ai beaucoup souffert de l'aridité des notations utilisées dans la littérature anglaise. Dans les équations, le choix des notations était plus ou moins heureux et les indices avaient une place variable suivant les auteurs. Plus tard, quand ce fut mon tour d'expliquer ce qu'est la modélisation stochastique à l'aide de HMM, j'ai regretté l'absence d'un document écrit qui n'effraie pas son lecteur en présentant les notions nécessaires de probabilité et de statistique. Ce chapitre essaie d'être ce que j'aurais aimé trouver pour apprendre et faire apprendre ces notions. J'ai adopté la notation du plus grand nombre. J'aimerais que ce chapitre soit introductif sans être simpliste, illustré sans ressembler à un livre d'histoire française à l'usage des écoles primaires des colonies d'outre-mer et qu'il indique les références pour ceux qui veulent approfondir le sujet. Il y en a toujours. Mais auparavant, il nous faut définir comment construire un système de reconnaissance fondé sur la théorie des probabilités et des statistiques. Pour ce faire, nous utiliserons un exemple simple.

2 Apprentissage et reconnaissance probabiliste

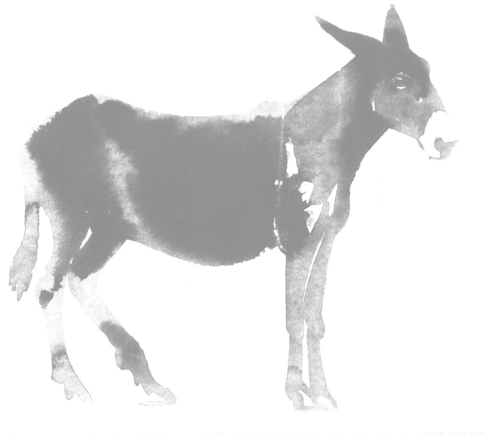


FIG. 2.1 – *Un âne*

Considérons le problème suivant. Un cultivateur achète un âne et décide de l'éduquer. Il doit lui apprendre deux mots que nous appellerons x_1 et x_2. L'âne ne peut entendre que deux sons, soient y_1 et y_2 ces deux symboles acoustiques capables d'émouvoir l'âne. Éduquer l'âne revient à lui apprendre une fonction de décodage :

$$f \ : \ y \to x$$

Dans notre situation, il existe 4 décodeurs possibles représentés par les 4 fonctions différentes de l'ensemble des symboles acoustiques constitué des 2 symboles y_1 et y_2 dans l'ensemble des mots constitué des éléments x_1 et x_2. Dans la suite de cet exposé, nous allons montrer comment définir ces fonctions à l'aide de différents critères dont certains sont fondés sur la théorie de l'information.

2.1 Entropie et information mutuelle

Une mesure intuitive de l'incertitude associée à la venue d'un évènement $X = x$ est le nombre moyen de bits nécessaire pour coder l'apparition de x. Cette mesure est l'entropie de X :

$$H(X) = -\sum_x prob(X = x)\log(prob(X = x))$$

De la même façon, une mesure de l'incertitude sur l'apparition d'un évènement $X = x$ alors que l'évènement $Y = y$ s'est déjà produit est l'entropie conditionnelle $H(X/Y)$:

$$H(X/Y) = -\sum_{x,y} prob(X = x, Y = y) \log(prob(X = x/Y = y))$$

On définit l'information mutuelle entre X et Y par :

$$\begin{aligned} I(X;Y) &= H(X) - H(X/Y) \\ &= \sum_{x,y} prob(X = x, Y = y) \log \frac{prob(X = x, Y = y)}{prob(X = x)prob(Y = y)} \end{aligned}$$

Cette quantité représente l'espérance du log de la probabilité d'occurrence du couple (x,y) normalisée par les probabilités marginales. Cette quantité vaut 1 si les variables X et Y sont indépendantes et l'information mutuelle est nulle dans ce cas. Cette normalisation est importante car elle permet de s'affranchir de la probabilité souvent inconnue $Prob(X = x)$ qui représente la probabilité d'apparition d'un mot du langage et dont la valeur influe beaucoup sur les performances du décodeur comme nous le verrons plus tard.

Quand on utilise ce critère d'apprentissage, on veut minimiser $H(X/Y)$ qui représente l'incertitude sur X une fois observé Y. Cela revient à rendre maximum I(X;Y).

2.2 Le processus de reconnaissance

Une fois éduqué, l'âne a appris à se méfier. Il probabilise la source des mots de son maître et considère que les mots prononcés par le cultivateur sont les réalisations d'une variable aléatoire X, de même pour les sons qu'il entend ; soit Y cette deuxième variable. Il se constitue donc un décodeur probabiliste.

L'âne sait qu'il peut confondre x_1 et x_2 car il peut, lorsque l'un ou l'autre de ces deux mots a été prononcé, entendre y1 ou y2 ; heureusement dans des proportions différentes. Il représente ces proportions par deux densités discrètes de probabilité. L'une représente ce qu'il peut entendre quand on lui prononce x_1 et l'autre représente ce qu'il peut entendre quand on lui prononce x_2. Ces deux densités sont représentées par deux valeurs :

$$p_1 = prob(Y = y_1/X = x_1)$$

$$p_2 = prob(Y = y_2/X = x_2)$$

On a bien sûr :

$$prob(Y = y_2/X = x_1) = 1 - p_1$$

$$prob(Y = y_1/X = x_2) = 1 - p_2$$

Pour diminuer l'indéterminisme, l'âne utilise aussi la connaissance qu'il a de son maître. Il sait que le cultivateur prononce x_1 et x_2 avec des fréquences différentes. Notons au passage que l'âne a bien de la chance car cette connaissance ne nous est pas donnée pour tous les mots du français. Si nous pouvions donner à nos machines toute l'avoine nécessaire, elle finiront bien par se comporter comme des ânes [5].

5. Note d'un rapporteur.

À partir de ces valeurs, l'âne peut construire une table de deux lignes et deux colonnes contenant les quatre probabilités d'observer conjointement x_i et y_j.

On sait que cette probabilité se calcule à partir de la formule de Bayes :

$$prob(X = x_1, Y = y_1) = prob(X = x_1) \times prob(Y = y_1/X = x_1)$$

Illustrons ceci sur un exemple en posant $p_1 = 0.6$ et $p_2 = 0.8$; l'âne reconnaît mieux x_2. Soient 0.6 et 0.4 les probabilités que le maître prononce x_1 et x_2 respectivement.

	y_1	y_2
x_1	0.36	0.24
x_2	0.08	0.32

Ainsi, si l'âne entend y_1, il choisira x_1 car la valeur de la probabilité de cet évènement est 0.36 que l'on compare à 0.08, valeur de $prob(X = x_2, Y = y_1)$. De même, s'il entend y_2, il choisira x_2 ; la valeur de la probabilité de cet évènement est 0.32 que l'on compare à 0.24. On connaît maintenant le taux de reconnaissance de l'animal : $0.36 + 0.32 = 0.68$!

Supposons que l'âne redevienne sauvage. Devenu onagre, il ne connaît plus les valeurs $prob(X = x_1)$ et $prob(X = x_2)$. Il les invente et les suppose donc égales respectivement à 0.2 et 0.8.

Examinons maintenant ce que devient la fonction de décodage pour l'onagre :

	y_1	y_2
x_1	0.12	0.08
x_2	0.16	0.64

Ainsi l'âne sauvage répond toujours x_2 quelque soit le symbole y. Son taux de reconnaissance tombe donc à 40 % qui est la probabilité d'apparition de x_2.

2.3 Les critères d'apprentissage

Le nombre de coups de bâton est inversement proportionnel au taux de reconnaissance et il s'agit de trouver les critères pour estimer les paramètres $\theta = \{p_1, p_2\}$ afin de maximiser le taux de reconnaissance quand on ne connaît pas les probabilités *a priori* $prob(X = x_1)$ et $prob(X = x_2)$. Plusieurs critères vont être définis à partir de statistiques calculées sur un corpus d'apprentissage.

Pour effectuer l'éducation de l'âne, il faut lui présenter une liste de couples (a_i, b_i). Les a_i sont les mots du langage : x_1 ou x_2 ; les b_i sont les symboles acoustiques entendus par l'âne : y_1 ou y_2. Les 4 nombres de couples différents sont regoupés dans le tableau :

	y_1	y_2
x_1	N_1	N_2
x_2	N_3	N_4

On pose aussi : $N = N_1 + N_2 + N_3 + N_4$. N est la taille du corpus d'apprentissage.

On appellera $prob_\theta(X = a_i, Y = b_i)$ la famille des probabilités d'observer le couple (a_i, b_i). Ces probabilités dépendent du paramètre θ.

Estimation selon le maximum de vraisemblance

Pour ce critère, on cherche θ qui rende maximale la quantité : $\prod_{i=1}^{N} prob_\theta(X = a_i, Y = b_i)$ qui représente la vraisemblance du corpus d'apprentissage. Ceci revient à rendre maximum :

$$\sum_{i=1}^{N} \log(prob_\theta(X = a_i, Y = b_i))$$

Appelons ω_1 et ω_2 les probabilités *a priori* $prob(X = x_1)$ et $prob(X = x_2)$ respectivement. Les termes de la somme sont de 4 natures différentes :

	y_1	y_2
x_1	$\omega_1 p_1$	$\omega_1(1 - p_1)$
x_2	$\omega_2(1 - p_2)$	$\omega_2 p_2$

Supposons ω_1 et ω_2 fixés et annulons les dérivées partielles en p_1 et p_2 dans la définition de la vraisemblance. Il vient très facilement :

$$p_1 = \frac{N_1}{N_1 + N_2}, \quad p_2 = \frac{N_4}{N_3 + N_4}$$

Ceci signifie que les probabilités sont estimées à partir des fréquences d'observations.

On remarque aussi que le sous corpus permettant l'estimation de p_1 est disjoint du sous corpus permettant l'estimation de p_2. Dans ce premier critère, l'estimation de p_1 ne tient pas compte des paramètres N_3 et N_4 qui interviennent dans l'estimation de p_2 et inversement.

Estimation selon le maximum d'information mutuelle

Supposons ω_1 et ω_2 fixés. Pour ce critère, on cherche θ qui rende maximum l'information mutuelle entre X et Y. Celle ci s'estime de la façon suivante :

$$I(X;Y) = \sum_{i=1}^{N} prob(X = a_i, Y = b_i) \log(\frac{prob_\theta(X = a_i, Y = b_i)}{prob_\theta(X = a_i) prob_\theta(Y = b_i)})$$

On ne connaît pas les valeurs de $prob_\theta(X = a_i, Y = b_i)$. On suppose que le corpus d'apprentissage est représentatif. On cherche donc à rendre maximum la quantité :

$$\sum_{i=1}^{N} \log(\frac{prob_\theta(X = a_i, Y = b_i)}{prob_\theta(X = a_i) prob_\theta(Y = b_i)})$$

Le calcul des dérivées partielles selon p_1 et p_2 peut être laissé à MAPLE. Les valeurs de p_1 et p_2 qui les annulent tiennent compte des valeurs de N_1, N_2, N_3 et N_4, contrairement aux valeurs déterminées selon le critère du maximum de vraisemblance.

Estimation *a posteriori*

L'âne ne connaît pas le vecteur θ qui définit, en partie, son décodeur. Il considère donc le vecteur θ comme la réalisation d'une variable aléatoire qui suit une loi *a priori* et il cherche à estimer les paramètres de cette loi à partir du corpus d'apprentissage. Il faut donc trouver la loi de probabilité *a posteriori* connaissant le corpus. On peut montrer que si θ suit une loi uniforme – c'est à dire s'il n'y a pas de densités plus probables que d'autres – alors ce critère est le même que le maximum de vraisemblance. Par contre, si on a “une idée” des valeurs de θ, c'est à dire des contraintes sur θ, alors l'estimation *a posteriori* permet de tenir compte du corpus d'apprentissage et de ces contraintes. L'âne peut utiliser ce dernier critère pour s'adapter à son maître.

2.4 Pour en savoir plus

Différents documents m'ont permis la rédaction de ce bref tour d'horizon des méthodes statistiques d'estimation. On peut trouver dans les soixantes premières pages d'un livre de biologie [Atlan 92] une trés bonne introduction à la théorie de l'information. La fin du livre est tout aussi intéressante et traite des applications aux problèmes biologiques.

Une thèse de Carnegie-Mellon University consacrée à la reconnaissance de la parole, m'a particulièrement aidé [Brown 87]. Elle pose les problèmes d'estimation par maximum de vraisemblance et maximum d'information mutuelle en décrivant bien les intérêts de chacune de ces deux méthodes ainsi que leurs relations.

En ce qui concerne l'estimation *a posteriori*, le livre de G. Saporta contient une bonne introduction aux méthodes d'estimation [Saporta 90].

Le dessin de l'âne provient du journal “Le Monde” qui publie parfois des articles sur les ânes. Dans l'imagerie populaire, l'âne représente aussi bien la bétise, donc l'impossibilité d'apprendre, que l'intelligence qui est la faculté de s'adapter. On dresse un cheval mais l'âne s'éduque. Dans cette partie consacrée à l'apprentissage, il fallait parler de cet animal.

3 Définitions des chaînes de Markov

3.1 Processus stochastique

Soit un ensemble T, engendré par l'indice t, et une variable aléatoire q_t indicée par t, on appelle processus stochastique l'ensemble : $(q_t, t \in T)$. En général, T est dénombrable et représente le temps, le processus est alors dit *discret*.

3.2 Suite stochastique

Soit un système pouvant se trouver dans un ensemble dénombrable d'états. On pose :

$$\mathbf{S} = \{s_1, s_2, \dots, s_N, \dots\}$$

Entre les instants t et $t+1$, le système passe aléatoirement de l'état s_i à l'état s_j. On appellera Ω l'ensemble des transitions notées (i, j) entre 2 états. Pour représenter ce processus aléatoire, on définit la variable q_t de la façon suivante : $q_t = s_i$ signifie que le système est dans l'état s_i au temps t. Par définition l'ensemble $(q_1, q_2, \dots q_t, \dots)$ est une suite stochastique à ensemble discret d'états.

3.3 Chaîne de Markov

Une chaîne de Markov est une suite stochastique à ensemble discret d'états telle que :

- Suite du premier ordre

$$\begin{aligned} Prob(q_t = s_k / q_{t-1} = s_j, q_{t-2} = s_i, q_{t-3} = \dots) \\ = Prob(q_t = s_k / q_{t-1} = s_j) \end{aligned} \tag{2.1}$$

– Suite du deuxième ordre

$$Prob(q_t = s_k/q_{t-1} = s_j, q_{t-2} = s_i, q_{t-3} = ...) \\ = Prob(q_t = s_k/q_{t-1} = s_j, q_{t-2} = s_i) \tag{2.2}$$

En d'autres termes, l'évolution du système entre 2 instants t et $t+1$ ne dépend que de l'état de ce système au temps $t-1$ (ordre un) ou aux instants précédents $t-1$ et $t-2$ (ordre deux).

Une chaîne est dite homogène si les probabilités de transition ne dépendent pas du temps. On peut alors poser : $a_{jk} = Prob(q_t = s_k/q_{t-1} = s_j)$ et $a_{ijk} = Prob(q_t = s_k/q_{t-1} = s_j, q_{t-2} = s_i)$.

Si l'ensemble des états **S** est fini et possède N états alors on dit que la chaîne est finie. Dans la suite, nous nous limiterons aux chaînes de cette sorte. Dans ce cas, on a les contraintes suivantes :

– Chaînes du premier ordre

$$\sum_{k=1}^{N} a_{jk} = 1 \quad , \ 1 \leq j \leq N$$

– Chaînes du second ordre

$$\sum_{k=1}^{N} a_{ijk} = 1 \quad avec \quad (i,j) \in \Omega$$

Dans un processus de Markov d'ordre 2, la probabilité de la suite d'états :

$$Q = q_1, q_2, ..., q_T$$

est définie par :

$$Prob(Q) = \Pi_{q_1} a_{q_1 q_2} \prod_{t=3}^{T} a_{q_{t-2} q_{t-1} q_t}$$

où Π_i est la probabilité de l'état s_i à $t = 1$ et a_{ij} est la probabilité de la transition $s_i \rightarrow s_j$ au temps $t = 2$. Dans la suite de cet exposé on représentera un état s_i par l'entier i quand il n'y aura pas d'ambiguïté.

4 Application à la reconnaissance de la parole

Les modèles de Markov utilisés en reconnaissance de la parole constituent un cas très particulier de modèles de Markov. Les ensembles T et S sont finis et on utilise deux processus stochastiques. L'un est qualifié de *visible*, il est constitué d'une suite de variables aléatoires O_t dont les réalisations sont les trames de parole. Ce processus stochastique est gouverné par un deuxième processus stochastique. En général, on suppose que c'est une suite de Markov d'ordre 1 (HMM1) ou 2 (HMM2). Les états de ce processus sont cachés et représentent les variables aléatoires O_t d'où sont issues les observations o_t.

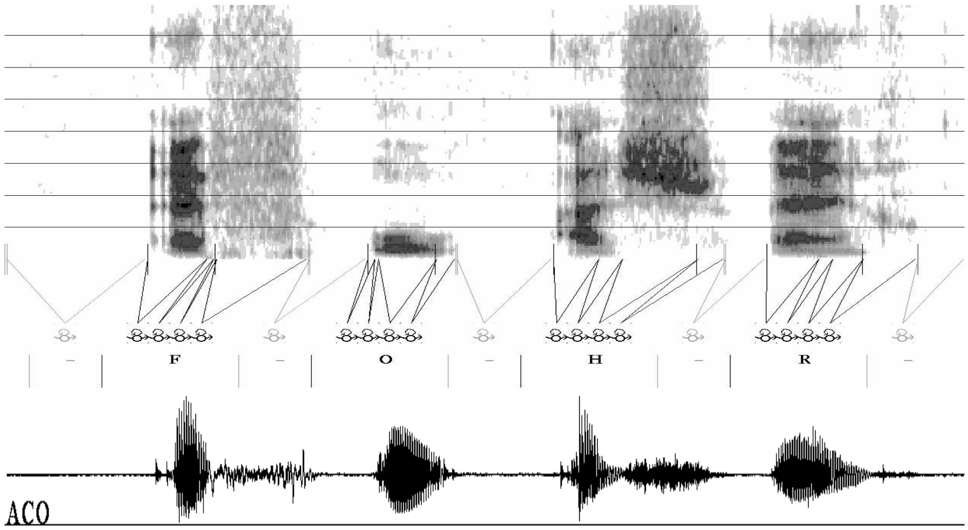


FIG. 2.2 – *Chaque état modélise un segment de parole*

L'exemple des urnes de Polya [Poritz 88] illustre bien les deux composantes d'un HMM. Dans cet exemple, nous avons plusieurs urnes contenant des boules de couleurs différentes. Ces urnes sont cachées derrière un voile. L'observateur ne voit pas de quelle urne est issue la boule qu'on lui présente. Le processus de tirage choisit l'urne dans laquelle se fera le tirage selon un processus Markovien ; le choix de l'urne suivante se fait uniquement en fonction de l'urne précédemment utilisée. Ce processus de tirage correspond au processus caché ; une urne correspond à un état. La boule de couleur issue de l'urne est la seule observation que l'on puisse faire sur ce processus particulier de tirage.

En reconnaissance de la parole à l'aide de HMM, on suppose que l'élocution peut être décomposée en une suite de segments dans lesquels chaque trame est issue du même processus aléatoire, c'est à dire possédant une moyenne et une variance qui ont été estimées pendant l'apprentissage. Dans la suite de ce chapitre, nous nous intéresserons à l'estimation selon le maximum de vraisemblance. Cet estimateur est efficace dans le cas d'un corpus exhaustif ; ce qui n'est jamais le cas dans notre pratique et justifie la constitution de corpus de plus en plus importants. Sur un corpus exhaustif, les variances des densités des HMM sont minimales. Un segment est modélisé par un état du processus de Markov caché. À chaque instant on ne sait pas de quel segment provient une trame car les densités associées aux états se recouvrent.

Dans la figure 2.2 on distingue les segments modélisés par chaque état de la chaîne de Markov. On représente l'alignement le plus probable entre la suite de trames représentant le signal de parole et la suite cachée d'états produite par la chaîne de Markov d'ordre deux. On distingue nettement que les premiers états de chacun des modèles ont capturé la partie vocalique tandis que les derniers états capturaient les parties consonantiques de la prononciation des lettres épelées du nom *FOHR*.

En modélisant ainsi les segments composant le signal de parole, on suppose que chaque trame est indépendante des trames voisines. Cette hypothèse ignore la réalité de l'articulation du conduit vocal mais a l'avantage de permettre la réalisation d'algorithmes d'apprentissage et de reconnaissance rapides.

Le problème de l'apprentissage sera d'estimer les paramètres des processus stochastiques à partir d'exemples d'élocutions. Le problème de la reconnaissance sera de reconstituer la suite de segments la plus probable connaissant la suite de trames observées.

4.1 Notations et définitions

Dans la suite nous appellerons :

- Q_t la suite de Markov finie d'ordre 1 ou 2, homogène, appelée *suite cachée*. Ce processus stochastique est entièrement défini par la donnée des probabilités initiales et de la matrice de transitions entre états. Dans le cas d'un processus d'ordre 1, la matrice (a_{ij}) donne la probabilité de transition entre les états s_i et s_j. Dans le cas d'un processus d'ordre 2, la matrice (a_{ijk}) donne la probabilité de transition entre les états s_i, s_j, s_k.

- O_t le processus stochastique visible. O_t prend ses valeurs dans R^d. La densité de chaque O_t est définie par la donnée des densités $b_i()$ représentant chacun des états s_i du processus de Markov caché. On modélise $b_i()$ par une somme pondérée de M lois normales $\mathcal{N}(\mu_{im}, \Sigma_{im})$ de moyenne μ_{im} et de matrice de covariances Σ_{im}.

$$b_i(O_t) = \sum_{m=1}^{M} c_{im}\mathcal{N}(O_t; \mu_{im}, \Sigma_{im}), \quad \text{avec} \quad \sum_{m=1}^{M} c_{im} = 1$$

Un modèle de Markov caché du deuxième ordre est défini par la donnée de :

- $\mathbf{S} = \{s_1, s_2, \ldots, s_N\}$, un ensemble fini de N états ;

- $\mathbf{A} = (a_{ijk})$ la matrice donnant les probabilités de transition entre états ;

- π le vecteur de probabilités initiales de la suite stochastique cachée ;

- $b_i(.)$ les lois des densités, souvent appelées abusivement *pdf*, associées aux états s_i du processus stochastique caché.

L'ensemble de ces paramètres $\lambda = \{S, A, \pi, b_i(.)\}$ constitue le modèle de Markov caché λ du deuxième ordre.

4.2 Probabilité d'un alignement temporel

Un alignement temporel entre une suite d'états et une suite de trames tel qu'il est montré figure 2.2 est une correspondance (état, trame). Sa probabilité est définie par :

$$Prob(Q, O) = \prod_t a_{q_{t-2}q_{t-1}q_t} b_{q_t}(o_t) \tag{2.3}$$

dans lequel le terme $a_{q_{t-2}q_{t-1}q_t}$ représente la probabilité de la contribution de la chaîne de Markov entre les instants t-2 et t (cas d'une chaîne d'ordre 2) qui définit la segmentation. Quant au terme $b_{q_t}(o_t)$ il représente la vraisemblance de la trame o_t conditionnée à la densité de l'état q_t.

Dans le cas des modèles de Markov gauche-droite décrits dans la figure 2.2, un alignement entre états et trames définit une segmentation.

Un grand intérêt des modèles de Markov cachés est qu'ils permettent lors de la reconnaissance une recherche de tous les alignements afin de retrouver la segmentation possédant la probabilité définie par l'équation 2.3 maximale. L'algorithme utilisé est *l'algorithme de Viterbi* qui est un algorithme de programmation dynamique [Forney 73].

5 Les algorithmes de reconnaissance et d'apprentissage

5.1 L'algorithme de Viterbi pour le premier ordre

Décrivons cet algorithme sur un exemple simple de modèle discret du premier ordre. Il est tiré de la thèse de A. Kriouile [Kriouile 90a] qui fut mon premier étudiant studieux dans ce domaine. Dans cet exemple, nous considérons un HMM1 "préhistorique" dans lequel chaque densité $b_i()$ est discrète, c'est à dire susceptible d'engendrer un nombre fini de symboles. Il y a 10 ans, tous les HMM utilisaient des densités discrètes afin d'assurer une vitesse raisonnable, sur les machines de l'époque, aux algorithmes d'apprentissage et de reconnaissance.

Soit le modèle de Markov du premier ordre de la figure 2.3, défini par la matrice de transitions entre états A telle que :

$$A = \begin{pmatrix} 0.3 & 0.5 & 0.2 \\ 0 & 0.3 & 0.7 \\ 0 & 0 & 1 \end{pmatrix}$$

et par la matrice B fixant les probabilités des 2 observations – notées a et b – sur chacun des trois états telle que :

$$B = \begin{pmatrix} 1 & 0 \\ 0.5 & 0.5 \\ 0 & 1 \end{pmatrix}$$

sans oublier le vecteur π donnant la distribution des probabilités initiales sur les trois états :

$$\pi = \begin{pmatrix} 0.6 \\ 0.4 \\ 0 \end{pmatrix}$$

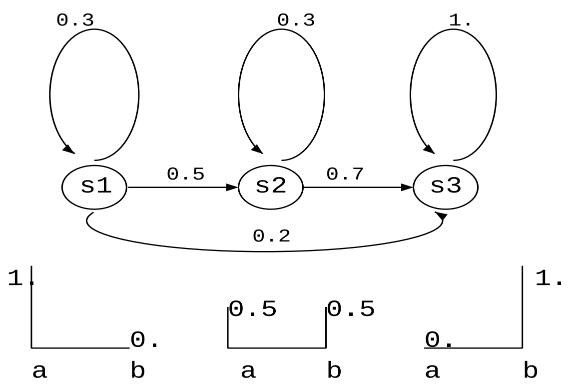


FIG. 2.3 – *Mon premier modèle de Markov*

Pour trouver le meilleur chemin de T états : $Q = (q_1, q_2, \ldots, q_T)$ associé à une suite de T observations : $O = (O_1, O_2, \ldots, O_T)$, on définit $\delta_t(j)$ qui est la probabilité du meilleur alignement état – observation entre les instants 1 et t et s'achevant sur s_j.

$$\delta_t(j) = \max_{q_t}[Prob(q_1, ...q_{t-2}, q_{t-1}, q_t = s_j, O_1, ..., O_t/\lambda)], \quad \begin{array}{l} 1 \quad \leq j \quad \leq N \\ 1 \quad \leq t \quad \leq T \end{array}$$

Une méthode récursive de calcul est donnée par la formule suivante :

$$\delta_t(j) = \max_{1 \leq i \leq N} [\delta_{t-1}(i) \cdot a_{ij}] \cdot b_j(O_t), \quad \begin{array}{lll} 1 & \leq j & \leq N \\ 1 & \leq t & \leq T \end{array} \tag{2.4}$$

avec l'initialisation suivante :

$$\delta_0(i) \quad = \quad \Pi_i, \qquad 1 \leq i \leq N$$

Le déroulement de l'algorithme est illustré par la figure 2.4.

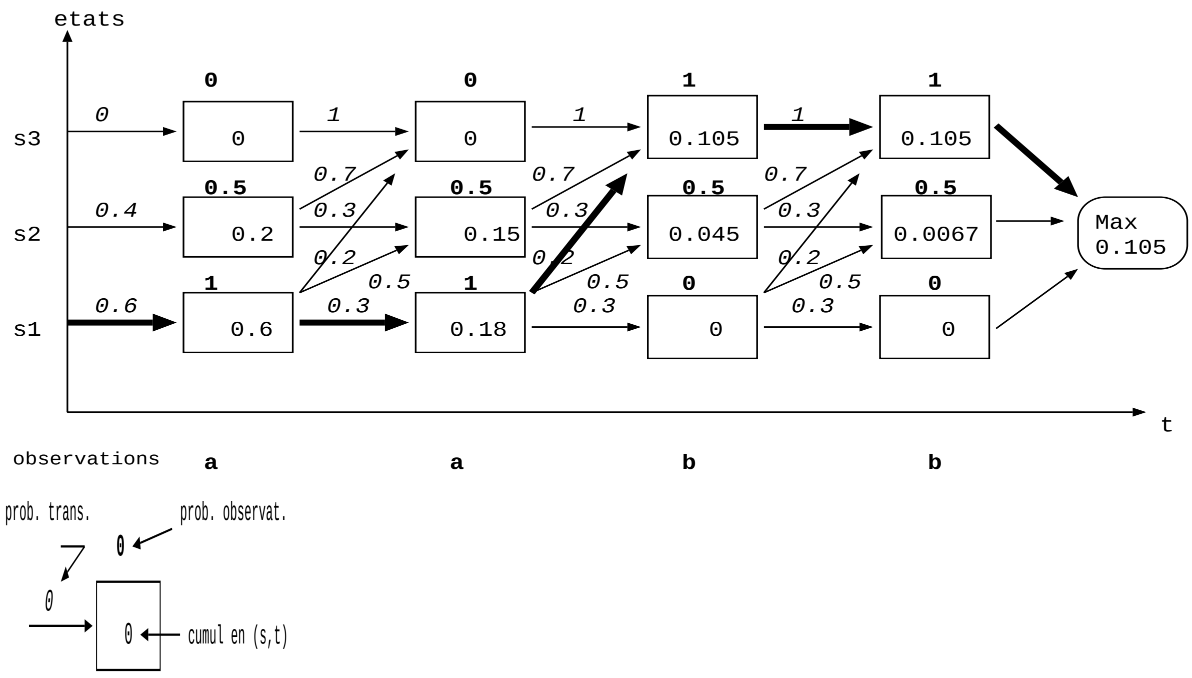


FIG. 2.4 – *Calcul de δ pour la suite d'observations* aabb

5.2 L'algorithme de Viterbi pour les suites de modèles

En reconnaissance de la parole continue, les frontières temporelles des unités à reconnaître ne sont généralement pas connues. On peut connecter les différents HMM dans un réseau plus vaste modélisant la suite d'unités à reconnaître. Le résultat de l'interconnexion reste un HMM sur lequel l'algorithme de Viterbi peut être utilisé pour retrouver la suite des états cachés correspondant à une élocution. La segmentation en unités est alors un produit secondaire de la reconnaissance. Un cas réel est présenté dans la figure 2.5. Dans cet exemple, on a représenté chaque phonème du français par un HMM2. La phrase prononcée est : « C'est en forgeant qu'on devient forgeron ». Une phonétisation possible de la phrase est donnée verticalement. La correspondance entre les phonèmes du français et leur représentation dans le programme d'alignement est donnée dans la table 2.1(d'après Calliope [Calliope 89]).

En niveaux de gris sont représentées les vraisemblances des trames calculées à l'aide des densités définissant les états. Comme ces vraisemblances sont normalisées pour que leur somme

				Consonnes				
[p]	paie	cl p	[t]	taie	cl t	[k]	quai	cl k
[b]	baie	vcl b	[d]	dais	vcl d	[g]	gai	vcl g
[m]	mais	m	[n]	nez	n	[ɲ]	gagner	
[f]	fait	f	[s]	sait	s	[ʃ]	chez	S
[v]	vais	v	[z]	zéro	z	[ʒ]	geai	Z
[ω]	ouais	w	[j]	yéyé	j	[l]	lait	l
[ʀ]	raie	R						
				Voyelles				
[i]	lit	i	[y]	lu	y	[u]	loup	u
[e]	les	e	[ø]	leu	2	[o]	lot	o
[ɛ]	lait	E	[ɔ]	lotte	O	[a]	là	a
[ə]	le	@	[ɛ̃]	lin	in	[ã]	lent	an
[õ]	long	on						

TAB. 2.1 – *Les phonèmes du français et leur représentations dans les alignements*

sur l'ensemble des modèles soit constante à chaque instant, on peut les interpréter comme des probabilités *a posteriori* des états connaissant les trames. On suppose alors les états *a priori* équiprobables.

5.3 L'algorithme *forward-backward* pour le premier ordre

Une des raisons du succès des HMM1 en reconnaissance de la parole est l'existence d'algorithmes d'apprentissage non-supervisé. L'algorithme de Baum-Welch, appelé aussi algorithme *forward-backward* [Baum 72], permet l'apprentissage selon le critère du maximum de vraisemblance. Cet algorithme est une variante de l'algorithme EM (Expectation-Maximization) d'estimation de variables cachées [Dempster 77]. À partir d'un modèle initial fixé *a priori*, l'algorithme permet de construire itérativement une suite de modèles markoviens associés à une unité – mot ou phonème – à l'aide d'un ensemble de répétition de cette unité.

Décrivons, encore par un autre exemple simple, le processus d'estimation selon le critère du maximum de vraisemblance.

Supposons donné le HMM1 discret représenté par la figure 2.6. Il ne nous est pas donné de connaître ce HMM1, par contre, nous connaissons sa topologie et nous avons collecté un corpus produit par ce HMM1. Ce corpus est représenté par la suite de répétitions construites à l'aide de l'ensemble des observations discrètes $\{a, b, c\}$ faites sur les états du HMM1 pour lequel 1 est le seul état initial.

$$TR = \begin{array}{llll} (b, & c, & a; & \\ c, & c, & a; & \\ b, & b, & b, & a; \\ b, & c, & b, & a; \\ c, & b, & b, & a; \\ c, & c, & b, & a) \end{array}$$

Cherchons à retrouver les matrices A et B de la page 22 à partir de TR et de la topologie connue du modèle. Considérons un modèle initial M_0 ayant la même topologie mais avec des

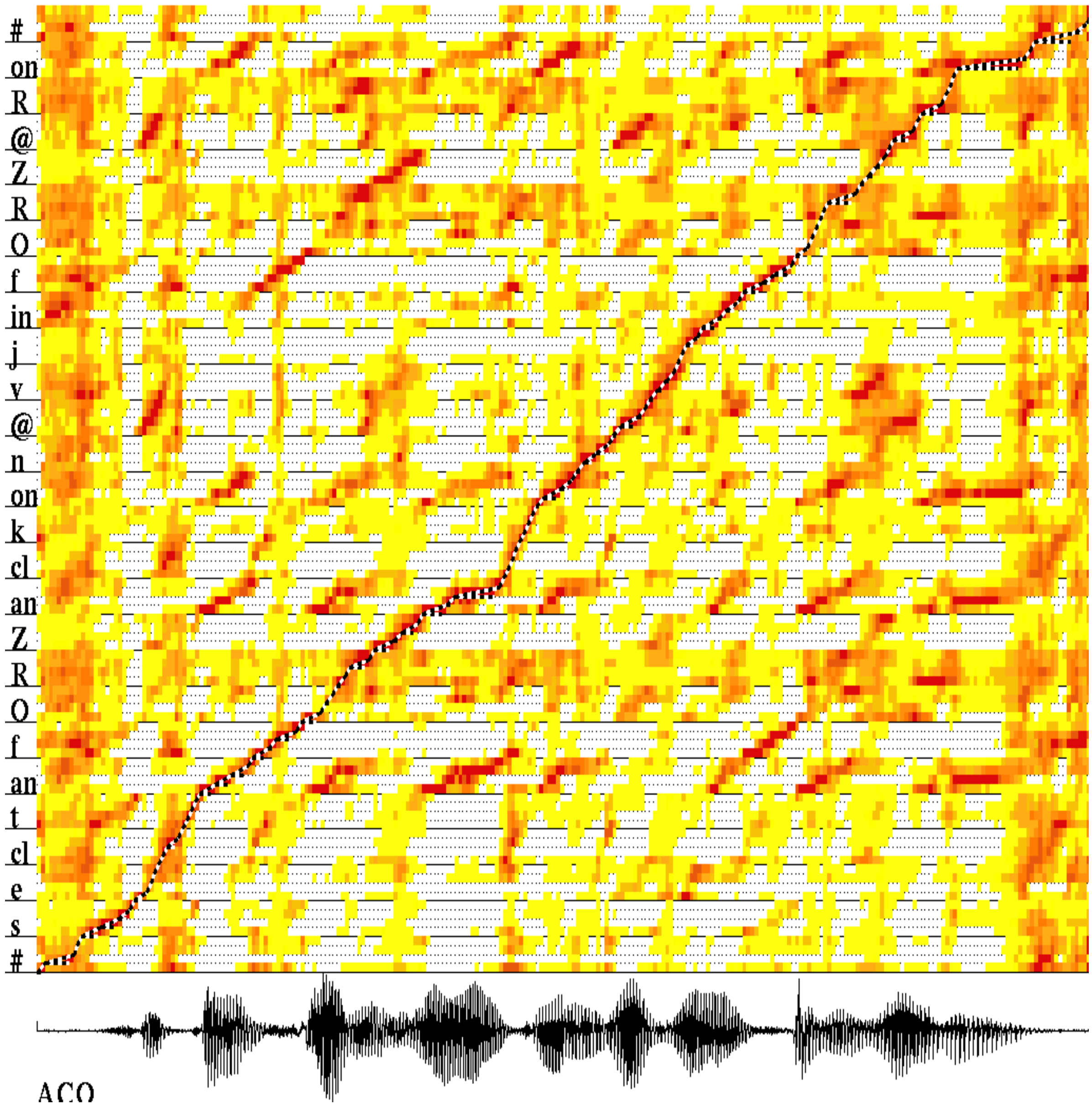


FIG. 2.5 – *Le meilleur chemin construit par l'algorithme de Viterbi détermine une segmentation en phonèmes dans la phrase :* «C'est en forgeant qu'on devient forgeron»

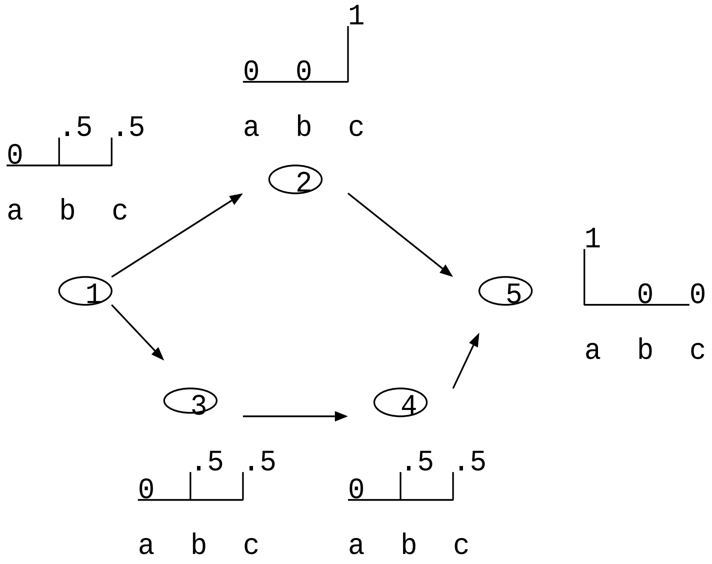


FIG. 2.6 – *HMM1 discret (5 états, 3 observations distinctes)*

densités de probabilités uniformes. Dans ce cas, les matrices initiales A_0 et B_0 sont égales à :

$$A_0 = \begin{pmatrix} 0 & .5 & .5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1. \\ 0 & 0 & 0. & 1. & 0 \\ 0 & 0 & 0 & 0 & 1. \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad B_0 = \begin{pmatrix} .33 & .33 & .33 \\ .33 & .33 & .33 \\ .33 & .33 & .33 \\ .33 & .33 & .33 \\ .33 & .33 & .33 \end{pmatrix}$$

Une itération de l'algorithme *forward-backward* détermine A_1 et B_1 à partir de A_0, B_0 et TR.

La détermination des suites d'états dans M_0 qui peuvent produire les élocutions de TR est simple. Les suites de 3 observations sont produites par un processus empruntant le chemin supérieur d'états 1,2,5 et les suites de 4 observations sont produites en empruntant le chemin inférieur 1,3,4,5.

Aprés ce qui a été dit au paragraphe 2.3 consacré aux critères d'apprentissage, nous admettrons que le critère d'estimation selon le maximum de vraisemblance revient à calculer des comptes – ou des fréquences – d'utilisation pour estimer les probabilités. Ainsi, nous réestimons A_0 en comptant combien de fois chaque transition est utilisée dans tout le corpus. Ainsi, on obtient la matrice suivante de comptes :

$$A1 = \begin{pmatrix} 0 & 2 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 \\ 0 & 0 & 0. & 4 & 0 \\ 0 & 0 & 0 & 0 & 4 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

En normalisant, on obtient :

$$A1 = \begin{pmatrix} 0 & 0.33 & 0.66 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1. \\ 0 & 0 & 0. & 1. & 0 \\ 0 & 0 & 0 & 0 & 1. \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

La normalisation est obtenue en rendant :

$$\sum_{j=1}^{n} A_1(i,j) \;=\; 1 \text{ où n est le nombre d'états.}$$

Le même travail est fait pour les observations réalisées à chaque état et on obtient $B_1 = B$.

Une nouvelle itération n'apporte pas de changements dans la définition des matrices A_2 et B_2 ; l'algorithme de réestimation a convergé.

Dans cet exemple, les matrices A_1 et A sont différentes parce qu'il y a 2 fois plus d'élocutions de 4 observations que de 3 observations. Pour obtenir des probabilités de transition (1,2) et (1,3) équiprobables ,il aurait fallu que TR comporte autant de suites de 4 symboles que de 3 symboles.

Il n'y a pas de suites d'états cachés dans cet exemple. La connaissance de la suite d'observations amène, sans ambiguïté, la connaissance de la suite d'états. Par contre, s'il existait une transition de l'état 2 sur lui-même, alors deux alignements seraient possibles pour les suites de 4 observations. Les comptes d'utilisation des transitions sont alors pondérés à l'aide des probabilités des alignements.

Sans avoir à considérer tous les alignements, l'algorithme *forward-backward* permet un calcul des comptes de toutes les transitions grâce aux calculs de deux quantités appelées traditionnellement α et β, ou aussi, *aller* et *retour*. Dans un HMM1, il suffit de calculer le compte de chaque transition (i,j) entre les instants t et $t+1$. Par extension au deuxième ordre, dans un HMM2, nous calculerons les comptes de transitions (i,j,k) entre les instants t et $t+2$. Ces calculs seront détaillés dans le paragraphe § 6.2.

6 Notre contribution : algorithmes du second-ordre

6.1 L'algorithme de Viterbi pour le deuxième ordre

La généralisation de cet algorithme a été proposée pour la première fois par Kundu *et al.* [Kundu 88] [Kundu 89]. En 1990, A. Kriouile [Kriouile 90c] donnait les formules de reestimation des HMM2.

L'extension de l'algorithme de Viterbi au deuxième ordre ne présente pas de difficultés majeures dans sa définition. On remplace simplement la référence à un état de l'ensemble **S** par une référence à un élément du produit cartésien **S x S**. La probabilité du meilleur alignement entre une suite partielle d'observations $o_1, o_2, \ldots, o_t$ et une suite d'états $q_1, ...q_{t-2}, q_{t-1} = s_j, q_t = s_k$ s'achevant par la transition $s_j \longrightarrow s_k$ entre les instants t-1 et t s'écrit :

$$\delta_t(j,k) = \max_{q_1,\ldots q_{t-2}} Prob(q_1, ...q_{t-2}, q_{t-1} = s_j, q_t = s_k, O_1, ..., O_t/\lambda), \quad \begin{array}{lll} 1 & \leq j & \leq N \\ 1 & \leq k & \leq N \\ 2 & \leq t & \leq T \end{array} \qquad (2.5)$$

Un calcul récursif peut se développer de la façon suivante :

$$\delta_t(j,k) = \max_{1\leq i\leq N}[\delta_{t-1}(i,j)\cdot a_{ijk}]\cdot b_k(O_t), \quad \begin{array}{lll} 1 & \leq j & \leq N \\ 1 & \leq k & \leq N \\ 3 & \leq t & \leq T \end{array} \tag{2.6}$$

A partir des initialisations suivantes aux temps 1 et 2 :

$$\begin{array}{rcll} \delta_1(i) & = & \Pi_i \cdot b_i(O_1), & 1\leq i\leq N \\ \delta_2(i,j) & = & \delta_1(i)\cdot a_{ij}\cdot b_j(O_2), & 1\leq i\leq N \\ & & & 1\leq j\leq N \end{array}$$

Pour éviter de faire ces initialisations fastidieuses, car le processus Markovien n'est d'ordre 2 qu'à partir de la troisième trame, on introduit deux états – notés 0 et 1 – dans lesquels aucune trame n'est consommée. Sans perte de généralité, on suppose que le processus débute sur l'état 2 au temps $t = 3$. On pose alors $a_{012} = 1$ et $\delta_2(0,1) = 1$; ce qui simplifie l'initialisation de l'algorithme.

6.2 L' algorithme *forward-backward* pour le second ordre

Pour estimer les paramètres d'un HMM2, nous définissons les nouvelles fonctions *forward* et *backward* sur lesquelles est basé l'algorithme de Baum-Welch étendu au deuxième ordre.

La fonction $\alpha_t(j,k)$ définit la probabilité d'avoir simultanément la suite d'observations partielle
$O_1, ..., O_t$ et la transition $s_j \longrightarrow s_k$ entre les instants $t-1$ et t.

$$\alpha_t(j,k) = Prob(O_1, O_2, ..., O_t, q_{t-1} = s_j, q_t = s_k/\lambda), \quad \begin{array}{lll} 2 & \leq t & \leq T \\ 1 & \leq j & \leq N \\ 1 & \leq k & \leq N \end{array}$$

Comme pour le premier ordre, $\alpha_{t+1}(j,k)$ peut être calculée à l'aide de $\alpha_t(i,j)$ où $s_i \longrightarrow s_j$ et $s_j \longrightarrow s_k$ sont deux transitions entre les instants $t-1$ et t d'une-part et t et $t+1$ d'autre-part.

$$\alpha_{t+1}(j,k) = \sum_{i=1}^{N}\alpha_t(i,j).a_{ijk}.b_k(O_{t+1}), \quad \begin{array}{lll} 2 & \leq t & \leq T-1 \\ 1 & \leq j & \leq N \\ 1 & \leq k & \leq N \end{array} \tag{2.7}$$

On définit aussi la fonction $\beta_t(i,j)$, comme étant la probabilité de la suite d'observations depuis $t+1$ à T, connaissant le modèle λ et la transition $s_i \longrightarrow s_j$ entre les instants t-1 et t.

$$\beta_t(i,j) = Prob(O_{t+1}, ...O_T/q_{t-1} = s_i, q_t = s_j, \lambda), \quad \begin{array}{lll} 2 & \leq t & \leq T-1 \\ 1 & \leq i & \leq N \\ 1 & \leq j & \leq N \end{array}$$

Tout comme pour le premier ordre, cette quantité peut être définie récursivement :

$$\beta_t(i,j) = \sum_{k=1}^{N} \beta_{t+1}(j,k) a_{ijk} b_k(O_{t+1}), \quad \begin{array}{lll} 2 & \leq t & \leq T-1 \\ 1 & \leq i & \leq N \\ 1 & \leq j & \leq N \end{array} \tag{2.8}$$

Etant donné un modèle λ et une suite d'observations O, on définit par $\eta_t(i,j,k)$ la probabilité de la transition $s_i \longrightarrow s_j \longrightarrow s_k$ entre les instants $t-1$ et $t+1$ pendant l'émission de la suite d'observations.

$$\eta_t(i,j,k) = Prob(q_{t-1} = s_i, q_t = s_j, q_{t+1} = s_k / O, \lambda), \quad 2 \leq t \leq T-1$$

On déduit facilement :

$$\eta_t(i,j,k) = \frac{\alpha_t(i,j) a_{ijk} b_k(O_{t+1}) \beta_{t+1}(j,k)}{P(O/\lambda)}, \quad 2 \leq t \leq T-1 \tag{2.9}$$

Tout comme dans le premier ordre, nous pouvons définir les quantités $\xi_t(i,j)$ et $\gamma_t(i)$:

$$\xi_t(i,j) = \sum_{k=1}^{N} \eta_t(i,j,k) \tag{2.10}$$

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) \tag{2.11}$$

$\xi_t(i,j)$ représente la probabilité *a posteriori* que le processus stochastique effectue la transition $s_i \rightarrow s_j$ entre les instants $t-1$ et t connaissant toute la phrase. Ces probabilités sont représentées en niveaux de gris sur la figure 2.7 pour la phrase d'apprentissage « *et le type de musique jouée* » pour les modèles de chacun des phonèmes issus de la transcription orthographique. On distingue ce qui sera le meilleur chemin trouvé par l'algorithme de Viterbi.

$\gamma_t(i)$ représente la probabilité *a posteriori* que le processus soit à l'état i au temps t connaissant toute la phrase.

Arrivé à ce point, pour obtenir des nouvelles estimées au sens du maximum de vraisemblance des HMM2, deux façons de normaliser sont à notre disposition : une façon qui transformera les modèles en HMM1 et une autre qui les gardera en HMM2.

La transformation en HMM1 se fait en moyennant les comptes $\eta_t(i,j,k)$ sur tous les états i visités au temps $t-1$. La quantité :

$$\eta_t^1(j,k) = \sum_{i=1}^{N} \eta_t(i,j,k) \tag{2.12}$$

est le compte classique de transitions entre 2 états d'un HMM1 entre les instants t et $t+1$.

Finalement l'estimée du premier ordre de $\overline{a_{ijk}}$ au sens du maximum de vraisemblance s'écrit donc :

$$\overline{a_{ijk}} = \frac{\sum_t \eta_t^1(j,k)}{\sum_{k,t} \eta_t^1(j,k)} = \frac{\sum_{i,t} \eta_t(i,j,k)}{\sum_{i,k,t} \eta_t(i,j,k)} \tag{2.13}$$

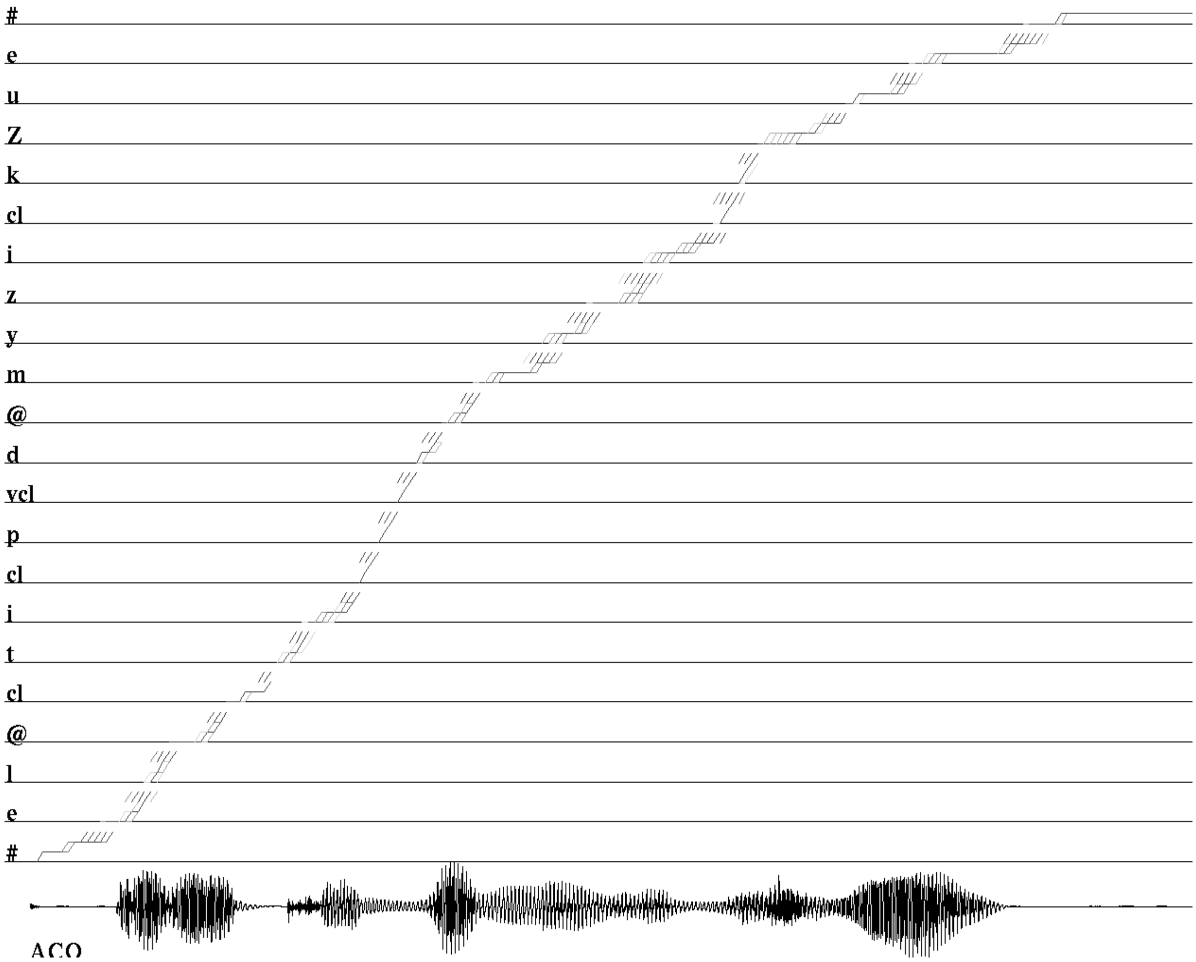


FIG. 2.7 – *Probabilités a posteriori des transitions connaissant toute la phrase : « et le type de musique jouée »*

Cette valeur est indépendante de i et peut s'écrire $\overline{a_{jk}}$.

L'estimée du second ordre de $\overline{a_{ijk}}$ au sens du maximum de vraisemblance est donnée par la formule :

$$\begin{aligned} \overline{a_{ijk}} &= \frac{\sum_t \eta_t(i,j,k)}{\sum_{k,t} \eta_t(i,j,k)} \\ &= \frac{\sum_{t=1}^{T-2} \eta_{t+1}(i,j,k)}{\sum_{t=1}^{T-2} \xi_t(i,j)} \end{aligned} \tag{2.14}$$

Dans un HMM2 continu, tel qu'il est défini dans le paragraphe 4.1, les observations sont issues de densités définies sous la forme d'un mélange de loies normales. Au sens du maximum de vraisemblance, les estimées des moyennes et matrices de covariances sont données par les formules :

$$\overline{\mu_i} = \frac{\sum_t \gamma_t(i) O_t}{\sum_t \gamma_t(i)} \quad (2.15)$$

$$\overline{\Sigma_i} = \frac{\sum_t \gamma_t(i)(O_t - \mu_i)(O_t - \mu_i)^t}{\sum_t \gamma_t(i)} \quad (2.16)$$

6.3 Complexité et implémentation

On appellera dans la suite N_{inp} le facteur de branchement d'entrée moyen du réseau formé de N états définissant la topologie d'un HMM. Typiquement dans le cas d'un réseau gauche-droite, où tous les états sont visités[6], cette valeur vaut deux. Le calcul de α et β dans le cas du deuxième ordre demande $N \times N_{inp} \times N_{inp} \times T$ opérations alors qu'il ne demandait que $N \times N_{inp} \times T$ dans le cas du premier ordre. Dans notre cas, il y a donc, *grosso modo* un facteur de 2 en termes d'espace mémoire et temps de calcul entre les performances comparées des HMM1 et des HMM2. En pratique, nous avons noté un accroissement de seulement 50 % puisque le système passe la moitié de son temps à calculer des vraisemblances de trames en supposant des mélanges de lois normales possédant des matrices de covariances pleines.

Afin de réduire le nombre d'opérations dans le calcul de α et β nous avons créé un ensemble de listes et de tables pour accélérer les accès aux éléments non nuls qui interviennent dans les calculs. La description de ces accès va illustrer ce paragraphe sur la complexité.

Calcul de α

Pour le calcul de α dans l'équation 2.7, il faut retrouver rapidement à partir du couple (j, k) l'ensemble des transitions (i, j, k) telle que a_{ijk} soit non nul. On utilise une table qui à chaque état, associe l'ensemble des transitions entrantes. Cette table possède N entrées. Chaque entrée pointe sur une liste de $N_{inp} \times N_{inp}$ éléments comme illustré ci-dessous :

$$k \rightarrow (i_0, j_0, k, a_{i_0 j_0 k}), (i_1, j_1, k, a_{i_1 j_1 k}), \ldots$$

Notons que le calcul de η dans l'équation 6.2 utilise les mêmes accès.

Calcul de β

Le calcul de β dans l'équation 2.8 est accéléré quand on accède facilement aux transitions (i, j, k) à partir d'une paire d'états (i, j) donnée :

$$(i, j) \rightarrow (i, j, k_0, a_{ijk_0}), (i, j, k_1, a_{ijk_1}), \ldots$$

Cette table possède $N \times N_{inp}$ entrées et chaque entrée pointe sur une liste de N_{inp} éléments.

Représentation des matrices α et β

La matrice α a une taille maximale de $N \times N \times T$ réels. Elle est calculée pendant la phase aller (*Forward* en anglais). Pendant la phase retour, elle ne sert qu'aux calculs des comptes représentés par $\eta_t(i, j, k)$. Après un minimum de calculs supplémentaires, nous l'utilisons pour stocker les valeurs de $\xi_t(i, j)$ en vue d'un affichage graphique comme dans la figure 2.14.

6. Des essais effectués sur des modèles où les états pouvaient être sautés nous ont montré que cette topologie dégradait les performances de reconnaissance. À l'heure actuelle, cette topologie a disparu des systèmes de presque toutes les équipes.

La matrice β est moins grosse puisque le calcul de β au temps t se fait en utilisant les seules valeurs de β au temps $t+1$.

Dans le cas de modèles gauche-droite, sans saut et avec une boucle sur chaque état, les probabilités d'un grand nombre de transitions sont nulles et les matrices sont très creuses car les seules probabilités non nulles sont associées aux transitions du type (i,i) et $(i,i+1)$. On ordonne ces transitions à l'aide de l'ordre lexicographique. Le rang d'une transition (i,j) est $i+j$. La transformation :

$$(i,j) \rightarrow i+j \quad , i=1,N, \quad j=i,i+1$$

permet de retrouver les probabilités associées aux transitions (i,j) en permettant leur rangement dans un tableau à une dimension. On ramène ainsi la taille des matrices à $N \times 2 \times T$ pour α et $N \times 2$ pour β. Nous en resterons là.

La mise en œuvre de ces deux algorithmes nécessite la résolution de plusieurs problèmes : la représentation des probabilités par des nombres réels infinitésimaux (de l'ordre de 10^{-100}) et la réduction du temps de calcul par des techniques de recherche en faisceaux (élagage en français québécois [Lacouture 95]). Ces problèmes sont maintenant bien documentés. La thèse de Brown [Brown 87] décrit comment implanter un seuillage dans les calculs de α et β. Ce seuillage accélère grandement les calculs et permet des expériences qui n'étaient pas possibles auparavant.

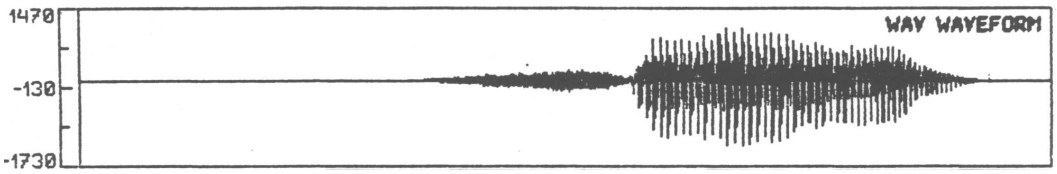
6.4 Algorithmes sous-optimaux de reconnaissance

Nous venons de voir que l'algorithme de Viterbi permet de retrouver le meilleur alignement entre une suite d'états et d'observations à l'intérieur d'un HMM représentant une unité à reconnaître. On peut constituer un réseau plus important en connectant plusieurs HMM de mots ou de phonèmes entre-eux, ce qui définit un nouveau HMM. Les règles de connexion se font à l'aide d'une grammaire définissant les transitions possibles entre unités : soit en séquence, soit en parallèle. On applique alors l'algorithme de Viterbi sur ce grand HMM. Le meilleur alignement dans ce réseau détermine une suite de mots ou de phonèmes.

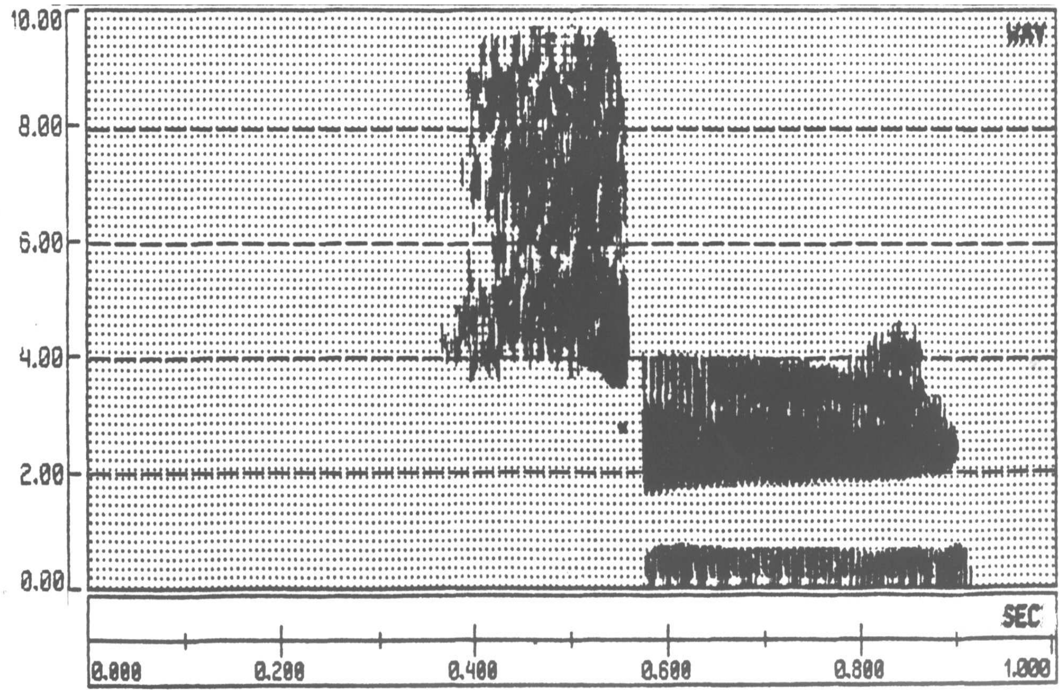
Recherche des N meilleurs chemins

L'algorithme de Viterbi a rapidement montré à ceux qui l'utilisaient que la solution optimale au sens stochastique du terme était fort singulière pour un expert en lecture de spectrogramme. Certains états de HMM recevaient de la parole qui aurait dû être modélisée par les états voisins. L'alignement entre les états et les trames acoustiques n'avait pas de sens phonétique. Ainsi une explosion de /t/ peut s'aligner avec un /s/ comme le montre l'exemple suivant.

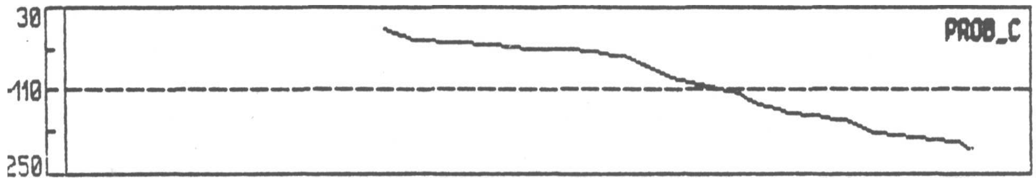
Exemple d'alignement singulier



Un locuteur nord américain prononce la lettre « C ». Le signal a été numérisé à 20 kHz. On effectue tout d'abord une détection parole/non parole pour localiser le début et la fin de l'élocution.

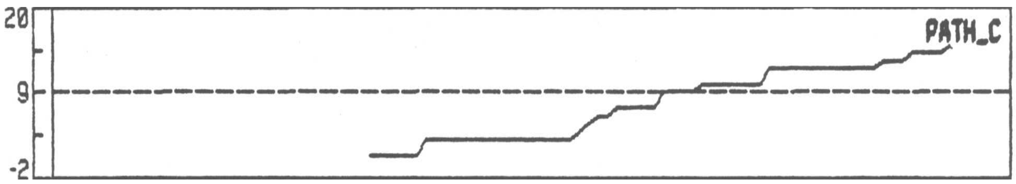


L'échelle verticale du spectrogramme s'échelonne entre 0 et 10 kHz. Le signal temporel et le spectrogramme montrent clairement les deux phonèmes : /s/ et /ɪ/.



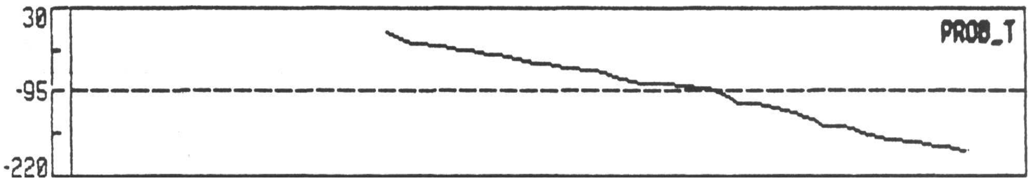
L'algorithme de Viterbi est utilisé sur deux HMM1 : celui associé à la lettre C (la bonne) et celui associé à la lettre T (la mauvaise qui est reconnue). Pour chaque modèle λ, on trace la courbe $PROB\_\lambda$ qui représente la quantité $\log(\delta_t(j))$ comme une fonction de t pour le meilleur chemin ; l'échelle verticale s'échelonne entre 30 et -250.

Sur le modèle de «C» la valeur de $\log(\delta_t(j))$: probabilité de l'alignement partiel (cf. eq. 2.6) diminue régulièrement. La valeur finale se situe au voisinage de -230.

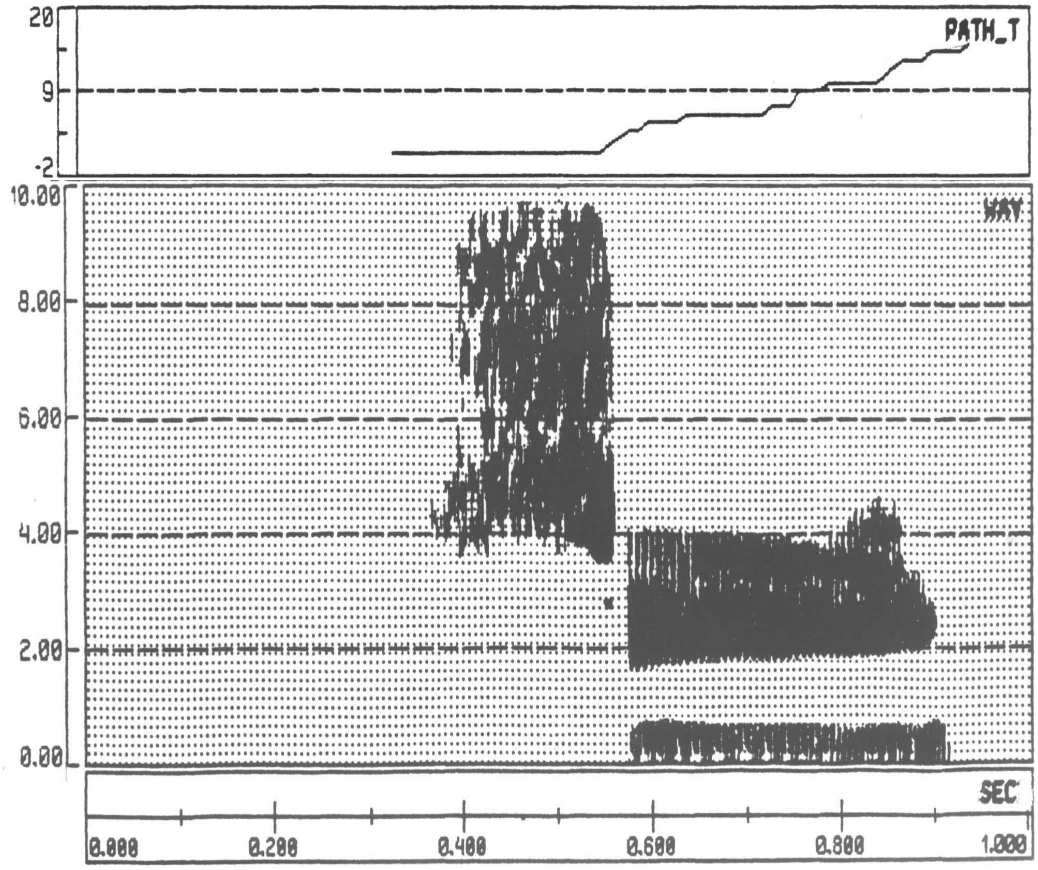


On représente aussi la suite des états du meilleur chemin par $PATH\_\lambda$; dans ce cas, l'échelle verticale représente les 15 états d'un modèle.

La sous-figure *PATH_ C* représente la suite des états de l'alignement complet. Les deux premiers états du modèle ont capté la partie fricative du premier /s/.



On utilise maintenant l'algorithme de Viterbi sur le modèle de « T ». La valeur finale de $\log(\delta_t(j,k))$ se situe aux alentour de -210 soit une valeur plus grande.



Le premier état du modèle du « T » chargé de modéliser l'explosion de l'occlusive /t/ a servi à aligner le segment de friction du /s/. Ceci ne pourrait pas avoir lieu si une contrainte de durée avait été associée sur cet état qui doit modéliser un segment court. On voit sur cet exemple que cette opération est très mal réalisée par un HMM1.

Modifier cet algorithme est une chose tentante. On peut ajouter facilement des durées minimales et maximales pour chaque état, mais la solution n'est plus optimale pour la suite de Markov. On se rend compte que la modélisation stochastique d'ordre un montre là une de ses faiblesses. Utiliser des modèles d'ordre deux permet d'éliminer certains alignements singuliers puisque les HMM2 modélisent explicitement la durée des courts séjours mais le problème demeure toujours.

Nous n'avons pas été originaux en décidant d'écrire l'algorithme de recherche des N meilleures solutions publié par R. Schwartz [Schwartz 90] afin d'utiliser les différentes segmentations en mots pour les re-classer en fonction d'autres sources d'informations.

Pourtant on retrouve dans la thèse de A. Kriouile [Kriouile 90c] un paragraphe consacré à la recherche des 3 meilleures solutions dans l'algorithme de Viterbi mais ce paragraphe concluait en disant que ces 3 meilleures solutions donnaient la plupart du temps la même segmentation en mots. R. Schwartz, au même moment, les considérait équivalentes et en considérait d'autres jusqu'à obtenir des segmentations différentes. Mais il n'est pas déshonorant de se faire doubler par R.Schwartz !

Cet algorithme nous a permis de créer des systèmes de reconnaissance hiérarchiques dans lesquels la segmentation en mots fournie par les HMM était contestée ou validée par des perceptrons multi-couches. A l'heure actuelle cette technique de *rescoring* fait encore l'objet de thèses.

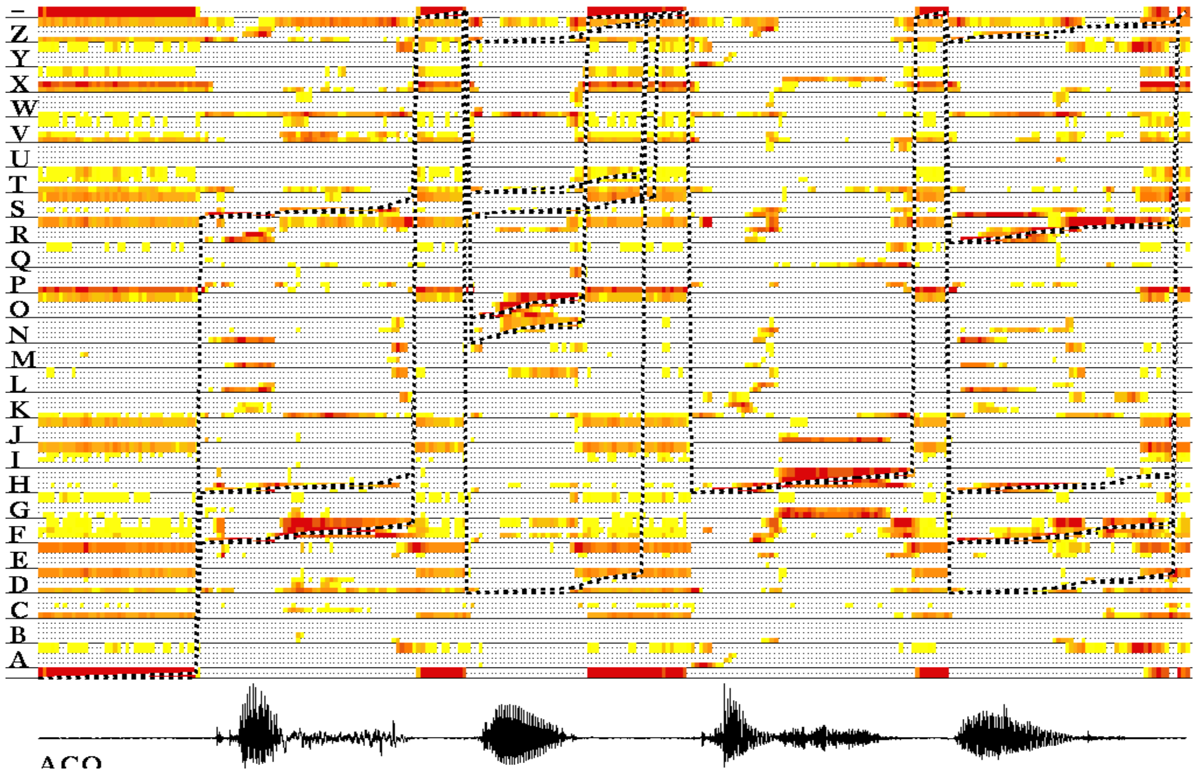


FIG. 2.8 – *Recherche des N meilleurs chemins dans le signal montré figure 2.2*

Algorithme de Viterbi-Bloc

L'algorithme de Viterbi-Bloc que nous avons proposé, provient de notre désir d'utiliser pendant la reconnaissance des critères basés sur une notion d'optimalité locale plutôt que globale [Haton 74]. En procédant localement, les algorithmes mettent en œuvre des structures moins volumineuses comme nous le verrons plus loin. Rappelons que l'algorithme de Viterbi, qui est un algorithme de programmation dynamique, construit et préserve à chaque instant, tous les chemins partiels s'achevant sur chaque état du HMM alors qu'un seul chemin, issu de l'état final, sera utilisé pour retrouver la suite des états empruntés.

Bizarrement, c'est par le biais des journées d'étude de la parole de Montréal dans lesquelles il avait été publié [Kriouile 90b] qu'il a été connu et référencé dans des travaux de Bell-Northern [Boulianne 94]. Pourtant, il avait été publié à ICASSP la même année [Kriouile 90c].

Cet algorithme effectue une segmentation simultanément avec la reconnaissance et donne une réponse en temps réel au fur et à mesure de la prononciation de la phrase sans attendre la fin de celle-ci. Le principe général consiste à utiliser une fenêtre glissante (*cf.* figure 2.9) de N trames de parole dans laquelle une décision est prise localement. Le processus de reconnaissance de Viterbi-bloc peut être décrit de la façon suivante.

`ViterbiBloc (O, T, M, K ) :`

entrée : $O_1, O_2, \ldots, O_T$ la suite de T trames représentant l'élocution

sortie : $m_1, m_2, \ldots, m_K$ la suite de K modèles qui maximise le critère choisi

I $\leftarrow$1 (I est le début de la fenêtre glissante de longueur N)
tant que $I + N \leq T$ **faire**
1) exécution de l'algorithme de Viterbi entre I et $I + N$ pour chaque HMM de référence
2) mémorisation des probabilités d'alignement dans la fenêtre de comparaison pour chaque modèle
3) comparaison de ces probabilités
4) choix du segment correspondant au modèle favorable (soit t_1 son extrémité) au sens d'un critère heuristique de choix prédéfini
5) éliminer tous les autres alignements
6) I $\leftarrow I + t_1 + 1$
fin-tant-que

Algorithme 2.1: Algorithme de Viterbi-bloc

Étape n° 1 : on applique l'algorithme de Viterbi entre les trames $[I, I+1]$, $[I, I+2]$, ... $[I, I+N]$.

Étape n° 2 : on sauvegarde les N probabilités d'alignement de chaque fenêtre.

Étape n° 3 : on étudie la dynamique de ces probabilités pour faire apparaître le meilleur modèle ainsi que la meilleure trame dans ce modèle. Soit t_1 cette trame. Les comparaisons sont faites à l'aide de critères heuristiques pour effectuer le choix du chemin localement optimal.

Étapes n° 4, 5 et 6 : on retient un modèle et on élimine les autres. Le processus continue à partir de la trame $I + t_1 + 1$.

De par la nature même des choix faits dans l'étape n° 3, l'algorithme de Viterbi-bloc est sous-optimal en ce sens qu'il ne garantit pas de retenir ainsi le chemin globalement optimal pour une phrase complète, c'est à dire, maximisant la probabilité *a posteriori* de la suite de modèles connaissant l'évidence acoustique. Ces comparaisons peuvent avantageusement utiliser des marques de segmentation qu'un niveau segmental pourrait avoir préalablement placées [André-Obrecht 88].

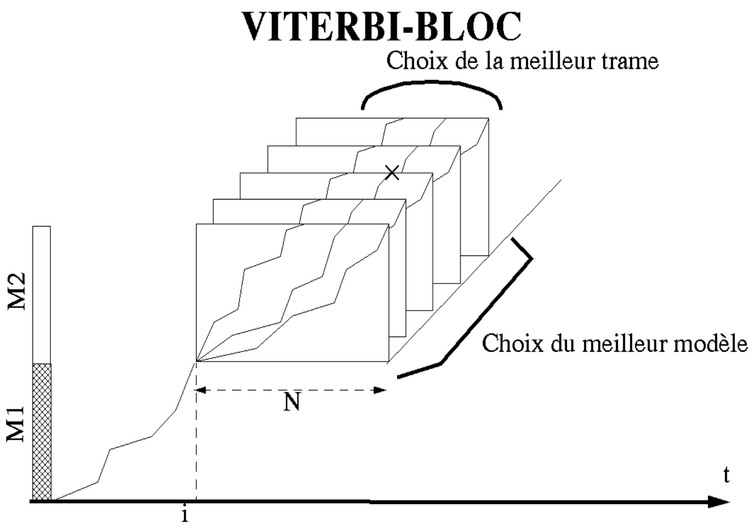


FIG. 2.9 – *L'algorithme de Viterbi Bloc (d'aprés [Kriouile 90c])*

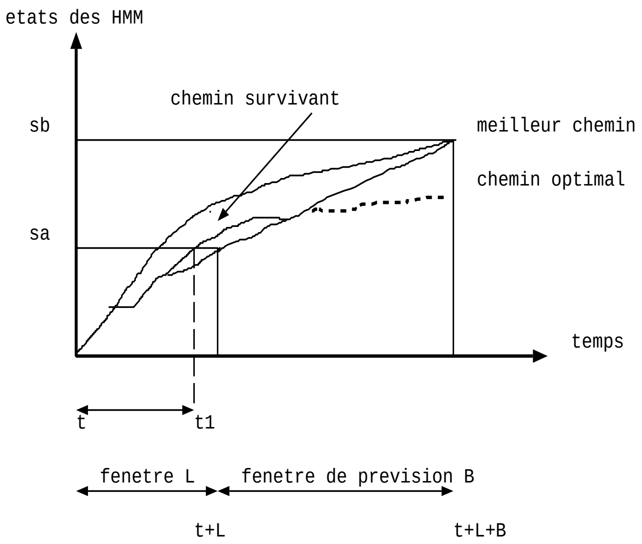


FIG. 2.10 – *Sélection du chemin optimal (d'aprés [Boulianne 94])*

L'algorithme utilisé en étiquetage par l'équipe de l'INRS-télécommunications et Bell Northern procède localement pour donner une transcription en phonèmes d'une phrase dont on connaît la transcription orthographique. Il utilise deux fenêtres : l'une, dite fenêtre de construction, de longueur L analogue à la fenêtre glissante de Viterbi-bloc et l'autre de longueur $L+B$, dite fenêtre de prévision, dans laquelle un ou deux phonèmes suivants sont recherchés (cf. figure 2.10). Il n'y a plus de comparaison de probabilités le long de l'arête supérieure de la fenêtre.

Supposons la segmentation effectuée jusqu'à la trame t. On appelle s_a l'état final du phonème suivant pour lequel on cherche les frontières temporelles. On sait seulement que ce phonème débute sur la trame t. On applique l'algorithme de Viterbi entre les trames $[t, t+1]$, $[t, t+2]$, ... $[t, t+L]$. Pour chaque trame de sortie $t+1$, $t+2$ ... $t+L$, alignée sur l'état final s_a, on poursuit le processus de construction à l'aide des phonèmes suivants jusqu'à la trame $t+L+B$. Soit s_b l'état possédant le meilleur score sur la trame $t+L+B$. A cet endroit, rien n'assure que l'on soit sur le chemin optimal donné par un classique algorithme de Viterbi. Mais, si la fenêtre de prévision est suffisamment grande –au plus, elle est égale à la portion restante de la phrase– on retrouvera le chemin optimal en reculant dans cette fenêtre car l'hypothèse expérimentale qui a motivé la spécification de l'algorithme de Viterbi-bloc est que le meilleur chemin s'achevant au point $(s_b, t+L+B)$ est optimal entre t et $t+L$. Finalement, on se retrouve à l'instant t_1 dans l'état s_a. t_1 est l'extrémité recherchée du premier phonème qui s'achève sur l'état s_a. C'est à partir de la trame t_1 qu'on continue le processus de segmentation. Les auteurs ont déterminé expérimentalement les tailles des fenêtres de construction et de prévision pour conserver le chemin optimum dans la construction des chemins partiels. Les durées de 0.6 secondes pour B et de 4 secondes pour $L+B$ conviennent dans le cas du décodage acoustico-phonétique.

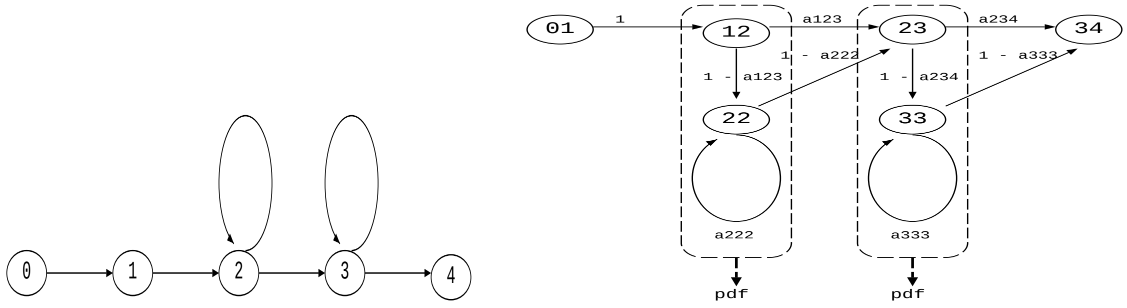
Puisque la recherche du meilleur chemin de décodage n'utilise qu'un critère d'optimalité locale, la reconnaissance n'attend plus la fin de la phrase, ainsi ces algorithmes peuvent délivrer une réponse au fur et à mesure de l'avancement du processus. Il n'est plus nécessaire de stocker des informations relatives à la totalité de la phrase. La place mémoire nécessaire à l'exécution ne dépend que de la taille de la fenêtre de construction, ce qui rend ces algorithmes intéressants pour la reconnaissance de longues phrases. Nous utilisons une variante de cet algorithme pour la construction d'un graphe de mots dans notre machine à dicter.

6.5 Modélisation de la durée au niveau des états

Chaque modèle de Markov du deuxième ordre possède un homologue du premier ordre sur le produit cartésien des états **S x S** mais la transformation augmente le nombre des états qui doivent être liés si on veut garder une estimation correcte des densités associées à chacun d'eux. Par exemple, la figure 2.11(b) montre le modèle du premier ordre équivalent au modèle du second ordre représenté dans la figure 2.11(a). Dans le modèle du premier ordre, les états dans une colonne partagent la même densité.

Dans un HMM2, la durée de séjour dans un état est gouvernée par deux paramètres : la probabilité d'entrer dans l'état et de n'y rester qu'une trame, et la probabilité de séjourner au moins deux trames. Cette dernière étant représentée par une loi géométrique comme le montrent les équations suivantes :

$$\begin{aligned} d_j(0) &= 0 \\ d_j(1) &= a_{ijk}, \quad i \neq j \neq k \\ d_j(n) &= (1 - a_{ijk}) \cdot a_{jjj}^{n-2} \cdot (1 - a_{jjj}), \quad n \geq 2 \end{aligned}$$



(a) original du 2e ordre (b) équivalent du 1er ordre

Fig. 2.11 – *Passage de l'ordre 2 à l'ordre 1 pour un HMM2*

Il est intéressant de noter qu'un HMM2 converge vers une topologie où les états sont dupliqués suivant la durée du segment qu'ils modélisent. Les états qui sont associés à des segments de durée importante sont automatiquement dupliqués et partagent la même densité, comme par exemple, l'état central du HMM2 du phonème /oy/ – comme dans *oily* – de l'exemple suivant. Par contre, les états transitoires ne sont pas dupliqués.

La technique qui consiste à dupliquer un état pour l'obliger à modéliser un segment long a déjà été utilisée par plusieurs équipes [Russell 87, Levinson 86] et peut être vue comme un moyen d'utiliser un modèle de Markov d'ordre supérieur. Un avantage indéniable des HMM2 est d'automatiser en partie ce traitement.

Exemple tiré de *TIMIT*

Les HMM2 qui modélisent les phonèmes dans *TIMIT* [7] ont 3 états. Ils sont indépendants du contexte. La figure 2.12 représente les densités de probabilité de séjourner sur l'état central en fonction du nombre de trames. Chaque trame dure 8 ms.

Deux courbes sont tracées, une pour le phonème anglais /oy/ et une autre pour le phonème /t/. Ce dernier phonème est un phonème court. Sa distribution est concentrée à gauche, tandis que /oy/ qui est une voyelle longue, voit sa distribution décalée à droite. Pour le HMM2 modélisant cette voyelle, la probabilité de ne rester qu'une fois dans l'état central est nulle ; cet état est dupliqué, ce qui n'est pas le cas du /t/. Dans un HMM1, il est impossible d'avoir un tel décalage entre densités car toutes les courbes ont leur pic en 1 puisque modélisées par une loi géométrique de raison inférieure à 1.

6.6 Modélisation de la durée au niveau supra-segmental

Par niveau supra-segmental, nous entendons suite de modèles. En général la modélisation de la durée se fait sur la base d'unités de reconnaissance. Le modèle de durée utilisé change quand le processus change d'unité. Il n'y a pas de traitement aux frontières d'unités et la corrélation entre les durées d'unités voisines n'est pas prise en compte.

Des études ont porté sur la modélisation de la vitesse d'élocution. Suaudeau [Suaudeau 93] définit un facteur d'élocution pour modifier dynamiquement la durée des unités à reconnaître en fonction des unités reconnues.

7. On peut se reporter page 6 pour se remémorer les caractéristiques du corpus *TIMIT*.

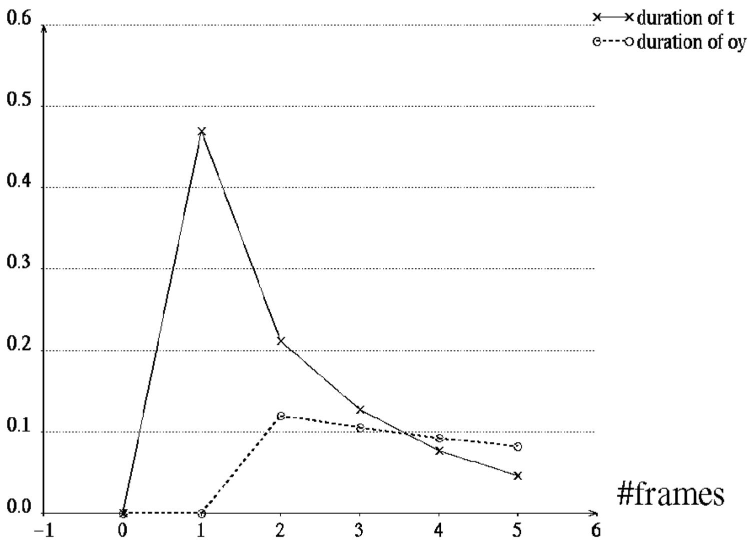


FIG. 2.12 – *Distributions des durées sur l'état central des phonèmes /t/ et /oy/*

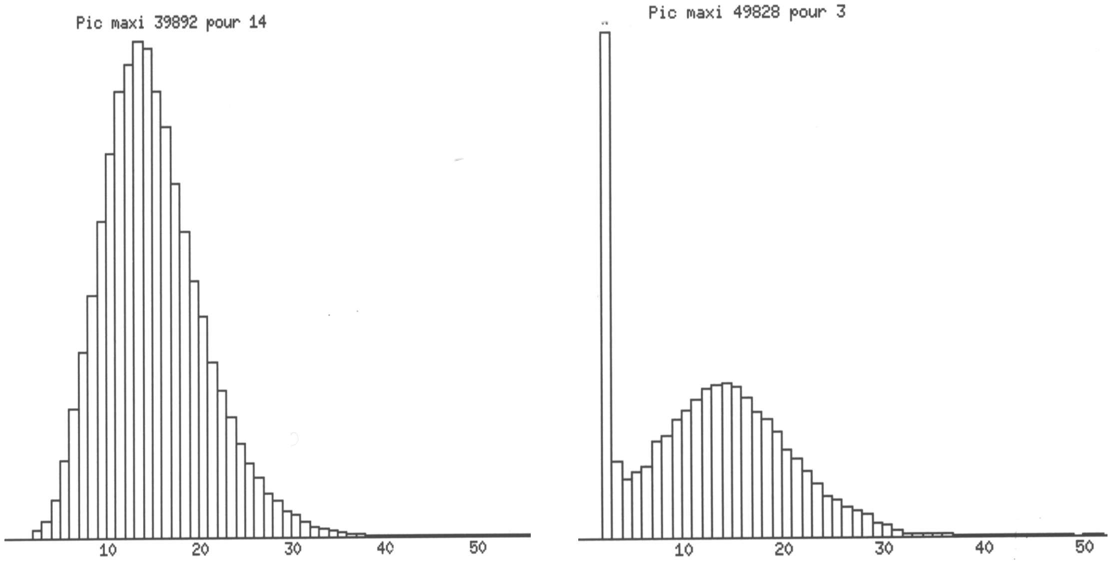
Dans un travail réalisé par un étudiant de DEA en 1994 [Binda 94], nous avons modélisé la durée de trois états consécutifs dans une suite mots. Ces états provenaient du même modèle ou bien étaient à cheval sur deux modèles de mots adjacents dans l'énoncé. Nous traitions ainsi de la même façon les mots et leurs frontières.

Ce travail n'était qu'un début de sujet de thèse sur l'utilisation d'informations segmentales à différents niveaux. Le reproche à faire à cette approche est que la durée de trois états n'est estimée qu'une fois pour toute sur tout le corpus d'apprentissage. Il n'y a pas d'ajustement en fonction de la vitesse d'élocution et de ses variations. Nous avions fait l'hypothèse que dans le cadre particulier de l'énoncé de suites de chiffres, comme c'est le cas du corpus *TI-NIST*, l'ensemble des locuteurs avait le même débit moyen. Par la suite, les résultats obtenus nous ont montré que cette hypothèse simple et vraisemblable peut conduire à d'excellents résultats.

Dans cette étude, la durée de trois états consécutifs dans la suite de modèles reconnus était utilisée pour répartir les alignements des mots sur les états des HMM en deux classes : les bons et les mauvais. Les trois états appartenaient à une fenêtre d'analyse qui en glissant parcourait l'ensemble des états de la suite des modèles de la réponse. Sur le corpus d'apprentissage de *TI-NIST*, deux densités ont été estimées pour la répartition du nombre de trames dans 3 états consécutifs : une pour les suites correctes ; une autre pour les suites incorrectes. Chaque chiffre est représenté par un HMM2 possédant 5 états. Les figures 2.13(a) et 2.13(b) montrent clairement que les alignements incorrects ont une grande probabilité d'avoir 3 états consécutifs qui ne capturent qu'une trame par état contrairement aux alignements corrects.

Les probabilités données par les modèles stochastiques et par le classifieur statistique étaient supposées indépendantes puis combinées dans un calcul heuristique, elles permettaient ensuite de ré-ordonner les réponses données par un algorithme de reconnaissance de type *N meilleurs*. De cette façon, le taux de reconnaissance de suites de chiffres passe de 97.6 % à 98.5 %, ce qui représente une réduction de 40 %.

Ce thème de recherche qui consiste à utiliser des informations segmentales soit en évaluant le degré de singularité (correct *versus* incorrect) des alignements temporels, soit en utilisant des informations de durée reste un thème d'actualité comme le témoignent les travaux récents



(a) suites correctes (b) suites incorrectes

FIG. 2.13 – *Répartition du nombre de trames dans 3 états consécutifs : estimation faite sur TI-NIST avec des modèles de chiffres à 5 états.*

[Lokbani 93, Jouvet 93, Moudenc 95] du CNET. Malheureusement notre contribution dans le domaine reste limitée faute de doctorants.

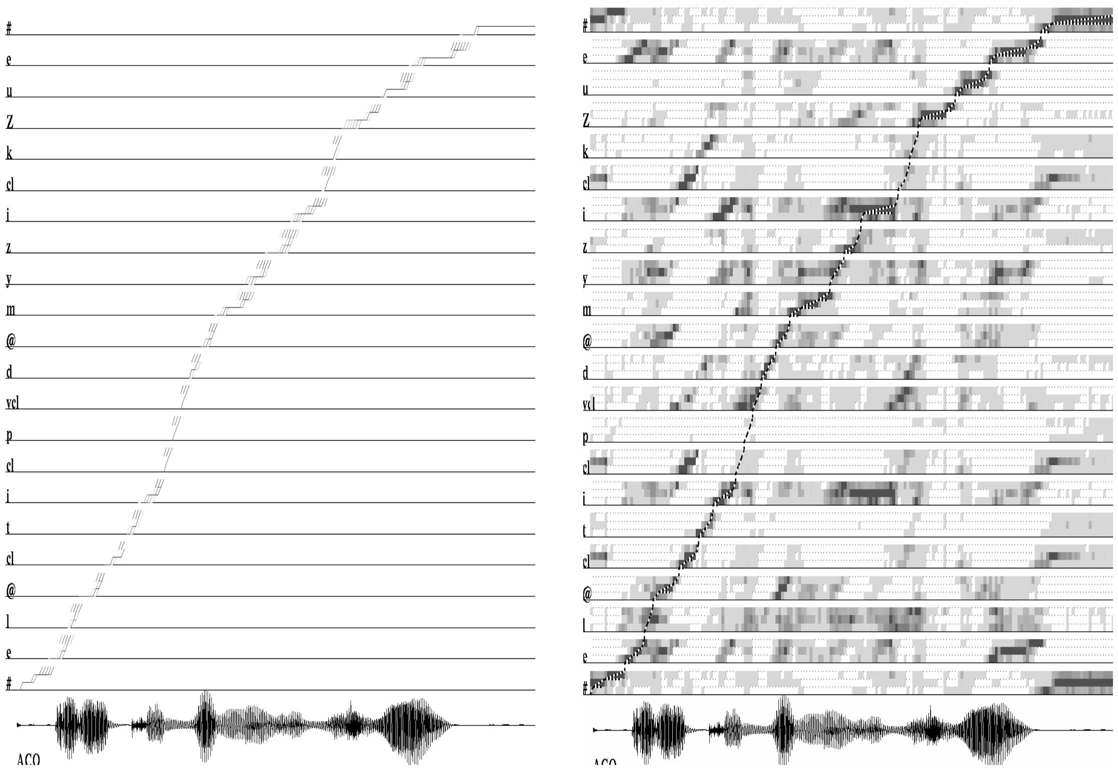
6.7 Apprentissage selon Baum-Welch ou Viterbi

En 1985, Rabiner introduisait l'approximation de Viterbi [Rabiner 85] pour effectuer l'apprentissage. Dans cette approximation, on recherche le meilleur chemin (*cf.* figure 2.14(a)) et on assigne aux comptes $\eta_t(i,j,k)$ la valeur 1. Nous avons noté, comme beaucoup, qu'il existe une grande dynamique, de l'ordre de 10^{30}, dans les valeurs de ces probabilités au voisinage proche du meilleur chemin. Cette dynamique se rencontre aussi bien dans un HMM1 que dans un HMM2, qu'il soit de mot ou de phonème. La contribution des chemins différents du meilleur est donc minime ce qui justifie *a posteriori* l'apprentissage avec l'algorithme de Viterbi.

Par contre, l'apprentissage selon Baum-Welch est intéressant pour obtenir une estimation initiale. Nous assignons à chaque trame la même vraisemblance. Cette valeur n'a pas d'importance car sa contribution est la même au numérateur et au dénominateur dans le calcul de $\eta_t(i,j,k)$. L'algorithme de Baum-Welch construit tous les chemins et pondère leur influence grâce à leur probabilité.

La mise en œuvre de la phase d'apprentissage est très simple, car le même algorithme sert à la fois pour l'estimation initiale et pour les itérations suivantes. Nous donnerons plus loin dans le chapitre 3 une description détaillée du processus d'apprentissage des modèles, qu'ils soient de phonèmes ou de mots.

Dans le progiciel HTK [Young 93], il existe un programme destiné à l'initialisation des modèles. Ce programme effectue une segmentation uniforme de chaque exemple et attribue chacun des segments à un état du HMM. Dans notre cas, l'algorithme de Baum-Welch envisage simplement toutes les segmentations pour constituer un modèle initial.



(a) Baum-Welch

(b) Viterbi

FIG. 2.14 – *Estimation selon Baum-Welch ou Viterbi*

On remarque sur la figure 2.14(b) que les vraisemblances des états des segments d'occlusion et d'explosion – /cl/ et /p/ – dans le mot «type» sont étrangement faibles. La phrase prononcée est «et le type de musique jouée». Dans cette phrase, le locuteur n'a pas prononcé de /p/ mais a voisé ce phonème en prononçant un /b/. Cette altération phonologique était nécessaire dans le contexte «type de». Il serait intéressant de retrouver l'ensemble des altérations du français en comparant la transcription graphèmes phonèmes avec les résultats d'un alignement.

7 Comparaisons entre HMM1 et HMM2

7.1 Méthodologie

Nous avons fait des comparaisons poussées afin d'évaluer l'influence de deux hypothèses couramment admises dans la modélisation de la parole à l'aide de HMM : l'utilisation de chaînes de Markov du premier ordre par opposition aux chaînes d'ordre deux et l'utilisation de matrices de covariances complètes par opposition à l'utilisation de matrices de covariances diagonales dans les mélanges de lois normales associées aux observations.

Nous avons utilisé deux corpus : *TI-NIST* et *TIMIT*. Sur le premier corpus, nous avons comparé une modélisation par mots en utilisant des HMM à 5 états. Il y a 23 modèles : 11 modèles de chiffres par genre (masculin et féminin) et un modèle de silence. Le système implémente ou non un *post* traitement visant à pénaliser les alignements incorrects. Ce *post* traitement optionnel nous sert à évaluer la modélisation des contraintes de durée par les HMM.

Sur le corpus *TIMIT*, la comparaison porte sur des HMM de phonèmes à 3 états représentant des mélanges de lois normales utilisant des matrices de covariances diagonales ou complètes. Dans tous les cas, il n'y a aucun *post* traitement. Sur ce corpus, nous utilisons 48 modèles de phonèmes indépendants du contexte. Le nombre total de densités est donné pour chaque expérience. Au cours du test, les phonèmes sont au nombre de 39, comme il est désormais d'usage depuis la thèse de K.F. Lee [Lee 89]. Dans l'évaluation des résultats pour déterminer le nombre d'insertions, omissions et substitutions, la pause compte comme un phonème. Cette pratique augmente artificiellement les performances – de l'ordre de 5 % – car la pause est un phonème fréquent – de l'ordre de 21 % – et toujours bien reconnu – de l'ordre de 90 % [Ljolje 94a]. Nous avons, nous aussi, évalué les résultats de cette façon afin de les comparer à ceux publiés couramment.

Les modèles du premier ordre sont obtenus grâce aux formules de normalisation données dans les équations 2.13, 2.15 et 2.16.

7.2 Analyse acoustique

Nous paramétrons traditionnellement le signal de parole par des coefficients cepstraux – *Mel Frequency Cepstral Coefficients* ou *MFCC* – qui ont toujours donné de bons résultats aussi bien en parole propre [Davis 80] que bruitée [Junqua 95a]. L'algorithme de calcul des coefficients cepstraux est celui publié par G. Chollet [Chollet 82] dans ses travaux sur les systèmes "zéros" d'évaluation de systèmes de reconnaissance. Récemment, de nouvelles variantes de calcul des coefficients cepstraux ont été publiées par le même auteur [Wassner 96] qui revendique une amélioration de taux de reconnaissance par rapport à son système "zéro". Dans le cas particulier de reconnaissance de cris, nous avons testé la variante qui consiste à filtrer le logarithme du spectre de puissance dans des canaux respectant l'échelle MEL plutôt que le spectre de puissance lui-même (cf. chapitre 5, page 81) sans retrouver la même amélioration.

Le signal de parole est enregistré dans un environnement calme et numérisé à une fréquence de 20 kHz pour *TI-NIST* et 16 kHz pour *TIMIT*. Une trame de 12 coefficients cepstraux est calculée toutes les 12 ms pour *TI-NIST* et 8 ms pour *TIMIT* à l'aide d'une fenêtre de 512 points. Le premier coefficient, représentant l'énergie est supprimé. Son utilisation dans des systèmes de reconnaissance amène une diminution de performances [Kenny 91]. Nous avons nous-même constaté le même résultat sur le corpus de *TI-NIST* [Baker 92]. Nous pensons que ce résultat provient de la variabilité importante de ce trait. A l'heure actuelle, beaucoup d'équipes normalisent ce coefficient en obligeant sa valeur maximale à être la même parmi toutes les phrases d'un corpus donné, donc en appliquant un facteur d'échelle différent sur l'amplitude du signal pour chacune des phrases. Nous n'avons pas encore testé ce nouveau cas particulier de normalisation.

Deux types de vecteurs de coefficients MFCC ont été utilisés. L'un comprend 24 coefficients regroupant les coefficients statiques et dynamiques du premier ordre (Δ) [Soong 88, Furui 81]. L'autre type est obtenu en complétant le précédent par les coefficients dynamiques du second-ordre ($\Delta\ \Delta$), ce qui constitue un vecteur de 35 coefficients. Les coefficients Δ et $\Delta\ \Delta$ sont calculés sur une fenêtre de 3, respectivement 5, trames autour de la trame courante.

7.3 Modélisation de la trajectoire dans l'espace des états

Dans un HMM, les trames sont supposées indépendantes une fois donnée la suite d'états du modèle. Les trames qui s'alignent sur un état proviennent de la même distribution et déterminent un segment dont la durée n'est pas corrélée avec la durée des segments voisins. Plusieurs études ont été faites sur la durée des états. Citons Levinson [Levinson 86], Wilpon [Wilpon 93] ou Russel [Russell 87, Russell 88]. Différentes familles de densités ont été utilisées : non paramétriques [Ferguson 80] ou paramétriques comme des lois gamma [Suaudeau 93, Levinson 86], log-normales [Kenny 91] ou Poisson [Russell 85].

L'étude des erreurs produites par l'algorithme de Viterbi montre qu'il est nécessaire d'avoir un mécanisme pour rejeter certaines réponses correspondant à des alignements incorrects. Ce mécanisme peut utiliser des contraintes d'alignements basées sur la durée relative des états successifs, un peu à la manière des contraintes de pente d'un algorithme de programmation dynamique.

Sur le corpus d'apprentissage de *TI-NIST*, nous avons noté que les durées respectives des états étaient fortement corrélées, ce qui laisse penser que la trajectoire de l'élocution dans l'espace des états suit un modèle précis.

Un alignement est obtenu à l'aide de l'algorithme de Viterbi. Il est représenté par un vecteur x constitué des durées relatives de chacun des N états. Comme la somme des composantes est égale à 1, on remplace une des composantes par le nombre total de trames alignées sur le modèle.

$$x = (d, \frac{d}{d_2}, \frac{d}{d_3}, ..., \frac{d}{d_N}) \tag{2.17}$$

d_i et d représentent respectivement le nombre de trames alignées avec l'état i et la durée du mot exprimée en nombre de trames.

Le modèle d'alignement correct est représenté par une loi normale $\mathcal{N}(g_\lambda, V_\lambda)$ de moyenne g_λ et de matrice de covariances V_λ, toutes deux dépendant du mot λ. Cette densité caractérise la classe des alignements corrects du mot λ. Elle est estimée sur le corpus d'apprentissage.

Après la reconnaissance par l'algorithme des N *meilleurs* [8], on mesure la distance d'un alignement fourni par l'algorithme de Viterbi à sa classe à l'aide de la métrique de Mahalanobis.

$$d^2(x, g_\lambda) = detV_\lambda{}^{1/N}(\mathbf{x} - \mathbf{g}_\lambda)^t \mathbf{V}_\lambda{}^{-1}(\mathbf{x} - \mathbf{g}_\lambda) \tag{2.18}$$

$detV_\lambda{}^{\frac{1}{N}}$ est un facteur de normalisation entre toutes les distances. Il donne à chaque matrice V_λ un déterminant de 1.

Cette distance vient pondérer la probabilité d'alignement donnée par l'algorithme de reconnaissance N *meilleurs*. Le score final est une combinaison linéaire du logarithme de cette probabilité et de la distance de Mahalanobis :

$$finalScore = A \cdot d^2(x, g_\lambda) - B \cdot \log(P(O/\lambda)) \tag{2.19}$$

A et B sont deux facteurs de normalisation de signe contraire que nous ajustons afin de rendre minimum le taux d'erreurs sur un corpus de développement.

Dans le cadre de la comparaison HMM1/HMM2, nous avons noté que les valeurs optimales des coefficients A et B dépendaient des systèmes. Dans un système à base de HMM1, la valeur de A est plus importante que dans un système à base de HMM2, ce qui montre que la trajectoire dans l'ensemble des états a besoin d'être plus contrainte après une reconnaissance à l'aide de HMM1. Nous pensons que ce résultat montre que les HMM2 modélisent mieux la trajectoire des états que les HMM1.

8. Dans ce cas, N représente le nombre de réponses calculées : typiquement 100, ce n'est pas le nombre d'états d'un HMM qui est égal à 5 dans ce cas.

A la différence des travaux de M.N. Lokbani [Lokbani 93] et Binda [Binda 94], nous n'utilisons pas de modèle d'alignements incorrects. Nos travaux sur la modélisation de la trajectoire dans l'espace des états sont antérieurs aux travaux effectués sur la modélisation du débit (cf. § 6.6). Il serait intéressant d'explorer cette piste de recherche dans le cas de la modélisation des trajectoires.

Il existe une analogie entre ce post traitement et la modélisation stochastique par trajectoires – *Stochastic Trajectory Modeling* ou *STM* – telle qu'elle est abordée par Y. Gong et M. Afify [Gong 94b, Afify 96]. En effet, le vecteur de durée x détermine une probabilité de durée pour le mot et de trajectoire dans l'espace des états pour son alignement. Dans une première approche, nous avons choisi de modéliser un seul alignement et une seule durée par la moyenne du vecteur x mais il est possible d'effectuer une classification[9] de l'ensemble des vecteurs x pour déterminer plusieurs couples (trajectoire, durée). L'analogie s'arrête là car les modélisations par HMM et STM présentent plusieurs différences :

- L'évidence acoustique n'intervient pas dans la probabilité d'une trajectoire dans un STM alors qu'elle intervient par le terme $d^2(x, g_\lambda)$ dans notre post traitement.

- Dans un STM, la durée est modélisée par une loi gamma alors qu'elle intervient à deux niveaux dans un HMM : sous forme du produit des probabilités des transitions a_{ijk} dont la valeur dépend du nombre de trames et en tant que composante du vecteur x.

- Un STM effectue un échantillonnage linéaire du segment à reconnaître en Q états représentant autant de lois normales dont les paramètres sont corrélés. L'ensemble des Q lois normales constitue une trajectoire, alors que l'algorithme de Viterbi effectue une segmentation non linéaire sur une seule suite d'états qui représente une seule trajectoire moyenne. Dans un HMM, la variabilité est reportée individuellement dans les états qui sont des mélanges de lois normales.

- L'échantillonnage linéaire en Q états dans un STM fait qu'il n'y a que Q vraisemblances calculées ; il s'agit dans ce premier cas d'une approche par segment, alors que le nombre de vraisemblances calculées est variable dans un HMM ; il s'agit dans ce dernier cas d'une approche centiseconde.

7.4 Comparaison

Les tables 2.2, 2.3 et 2.4 résument les résultats déjà publiés dans [Mari 94b]. Elles représentent les taux de reconnaissance de chaînes de chiffres obtenues sur *TI-NIST* en modélisant un chiffre à l'aide de HMM à 5 états. Deux catégories de systèmes sont comparées : la première à base de 23 HMM1 – un modèle de silence, 11 modèles de chiffres[10] pour les hommes, 11 modèles de chiffres pour les femmes – et la seconde à base de 23 HMM2 possédant la même topologie que leurs homologues du premier ordre. Nous comparons 8 systèmes suivant la taille du vecteur d'analyse acoustique, l'ordre des HMM et la présence ou l'absence d'un post traitement appliqué sur la trajectoire dans l'espace des états. Nous donnons entre parenthèses les intervalles de confiance au risque de 5 %. Tous ces résultats sont obtenus sans utiliser le traitement modélisant le débit des locuteurs tel qu'il est expliqué page 41.

9. Voilà une idée à développer pour unifier l'approche à base de HMM et de STM !
10. En anglais, pour dire *zero*, on peut aussi dire *Oh*.

Paramètrisation	hommes + femmes	
	HMM1	HMM2
$11MFCC + 11\Delta$ $+\Delta E + \Delta\Delta E$	4.5% (4.1 5.0)	2.4% (2.1 2.7)
$11MFCC + 11\Delta$ $+\Delta E + \Delta\Delta E +$ $11\Delta\Delta$	3.7% (3.3 4.1)	2.4% (2.1 2.7)

TAB. 2.2 – *Taux d'erreurs sur les chaînes (sans post traitement)*

Paramètrisation	hommes + femmes	
	HMM1	HMM2
$11MFCC + 11\Delta$ $+\Delta E + \Delta\Delta E$	2.8% (2.5 3.2)	2.2% (1.9 2.5)
$11MFCC + 11\Delta$ $+\Delta E + \Delta\Delta E+$ $11\Delta\Delta$	2.3% (2.0 2.6)	2.1% (1.8 2.4)

TAB. 2.3 – *Taux d'erreurs sur les chaînes (avec post traitement)*

	HMM1	HMM2
Insertions	174	159
Omissions	14	20
Substitutions	31	34
Tx. d'erreurs de chaînes	2.3 %	2.4 %
% correct	99.8	99.8
Tx. reconnaissance de mots	99.2	99.2

TAB. 2.4 – *Comparaison entre HMM1 (avec post traitement) et HMM2 (sans post traitement)*

A leur lecture, plusieurs constatations peuvent être tirées :

1. quand il n'y a pas de post traitement, les HMM2 fournissent de meilleurs résultats que les HMM1 ;
2. l'écart de performances se réduit avec l'utilisation d'un post traitement ;
3. les coefficients de régression de second ordre n'améliorent pas les performances des HMM2 d'une façon significative.

Les points 1 et 2 peuvent s'expliquer par les capacités qu'ont les HMM2 de modéliser les courts séjours dans certains états. Ainsi la trajectoire de la parole dans l'espace des états est déjà modélisée dans un HMM2. Dans le cas où chaque état capte au plus un nombre réduit de trames (de l'ordre de 2 ou 3), la durée de chaque état est correctement modélisée par les transitions a_{ijk} du HMM2 (cf. équations 2.17). La corrélation des trames entre elles est grossièrement représentée par la corrélation des états ce qui peut expliquer le point 3. Rappellons que les coefficients de régression sont utilisés pour modéliser la corrélation entre trames.

La comparaison est différente sur *TIMIT* (cf. tables 2.5 et 2.6). En effet, on observe un écart plus petit entre les performances des HMM1 et des HMM2. Une raison peut être avancée : tout en étant de même topologie – modèles gauche-droite, sans saut, avec une boucle sur chaque état – les modèles diffèrent sur le nombre d'états ; ils ont 5 états sur *TI-NIST* et 3 états sur *TIMIT*. L'influence de l'état d'où l'on vient se rencontre plus souvent dans un modèle à 5 états que dans un modèle à 3. En effet, un HMM2 possédant 3 états actifs n'a que 8 valeurs de probabilité de transitions a_{ijk} alors qu'un HMM1 de même topologie contient 6 valeurs de probabilité de transitions a_{ij}. Cette constatation a déjà été faite dans le domaine de l'écriture et de la reconnaissance de mots manuscrits [Kundu 89]. De plus, il n'y a plus de modélisation possible de trajectoires dans un si petit ensemble d'états. Il n'est pas possible d'augmenter le nombre d'états car celui-ci est en relation avec la durée d'une unité et certains phonèmes sont déjà trés courts en moyenne. La solution serait, par contre, de modéliser des unités plus longues comme des polyphones[11].

En conclusion, sur *TIMIT* et avec des modèles à 3 états actifs, les HMM2 et HMM1 se comportent de la même façon. L'usage de coefficients du second ordre améliore les performances dans tous les cas. Par contre, sur *TI-NIST* et avec des modèles de 5 états, les HMM2 se détachent nettement de leurs homologues du premier ordre et n'ont plus besoin de coefficients de régression du second ordre car la corrélation à long terme entre trames est déjà modélisée.

$\Delta\Delta$	HMM1	HMM2
Diagonale	65.4	65.7
Complète	70.5	70.6

(a) test

$\Delta\Delta$	HMM1	HMM2
Diagonale	66.0	66.2
Complète	69.8	69.9

(b) core test

TAB. 2.5 – *Taux de reconnaissance comparés sur TIMIT (35 coef.et 1111 Pdf)*

Δ	HMM1	HMM2
Diagonale	65.4	65.6
Complète	69.4	69.4

(a) test

Δ	HMM1	HMM2
Diagonale	65.3	65.9
Complète	69.4	68.8

(b) core test

TAB. 2.6 – *Taux de reconnaissance comparés sur TIMIT (24 coef. et 1329 Pdf)*

7.5 Influence de la corrélation dans une trame d'analyse

Nous avons poursuivi la comparaison HMM1/HMM2 en testant 4 types de modélisation sur *TIMIT* : HMM1 et HMM2 d'une part et matrices de covariances complètes et diagonales dans les mélanges de lois associées aux observations d'autre part.

Nous modélisons les 48 allophones de la langue anglaise par autant de HMM possédant chacun 3 états comme il a été dit auparavant.

Les tables 2.5 et 2.6 donnent les taux de reconnaissance[12] de phonèmes obtenus avec l'aide d'un modèle bigramme . A l'étude des résultats donnés, on distingue deux types de systèmes : ceux qui utilisent des matrices de covariances complètes et ceux qui utilisent des matrices de covariances diagonales ; les premiers ayant 5 % d'erreurs en moins[13] pour le même nombre de densités. Cet écart de performances s'explique par le facteur 10 entre le nombre de paramètres des deux systèmes ; le nombre de densités étant le même dans tous les systèmes mais pas leurs représentations. Ces résultats rejoignent ceux publiés par Ljolje en 1994 avec des HMM1 [Ljolje 94b].

Comme la vraisemblance moyenne d'une trame du corpus d'apprentissage est bien plus faible (de l'ordre de 500) quand on utilise des matrices diagonales, le facteur linguistique qui surestime l'importance des transitions entre phonèmes a été lui aussi modifié. De cette façon, on gagne 0,2 % en taux de reconnaissance (*accuracy* en anglais).

11. Les « ions, tre, dre, gre, yeux » qui font le malheur des jeunes enfants et le bonheur des orthophonistes.
12. Taux de reco. = 1 - % substitutions - % omissions - % insertions
13. Ou 15 % si on ramène ces 5 % aux 30 % d'erreurs restantes, comme cela se fait aux USA.

Toutefois, dans chaque cas de figure, les HMM2 donnent des performances supérieures aux HMM1 grâce à une meilleure modélisation de la durée bien que cette amélioration ne soit pas statistiquement significative. On voit que la contrainte de normalisation supplémentaire donnée par l'équation 2.12 introduit un lissage des probabilités de transitions au détriment de la capacité de discrimination des modèles. De plus, ce lissage se ressent dans la vraisemblance du corpus d'apprentissage qui a été maximisée par l'algorithme *Forward-Backward*. Celle-ci est maximale avec les HMM2 implémentant des matrices complètes. Expérimentalement on a trouvé la suite d'inégalités sur les vraisemblances du corpus d'apprentissage :

$$HMM1_{diag} \leq HMM2_{diag} \leq HMM1_{full} \leq HMM2_{full}$$

Nos meilleurs résultats ne sont pas éloignés mais restent inférieurs aux derniers résultats publiés en 1993[14] par [Lamel 93] qui utilisaient 500 HMM1 dépendants du contexte et avec des matrices de covariances diagonales. Avec l'aide d'un modèle bigramme, ce système atteint un taux de reconnaissance de 72.9 %. Ce système possède beaucoup plus de paramètres ce qui explique les meilleures performances. L'utilisation de modèles dépendants du contexte ou de polyphones constitue notre préoccupation actuelle.

8 Conclusions

Dans le cadre de la reconnaissance de la parole, ce chapitre décrit différentes extensions des algorithmes classiques de reconnaissance et d'apprentissage fondés sur une approche markovienne.

Au prix d'une complexité légèrement supérieure, les HMM2 modélisent mieux un signal temporel complexe comme la parole. Les meilleures performances peuvent être attribuées à une modélisation plus fine de la durée au niveau des segments et de la trajectoire de la parole dans l'ensemble des états.

Dans un HMM2, la durée de séjour dans un état est gouvernée par deux paramètres : la probabilité d'entrer dans l'état et de n'y rester qu'une trame, et la probabilité de séjourner au moins deux trames. Cette dernière étant représentée par une loi géométrique. Cette modélisation simple permet de s'affranchir de post-traitements dans lesquels on re-ordonne les alignements calculés par l'algorithme de Viterbi dans le but d'éliminer les singuliers.

Dans le domaine de la reconnaissance, l'algorithme de Viterbi-bloc permet de traiter les données au fur et mesure de leur arrivée, sans attendre la fin de l'énoncé. Cette propriété est intéressante quand la longueur des phrases ou la taille importante du vocabulaire ne permettent pas un décodage complet de l'énoncé.

Nous avons vu au paragraphe 6.6, que la modélisation du débit d'élocution permet d'améliorer les performances d'un système de reconnaissance de suites de chiffres.

Nous avons évalués sur différents corpus les performances en reconnaissance des modèles d'ordre deux. Dans tous les cas, ceux-ci donnent de meilleures performances que leurs homologues du premier ordre.

Les meilleurs résultats dûs à l'utilisation de matrices de covariances complètes contredisent quelque peu l'hypothèse d'indépendance des coefficients cepstraux. Nous avons observé une vraisemblance de l'espace d'apprentissage plus élevée en utilisant de telles matrices pour estimer des lois normales.

14. Il devient difficile de trouver des performances récentes. Les équipes préfèrent donner les taux de reconnaissance de mots

Chapitre 3

Mise en œuvre de HMM dans un système de reconnaissance

1 Introduction

Ce paragraphe présente de façon succincte les différentes étapes nécessaires à l'utilisation de progiciels "tout markoviens" dans le domaine de la reconnaissance automatique de la parole. Nous décrirons tout d'abord la création d'un système de reconnaissance de mots enchaînés à l'aide de modèles de Markov cachés : construction des modèles initiaux, estimations en mode isolé puis en mode connecté, éclatement en plusieurs lois normales et enfin reconnaissance. Nous envisagerons aussi, en seconde application, l'utilisation de HMM2 pour le décodage acoustico-phonétique du français dans le cadre de la réalisation d'une machine à dicter.

Dans la première application, il s'agit de reconnaître le nom ou le prénom d'une personne dans un annuaire. Le vocabulaire est limité aux 26 lettres de l'alphabet, mais les perturbations dues au canal de transmission, la qualité des lignes (interurbain ou appel longue distance), l'utilisation de combinés téléphoniques variés et la présence de bruit ambiant représentent des difficultés qu'il faudra prendre en compte dès les premiers traitements. Les expériences décrites dans ce chapitre utilisent le corpus de lettres épelées distribué par *OGI* (*cf.* page 7).

Dans la seconde application, le signal acoustique est de bonne qualité, car l'acquisition s'est déroulée dans un milieu calme et avec un seul microphone. En revanche, le vocabulaire utilisé est supérieur à 20000 mots et la grammaire est peu contrainte puisqu'il s'agit de textes issus du journal ≪Le Monde≫. Le corpus associé est connu sous le nom de *BREF80* (*cf.* page 6).

Les outils dont nous décrivons l'utilisation sont développés dans notre équipe depuis 1994. Ils sont encore en évolution et n'ont pas la facilité d'utilisation de logiciels commerciaux du même domaine comme HTK [Young 93]. La démarche de création d'un système de reconnaissance à base de progiciels de type HTK se ferait de la même façon.

2 Choix des paramètres du modèle

2.1 Nombre d'états et de densités

Pour réaliser un système de reconnaissance à base de HMM, il faut s'assurer au préalable de l'existence d'un corpus de taille suffisante. Les modèles de Markov nécessitent l'estimation de moyennes et d'écart-type de variables aléatoires, il faut donc disposer d'un gros corpus pour que cette estimation soit statistiquement valable. La taille du corpus va influer sur le nombre de paramètres des modèles. On peut distinguer deux types de paramètres dans un HMM : le nombre d'états d'un modèle et le nombre de densités par états. Le premier est en relation avec la durée et la structure de l'unité modélisée. A l'heure actuelle, lorsqu'il s'agit de reconnaître des unités phonétiques, la majorité s'accorde sur une valeur de trois états par modèle mais s'il s'agit de mots, unités moins homogènes, certaines équipes utilisent des nombres plus élevés (de 6 à 26) [Jouvet 94, Normandin 95]. Quant au nombre de densités par états, celui-ci est directement lié à la taille du corpus. Sur une partie du corpus de lettres épelés de *OGI* comportant une moyenne de 600 prononciations pour chacune des 26 lettres, nous avons estimé 26 modèles comportant 6 états (sauf pour le W qui en comporte 12). Chaque état possède en moyenne 3 densités.

La pratique montre que la variation des performances en fonction du nombre d'états des modèles est lente [Rabiner 95] et il est souvent nécessaire de lier certains états entre eux pour accroître la qualité de l'estimation [Digalakis 94].

2.2 Analyse acoustique

Comme cela a été déjà dit plus haut, une méthode qui donne de bons résultats tant en parole propre que bruitée est le calcul de cepstres avec une échelle Mel (MFCC). Pour capturer les phénomènes transitoires, une amélioration consiste à calculer des coefficients de régression du premier et du second ordre sur des trames adjacentes. On capture ainsi des événements sur une plus grande fenêtre de parole. Mais le fait d'utiliser des vecteurs de plus grande dimension nécessite l'estimation d'un plus grand nombre de paramètres et donc de plus grand corpus. On a donc les choix suivants :

- pas de coefficient de régression, on utilise les coefficients cepstraux, dits coefficients statiques, moins le premier coefficient qui est l'énergie. Dans ce cas, on obtient rapidement des résultats médiocres ;

- on adjoint aux coefficients statiques des coefficients de régression du premier ordre, ce qui se traduit par de gros progrès dans la reconnaissance mais parallèlement par un net ralentissement du système de reconnaissance car la complexité de l'algorithme de calcul de la vraisemblance sous l'hypothèse de normalité est en n^2 dans le cas d'utilisation de matrices de covariances complètes ; n étant la dimension d'une trame de parole ;

- on augmente encore la taille de la trame et la fenêtre temporelle d'analyse en ajoutant les coefficients de régression du second ordre. Si les densités sont toujours aussi bien estimées, on peut s'attendre, après un temps de calcul encore plus important à des résultats encore meilleurs. Cette course à la dimension s'arrête là, les résultats utilisant une analyse discriminante linéaire (LDA) sur des vecteurs capturant une plus longue fenêtre de parole [Beulen 95] ne montrent pas une nette amélioration.

Un bon compromis est d'utiliser les coefficients statiques auxquels on ajoute les coefficients de régression du premier ordre et l'accélération de l'énergie sur une fenêtre de 80 ms.

Quand le canal de transmission est variable, ce qui est le cas pour le téléphone, des traitements adaptés peuvent être utilisés, parmi lesquels ont peut citer : l'analyse *Rasta* et la soustraction de la moyenne à long terme des cepstres. Sur la base de donnée *OGI*, c'est cette dernière technique qui nous a donné les résultats les meilleurs [Junqua 95b].

3 La phase d'amorçage

Après avoir fixé la topologie des modèles et enregistré le corpus, il faut étiqueter une partie de ce corpus. Il faut une moyenne de 60 prononciations différentes de la même unité dans un cadre multi-locuteurs[15]. Le plus simple est d'utiliser un éditeur de signal pour poser les frontières temporelles de début et de fin de mot pour chacune des prononciations. Par exemple si vous voulez reconnaître les 26 lettres épelées alors étiquetez 60 alphabets. Que faire des données restantes? on peut penser que les HMM estimés avec ces 60 alphabets étiquetteront le reste. Ceci est vrai pour des corpus dont on connaît exactement la transcription comme dans le cas de noms de personnes épelés. En revanche, pour étiqueter en phonèmes une phrase dont on ne connaît que le texte écrit, il faut résoudre le problème de la transcription graphèmes-phonèmes (cf. §5, page 51).

4 La phase d'apprentissage

4.1 Création des modèles "zéro"

La petite partie du corpus qui vient d'être étiquetée à la main sert de corpus d'apprentissage pour des modèles de base. Chaque modèle est estimé isolément en donnant les frontières temporelles de chaque exemple d'apprentissage. Pour cela, nous utilisons un modèle initial où toutes les transitions sont équiprobables et où chaque état est représenté par une loi uniforme [Mari 94b]. L'algorithme d'apprentissage envisage tous les alignements pondérés par leurs probabilités comme le montre la figure 3.1(a). Ce modèle est ensuite ré-estimé par l'algorithme de Baum-Welch décrit au § 6.2 du chapitre 2 en supposant que chaque état représente une densité normale avec une matrice de corrélation complète. La figure 3.1(b) montre les alignements les plus probables retenus.

La matrice de corrélation de la loi normale montre que les coefficients des cepstres sont peu corrélés néanmoins elle présente quelques irrégularités qui justifient son utilisation par rapport à l'utilisation d'une matrice diagonale.

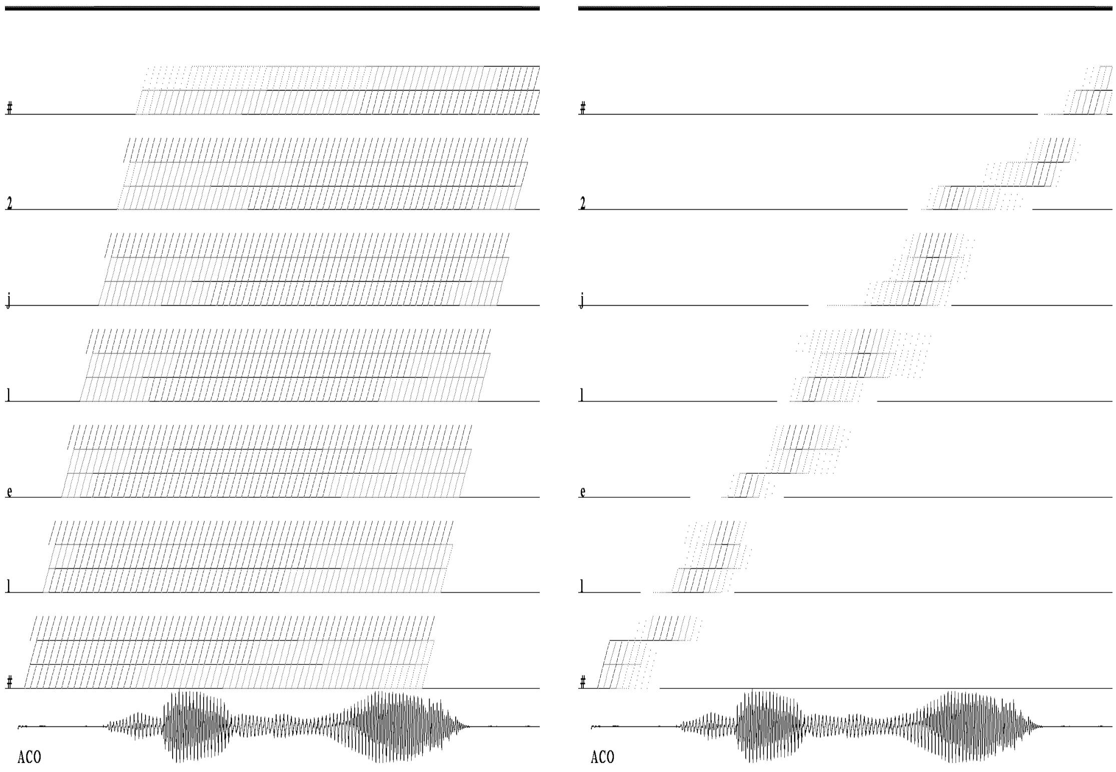
Après avoir construit un modèle par unité à reconnaître et un modèle de bruit – ou de silence – on utilise ces modèles dans une reconnaissance guidée pour étiqueter le corpus restant morceau par morceau selon une difficulté croissante. Par exemple, on peut étiqueter tous les alphabets, puis se servir de ces modèles pour étiqueter toutes les épellations réussies et enfin, terminer par un étiquetage des appels qui ne respectaient pas les consignes. Nous n'en sommes pas encore là ! Nous avons essayé d'étiqueter des événements acoustiques qui ne sont pas de la parole, tels que raclements de gorge, toux, onomatopées d'hésitation mais sans résultats concluants en reconnaissance.

5 L'étiquetage du corpus

5.1 Cas de mots enchaînés

Dès lors que l'on a un modèle par mot à reconnaître, on peut connecter ces modèles pour la reconnaissance d'une phrase. Quand l'utilisateur épelle « F O H R» on lancera l'algorithme de Viterbi sur le gros modèle constitué du modèle HMM de F connecté à celui de O et ainsi de suite jusqu'à celui du R. Comme on ne sait pas où sont les frontières entre la parole et le

15. *speaker independent* en anglais.



(a) Alignement initial (b) Alignement final

Fig. 3.1 – *Alignements initiaux et finaux de la phrase « Les lieux » phonétisée comme : # l e l j 2 #. La table définissant les phonèmes utilisés pour l'affichage est donnée page 24.*

silence, on encadre chaque modèle de lettre par un HMM de silence optionnel. Ce gros modèle est appelé le modèle de prononciation, parfois même modèle linguistique.

Si les modèles "zéro" sont suffisamment bons, alors l'algorithme de Viterbi trouvera le meilleur alignement entre les états et les trames définissant ainsi une segmentation donnant de bons résultats par la suite.

5.2 Cas de la reconnaissance phonétique

Afin de limiter le travail d'étiquetage, notre but est de réaliser l'apprentissage d'un modèle de Markov en utilisant un corpus de parole dont on n'a que la transcription orthographique. A partir des modèles "zéro", l'apprentissage se fera en deux étapes :

– étiquetage automatique de l'ensemble du corpus grâce aux premiers modèles ;

– apprentissage des modèles plus précis à l'aide de l'ensemble du corpus.

Pour étiqueter automatiquement des fichiers de parole, nous utilisons un modèle de Markov appris à l'aide du petit corpus étiqueté à la main. La segmentation est immédiate si on possède la transcription en phonèmes de la phrase. Il suffit d'utiliser l'algorithme de Viterbi comme le montre la figure 2.5 de la page 25.

Dans le cas de *BREF80*, les phrases proviennent de la lecture d'articles de journal (c.-à-d. Le Monde) et nous ne possédons que le texte écrit de l'article. Il est nécessaire d'effectuer au préalable une transcription de graphèmes en phonèmes. A l'aide du dictionnaire BDLEX [Pérennou 92] qui contient la transcription phonétique, la finale phonologique et des règles de liaison nous pouvons générer un ensemble de transcriptions phonétiques possibles.

5.3 Transcription graphèmes-phonèmes : problèmes rencontrés

Le problème de la transcription graphème-phonème pour l'étiquetage est sensiblement différent de celui posé pour la synthèse de parole. En effet, pour la synthèse, il suffit de trouver UNE prononciation acceptable. Pour l'étiquetage, il faut aligner LA prononciation que le locuteur a utilisée. Par exemple « 1900 » peut se prononcer « mille neuf cents » ou « dix neuf cents » Il faut donc engendrer toutes les prononciations possibles d'une phrase, les aligner et celle qui donne le meilleur score d'alignement est utilisée pour réaliser l'étiquetage.

D'après D. Fohr [Fohr 96], voici une liste, non exhaustive, des problèmes rencontrés lors de l'étiquetage de *BREF80* :

- Les pauses entre les mots et les ≪e muet≫ doivent être systématiquement prévus.
- Les liaisons doivent toutes être envisagées.
- Il faut tenir compte des mots d'origine étrangère et des noms propres pour lesquels plusieurs prononciations, même étranges, doivent être imaginées. Par exemple, l'ex-président américain Bush a été prononcé : /buʃ/ (ce qui semble naturel) /byʃ/ et même /bøʃ/ !
- Les acronymes peuvent s'épeler ou se prononcer comme un seul mot : par exemple le CRIN.
- Dans le cas des chiffres, il faut tenir compte du fait que certaines finales sont optionnelles, 500 peut se prononcer / s ɛ̃ k s ɑ̃ / ou / s ɛ̃ s ɑ̃ /.
- La virgule doit être traitée différemment suivant qu'elle représente un signe de ponctuation ou une virgule décimale : dans le premier cas, elle ne se prononce pas tandis que dans le dernier cas elle se prononce par le mot « virgule » ou, pour les locuteurs anglophones, « point ».
- Certaines initiales peuvent, suivant le contexte se prononcer de manières très différentes : par exemple V peut être prononcé / s ɛ̃ k j e m / dans « V^{e} république », / v e / dans « V. Dupont » et / v i k t ɔ R / dans « V. Hugo ».

Toutes ces difficultés montrent à quel point il est difficile d'envisager une procédure purement automatique pour l'étiquetage. D. Fohr [Fohr 96] propose de définir plusieurs critères pour trouver les étiquetages erronés :

- rejeter les phrases qui s'alignent mal lors de l'apprentissage ;
- calculer la probabilité *a posteriori* d'un phonème connaissant ses limites. Si une suite de phonèmes a une probabilité faible, cette phrase est aussi rejetée ;
- faire transcrire à l'écoute par un expert toutes les phrases ainsi rejetées et, ensuite, les réaligner automatiquement.

Notre article publié aux Journées d'Etudes sur la Parole [Fohr 96] contient une évaluation de cet algorithme d'étiquetage.

6 Apprentissage en parole continue

En parole continue, les frontières entre unités ne sont pas connues. On laisse alors l'algorithme d'apprentissage Baum-Welch tenter tous les alignements et ne retenir que les plus probables. Les nombres de transitions entre états et de trames observées sur chaque état sont pondérés en fonction de la vraisemblance de l'alignement. Encore une fois, si l'initialisation est bien faite, les bonnes segmentations seront les plus probables. Par cette méthode, on capture les phénomènes de co-articulation entre deux modèles.

7 Éclatement en plusieurs densités

Pour approcher une densité quelconque, on peut soit créer un réseau de neurones, soit trouver une base de densités gaussiennes sur laquelle on effectue une combinaison linéaire. Si tous les états partagent la même base, ils sont appelés semi-continus [Huang 90]. En revanche, les HMM continus utilisent une base différente de densités pour chaque état. La construction des bases de densités [16] a été un sujet de recherche entre 1987 et 1994. On a vu définir différents mots recouvrant les même notions : *clustering* des trames s'alignant sur des états et partage des paramètres des densités associées aux états. En 1988, L. Bahl [Bahl 88] introduit les *fenones* qui sont les segments stationnaires modélisés par les états de HMM. Les chercheurs d'IBM utilisent les *fenones* plutôt que les phonèmes pour étiqueter les mots. En 1992, H.Hwang [Hwang 92] définit les *senones* qui sont des unités sub-phonétiques correspondant aux états de HMM. Les senones peuvent être partagés par des HMM de phonèmes ou de mots. *Fenones* et *senones* recouvrent les même notions. Enfin, en 1994, Digalakis [Digalakis 94] définit les *genones* qui correspondent aux densités gaussiennes qui sont partagées par les différentes lois de probabilité d'observations associées aux différents états.

L'algorithme que nous utilisons pour trouver l'ensemble des bases – ou l'ensemble de *genones* – utilise la notion de trajectoire de la parole dans l'espace des paramètres et des états d'un HMM2. Il peut se décrire de la façon suivante.

Effectuons préalablement une quantification vectorielle des trames – représentées par des coefficients statiques et dynamiques – contenues dans un corpus représentatif de tout le vocabulaire. Cette quantification vectorielle définit un pavage de l'espace de parole qu'une trajectoire de trames va traverser. À chaque trajectoire correspondent plusieurs suites d'états cachés pondérées par leur probabilité d'alignement. Pour calculer toutes ces suites, nous utilisons l'algorithme *forward-backward* sur les modèles à une densité par état. Une trajectoire est subdivisée en portions qui correspondent aux différents états.

Le couple (état, pavé) définit une nouvelle classe modélisée par une densité gaussienne estimée à partir des trames de la portion de trajectoire alignée sur cet état et qui traverse ce pavé. Le représentant de cette classe est le barycentre de ces trames pondérées par la probabilité de la suite cachée. Ceci correspond à une estimation selon le maximum de vraisemblance.

Le nombre maximum de densités ainsi réestimées est égal au nombre de pavés – typiquement une centaine – multiplié par le nombre d'états. Cette étape peut être suivie d'un regroupement des densités, état par état. Le regroupement peut s'effectuer en utilisant deux critères : le nombre minimal de trames utilisées pour estimer cette densité et/ou la distance au sens de Kullback-Leibler [Tou 74] entre deux gaussiennes. Le modèle obtenu ainsi par éclatement est ensuite réestimé par plusieurs itérations avec l'algorithme *forward-backward* ce qui permet aux densités de se particulariser état par état.

16. *State clustering* en anglais.

À nos yeux, l'intérêt de notre méthode est dans l'utilisation d'un *codebook* initial, correctement estimé sur un nombre important de trames. Ce *codebook* est constitué par une division successive du corpus par l'algorithme de Buzo-Gray [Buzo 80] et il est ensuite réduit par un regroupement des classes selon un critère de variation de l'inertie inter-classe. Cette étape, dite *étape de consolidation*, permet d'éviter les divisions inutiles [Lebart 95] et de garder éloignés les représentants des classes. Les moyennes représentant les nouvelles classes obtenues par l'intersection de la classe (état, pavé) et de la trajectoire restent à l'intérieur de l'hypercube représentant la classe (état, pavé) et demeurent représentatives d'une portion de trajectoire et d'une portion de pavé alors qu'une classification des trames associées à l'état aurait été singulière en cas de trop petit nombre de trames. On peut voir les *genones* ainsi définis comme représentatifs de portions de trajectoires. Pendant la phase de regroupement, nous contrôlons la robustesse de l'estimation grâce au nombre minimum de trames qui doivent réestimer une classe et nous laissons à l'algorithme le soin de définir le nombre optimal de densités par état.

8 Reconnaissance

Les modèles de Markov doivent s'enchaîner les uns aux autres dans un réseau linguistique définissant la grammaire du langage à reconnaître. Dans le cas de la reconnaissance du corpus *OGI*, la grammaire autorise n'importe quelle suite de lettres en valuant la transition d'une lettre à l'autre par la probabilité d'apparition de cette transition estimée à partir de fréquences d'observations. La reconnaissance se fait par la recherche des N meilleurs chemins dans le réseau linguistique. Selon la grammaire utilisée, on a soit un algorithme de segmentation et d'étiquetage si on connaît la suite d'unités prononcées, soit un algorithme de reconnaissance. Les réponses peuvent être ensuite ré-ordonnées en fonction d'autres sources d'information comme celles mises en œuvre par des réseaux de neurones [Junqua 95b]. La suite de lettres peut être ensuite comparée d'une façon souple à tous les noms de l'annuaire (plus de 20000 entrées) pour déterminer le nom le plus proche.

Dans le cas de *BREF80*, à partir de phonèmes il faut reconnaître des mots pour construire des phrases. Ceux-ci sont représentés par leurs transcriptions phonétiques. La syntaxe est représentée par une matrice de bigrammes ou trigrammes qui représente la probabilité qu'un mot suive un mot donné (ou une suite donnée de deux mots dans le cas d'une matrice de trigrammes). Il faut noter que la langue française se laisse moins bien modéliser de la sorte que la langue anglaise. Dans le cadre du projet SQALE [Aubert 95, Dugast 95, Lamel 95] de reconnaissance de la parole continue multilingue, il a été noté que le grand nombre d'homophones présent en français comptait pour 21 % des erreurs lexicales du système. La langue anglaise, bien que plus difficile à reconnaître au niveau phonétique, permet des taux de reconnaissance plus élevés au niveau lexical.

9 Partage de paramètres

Afin d'améliorer la qualité de l'estimation des densités, celles-ci peuvent être partagées de différentes façons :

- pour obliger un modèle à capturer une plus longue durée de parole, une solution simple consiste à dupliquer les états qui vont partager la même densité ;
- pour modéliser des phonèmes en contexte, on peut décider de partager l'état central entre tous les contextes d'un même phonème ;

– pour des mots qui débutent ou finissent par un même phonème, on peut lier les états initiaux ou finaux des modèles correspondants; par exemple pour la reconnaissance de lettres épelées P et T on peut lier les deux derniers états.

Le partage peut se faire à différents niveaux de granularité :

– au niveau des états, comme nous venons de le voir ;

– à l'intérieur des états, en définissant des génones qui contiennent les lois normales qui servent à la définition des mélanges [Digalakis 94] ;

– dans la composante définissant le mélange, par exemple en partageant les matrices de covariances [Beulen 95].

Pour réaliser le partage de paramètres, il est souvent nécessaire de réaliser une classification et donc de définir une distance. Dans le cas de lois normales, la distance de Kullback-Leibler [Tou 74] est souvent utilisée.

10 Conclusion

Nous venons de voir les différentes phases et les multiples problèmes rencontrés lors de la réalisation d'un système de reconnaissance de la parole markovien. Le soin apporté au choix du corpus et à son étiquetage, à la définition des topologies des modèles, aux techniques de paramétrisation et d'apprentissage conditionne le résultat final. Contrairement à ce qui est admis, nous pensons que la qualité de l'estimation initiale des modèles influe sur le résultat. J'ai acquis dans ce domaine une expérience jamais explicitée dans des publications et néanmoins indispensable à l'équipe pour la mise en place d'applications pratiques.

Chapitre 4

Applications à la perception de signaux complexes

1 Application sur différents corpus de parole

1.1 Tests en reconnaissance

Nous avons utilisé les HMM2 pour construire différents systèmes de reconnaissance de la parole en anglais et en français. La modélisation se faisait au niveau du chiffre ou de la lettre épelée comme dans *TI-NIST* ou *OGI* ou au niveau du phonème comme dans *TIMIT* ou *BREF80*. Deux types de parole ont été utilisés : de la parole à grand rapport signal/bruit et de la parole téléphonique. Tous ces résultats ont été publiés dans les congrès internationaux *Eurospeech* [Mari 94b, Mari 94a, Junqua 95b] ou *Icassp* [Junqua 95c, Mari 96] entre les années 1994 et 1996. Pendant cette période de trois ans, nous avons pu régler soigneusement les détails de l'analyse acoustique. Le décalage temporel entre trames et taille de la fenêtre pour le calcul des coefficients de régression en sont les principaux. La modélisation stochastique a aussi progressé pendant cette période. Nous avons développé des outils permettant la ligature d'états (*tied states* en anglais) et la classification de trames, d'états ou de HMM. A chaque résultat de test il faut associer la date de l'expérience. Beaucoup de tests, s'ils étaient refaits, donneraient des résultats meilleurs. Les résultats ci-dessous témoignent seulement de nos efforts au cours des trois dernières années pour continuer à construire et à évaluer des systèmes de reconnaissance opérationnels. D'autres systèmes n'ont pas pu être testés ; je pense tout particulièrement au système de reconnaissance de mots isolés fondé sur le modèle psychologique d'accès lexical des cohortes et sur une théorie de la perception des corrélats acoustiques [Mari 84].

Corpus	nature	signal/bruit	analyse acoustique	% modèles	%phrases
TI-NIST	suites de chiffres	élevé	$\Delta\Delta$	99.2	98.5 (99.4)
OGI	noms épelées	téléphone	Δ	89	96
TIMIT	phonèmes en paro. cont.	élevé	$\Delta\Delta$	70.6 (75.0)	n.d.
BREF80	phonèmes en paro. cont.	élevé	$\Delta\Delta$	74.0 (78.7)	n.d.

TAB. 4.1 – *Nos meilleurs résultats sur les corpus actuels à l'aide de HMM2*

La table 4.1 donne les taux de reconnaissance (*accuracy*) de modèles et de phrases vues comme des suites de modèles. Le sigle *n.d.* signifie que cette donnée n'est pas disponible dans notre système. Nous mettons entre parenthèses la meilleure performance mondiale dans le domaine.

Les meilleures performances sur *TI-NIST* ont été obtenues par R. Haeb-Umbach et Y. Normandin [Haeb-Umbach 93, Normandin 94] qui utilisent des techniques d'analyse acoustique évoluées (utilisation d'analyse linéaire discriminante pour capturer la corrélation temporelle des trames) et d'apprentissage par maximum d'information mutuelle. Nous n'avons pas ce dernier outil ..., du moins pas encore. Mais les travaux menés par H. Li dans notre équipe [Li 95] devraient combler ce manque.

Aucun laboratoire n'a donné de résultats sur le corpus *OGI*. Nous nous intéressons à la parole téléphonique en raison de son importance dans les systèmes d'information. Pour cette raison nous sommes présents dans les projets COST 249 et COST 250 qui sont des groupes de réflexion et d'échanges concernant l'implémentation de systèmes de reconnaissance de la parole téléphonique. Le taux de reconnaissance de noms donné dans la dernière colonne est calculé en utilisant un dictionnaire de 4000 noms dans lequel est recherché le nom le plus proche de la suite de lettres reconnues.

Les résultats concernant *TIMIT* et *BREF80* sont intéressants car ils montrent que la modélisation stochastique à l'aide de HMM n'utilisant pas les approximations classiques : ordre un, apprentissage par algorithme de Viterbi, matrices de covariances diagonales, donnent des résultats prometteurs puisque nos HMMs de phonèmes sont encore indépendants du contexte. L'écart des performances entre *TIMIT* et *BREF80* ne dépend pas du système mais dépend de la difficulté de la langue. Un tel écart a déjà été remarqué par différentes équipes participant au projet de reconnaissance multi-linguale SQALE [Aubert 95, Dugast 95, Lamel 95].

1.2 Tests en étiquetage de corpus

Pour réaliser un système de dictée automatique, il est nécessaire d'augmenter les performances du système acoustique en utilisant des modèles plus précis comme des modèles dépendant du contexte et d'utiliser des coefficients de régression du deuxième ordre ou davantage de densités par états. Ceci implique l'estimation d'un plus grand nombre de paramètres et donc l'utilisation de plus gros corpus qui ne pourront plus être étiquetés manuellement. Notre but est de réaliser l'apprentissage d'un modèle de Markov en utilisant un corpus de parole dont on n'a que la transcription orthographique comme cela a été déjà décrit au § 5.3. Ce travail a été réalisé en collaboration avec Dominique Fohr [Fohr 96] à partir des programmes d'apprentissage et de reconnaissance développés plus haut.

Performances

Pour évaluer les performances de cette approche, il est nécessaire de posséder un corpus étiqueté qui va servir de référence. Nous avons choisi *TIMIT*, un corpus entièrement étiqueté manuellement. A l'aide de modèles de phonèmes et de la suite des phonèmes constituant la transcription de chacune des 1344 phrases de test, nous avons placé automatiquement les frontières temporelles des segments sur ce corpus de test. Les résultats sont donnés dans la table 4.2. Dans cette table, on représente le pourcentage de frontières trouvées autour de la frontière manuelle posée par l'expert du *MIT*. Ce pourcentage est donné comme une fonction de l'écart toléré de chaque coté de la frontière manuelle. Dans plus de 93 % des cas, la frontière est placée à moins de 30 ms de la frontière placée par l'expert du *MIT*, ce qui nous paraît un résultat raisonnable compte tenu de la difficulté de la tâche et du fait que un nombre important de frontières auraient pu être placées de façon différentes. Ce résultat montre que,

% de frontières dont l'écart est ≤ à 20 ms.	86.1%
% de frontières dont l'écart est ≤ à 30 ms.	93.2%
% de frontières dont l'écart est ≤ à 40 ms.	96.2%
% de frontières dont l'écart est ≤ à 50 ms.	97.7%

TAB. 4.2 – *Répartition des frontières en fonction de l'erreur tolérée (d'aprés [Fohr 96])*

si on est capable de produire la bonne séquence de phonèmes, notre procédure permet de placer automatiquement et de façon fiable les frontières phonétiques. En réalité, il n'existe pas de bonne séquence de phonèmes : s'agit-il de la suite que le locuteur a voulu produire ou de la suite qu'il a effectivement produite ? Dans *TIMIT*, nous avons utilisé ce dernier cas. Par contre, dans *BREF80*, nous nous rapprochons du premier cas puisque nous avons utilisé une transcription en phonèmes à partir de la transcription orthographique du texte. Ce problème est loin d'être trivial et aucune équipe ne donne de résultats sur le taux de reconnaissance en phonèmes excepté sur *TIMIT* qui possède une transcription normative. Sur les nouveaux corpus tels que *WSJ* [17] ou *BREF80*, c'est le taux de reconnaissance au niveau lexical qui est utilisé. Dans le cadre de l'évaluation des HMM2 de phonèmes à l'aide de *BREF80*, nous donnons des taux de reconnaissance de phonèmes calculés à partir d'une transcription graphème-phonème standard de la phrase [Fohr 96].

2 Partage des états et recherche d'allophones

Les meilleurs résultats en reconnaissance de la parole ont été obtenus avec des modèles de phonèmes en contexte : les allophones. Cette modélisation se fait en effectuant une classification hiérarchique des réalisations de chaque phonème en fonction de leurs contextes droit et gauche afin d'aboutir à la détermination de classes de contexte pour chaque phonème. Ceci a abouti à la définition de *triphones généralisés* [Lee 89].

2.1 Ligature des états de HMM

Dans le but d'obtenir des modèles de phonèmes en contexte, nous avons, nous aussi, réalisé différents outils de modélisation stochastique. La construction de triphones généralisés implique la maîtrise du partage des états des HMM ainsi que la comparaison de deux HMM. Ces techniques mettent en œuvre des algorithmes de classification et de calcul de distances entre états.

Grossièrement, un HMM peut être vu comme un vecteur d'états qu'il faut classer. Deux cas de figure se présentent :

- Les classes sont connues et on oblige certaines composantes du vecteur a être identiques ; cela revient à projeter le long d'un segment ; c'est le cas présenté dans le corpus *OGI* où les classes sont associées aux lettres articulées à l'aide de la même voyelle ; on projette le long de la voyelle sur le plan de la consonne.
- Les classes ne sont pas connues. Le processus de catégorisation [18] est effectué automatiquement sur les HMM ; c'est le cas présenté dans le corpus *BREF80* où nous recherchons des classes de phonèmes en contexte dont l'interprétation sera laissée au phonéticien.

17. Wall Street Journal : un corpus nord-américain équivalent à *BREF80*.
18. La catégorisation est le processus de recherche automatique des classes tel que le définit le chercheur d'IA.

Distances entre deux lois normales

Dans le cas où deux classes contiennent des populations normales, la divergence entre ces deux classes donne une mesure de leurs similarités. La notion de divergence est très clairement introduite dans l'ouvrage [Tou 74] aux pages 291 – 298. On l'appelle aussi la mesure de Kullback-Leibler. Nous avons choisi de calculer la distance entre deux lois normales associées à deux états différents par la mesure de Kullback-Leibler. Dans le cas où les deux lois ont la même matrice de covariances, une distance euclidienne entre les deux vecteurs moyenne suffit. La distance de Kullback-Leibler permet de généraliser au cas où les deux lois normales n'ont plus la même matrice de covariances. Par contre, il n'existe pas de distance aisément calculable sous forme analytique entre deux lois représentées par des mélanges de lois normales. C'est pour cette raison que la ligature des états se fait sur des HMM ne possédant qu'une densité par état.

Soient x un individu, w_i et w_j deux classes dans lesquelles x peut être classé. On note $p_i(x) = p(x/w_i)$ la probabilité d'apparition de x quand celui-ci appartient à la classe w_i. On définit de même $p_j(x) = p(x/w_j)$. L'information discrimante entre ces deux évènements se mesure par :

$$u_{ij} = \ln \frac{p_i(x)}{p_j(x)}$$

L'espérance de cette mesure pour la classe w_i s'écrit :

$$I(i,j) = \int_x p_i(x) \ln \frac{p_i(x)}{p_j(x)} dx$$

Pour avoir une mesure de ressemblance qui ne dépende pas de l'ordre des classes w_i et w_j, on pose :

$$Jij = I(i,j) + I(j,i)$$

Jij est la divergence entre les classes w_i et w_j. Cette quantité a une forme analytique quand les classes ont une répartition gaussienne :

$$p_i(x) = \mathcal{N}(\mathbf{x}; \mu_\mathbf{i}, \Sigma_\mathbf{i}) = \frac{1}{\sqrt{(2\pi)^p \det \Sigma_\mathbf{i}}} \exp^{-\frac{1}{2}(\mathbf{x}-\mu_\mathbf{i})^t \Sigma_\mathbf{i}{}^{-1}(\mathbf{x}-\mu_\mathbf{i})}$$
$$p_j(x) = \mathcal{N}(\mathbf{x}; \mu_\mathbf{j}, \Sigma_\mathbf{j}) = \frac{1}{\sqrt{(2\pi)^p \det \Sigma_\mathbf{j}}} \exp^{-\frac{1}{2}(\mathbf{x}-\mu_\mathbf{j})^t \Sigma_\mathbf{j}{}^{-1}(\mathbf{x}-\mu_\mathbf{j})}$$

Dans ce cas J_{ij} s'écrit :

$$\begin{aligned} J_{ij} = & \frac{1}{2} tr[(\Sigma_i - \Sigma_j)(\Sigma_j{}^{-1} - \Sigma_i{}^{-1})] \\ & + \frac{1}{2} tr[(\Sigma_i{}^{-1} + \Sigma_j{}^{-1})(\mu_i - \mu_j)(\mu_i - \mu_j)^t] \end{aligned}$$

Un cas particulier est intéressant :

$$\Sigma_i = \Sigma_j = \Sigma$$

Dans ce cas J_{ij} devient la distance de Mahalanobis :

$$J_{ij} = (\mu_i - \mu_j)^t \Sigma^{-1} (\mu_i - \mu_j)$$

Implémentation

Une implémentation de ces algorithmes a été faite sur le corpus de lettres épelées du corpus *OGI*. Ce corpus nous a servi de corpus de validation avant de tenter une recherche automatique d'allophones sur *BREF80*.

Les classes *a priori* envisagées étaient les ensembles de lettres anglaises articulées à l'aide du même segment vocalique tels que :

$$ESET = \{B, C, D, E, G, P, T, Z, V\}$$

ou

$$ASET = \{A, H, J, K\}$$

ou encore :

$$\{L, M, N, S, X\}$$

suivant le segment vocalique utilisé pour prononcer cette lettre : la voyelle /ɪ/ pour le ESET, la diphtongue /e i/ pour le ASET ou la voyelle /e/ pour $\{L, M, N, S, X\}$.

Au sein d'une même classe, l'algorithme de comparaison lie, suivant les cas, les derniers ou les premiers états des modèles en fonction de la place du segment vocalique qui y est tenu. Ces résultats ont été publiés à *ICASSP'96* [Mari 96]. Cette façon d'autoriser la ligature d'états en fonction d'une connaissance phonétique donnée *a priori* a permis une amélioration de 20 % du taux de reconnaissance car les modèles étaient plus discriminants. Les états associés au segment vocalique lorsqu'ils sont liés, participent plus à la séparation entre deux modèles de la même classe car la comparaison porte sur la partie informative du modèle : la consonne. Nous intégrons simplement de cette façon des connaissances phonétiques *a priori* dans un processus d'apprentissage stochastique.

Construction de HMM à états liés

Nous présentons ci-dessous, la démarche supervisée de construction de HMM de lettres épelées utilisée sur le corpus *OGI* ainsi que l'algorithme déterminant quels états doivent être liés.

Tous les modèles de lettres (excepté le W) ont 6 états. On appelle HMM^{α}, le HMM associé à la lettre α et on notera s_i^{α} l'état n° i du modèle associé à la lettre α. Deux lettres α et β sont dans la même classe phonétique si elles sont articulées sur la même voyelle (ou diphtongue).

Le démarche de construction peut être représentée par l'algorithme 4.1.

Le seuil R_0 contrôle la taille des listes. Nous le choisissons en fonction de la distance $d(s_i^{\alpha}, s_i^{\beta})$ évaluée sur les 2 derniers états d'un modèle du ESET et du modèle du /ɪ/. Plus R_0 est faible, moins d'états seront liés.

Dans cet algorithme, la ligature des états permet une meilleure discrimination entre modèles par l'utilisation de connaissances phonétiques. L'apport de connaissances se traduit par l'introduction de contraintes sur les ligatures, ce qui constitue une originalité par rapport à un apprentissage non supervisé ou les listes L_i ne sont pas modifiées par un expert. Dans le paragraphe suivant, la ligature des états permettra de dégager des classes d'allophones permettra une meilleure estimation de modèles de phonèmes en contexte puisque chaque modèle sera appris sur un corpus d'apprentissage plus vaste.

`Ligature (corpus, classes phonétiques, HMM`$^{\mathtt{A}}$`, HMM`$^{\mathtt{B}}$`, ... , HMM`$^{\mathtt{Z}}$`)`

entrée : un corpus de lettres épelées, l'ensemble des classes phonétiques.

sortie : les 26 HMM de lettres avec états liés.

1. Apprendre un modèle par lettre. Chaque état est représenté par une *pdf* normale.

2. **pour** i = 1 à 6 **faire**
   $L_i \leftarrow ()$ /* liste vide */
   pour $\alpha \in \{A, B, C, \ldots, Z\}$ **faire**
      pour $\beta \in \{A, B, C, \ldots, \alpha\}$ **faire**
         Calculer la distance de Kullback-Leibler $d(s_i^\alpha, s_i^\beta)$
         si $d(s_i^\alpha, s_i^\beta) \ \leq \ R_0$ **et** α et β sont dans la même classe **alors**
            ajouter s_i^α, s_i^β dans la liste L_i
         fin
      fait
   fait
   fait

3. Donner ces 6 listes à un expert qui décide pour chaque couple (s_i^α, s_i^β) si l'état n° i de HMM^α et HMM^β peut capturer le même segment vocalique. Cette décision peut se faire en visualisant la spécialisation de l'état (cf. figure 4) ou en regardant la place de l'état dans le modèle : début ou fin. Le résultat du travail de "l'expert" est une liste expurgée de tous les couples dus au manque de résolution temporelle des HMM2 ne possédant qu'une *pdf* par état. Cette liste détermine quels HMM2 partagent l'état n° i.

4. Effectuer une estimation à l'aide de l'algorithme de Baum-Welch pour rendre optimaux au sens du maximum de vraisemblance ces nouveaux HMM.

5. Éclater chaque densité en plusieurs à l'aide de la méthode donnée au chapitre 3, page 54.

6. Effectuer une dernière estimation à l'aide de l'algorithme de Baum-Welch.

Algorithme 4.1: Création de modèles partageant des états sur *OGI*

2.2 Recherche d'allophones

Dans le cadre de la recherche d'allophones, nous n'avons publié aucun résultat. Nous utilisons un algorithme identique à celui de S. Young [Young 94]. Toutefois, nous avons déjà retrouvé pour certaines voyelles, les classes d'allophones suivant leur contexte gauche. La table 4.3 donne les classes de contexte gauche pour le phonème /a/. Ces résultats ont été obtenus à l'aide de *BREF80*.

antérieures	i+a e+a j+a k+a g+a
dentales et chuintantes voisées	ə+a y+a ø+a l+a z+a d+a −+a
postériorisées	o+a u+a ã+a ɔ̃+a R+a w+a
nasales	ɛ̃+a m+a n+a
labiales sourdes	f+a p+a
dentales et chuintantes sourdes	s+a ʃ+a t+a
labiales sonores	v+a b+a

TAB. 4.3 – *Les différents allophones du phonème /a/*

Il est intéressant de noter que la classification opérée par l'algorithme a un sens pour un phonéticien. Ainsi, par une étude statistique d'un gros corpus d'apprentissage, les processus stochastiques permettent de retrouver des régularités de comportement d'allophones en dégageant des classes de contexte. Nous retrouvons une information d'ordre symbolique : les classes de contexte à partir d'une information d'ordre numérique : le corpus. Il s'agit d'un problème de fouille de données tel qu'il est abordé en Intelligence Artificielle.

3 Application à la reconnaissance d'autres signaux

La parole peut être définie comme un signal pseudo-stationnaire d'origine biologique pour lequel la modélisation stochastique à l'aide de HMM fournit de bons résultats en reconnaissance. Parallèlement à cette activité, nous avons mené une étude visant à classer des sons sous-marins d'origine mécanique ou biologique en les modélisant à l'aide de HMM2. Une autre étude a été menée sur un autre type de signaux complexes. Il s'agit de la reconnaissance de l'écriture sous forme cursive et imprimée.

3.1 Reconnaissance de bruits sous-marins

Cette étude s'est faite en collaboration avec la direction des constructions navales (DCN). Un ingénieur expert, embauché dans le cadre d'un contrat a utilisé les HMMs du second-ordre pour reconnaître et classer des bruits sous-marins réels ou simulés.

Les figures 4.1 et 4.2 montrent deux exemples de bruits sous-marins : des sifflements modulés de dauphins et une série de chocs secs et graves.

Les échelles de fréquences sont différentes dans les deux figures. Alors que la figure 4.1 montre des sifflements dans la gamme de 3000 à 5000 Hertz, les chocs montrés dans la figure 4.2 se situent entre 100 et 1000 Hertz. Résumons en quelques points les difficultés de traitement dues à la nature des signaux.

Bande passante : les dauphins peuvent moduler entre mille et plus de seize mille Hertz. Quant au cachalot, il imite très bien le choc sourd, résonnant et riche en très basses fréquences (100 Hertz ou moins).

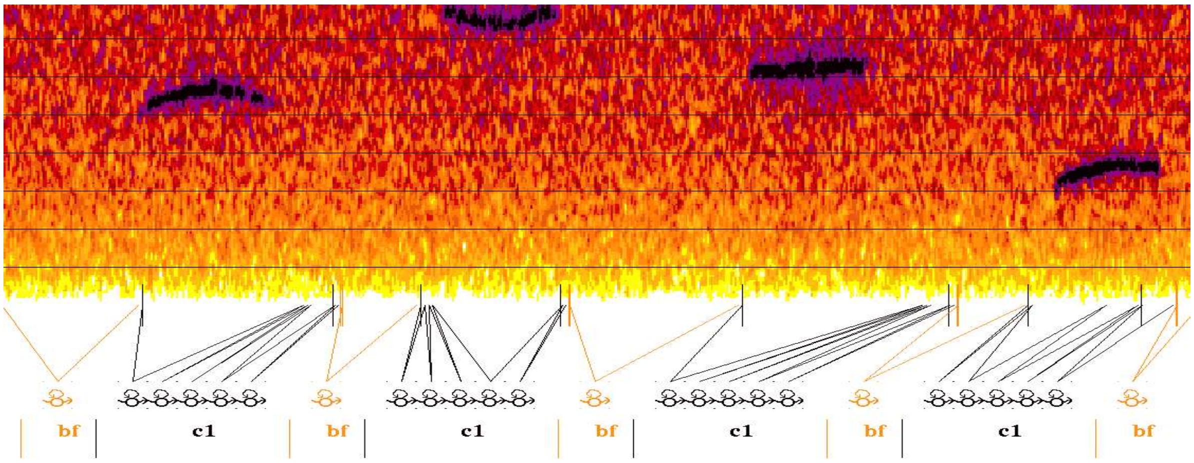


FIG. 4.1 – *Sifflements modulés biologiques*

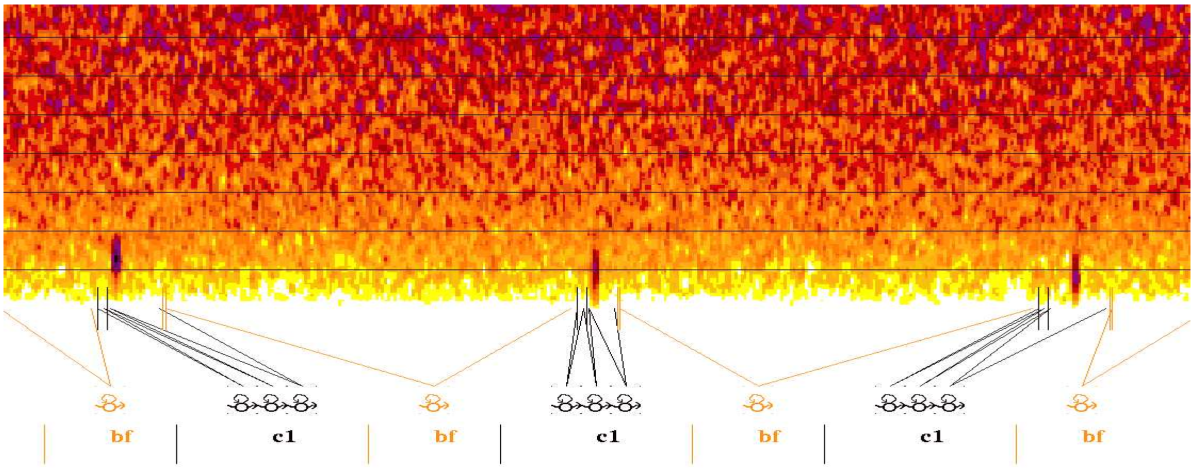


FIG. 4.2 – *Série de chocs secs et graves*

Variabilité intra-classe : les chocs peuvent se produire dans différentes gammes de fréquences sans que celles-ci soient significatives. La durée des événements peut être variable.

Variabilité inter-classes : parmi les signaux on peut distinguer des signaux de type impulsif comme les chocs, des signaux à large bande et des signaux à bande étroite modulés comme les sifflements de dauphins.

Présentation des signaux

Le choix des classifieurs a été fait en fonction des caractéristiques de chacune des classes à discriminer. Comme il a été mentionné plus haut dans la présentation, la variété des sources sonores : dauphins, crevettes, cachalots pour les bruits biologiques ainsi que les bruits mécaniques de toutes origines crée un univers sonore bien plus vaste que l'espace sonore dans lequel notre cerveau perçoit de la parole. La principale différence avec la parole est que les classes homogènes restent à découvrir, alors que les études linguistiques en parole nous ont permis de choisir *a priori* une modélisation en mots ou en phonèmes. A l'heure actuelle, nous ne connaissons pas de manière exhaustive le nombre de classes d'où sont issus ces signaux. Nous utilisons une approche incrémentale qui consiste à définir et entraîner des nouveaux modèles au fur et à mesure que la nécessité s'en fait sentir.

La topologie des HMM modélisant chaque classe doit tenir compte du caractère impulsif et/ou répétitif de certains signaux. Dans cette optique, nous testons des topologies qui ne sont plus simplement gauche-droite et dans lesquelles un état peut être re-visité.

Présentation des traitements

Les traitements sont analogues à ceux effectués en reconnaissance de la parole. La paramétrisation des signaux se fait à l'aide de cepstres sans utiliser d'échelles Mel mais en adoptant des largeurs de bande différentes pour tenir compte des possibilités ultra sonore ou infra sonore de certaines sources. La soustraction spectrale donne sur ce type de signaux des résultats intéressants. Nous utilisons aussi des coefficients de régression d'ordre un et deux qui montrent leur aptitude à capturer la dynamique du signal.

L'utilisation de l'analyse discriminante linéaire (*Linear Discriminant Analysis* ou LDA) comme méthode de projection dans un espace de dimension moindre a été utilisée en reconnaissance de la parole continue par K. Paliwal [Paliwal 92], H. Ney [Haeb-Umbach 93] et dernièrement par K. Beulen [Beulen 95]. Son principal intérêt est de permettre une estimation plus robuste des paramètres des modèles.

Sur un petit corpus de test comportant 150 signaux simulés mais validés et étiquetés par un expert, nous avons comparé les résultats d'une reconnaissance en 5 classes à l'aide de trames d'analyse dont la dimension avait été réduite grâce à une analyse linéaire discriminante. Dans cette expérience nous n'utilisons qu'une seule classe de projection.

dimension d'une trame	24	20	16	9
taux de reconnaissance	77.8	78.1	79.6	78.4

TAB. 4.4 – *Performances comparées suivant la dimension d'une trame d'analyse*

La table 4.4 montre les performances comparées sur ce corpus suivant la dimension d'une trame d'analyse constituée initialement des 11 coefficients statiques, 12 coefficients de régression du premier ordre et de l'accélération de l'énergie. Les résultats montrent une amélioration quand on réduit la taille de la trame d'analyse à 16 coefficients. Il nous reste à explorer l'influence de la LDA en reconnaissance du discours continu. Jusqu'à présent, les corpus d'apprentissage utilisés nous ont paru suffisants pour ne pas à avoir à utiliser cette technique. De plus, les derniers résultats publiés par K. Beulen [Beulen 95] en reconnaissance de la parole, montrent que l'information contextuelle est aussi bien modélisée par les désormais classiques coefficients de régression que par un grand vecteur obtenu par concaténation de plusieurs trames de coefficients statiques puis réduit par LDA. Une petite amélioration peut être obtenue par l'utilisation simultanée des deux méthodes.

3.2 Reconnaissance de l'écriture

Tout comme pour la parole, les méthodes stochastiques à base de modèles de Markov cachés peuvent s'appliquer à la reconnaissance de textes écrits aux niveaux morphologique, lexical et syntaxique. En effet, il existe une grande similitude dans les comportements d'un phonème et d'un caractère quand ceux-ci sont placés dans un contexte. Particulièrement dans l'écriture cursive, la forme d'un caractère est affectée par ses voisins. On peut extraire des régularités de comportement par l'étude statistique de corpus d'apprentissage représentatifs de l'ensemble des contextes possibles. Le problème est différent dans le cas de caractères imprimés dont la forme est invariable. Dans ce domaine, nous nous sommes intéressés à la reconnaissance de textes imprimés multifontes en caractères latins [Anigbogu 92] et arabes [Amin 85, Amin 89].

Dans ces systèmes de reconnaissance, le signal issu du scanner est tout d'abord traité à des fins de segmentation : segmentation du texte en lignes, segmentation de lignes en mots et segmentation de mots en caractères. Ces derniers sont ensuite représentés par une succession de primitives graphiques comme les codes de Freeman qui représentent les 8 directions spatiales d'un segment de droite. Après ces traitements, le caractère à modéliser est représenté par une suite de symboles discrets. Par convention, le codage débute toujours au même endroit dans le caractère : en haut à droite pour l'arabe ; en haut à gauche pour le latin. Ceci transforme le signal bidimensionel issu du scanner en une suite monodimensionelle.

Pendant longtemps, les méthodes de représentation des caractères en primitives de base ont fait que l'écriture était vue comme un processus monodimensionnel qui transformait l'image des mots en une séquence ordonnée de symboles. Cette description limitée est actuellement contestée par l'utilisation de modèles pseudo-1D ou complètement 2D tirés de l'analyse d'images [Saon 96].

A. Amin [Amin 89] n'utilise des modèles de Markov discrets que pour la reconnaissance des mots à partir de la suite de caractères multifontes. La reconnaissance des caractères à partir d'un texte imprimé multifonte est réalisée par deux niveaux distincts : un module de reconnaissance de caractères travaille sur le signal issu du scanner pour le segmenter et identifier les caractères par des méthodes structurelles basées sur une description morphologique des caractères. Le niveau lexical est fondé sur l'algorithme de Viterbi qui fonctionne sur la suite de caractères et peut apporter une correction à l'identification en caractères effectué par le niveau inférieur. En effet, à chaque mot est associée une source de Markov produisant des caractères. En 1989, la vitesse de traitement du système de reconnaissance était proche de 3 caractères par seconde avec un taux de reconnaissance de mots de 90 %.

J. Anigbogu [Anigbogu 92] utilise un type de paramétrisation différent dans des modèles de Markov discrets pour reconnaître des caractères. Le contour extérieur du caractère délimitait une région fermée du plan qui était partagée en 8 parties : les octants. Dans chaque octant on calculait un vecteur de paramètres en rapport avec la forme du caractère dans cet octant. A chaque caractère était ainsi associée une source de Markov d'ordre un ou deux générant 8 vecteurs. Il est intéressant de noter que plusieurs topologies ont été essayées et que la topologie ergodique convergeait vers une topologie gauche-droite.

Plusieurs expériences [Anigbogu 95] ont été réalisées sur des ensembles de 15 000 symboles dans chacune des 10 fontes courantes : *Chicago, Geneva, New York,* ... . Les performances dépendent de la fonte utilisée et du nombre d'états du HMM2. Elles oscillent entre 94.01 % pour la fonte *courrier* et un modèle à 4 états et 99.83 % pour la fonte *Geneva* et un modèle à 8 états. Un système à base de HMM1 donne des résultats semblables, ce qui est normal puisque les meilleurs résultats sont obtenus avec un nombre d'états sensiblement égal au nombre d'observations. Par contre, à faible configuration – 4 ou 5 états – les HMM2 donnent de meilleurs résultats que les HMM1.

4 Conclusions

Nous venons de décrire nos travaux en modélisation stochastique à l'aide de modèles de Markov cachés et leur application à la reconnaissance de signaux monodimensionels comme la parole, les bruits sous-marins ou l'écriture. Notre apport dans ce domaine se situe principalement dans la modélisation à l'aide de modèles de Markov d'ordre supérieur. Dans le cadre de l'apprentissage par maximum de vraisemblance, l'espace d'apprentissage a une vraisemblance supérieure avec des modèles d'ordre deux comparés aux modèles d'ordre un. Nous n'avons pas noté de phénomènes de sur-apprentissage puisque le même écart de performances se retrouve sur les corpus de tests. Nous pensons que les HMM2 modélisent mieux la durée des segments transitoires et stationnaires qu'ils soient d'origine biologique ou mécanique, ce qui explique leurs meilleures performances.

La modélisation stochastique implique l'utilisation de nombreux outils statistiques comme les algorithmes de classification, d'analyse discriminante, d'apprentissage selon différents critères et de reconnaissance. La constitution de cette boîte à outils a été notre préoccupation ces 5 dernières années ainsi que son évaluation dans différents domaines de la reconnaissance des formes ou de l'intelligence artificielle.

Dans notre souci de modéliser la variabilité temporelle sans chercher à l'expliquer, la modélisation stochastique nous a montré rapidement ses points forts : le caractère mathématique de ses algorithmes et les possibilités d'apprentissage automatique. Dans l'application de reconnaissance de lettres épelées, les HMM ont montré qu'ils pouvaient utiliser une connaissance phonétique donnée *a priori* sous forme d'une liste de contraintes représentées par des ligatures d'états. Inversement, dans le cas de la recherche d'allophones, les HMM ont construit un ensemble de classes d'allophones dont le comportement présentait certaines régularités. On peut donc voir les HMM comme des outils adaptés à la fouille des données de signaux temporels ; ils constituent une réponse à la perception de signaux complexes.

Il reste que nos méthodes doivent être améliorées et diversifiées. L'utilisation de nouveaux critères d'apprentissage, l'approximation de lois de probabilité par des réseaux de neurones ou encore la reconnaissance dans plusieurs bandes de fréquences menée en parallèle [Bourlard 96] sont des pistes de recherche à exploiter. Malgré tout, nous pensons que la reconnaissance de la parole ne doit pas se limiter à l'utilisation exclusive de modèles stochastiques dont l'appétit en corpus ne décline toujours pas. Sommairement parlant, on aimerait que l'intelligence artificielle propose des modèles pour représenter des connaissances. Ainsi, en introduisant des nouvelles contraintes, il sera possible diminuer la taille des corpus [Emptoz 96].

Chapitre 5

Applications à l'interaction homme-machine

1 Introduction

Ce chapitre présente trois systèmes de reconnaissance de la parole dont j'ai contrôlé la réalisation de plus ou moins loin. Le premier système que nous décrivons, intègre la reconnaissance de la parole dans un système plus vaste d'interprétation de requêtes pour W3. Il a été réalisé pendant le stage d'un élève ingénieur que j'ai encadré. Les deux autres systèmes ont été réalisés à partir du progiciel HMM2 que j'ai développé ces 6 dernières années. L'implémentation a été faite par des stagiaires de différentes filières d'enseignement ou des ingénieurs sous contrat.

Parmi ces systèmes, deux ont été réalisés dans le cadre du projet régional IRMA. Ce projet régional, présenté dans le cadre des projets de recherche du contrat de plan État-Région lancé en 1992, n'a pas été retenu dans les affectations budgétaires de ce contrat mais a reçu une aide financière de la Région Lorraine pendant la période 1994-1995.

IRMA est un acronyme signifiant "Intégration Répartie Multimodale pour le multimédiA". L'objectif de ce projet est la définition et la mise en œuvre d'interactions multimodales homme-machine pour un accès intelligent à des informations multimédias réparties. Les mots clés de ce projet sont : interaction par le geste et la parole, adaptation aux utilisateurs pour fournir des accès tolérants aux informations multimédias, dialogues naturels, nouveaux protocoles de communication pour gérer sur réseau des échanges d'informations multimédias dans le cas d'interactions réparties.

Ce projet a poursuivi une politique de concertation avec des partenaires industriels de la Région Lorraine. Cette concertation a débouché sur l'organisation de deux séminaires au cours desquels les équipes participantes ont présenté leurs compétences dans ces domaines. C'est ainsi que notre équipe s'est proposée pour réaliser le maillon de reconnaissance de la parole intervenant dans la communication orale avec un système multimodal d'accès intelligent à des informations multimédias réparties. Ce maillon doit permettre à terme la localisation des mots dans un continuum sonore représentant de la parole spontanée éventuellement bruitée puis fournir une représentation symbolique utilisable par les niveaux supérieurs d'interprétation du système multimodal. Les deux systèmes présentés dans le cadre du projet IRMA n'ont pas cette prétention. Ils témoignent simplement de nos compétences dans le domaine de la reconnaissance de la parole.

Le système d'interprétation de requêtes pour W3 permet à un utilisateur de demander oralement des documents tout en utilisant un logiciel d'accès à Internet comme Mosaic. Il a été publié aux journées d'études sur la parole [Thiébaut 96]. Il a aussi permis de répondre à un appel d'offres national lancé par le CNET sur l'étude de la multimodalité dans un système multimédia.

Pendant la réalisation de ce système, nous nous sommes rendus compte que la conception d'un système d'interprétation de requêtes à partir d'un système de reconnaissance déjà opérationnel – Vinics – et d'un système d'accès à W3 – Mosaic – était un problème ardu mettant en œuvre des méthodes élaborées de construction de logiciels comme la gestion automatique des différentes versions des fichiers source d'un projet parmi un ensemble de programmeurs ou l'utilisation de primitives de synchronisation entre tâches (*remote procedure calls*) afin de ré-utiliser au mieux les programmes existants. Ce projet a constitué un investissement de 4 mois de la part d'un ingénieur stagiaire qui a réussi à ré-utiliser les codes des programmes Vinics et Mosaic. La taille de l'interpréteur est du même ordre que ces deux derniers programmes. Cette volonté de ré-utilisation se traduit naturellement par un accroissement du temps de réponse, donc une diminution de convivialité entre le système et l'utilisateur. Pourtant ce système constitue un système multimédia qui doit nous permettre de découvrir la place de la parole dans l'interaction entre un usager et W3.

Le deuxième système est né à la suite d'une visite du laboratoire par une petite société de service régionale. Celle-ci était intéressée par la reconnaissance de la parole pour permettre à un serveur de prendre des commandes dans une salle de restaurant. Ces deux applications utilisent des méthodes originales de modélisation stochastique développées dans notre équipe et sont des maquettes de démonstration qui doivent s'intégrer dans des systèmes de communication homme-machine. Différents aspects de l'interaction orale dans la communication homme-machine sont mis en valeur dans ces deux systèmes : spontanéité et simultanéité de la parole avec d'autres formes d'interaction. Ainsi le projet IRMA nous a permis de présenter ces deux systèmes en les rattachant au thème de l'interaction homme-machine.

Le troisième et dernier système présenté dans ce chapitre concerne l'audio-surveillance des couloirs du métro. Il se différencie des deux précédents pour au moins deux raisons : l'entreprise partenaire n'est pas lorraine et il est difficile de parler d'interaction entre l'homme et la machine ; l'homme ou la femme crie et la machine décide s'il s'agit d'une agression. Disons qu'il s'agit d'un système d'aide qui est une forme particulière d'interaction.

2 Une souris vocale pour W3

Internet est un ensemble mondial de réseaux inter connectés. Divers protocoles de communications cohabitent sur celui-ci, et notamment World Wide Web plus connu sous ses abréviations WWW ou encore W3. Divers logiciels permettent d'exploiter le réseau de W3, citons entre autres Mosaic et Netscape. Ces logiciels permettent à l'utilisateur de visualiser des documents à l'aide d'une interface graphique. Par un simple "click" de la souris sur une phrase soulignée ou une image, l'utilisateur peut alors visualiser un nouveau document. A l'heure actuelle, grâce à l'apparition des systèmes multimédia, l'interaction homme-machine peut se faire à l'aide de nouveaux modes de communication plus naturels et spontanés tels que le geste ou la parole. Dans cet ordre d'idée, il est séduisant d'envisager un dialogue oral pour accéder à W3. Pour tester l'intégration de la parole dans un système multimodal, nous avons modifié Mosaic pour qu'un utilisateur puisse désigner des documents oralement grâce à un logiciel de reconnaissance de la parole continue développé au CRIN : Vinics [Gong 94a].

2.1 Interaction homme-machine durant une session W3

Comme tout système de reconnaissance de parole continue, Vinics nécessite l'apport d'une grammaire. Dans le contexte de notre application, cette grammaire décrit un ensemble de requêtes que l'utilisateur peut prononcer. Celle-ci est constituée de deux parties : une partie statique qui introduit la requête comme «je voudrais ... passez moi ...» et une partie dynamique qui particularise la requête en désignant le point d'ancrage souhaité. Une des limitations actuelle de l'interface est de ne pouvoir désigner que des points d'ancrage représentés par du texte. On ne peut pas demander d'image, surtout s'il y en a plusieurs !

La maquette réalisée (cf figure 5.1) nous a permis d'identifier plusieurs comportements des utilisateurs face au système, ce qui nous a conduit à définir plusieurs modes d'interaction :

- certaines personnes ne connaissent pas le contenu des pages ou la localisation de la page qui correspond à leur recherche. Ainsi, le plus simple pour celles-ci est de donner un mot-clé indiquant leur recherche ; c'est le mode « débutant » ;
- un deuxième type de personnes naviguant fréquemment sur le réseau et par conséquent connaissant bien quel point d'ancrage est lié au document recherché peut simplement faire une requête comportant exactement la suite de mots correspondant à ce point d'ancrage ; c'est le mode « expert » ;
- d'autres personnes, bien que non expertes, souhaitent avoir plus de possibilités que n'en propose le mode débutant. Par exemple, elles veulent faire une recherche par rapport à une ancienne page déjà analysée. Le mode « choix » permet ce type de recherche, il permet aussi de choisir entre plusieurs adresses de page (URLs) lorsque le mot-clé prononcé en réfère plus d'une.

Un des avantages principaux de l'entrée vocale, telle que nous l'avons implantée, est de pouvoir demander des documents dont la référence n'est pas présente à l'écran, mais qui se situent dans une périphérie proche de la page courante, soit sur les pages précédemment affichées, soit sur des pages pointées par des liens contenus dans la page courante. Cette possibilité de naviguer indépendamment des choix présentés à l'écran permet d'exploiter le parallélisme entre le geste qui conduit la souris et la parole qui demande une référence de la mémoire. Ceci doit conduire à des interactions homme-machine plus riches.

2.2 Modification de Mosaic

Lorsque l'utilisateur demande un document, Mosaic envoie une requête au serveur le possédant. Si ce document est une page HTML, il est possible de récupérer les phrases des points d'ancrage. La recherche se poursuit avec les points d'ancrage de ces nouvelles pages dans une limite de profondeur donnée (actuellement 2). Ainsi, un utilisateur pourra accéder oralement à des pages sans voir leurs points d'ancrage, ce qui constitue une possibilité supplémentaire par rapport à une désignation par souris comme nous l'avons déjà indiqué.

2.3 Modification de Vinics

Vinics est un ensemble de processus séparés permettant la reconnaissance de phrases appartenant à un langage régulier. Cet ensemble de processus regroupe l'analyse acoustique, la reconnaissance, la compilation et la génération automatique des grammaires. Il existe une

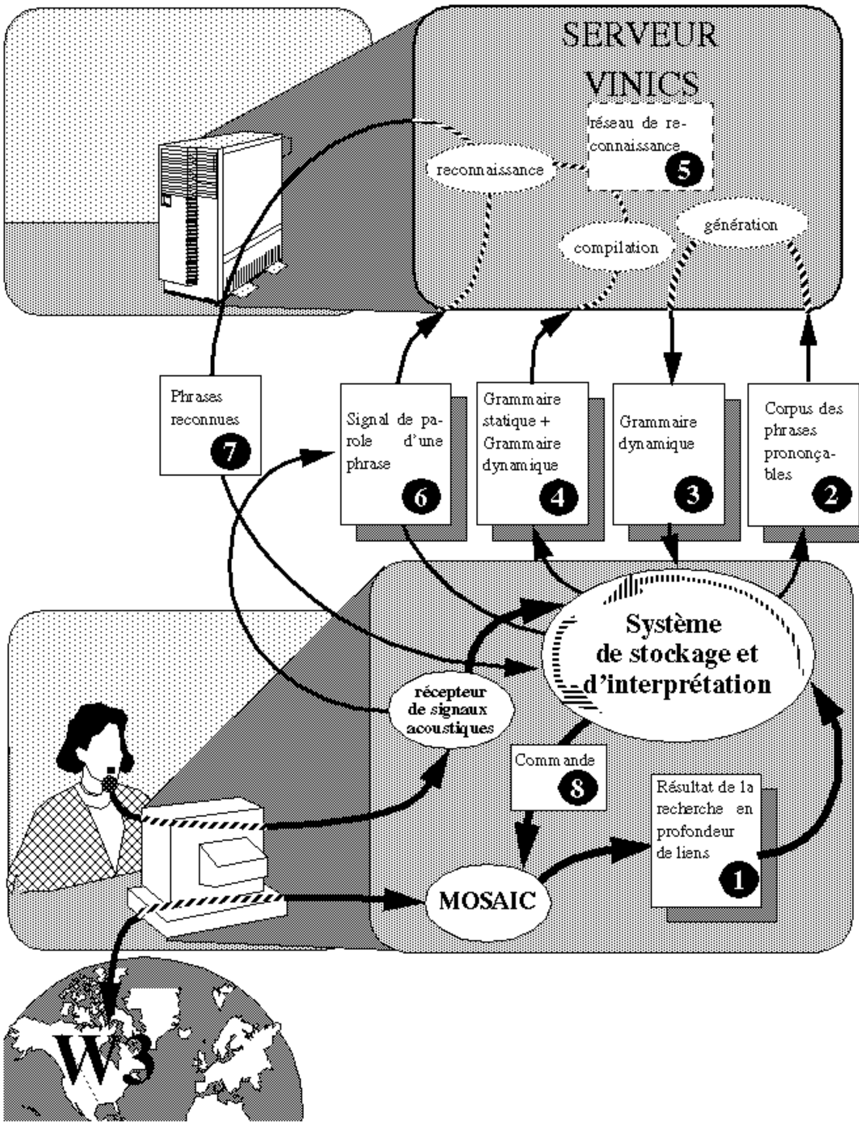


FIG. 5.1 – *Dépendances inter-processus entre Vinics et Mosaic*

liaison de communication entre le système de reconnaissance et l'application permettant une nouvelle compilation de grammaire à chaque nouvelle page.

Vinics effectue une reconnaissance de phrases en commençant par une reconnaissance phonétique dont les paramètres dépendent du locuteur. La recherche lexicale est conduite par un analyseur syntaxique qui valide des hypothèses de mots à l'aide des transcriptions en phonèmes stockées dans le dictionnaire BDLEX [Pérennou 92]. La figure 5.1 décrit le fonctionnement de VINICS et son interaction avec Mosaic.

Enregistrement d'un utilisateur : l'enregistrement d'un utilisateur nécessite que la phase d'apprentissage ait déjà été réalisée. Ce service permet l'initialisation de l'ensemble des paramètres propres à l'utilisateur. Il comporte donc une base de données contenant, pour chaque utilisateur potentiel, ses paramètres caractéristiques, comme par exemple les modèles de phonèmes.

Génération et compilation automatique d'une grammaire : la reconnaissance d'une phrase exige la présence d'un réseau de reconnaissance (l'élément N° 5 dans la figure 5.1). Ce réseau provient de la compilation de la grammaire (4), possédant une partie statique et une partie dynamique (3). Cette deuxième partie est générée automatiquement à partir d'un corpus de phrases (2), lui-même élaboré à partir de l'ensemble des liens (1) que Mosaic a découvert lors de sa recherche dans W3.

Reconnaissance d'une phrase : la reconnaissance d'une phrase est menée à partir du signal de parole (6) représentant cette phrase. Elle utilise le dernier réseau généré lors de la compilation ou du chargement d'une grammaire.

Le résultat renvoyé est une suite de phrases classées par ordre décroissant de taux de reconnaissance. Une fois les phrases reconnues (7), celles-ci sont interprétées de manière à livrer à Mosaic une commande (8).

2.4 Interprétation d'une requête provenant de Vinics

Les requêtes proviennent de Vinics sous forme de phrases. Ces requêtes sont de deux type : soit il s'agit d'un changement de mode de l'interpréteur ; soit il s'agit d'une demande de document. Dans une première étape, le début de phrase est analysé pour déterminer ce type. S'il s'agit d'une commande, celle-ci est alors soit exécutée dans le cas d'un changement de mode, soit transférée vers Mosaic. Si la requête est une demande de page, elle est alors analysée de manière à obtenir la phrase clé qui va permettre de rechercher à travers le graphe des données l'URL correspondant au document souhaité. Suivant le mode dans lequel se trouve l'application, cette phrase clé est une série de mots ou un mot seul dont l'interprétation est fonction de ce mode. Nous allons donc, pour chacun de ces modes, donner un exemple d'interprétation de requête.

2.5 Exemples d'utilisation

On considère une consultation au serveur du CRIN. La page donnée dans la figure 5.2 est située juste sous la page d'accueil.

Mode débutant

L'utilisateur a sous les yeux la page ci-dessus. S'il prononce la phrase « je voudrais les axes », c'est le point d'ancrage *Axes Recherche* qui est choisi car c'est la première occurrence de ce point d'ancrage qui est retrouvé dans le texte HTML. Si ce mot avait été absent, il aurait été recherché dans les pages connexes (cf. N° 4 dans la figure 5.1).

Mode expert

Dés la page d'accueil, dans ce mode, l'utilisateur peut prononcer « je voudrais la recherche fondamentale et appliquée en informatique » ou «je voudrais les axes recherche». L'effet est le même que si l'utilisateur avait désigné ce point d'ancrage dans la page ci-dessus. Par contre, l'utilisateur est obligé de prononcer l'intégralité du point d'ancrage souligné par opposition au mode débutant ou seulement une partie suffisait.

Mode choix

Après avoir prononcé la phrase «je voudrais la recherche», le système analyse toutes les

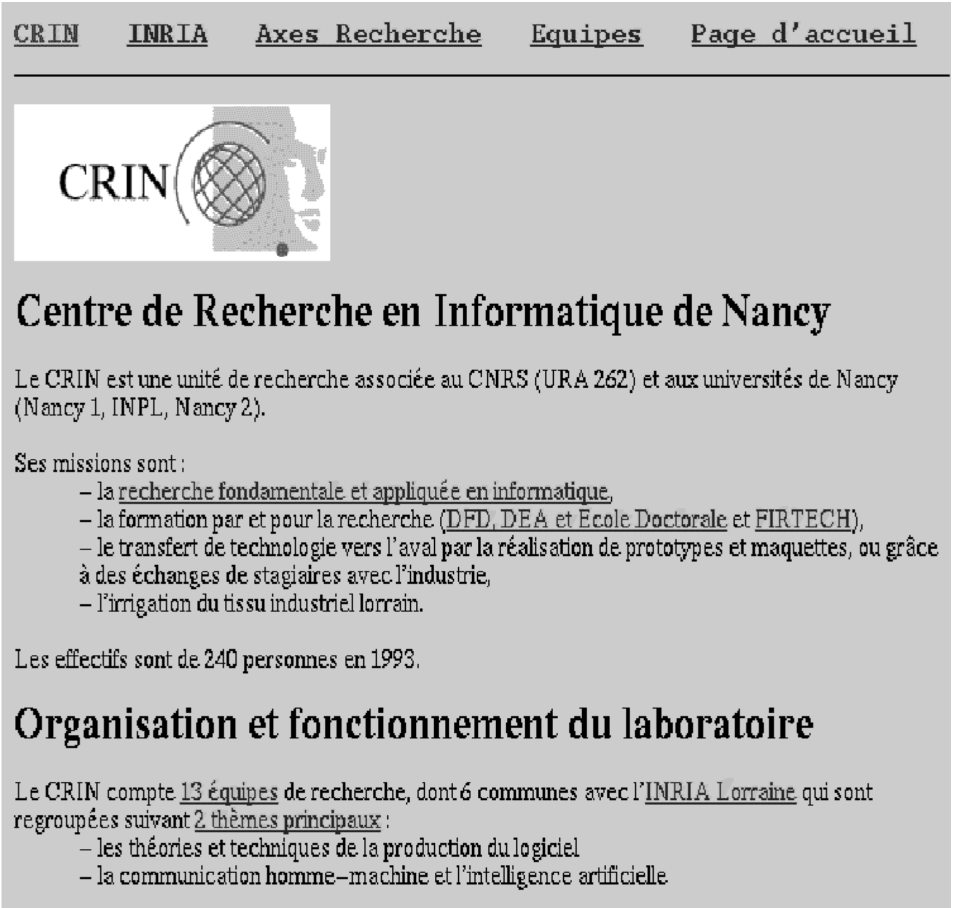


FIG. 5.2 – *Une page du serveur du CRIN*

occurrences du mot «recherche» depuis la page d'accueil et les pages déjà accédées. Il propose ensuite les choix suivants :

1. recherche fondamentale et appliquée en informatique ?
2. axes recherche ?
3. ...

2.6 Intérêt de cette maquette

Ce système constitue un bon terrain d'expériences pour mettre à jour et résoudre les problèmes de demande de renseignements dans les systèmes de demain et d'après-demain combinant plusieurs modalités d'interaction. A l'heure actuelle, nous ne savons pas comment les utilisateurs s'approprieront cette nouvelle forme d'interaction. Nous pensons que cette maquette deviendra un outil pour découvrir et prédire quels sont les comportements des usagers dans une interaction alliant le geste et la parole. Cet outil devrait intéresser tous ceux qui travaillent sur l'interaction multimodale homme-machine.

A n'en pas douter, les concepteurs de pages HTML devront tenir compte de cette nouvelle interaction quand ils spécifieront les points d'ancrage. Trop souvent, ces mots ne correspondent pas au sens général de la page qu'ils référencent. De plus, les points d'ancrage sont souvent constitués d'un ou deux mots, ce qui peut engendrer des ambiguïtés. En fonction des demandes des utilisateurs, nous pourrons décider quel type de reconnaissance (mots clés, parole continue, langage artificiel ou quasi-naturel) sera adéquat et ergonomique.

3 Prise de commandes les mains libres

Toujours dans le but d'explorer la place de la parole dans un système multimodal, nous avons réalisé un prototype de machine destinée à aider un serveur de restaurant à saisir la commande d'un client. Ce système permet au serveur qui prend une commande de conserver les mains libres.

La parole permet, pour un utilisateur, d'interagir avec un système en parallèle avec d'autres formes d'interaction. Ce système fait partie de la famille des systèmes permettant la rédaction de comptes rendus d'activités pendant l'accomplissement de cette activité. Dans cette famille on peut citer la rédaction de comptes rendus d'opérations de maintenance par un agent en activité ou bien la rédaction de comptes rendus d'opérations médicales simultanément avec l'intervention elle-même. Ces deux types d'activités sont envisagées actuellement dans notre équipe et ont donné naissance à des systèmes de rédaction réalisés à l'aide du système de reconnaissance Vinics implanté sur une station de travail UNIX.

Dans l'application destinée à aider un serveur de restaurant, la syntaxe du langage est très simple puisque il s'agit d'une commande par mots-clés : les items de la carte.

Nous avons réalisé ce système sur un PC équipé d'une carte d'acquisition bas de gamme du type *Sound Blaster 16*. Le matériel et le système sont donc du type grand-public contrairement au matériel utilisé pour la rédaction de comptes rendus.

J'ai personnellement écrit les tâches MS-DOS de bas niveaux effectuant l'acquisition, la paramétrisation en cepstres et la segmentation parole/non parole. Nous traiterons de ce point particulier dans le paragraphe suivant puisque il s'agit toujours d'un sujet de recherche surtout en parole bruitée plus ou moins spontanée [Mauuary 93, Junqua 94].

3.1 Segmentation parole/non parole

Une des principales causes d'erreurs en reconnaissance de mots isolés provient d'une segmentation parole/non parole imprécise ou erronée. Dans une application réelle [Junqua 94], utilisant un système de reconnaissance de mots isolés, on a noté que la moitié des erreurs de reconnaissance peut être attribuée à des erreurs de localisation des extrémités de mots.

Il est donc primordial de séparer les segments de parole du bruit ambiant d'une façon fiable. En reconnaissance de la parole continue, si on accepte que le traitement ne débute que lorsque l'acquisition est terminée, on peut modéliser le bruit de fond comme un mot particulier et utiliser une grammaire de suites de mots pour reconnaître et localiser les frontières temporelles de la suite des segments. Le problème de la localisation des extrémités temporelles d'une phrase ou d'un mot (en anglais *end point detection*) est différent et complémentaire du problème de l'activation par la parole de l'algorithme d'acquisition (en anglais *voice activation algorithm*). Ce dernier effectue une segmentation grossière tout en déclenchant puis arrêtant l'acquisition tandis que le premier, grâce à une analyse de toute la portion de signal acquis, effectue un ajustement plus précis des frontières des mots présents dans l'élocution car l'analyse de toute la phrase donne plus de possibilités de normalisation et d'adaptation.

Différents algorithmes ont été proposés ces 20 dernières années. Parce que les machines de l'époque étaient lentes comparées à celles d'aujourd'hui, tous les algorithmes calculaient efficacement l'énergie, parfois dans différentes bandes de fréquence, le taux de passage par zéro et la fréquence fondamentale sur des fenêtres successives du signal puis mesuraient les durées des segments homogènes. Un ensemble de règles heuristiques permet de localiser les frontières des syllabes initiale et finale. Une bonne description des différentes méthodes de détection de frontières de mots a été faite récemment par J-C. Junqua dans [Junqua 94].

Dans notre application, nous voulons que l'acquisition du signal soit gouvernée directement par l'élocution et n'utilise pas d'artifice comme un bouton à presser afin de laisser les mains libres à l'utilisateur. L'acquisition et la segmentation sont donc simultanées. Nous avons opté pour une activation guidée par la voix (*voice activation algorithm*) suivie d'une recherche plus précise de frontières des mots (*end point detection*).

Activation par la voix

La vitesse de traitement d'un PC Dx2 66 MHz permet le calcul des vecteurs constitués de 8 coefficients cepstraux (c.-à-d. les trames) en temps réel et en simultanéité avec l'acquisition par la carte jusqu'à une fréquence de 12 kHz. L'acquisition est gouvernée par un automate qui commence par classer les cepstres en deux catégories : les trames de parole et les trames de silence. Le classement se fait en calculant la distance du cepstre courant à un cepstre de silence estimé pendant les deux secondes qui suivent le lancement du programme. La distance utilisée est la distance de Mahalonobis estimée sur un échantillon du bruit ambiant. On tient ainsi compte de la variance de chaque coefficient cepstral et des caractéristiques du bruit. Les meilleurs résultats ont été obtenus en négligeant le coefficient cepstral C_0 qui représente le logarithme de l'énergie du signal et en lissant les distances entre la trame du silence et les trames successives :

$$d_n = (1-\alpha) \times d(\overline{noise}, cepstre_{n-1}) + \alpha \times d(\overline{noise}, cepstre_n)$$

avec :

- $cepstre_n$ le vecteur de coefficients cepstraux au temps n ;
- $\overline{noise}$ le vecteur moyen de coefficients cepstraux du bruit ambiant ;
- d_n la distance lissée au temps n ;
- $d(.,.)$ la distance de Mahalanobis entre deux vecteurs ;
- α un coefficient de lissage entre 0 et 1 ajusté empiriquement.

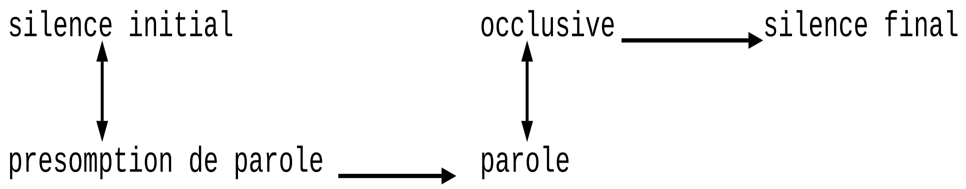


FIG. 5.3 – *Topologie de l'automate d'acquisition*

Des règles heuristiques de durée sont utilisées pour éviter les activations sur des bruits trop courts et arrêter l'acquisition quand le silence revient à nouveau. Elles sont implantées dans un automate contrôlant l'acquisition et décrit par la figure 5.3. Il possède les états suivants :

Silence initial : les trames acquises ont été classées tour à tour comme du silence. Elles ne sont pas sauvegardées.

Présomption de parole : la trame courante est de la parole, elle est sauvegardée à la suite des autres mais l'ensemble n'est pas suffisamment long et peut être encore un bruit court. Quand l'ensemble a assez duré, on passe dans l'état **parole**. Par contre, une seule trame de silence nous fait passer de cet état à l'état **silence initial**. Les consonnes occlusives voient ainsi disparaître le segment précédant l'explosion. Mais l'acquisition et le stockage du signal temporel se font. tampons après tampons. d'une façon synchrone. Le segment occlusif situé en milieu de tampon sera donc sauvé.

Parole : quand il y a suffisamment de trames dans l'état précédent, on passe dans l'état **parole** et on continue à acquérir le signal en provenance de la carte.

Occlusive : une trame n'est plus de la parole, on la sauvegarde toujours car on peut alors traiter un segment d'occlusion. Si le segment est assez long, on déduira une fin d'élocution sinon on retourne à l'état **parole**.

Silence final : l'acquisition est stoppée.

Cet automate utilise plusieurs seuils fixés après une série d'essais : le seuil de distance entre la trame courante et la trame de bruit estimé ; la durée minimum d'un mot et la durée maximum d'une occlusive. Nous utilisons un microphone différentiel à réduction de bruit qui élimine en partie les bruits ambiants. Dans des conditions de démonstration normales, en l'absence d'effet Lombard mais en présence d'un bruit de réception [19] raisonnable, nous n'avons jamais senti la nécessité de les adapter aux conditions de bruit ambiant.

Détection des frontières

Un tel automate de décision fournit une segmentation grossière surtout quand les premiers phonèmes ne sont pas des voyelles énergétiques. Afin de ne pas hypothéquer les chances de l'algorithme de reconnaissance qui entre en fonction dès que l'acquisition est terminée, nous sauvegardons sur disque toutes les segments de signal qui proviennent du DMA de la carte Sound Blaster dès que l'automate de décision quitte l'état **silence initial**. Dés que l'automate y retourne toutes ces zones sont détruites. On obtient donc en fin d'acquisition un fichier nécessairement plus grand que l'élocution acquise, ceci parce que la probabilité est faible d'avoir une élocution qui débute et s'achève sur des cycles du DMA. La détection précise des frontières des mots se fait pendant la reconnaissance en utilisant un modèle de bruit qui intervient dans la grammaire du langage à reconnaître.

3.2 Description de l'application

Une carte de restaurant est structurée logiquement en plusieurs sous-menus : les entrées, les plats, les desserts et les boissons. Un serveur qui annonce «Ça roule» fait savoir à tout le monde (donc au système) que la commande est bien prise. A l'aide de ces quelques constatations, nous avons défini plusieurs vocabulaires associés à autant de langages artificiels différents que de sous-menus. Il y a un langage de commande pour chaque partie de la carte. Le serveur peut passer d'un sous-menu à un autre en utilisant une commande par mots-clés : entrée, plat, boisson, ... La reconnaissance du mot « Ça roule » termine la commande et imprime l'addition. Chaque item de la carte peut être précédé d'un nombre et pour certains plats quelques qualificatifs peuvent être rajoutés ; par exemple, le mode de cuisson peut être précisé pour les viandes.

19. Cocktail party.

De cette façon, le facteur de branchement reste faible et la reconnaissance s'en trouve accélérée. Chaque mot est représenté par un HMM2 possédant un nombre variable d'états suivant la durée moyenne du mot et une densité gaussienne par état. Le changement de vocabulaire se fait par changement de la grammaire modélisant la partie de la carte dans lequel est prise la commande. Chaque grammaire est représentée par un automate décrivant un langage régulier très simple. L'ensemble des vocabulaires est fait de 23 mots et chaque automate a un facteur de branchement de l'ordre de 5.

3.3 Conclusions

Il faudrait évaluer ce système dans des conditions réelles avec des vrais utilisateurs. Nous reportons dans ce système régulièrement les améliorations validées en traitement du signal : soustraction spectrale pour supprimer le bruit additif, soustraction de la moyenne des cepstres pour éliminer le bruit convolutif. Actuellement, le système est porté sur Windows95 afin de profiter pleinement de la totalité de la mémoire des nouveaux Pentium. Nous avons ainsi un système disponible pour gérer des applications plus complexes.

4 Reconnaissance de cris de détresse

Dans le cadre de son projet "Sécurité", la RATP souhaite développer un système capable d'acquérir et d'interpréter des émissions sonores afin de détecter des situations d'agression.

Un tel système se situe à la limite supérieure des capacités actuelles des systèmes de traitement automatique de sons (parole et bruits) et fait appel aux techniques avancées de traitement du signal et de reconnaissance des formes. La solution proposée par notre équipe s'inscrit dans une telle approche. Elle repose sur :

le traitement du signal bruité par des méthodes cepstrales, de soustraction spectrale et de classification ;

l'analyse phonétique du corpus de parole criée ;

la modélisation stochastique des bruits et cris par des modèles de Markov, des modèles de trajectoires et des réseaux de neurones ;

la localisation d'événements dans un signal par des méthodes de *word spotting*.

4.1 Présentation technique de la proposition

Notre proposition fait appel aux techniques complémentaires de modélisation stochastique et d'analyse phonétique. Elle s'articule autour de quatre points :

- l'enregistrement *in situ* du corpus de données en collaboration avec la RATP, son analyse acoustique et son étiquetage ;
- le traitement du signal bruité par des méthodes d'analyse cepstrale, de soustraction spectrale et de classification mixte descendante/ascendante et consolidation des classes ;
- la modélisation stochastique des bruits et des mots à identifier à l'aide de modèles markoviens gauche-droite, parallèles (PMC ou Parallel Model Combination) ; le but de cette étude n'est pas de réaliser un système opérationnel de détection de cris de détresse mais

de réaliser une étude comparative de différentes méthodes de traitement et de reconnaissance de la parole bruitée ;

- la localisation d'entités apprises : bruits, mots à l'aide de méthodes de types *word spotting*. Cette localisation constitue le cœur du système proposé car celui-ci doit être en écoute permanente et détecter les événements significatifs dans les signaux fournis par les capteurs.

Ce contrat industriel a donné naissance à plusieurs produits :

- une maquette de système fonctionnant en ligne sur un PC ;
- un rapport de fin de contrat détaillé ;
- des spécifications du système industriel final.

4.2 Description du système

Le système se compose d'un microphone relié à un ordinateur chargé d'identifier des cris de détresse dans une zone de station de métro. La détection d'un cri alerte le personnel situé dans un poste central de surveillance et sélectionne la caméra vidéo couplée avec le microphone. Un nombre plus réduit de surveillants peut contrôler plus efficacement un nombre plus important de zones.

Dans la station “Le Havre-Caumartin”, cinq zones ont été choisies correspondant à différentes ambiances sonores : bord des voies, couloir, escalier roulant, voies d'accès, hall.

Pour reconnaître un cri de détresse dans l'univers sonore d'une station de métro, un simple filtrage fréquentiel suivi d'une détection par seuillage d'énergie ne peut pas convenir. Une analyse fréquentielle des sources de bruits montre que grand nombre d'entre elles ont un spectre qui se recouvre avec le spectre de la parole. Il paraît nécessaire d'effectuer une reconnaissance des entités sonores pour décider si un ou plusieurs cris ont été poussés. Tout comme pour la reconnaissance de bruits sous-marins, nous avons opté pour une approche de type “reconnaissance de la parole” plutôt que “localisation de la parole” et nous avons choisi de modéliser par des HMM2 deux types de sons : les voyelles criées ainsi que certaines expressions d'une-part et les bruits caractérisant l'ambiance sonore du lieu d'autre-part.

Périodiquement, le système échantillonne l'ambiance sonore et fait une reconnaissance sur le signal sonore prélevé. Une alarme est présentée au poste de surveillance si un modèle de cri est présent parmi l'ensemble des modèles reconnus. Il est donc nécessaire d'avoir pour chaque zone, une liste des modèles de bruits représentatifs de cette zone.

4.3 Description des corpus

Cette section présente l'ensemble des entités sonores utilisées pour l'apprentissage des modèles de Markov et pour les tests effectués sur ceux-ci.

Le corpus de bruits

Ces enregistrements ont été réalisés par la RATP. La base de bruits est constituée de 180 minutes d'enregistrement du bruit de fond du métro pour les cinq zones d'implantation du système. La répartition est la suivante :

Zone 1 : 38 minutes ;

Zone 2 : 37 minutes ;

Zone 3 : 43 minutes ;

Zone 4 : 41 minutes ;

Zone 5 : 35 minutes.

Pour chacune des cinq zones, trois enregistrements ont été réalisés à des heures différentes de la journée : matin, midi et soir. La durée d'un enregistrement est de 10 minutes.

Cette base a été ensuite divisée en deux parties de taille égale : une servant à l'apprentissage des modèles de bruit, l'autre servant au test du système.

Le corpus d'apprentissage a été écouté puis segmenté manuellement en bruits caractéristiques. Nous nous sommes trouvés devant le même problème que pour l'identification de bruits sous-marins, c'est à dire sans connaissance *a priori* des classes à construire. Lors de cet étiquetage, on constate qu'il n'y a pas de différences importantes entre les plages horaires d'une même zone quant à la variété des bruits possibles. Par contre, il y a des différences de bruits suivant les zones. En conséquence, chaque zone aura ses propres modèles, mais ceux-ci seront toujours les mêmes quelle que soit l'heure.

Les différents bruits étiquetés sont pour les zones 1, 2, 3 et 5 ;

- le bruit de fond ;
- les passages de rames de métro ;
- les bruit de pas ;
- les chocs métalliques.

Le passage de rames de métro n'est pas audible dans la zone 4, par contre, on constate la présence d'un escalier mécanique bruyant.

Ce corpus présente un défaut ; certains bruits caractéristiques sont présents en trop petit nombre ce qui empêche leur modélisation par des HMM. D'autre part, il n'y a pas que du bruit d'origine mécanique sur ce corpus. On distingue en effet les signaux d'origines diverses : des aboiements, des éternuements, des sifflements, des personnes parlant fort en passant à coté du micro, ...

Le corpus de cris

Dans la cassette audio fournie par la RATP, les cris donnés étaient des séquences relativement longues. Ces séquences ont été segmentées individuellement en cris. On obtient ainsi 20 cris qui durent en moyenne 2 secondes. Ces cris proviennent soit de la victime, soit des agresseurs.

Ce corpus est trop petit pour servir à l'apprentissage de HMM. Nous avons donc décidé d'enregistrer un corpus de cris afin d'estimer plusieurs modèles. La base servant à la construction des modèles a été construite au laboratoire. Des membres volontaires du laboratoire (12 hommes et 8 femmes) ont crié comme s'ils étaient agressés. Les enregistrements ont eu lieu dans le parking sous-terrain du laboratoire. Les locuteurs étaient à trois mètres du micro. L'interprétation était libre. Cependant tous les locuteurs ont crié « Au-secours » et « A l'aide » au moins une fois. On a ainsi obtenu 1320 cris représentant au total 15 minutes de cris. Contrairement aux cris fournis par la RATP qui sont utilisés pour la détection, on n'a ici que des cris de personnes agressées.

Pour chaque zone, on mixe les cris sur le bruit d'apprentissage de la zone. Les cris sont mixés au bruit au rapport 0 dB.[20]

En fin de compte, on dispose, pour l'apprentissage, d'une base de bruit constituée de cinq minutes étiquetées par zone et d'une base de bruit avec cris où, pour chaque zone, les 1320 cris sélectionnés ont été mixés dans les 5 minutes de bruit.

4.4 Choix des modèles de bruits

Parmi les modèles de bruits, nous avons retenu pour l'ensemble des cinq zones :

1. les bruits de pas ;
2. le bruit permanent du système de ventilation ;
3. le bruit de passage des rames de métro pour les zones 1, 2, 3 et 5 ;
4. le grincement permanent de l'escalier roulant de la zone 4 ;
5. un bruit de tintement métallique.

4.5 Choix des modèles de cris

Parmi les modèles de cris, nous avons retenu :

1. les voyelles criées /o/, /ou/ et /a/ ;
2. certaines expressions de détresse présentes en suffisamment grand nombre dans le corpus d'apprentissage comme « à l'aide, au secours, aux voleurs, arrêtez le » ;
3. un modèle de cri, tous cris confondus.

4.6 Traitement du signal

Sur la maquette fonctionnant en temps réel, le signal est filtré à 4000 Hz puis numérisé à 8 kHz sur 16 bits. Toutes les 16 ms, une FFT est calculée sur une fenêtre de 256 points et détermine un spectre de puissance. Pour éliminer le bruit additif stationnaire, nous avons appliqué une technique de soustraction spectrale. Le calcul des coefficients cepstraux se fait sans utiliser d'échelle Mel afin de préserver la remarquable structure harmonique d'une voyelle criée dans laquelle apparaissent régulièrement espacées, et beaucoup plus nettement que dans la voix parlée, les harmoniques de la fréquence fondamentale (cf. figures 5.4(b) et 5.4(a)).

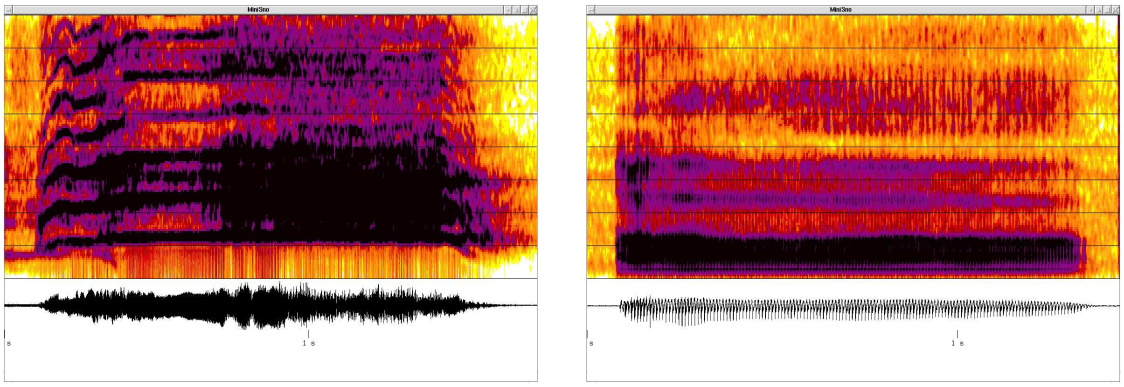
Soustraction spectrale

La soustraction spectrale est une méthode destinée à supprimer un bruit de fond additif stationnaire [Lim 78, Boll 79]. On considère en effet que ce que l'on perçoit dans le métro est de la forme :

$$Signal\_Final(t) = Bruit\_De\_Fond\_Stationnaire(t) + Bruit\_Special(t) + Cris(t)$$

20. **Le rapport signal/bruit** : Soit un cri $C(t)$ d'une durée d mixé sur un signal de bruit $B(t)$. On définit le rapport signal sur bruit par :

$$R = 10 \times \log\left(\frac{\int_0^d C^2(t)\,dt}{\int_0^d B^2(t)\,dt}\right)$$



(a) Un "AH" crié (b) Un "AH" parlé

FIG. 5.4 – *Spectrogrammes comparés d'une voyelle criée et parlée*

Le bruit de fond existe en permanence et varie assez peu, on va donc en estimer une moyenne et le soustraire au signal. Quant aux bruits transitoires connus, nous avons décidé de les modéliser à l'aide de HMM2.

L'algorithme de soustraction spectrale

Dans cet algorithme, le signal est la somme de deux processus aléatoires non corrélés : le bruit de fond et le son à distinguer.

$$signal[t] = b[t] + s[t]$$

On a donc :

$$\Phi_{signal}(\omega) = \Phi_b(\omega) + \Phi_s(\omega)$$

où Φ est la densité spectrale de puissance.

Cette densité spectrale de puissance peut être approximée à court terme par la transformée de Fourier en écrivant :

$$\Phi_{signal}(\omega) = \frac{|x(\omega)|^2}{n^2}$$

avec $|x(\omega)|$ amplitude spectrale du signal et n taille de la fenêtre.

On peut donc maintenant approximer le signal débruité en écrivant :

$$\Phi_s(\omega) = \Phi_{signal}(\omega) \;-\; \Phi_b(\omega)$$

Cependant cette formule n'est pas directement applicable car elle peut amener à une estimation négative de la densité spectrale de puissance.

On utilise donc dans la pratique :

$$|x(\omega)| = \sqrt[\alpha]{max(|Signal(\omega)|^2 - \beta \times |Bruit(\omega)|^2, SEUIL)}$$

où β est utilisé pour sous-estimer ou surestimer le bruit.

Avec $\alpha = 2$, on parle de soustraction spectrale de puissance. Avec $\alpha = 1$, on parle de soustraction spectrale en amplitude.

Plusieurs méthodes sont utilisées et actuellement évaluées pour estimer le spectre du bruit stationnaire. On peut l'estimer d'une façon continue sur une fenêtre glissante afin de capturer une lente variation temporelle mais cette fenêtre peut contenir des cris. On peut aussi estimer le spectre du bruit sur les portions de signal reconnues comme étant du bruit et itérativement adapter le spectre de bruit ainsi que les modèles. On peut, plus simplement, estimer ce bruit une fois pour toute à différents moments de la journée. Une campagne d'évaluation sur site permettra de mieux cerner l'intérêt de chaque méthode.

4.7 Comparaison avec une approche avec PMC

Le but de cette étude n'est pas de réaliser un système opérationnel de détection de cris de détresse mais de réaliser une étude comparative de différentes méthodes de traitement et de reconnaissance de la parole bruitée.

A côté de la méthode précédente qui consiste à effectuer un apprentissage de modèles de cris dans le bruit, on peut essayer d'effectuer un apprentissage de modèles de cris dans une ambiance calme et reconstruire des modèles bruités pendant la reconnaissance à partir du bruit courant. Une méthode qui a donné de bons résultats est la méthode de combinaison de modèles (Parallel Models Combination : PMC) [Gales 94].

Plusieurs hypothèses sont nécessaires pour déterminer l'expression des moyennes et matrices de covariances du HMM de parole bruitée :

- les signaux de parole et de bruit sont additifs dans le domaine des densités spectrales de puissance, ce qui constitue une hypothèse couvrant un grand nombre de situations pratiques ;
- la présence du bruit ne modifie pas la localisation des trames de parole ; une observation de parole propre affectée à l'état i d'un HMM lors de l'apprentissage, reste affectée à l'état i du HMM après perturbation par le bruit ;
- les signaux de parole et de bruit sont indépendants ;
- la somme de deux vecteurs aléatoires de distribution log-normale est un vecteur aléatoire de distribution log-normale. Cette condition revient à considérer que lorsque parole propre et bruit sont modélisés par des lois normales dans le domaine cepstral, les observations de parole bruitée sont également distribuées selon une loi normale.

Par contre, aucune hypothèse n'est faite sur la stationnarité du bruit. Un bruit non stationnaire peut être modélisé par un HMM ergodique à plusieurs états, et la combinaison des HMM donne naissance à un HMM de parole bruitée.

Cette approche présente l'intérêt d'être non supervisée dès lors que l'on est capable de détecter les zones d'absence de parole dans le signal, afin de réestimer les modèles de bruit. Il est ainsi possible d'obtenir des modèles capables de s'adapter automatiquement aux conditions de bruit. Elle présente l'inconvénient majeur de ne pas accepter une paramétrisation du signal par des coefficients dynamiques (Δ). Nous avons vu que ce mode de représentation permet la modélisation de la corrélation entre trames et apporte une nette amélioration des performances. Ainsi, nous avons préféré utiliser une méthode à base de HMM2 entraînés dans le bruit plutôt qu'une approche à base de construction dynamique de modèles telle que la PMC le permet.

4.8 État d'avancement des travaux

Le projet entre dans sa phase terminale. La maquette sur PC accepte en temps réel de mixer le signal d'un microphone avec le signal audio provenant d'une cassette de bruits de fond fournie par la RATP et valide notre approche – moins de 5 % de fausses alertes et de faux rejets – pendant des démonstrations faites dans les sous-sols de l'université. L'écriture de la version finale du rapport n'est qu'une affaire de temps.

5 Conclusions

Les progrès réalisés en matière de décodage acoustico-phonétique par des méthodes stochastiques permettent la réalisation de systèmes dans lesquels une interaction vivante peut se produire. Nous avons présenté trois systèmes de reconnaissance de la parole basés sur des méthodes stochastiques : modèles de Markov et modèles de trajectoires. L'usager est directement concerné par l'utilisation des deux premiers systèmes dans lesquels il a un rôle actif. Nous avons opté pour une approche ascendante qui consiste à créer un système afin de voir comment l'utilisateur se l'appropriera. Cette démarche n'a de sens que si d'autres équipes plus concernées par des problèmes de dialogue ou d'interaction homme-machine s'intéressent aux problèmes que nous traitons. Le dernier système d'audio-surveillance est un exemple où les techniques de reconnaissance de la parole peuvent être utilisées sans donner naissance à un système de dialogue. Notre investissement dans ce projet est motivé par notre désir de garder une finalité pratique à la recherche dans ce domaine.

Chapitre 6

Bibliographie personnelle

Publications dans des revues spécialisées avec comité de lecture

[81-R-048] J. F. MARI. – « Reconnaissance du discours continu par îlots de confiance ». – *R.A.I.R.O. 15*, 2 (1981), pp. 167 – 196.
Cet article dans une revue décrit un analyseur syntaxique de langages artificiels adapté à la reconnaissance des formes qui constituait une partie de la thèse de troisième cycle. Dans le cas de la reconnaissance de la parole continue telle qu'elle était abordée dans MYRTILLE 1, l'analyse syntaxique peut se faire par îlots de confiance plutôt que de la gauche vers la droite. Cette technique qui subsume la précédente, permet une souplesse plus grande particulièrement dans le cas de la parole spontanée.

[89-R-272] A AMIN, J.-F. MARI. – « Machine Recognition and Correction of Printed Arabic Text ». – *IEEE Transactions on Systems, Man, and Cybernetics 19*, 5 (septembre-octobre 1989), pp. 1300–1306.
Le problème de la reconnaissance de l'écriture cursive arabe y est abordé à l'aide des mêmes méthodes que la reconnaissance de la parole : des modèles de Markov cachés.

[90-R-134] A. BOYER, J. DI MARTINO, P. DIVOUX, J.-P. HATON, J.-F. MARI, K. SMAILI. – « Statistical Methods in Multi-Speaker Automatic Speech Recognition ». – *Applied Stochastic Models and Data Analysis 6*, 3 (1990), pp. 143–155.
Cet article décrit et compare plusieurs méthodes statistiques de reconnaissance des formes : utilisation de la programmation dynamique, classification de mots par des méthodes globales, utilisation d'automates et de modèles de Markov cachés. Toutes les comparaisons sont faites sur le même corpus et montrent la supériorité, en termes de taux de reconnaissance, des modèles de Markov cachés.

Ouvrages de synthèse

[94-R-152] Y. GONG, J.-P. HATON, J.-F. MARI. – « Issues in Acoustic Modeling of Speech for Automatic Speech Recognition ». – *Dans : Progress and Prospects of Speech Research and Technology*, H. Niemann, R. de Mori, et G. Hanrieder (réd.). Infix, 1994, pp. 34–44.
Dans ce chapitre de livre, nous présentons les différents atouts de deux méthodes stochastiques développées dans notre équipe. La modélisation stochastique est une

méthode souple pour tenir compte de la grande variabilité de la parole. Contrairement à la programmation dynamique qui utilise des méthodes heuristiques pour construire des formes de référence robustes, les modèles stochastiques permettent un apprentissage rigoureux reposant sur la théorie des probabilités. Ce chapitre décrit des techniques stochastiques adaptées aux phénomènes transitoires propres à la parole. Il présente deux apports de l'équipe RF-IA au problème : les modèles de Markov du second-ordre et le modèle stochastique de trajectoire.

Colloques internationaux avec actes et comité de lecture

[Haton 77] J.-P. Haton, J.-M. Pierrel, J.-F. Mari. – « Research towards understanding Models for artificial and natural Languages». – pp. 807 – 810, – *Dans : Actes IEEE ICASSP*. – Hartford (USA), mars 1977.
Cet article décrit l'architecture du système Myrtille 1, système de reconnaissance et compréhension de langages artificiels développé, à l'époque dans notre équipe. Deux analyseurs syntaxiques chargés d'élaborer des hypothèses au niveau lexical sont comparés : le premier fonctionne classiquement de la gauche vers la droite tandis que le second fonctionne par îlots de confiance grâce à un algorithme de localisation de mots dans un treillis de phonèmes. Cette dernière méthode d'analyse syntaxique constitue le sujet de ma thèse de troisième cycle.

[84-R-001] J.-F. Mari, J.-P. Haton. – « Some experiments in automatic recognition of a thousand word vocabulary». – *Dans : Actes IEEE ICASSP*. – San Diego (USA), mars 1984.
Dans cet article, nous présentons nos travaux dans le domaine de la reconnaissance de grands vocabulaires. En 1984, un vocabulaire de 1000 mots était un grand vocabulaire. Le lexique était indexé par des patrons phonétiques de chaque mot : la liste des grandes classes phonétiques retrouvées dans ce mot. On appelait *cohorte* l'ensemble des mots possédant le même patron. Les traits phonétiques étaient retrouvés à l'aide d'une système à base de règles dont la mise au point s'avéra délicate. J'ai abandonné ce travail en allant chez BBN apprendre les techniques de modélisation stochastique. La nécessité d'avoir un corpus pour évaluer un système s'est montrée clairement à cette époque là. À la même époque, des travaux analogues étaient réalisés au MIT par l'équipe de V. Zue. Ils ont été, eux-aussi, abandonnés en 1985. La recherche d'une organisation optimale de lexique se fait actuellement par des méthodes stochastiques à l'aide d'arbres de décision.

[85-R-049] J.-F. Mari, S. Roucos. – «Speaker independent connected digit recognition using Hidden Markov Models». – *Dans : Proceedings Congress "Speech Technology"*. – New York (USA), avril 1985.
La trace écrite de mon passage chez BBN ; ma première publication sur ce sujet. Cet article décrit un système de reconnaissance de suites de chiffres sur une partie de la base de *TI-NIST*. Chaque chiffre est modélisé par un HMM1 discret à 15 états.

[87-R-020] J.-P. Haton, N. Carbonell, D. Fohr, J.-F. Mari, A. Kriouile. – « Interaction between stochastic modeling and knowledge-based techniques in acoustic-phonetic decoding of speech». – *Dans : Proceedings IEEE-ICASSP 87*. – Dallas (Texas), avril 1987.

Cet article est la première marque de notre réflexion sur les systèmes hybrides à base de HMM et de systèmes experts. La coopération entre un système stochastique et un système expert de décodage se fait grâce à un tableau noir. L'absence de techniques d'apprentissage automatique de stratégies d'interaction rend la mise au point délicate, voire même impossible.

[88-R-116] A. Boyer, J. di Martino, P. Divoux, J.-P. Haton, J.-F. Mari, K. Smaili. – « Statistical Methods in Multi-speaker Automatic Speech Recognition». – *Dans : Proceedings 4th International Symposium on Applied Stochastic Models and Data Analysis.* – Nancy, décembre 1988.
Cet article a beaucoup intéressé la communauté de mathématiciens en analyse des données par ses cotés applicatifs. Il a donné ensuite naissance à un article dans une revue de statistique [90-R-134].

[89-R-173] C. Bourjot, A. Boyer, J.-F. Mari. – «Methodology about Assessment of Large Vocabulary Systems». – *Dans : Proceedings 7th FASE.* – Edinburgh (Scotland), août 1989.
Cet article contient nos réflexions sur l'évaluation des systèmes de reconnaissance de grands vocabulaires. Il représente une partie de notre contribution au projet *Speech Assessment Method* qui faisait lui-même partie du projet européen ESPRIT.

[90-R-017] A. Kriouile, J.-F. Mari, J.-P. Haton. – « Some Improvements in Speech Recognition Algorithms based on HMM». – *Dans : Proceedings IEEE ICASSP'90 (International Conference Acoustics Speech and Signal Processing).* – Albuquerque, New Mexico (USA), avril 1990.
Cet article marque la naissance des HMM du second ordre et de l'algorithme de Viterbi à optimalité locale. Cet algorithme, appelé Viterbi-bloc, permet une segmentation et un étiquetage au fur et à mesure de l'énonciation de la phrase, contrairement à l'algorithme de Viterbi qui doit attendre la fin de la phrase. Nous comparons différents critères utilisant les probabilités partielles des chemins de recalage temporel. Une variante très intéressante de cet algorithme a été proposée par la suite par des chercheurs de BELL-Northern pour étiqueter des corpus.

[91-R-197] A. Kriouile, J.-F. Mari, J.-P. Haton. – «Speech Recognition based on Second Order HMM». – *Dans : Proceedings Fifth International Symposium on Applied Stochastic Models and Data Analysis*, R. Gutiérrez, M. J. Valderrama (réd.), World Scientific, pp. 360–370. – Granada (Spain), avril 1991.

[94-R-083] J. di Martino, J.-F. Mari, B. Mathieu, K. Perot, K. Smaïli. – «Which Model for Future Speech Recognition Systems : Hidden Markov Models or Finite-State Automata». – *Dans : Proceedings International Conference on Acoustics, Speech and Signal Processing*, IEEE, pp. II-633 – II-635 – Adelaïde (South Australia), avril 1994.
Nous comparons deux méthodes de reconnaissance de la parole : l'une, désormais classique, à base de HMM et l'autre, plus originale, à l'aide d'automates qui permettent un apprentissage incrémental. Nous avons implémenté cette dernière approche dans le cas discret pour la reconnaissance de suites de mots. Les vecteurs cepstraux quantifiés représentant un mot sont assignés à des chemins d'un automate. Une nouvelle élocution d'un mot provoque soit l'ajout d'une branche dès qu'il n'est

plus possible d'aligner cette suite de vecteurs dans l'automate, soit le renforcement d'une branche existante. La reconnaissance se fait par programmation dynamique.

[94-R-204] J.-F. Mari, J.-P. Haton. – « Automatic Word Recognition Based on Second-Order Hidden Markov Models ». – *Dans : Proceedings International Conference on Spoken Language Processing*, p. S07.17. – Yokohama (Japan), septembre 1994.
Les HMM du second-ordre sont devenus continus. Cet article compare leurs performances avec leurs homologues du premier ordre sur le corpus de chiffres de *TI-NIST*.

[94-R-205] J.-F. Mari, D. Fohr, Y. Anglade, J.-C. Junqua. – « Hidden Markov Models and Selectively Trained Neural Networks for Connected Confusable Word Recognition ». – *Dans : Proceedings International Conference on Spoken Language Processing*, p. S26.11. – Yokohama (Japan), septembre 1994.
Cet article décrit un système hiérarchique de reconnaissance de lettres épelées au téléphone. Les HMM du second ordre effectuent tout d'abord une segmentation du signal ainsi qu'un étiquetage en grandes classes ; les réseaux de neurones effectuent ensuite une discrimination plus fine à l'intérieur des classes de lettres connues pour être proches. La recherche des indices acoustiques qui sont utilisés en entrée des ANN se fait à l'aide d'algorithmes issus des systèmes experts de décodage acoustico-phonétique développés dans notre équipe dans les années 85. Ce système met en œuvre, dans une stratégie hiérarchique, des techniques stochastiques et une représentation de connaissances.

[95-R-278] J.-C. Junqua, S. Valente, D. Fohr, J.-F. Mari. – «An N-Best Strategy, Dynamic Grammars and Selectively Trained Neural Networks for Real-Time Recognition of Continuously Spelled Names over the Telephone ». – *Dans : Proceedings IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 852–855. – Detroit (Michigan, USA), mai 1995.
Cet article, tout comme le suivant, est le résultat d'une collaboration serrée entre les laboratoires STL (USA) et le CRIN en matière de reconnaissance de noms épelés au téléphone à l'aide de HMM1 et HMM2. Dans cet article, nous explorons deux dimensions de la reconnaissance : l'évaluation de plusieurs vecteurs de paramètres d'entrée comme des coefficients cepstraux normalisés ou des coefficients PLP-RASTA ainsi que la spécification de stratégies de décodage lexical basées sur plusieurs passages sur la représentation du signal d'entrée en y appliquant progressivement des nouvelles contraintes. La stratégie donnant les meilleures performances sur une application de recherche de noms de personne dans un annuaire téléphonique à partir d'une épellation s'élabore à l'aide de trois passages dont le dernier est réalisé à l'aide d'une grammaire dynamique de noms de personnes phonétiquement très proches qu'un système de reconnaissance à base de HMM1 de lettres doit décoder. Une comparaison entre HMM1 et HMM2 est aussi donnée sur la même tâche de reconnaissance mais sur deux sites d'expérimentation différents afin de réduire le biais dû à la paternité de chacun des systèmes.

[95-R-279] J.-C. Junqua, D. Fohr, J.-F. Mari, T. H. Applebaum, B. H. Hanson. – « Time derivatives, Cepstral Normalization, and Parameter Filtering for Continuously Spelled Names over the Telephone ». – *Dans : Proceedings 4th European Conference on Speech Communication and Technology*, *1*, pp. 1385–1388. – Madrid (Spain), septembre 1995.

Quand on tient une idée, on ne la lâche pas ! Il restait encore quelques comparaisons à faire en faisant varier la longueur de la fenêtre de régression, en utilisant les dérivées secondes et normalisant ou filtrant les spectres ou cepstres. Dans cet article, réalisé avec l'équipe de STL, nous nous intéressons au filtrage des paramètres spectraux afin de diminuer la dissimilarité entre les données de reconnaissance et d'apprentissage. Nous avons étudié différentes combinaisons de paramètres spectraux ainsi que leurs dérivées. Les influences du traitement RASTA, de la normalisation cepstrale à plus ou moins long terme et de la quantité de données d'apprentissage sont aussi évaluées. En conclusion, nous retenons les bonnes valeurs de durée pour calculer une régression sur le signal d'entrée et remarquons que les cepstres avec des HMM2 donnent les meilleures performances même en parole bruitée.

[Pican 96] N. Pican, D. Fohr et J.-F. Mari. – HMMs and OWE Neural Network for Continuous Speech Recognition. *Proceedings International Conference on Spoken Language Processing*, Philadelphie, 1996.
Nous utilisons un nouveau type de réseau connexionniste pour identifier des phonèmes en contexte. Dans le cadre de la reconnaissance de l'anglais lu, cet article décrit un système hiérarchique de reconnaissance phonétique. Les HMM de phonèmes indépendants du contexte effectuent une segmentation et une reconnaissance. Tous les segments occlusifs sont re-classés à l'aide du réseau OWE dont les poids dépendent du contexte phonétique du segment considéré. Les résultats publiés sont intéressants car il devient difficile de faire mieux que les HMM.

[Mari 96] J.-F. Mari, D. Fohr et J.-C. Junqua. – A Second-Order HMM for High Performance Word and Phoneme-Based Continuous Speech Recognition. *Proceedings IEEE-ICASSP*, Atlanta, 1996.
Dans le cadre de la reconnaissance de l'anglais lu, les capacités des HMM du second ordre pour modéliser la durée des unités sont décrites en détail.

Colloques nationaux avec actes et comité de lecture

[78-R-048] J. P. Haton, J. F. Mari, J. M. Pierrel. – « Un système de compréhension du discours parlé ». – *Dans : Congrès AFCET-INRIA*. – 1978.
Cet article décrit les recherches menées dans notre équipe sur la compréhension de la parole. Nous mettons l'accent sur l'analyse syntaxique de langages artificiels et quasi-naturels. Deux analyseurs syntaxiques adaptés à ces deux catégories de langage y sont décrits.

[94-R-208] J.-F. Mari, D. Fohr, J.-P. Haton. – « Modèles stochastiques d'ordre 1 et 2 ». – *Dans : Actes XXèmes Journées d'Etude sur la Parole*, pp. 199–202. – Tregastel, juin 1994.

[Thiébaut 96] E. Thiébaut, J.-F. Mari, J.-P. Haton, Y. Gong et D. Fohr. – Utilisation d'un système de reconnaissance de la parole pour accéder à W3 . *XXIèmes Journées d'Etude sur la Parole*, Avignon, 1996, pp. 429 – 432.
Pour ajouter un micro à sa souris, il ne faut pas avoir un chat dans sa gorge ! Voilà la conclusion de l'exposé de cet article aux *XXIèmes Journées d'Etude sur la Parole*. Nous décrivons un système permettant d'accéder à W3 à l'aide de commandes orales. Ce système augmente les possibilités de MOSAIC en permettant de désigner oralement des documents en plus d'une désignation à l'aide d'une souris.

[Fohr 96] D. Fohr, J.-F. Mari et J.-P. Haton. – Utilisation de modèles de Markov pour l'étiquetage automatique et la reconnaissance de BREF80. *XXIèmes Journées d'Etude sur la Parole*, Avignon, 1996, pp. 339 – 342.
Notre première contribution au projet AUPELF de dictée vocale. Nous sommes prêts à diffuser l'étiquetage de la base pour la communauté de chercheurs.

Colloques avec actes sans comité de lecture

[95-R-300] J.-F. Mari, D. Fohr. – « Présentation de progiciels HMM ». – *Dans : Actes Ecole Thématique sur les Fondements et Perspectives en Traitement Automatique de la Parole*, H. Méloni (réd.), GDR-PRC Communication Homme-Machine - Pôle Parole. – Marseille, juillet 1995.
L'art ou la manière pour construire des systèmes de reconnaissance de la parole à partir de HMM. Les différentes étapes de construction de modèles de mots ou de phonèmes sont décrites. Nous étudions aussi l'influence des paramètres du système : nombre d'états dans la topologie des modèles, caractéristiques des paramètres acoustiques et partage d'états.

Colloques sans actes ou avec actes à diffusion restreinte

[80-P-046] Haton J. P, J. F. Mari, J.-M. Pierrel, S. Sabbagh. – « Représentation et mise en oeuvre de contraintes syntaxiques et sémantiques en reconnaissance du discours continu ». – *Dans : Séminaire GALF-AFCET*, Quinton Haton, Pierrel (réd.), pp. 149 – 167. – septembre 1980.

[95-R-285] D. Fohr, J.-F. Mari, J.-C. Junqua. – «Feature Extraction for Recognition over the Telephone». – *Dans : Project European COST 149*. – septembre 1995.

[94-R-209] J.-F. Mari, D. Fohr, J.-P. Haton. – «Quelques aspects des modèles de Markov en reconnaissance de la parole». – *Dans : Actes Séminaire sur la Reconnaissance Automatique de la Parole*. – Nancy, mars 1994.

Thèses

[79-T-048] J. F. MARI. – *Contribution à l'analyse syntaxique et à la recherche lexicale en reconnaissance du discours continu.* – Thèse de doctorat, Université de NANCY 1, 1979.

Notes internes

[85-R-048] J.-F. Mari. – «Rapport de stage INRIA effectué chez BBN (Bolt, Beranek, NewMan)», 1985.

[85-R-080] J.-F. Mari. – «Reconnaissance de mots enchaînés à l'aide de modèles markoviens discrets», 1985.

Manuels

[78-E-042] J. P. Finance, J. F. Mari. – *Méthodes de programmation déductives et structures de données, cours d'algorithmique*, 1978.

[CLA 96] J.F. Mari et A. Napoli éditeurs. – Actes du séminaire “Aspects de la classification”. Rapport de Recherche CRIN 96-R-072, Nancy, France, 1996.
Les techniques de classification numérique ont toujours été présentes en reconnaissance des formes. Les réseaux de neurones montrent chaque jour leurs (très?) bonnes propriétés de classification, et la classification se fait de plus en plus présente en représentation des connaissances. Ainsi, ce rapport présente, simplement dans un but introductif, les aspects mathématiques, statistiques, neuromimétiques et cognitifs de la classification.

Rapports de fin de contrats

[96-R] F. Botella, F. Alexandre, D. Fohr, Y. Gong, J.-F. Mari, O. Siohan – Classification de signaux transitoires sous-marins. Rapport de fin de contrat avec la Direction des Constructions Navales – convention Num : C94 50 572 00 405 75 97 – Nancy, France, 1996.
Où il est question de cris de dauphins, cachalots et autres crevettes qu'il fallait différencier des bruits d'hélices, chasses et autres bruits mécaniques.

[96-R] C. Antoine, F. Fauchier, D. Fohr, Y. Gong, J.-P. Haton, J.-F. Mari, O. Siohan – Détection de cris en milieu bruité. Rapport de fin de contrat avec la Régie Autonome des Transports Parisiens – Nancy, France, 1996.
Criez, on vous regarde ! Ce rapport décrit une solution au problème de l'audiosurveillance dans les couloirs du métro. L'analyse de la voix criée permet sa reconnaissance au milieu d'autres voix et bruits mécaniques.

Chapitre 7

Conclusions et perspectives

1 Conclusions

Dans le chapitre 1 nous avons défini notre thème de recherche : la perception de signaux complexes à l'aide de modèles stochastiques. Nous avons montré dans le chapitre 2 quelle a été notre contribution dans ce domaine : l'utilisation de modèles de Markov du deuxième ordre et l'utilisation d'algorithmes de reconnaissance à optimalité locale. L'influence d'hypothèses simplificatrices comme l'utilisation de matrices de corrélation diagonales ou de chaînes de Markov du premier ordre a été évaluée sur différents corpus de parole "propre" ou téléphonique. Il en ressort que les matrices non-diagonales permettent d'atteindre des vraisemblances 500 fois plus importantes sur les corpus d'apprentissage. Nous pensons que leur utilisation se généralisera dans les systèmes de reconnaissance futurs. Dans le cas de modèles comportant un grand nombre d'états, les modèles de Markov du deuxième ordre ont de meilleures performances que leurs homologues du premier ordre. Ce résultat a été montré sur des systèmes de reconnaissance de noms de personnes épelés au téléphone et sur des suites de chiffres. Il corrobore le résultat trouvé en reconnaissance de l'écriture cursive.

En choisissant de développer notre propre boîte à outils par opposition à la grande majorité des autres laboratoires qui utilisent des progiciels standards, nous avons acquis une maîtrise dans la spécification et l'implémentation de systèmes de reconnaissance. Le chapitre 3 résume cette expertise.

Au fur et à mesure que ce projet prenait corps et était validé sur des signaux différents, nous avons progressé dans la standardisation des traitements afin de réutiliser au mieux les différents programmes de paramétrisation du signal, d'apprentissage et de reconnaissance des modèles stochastiques. Ceci nous a permis d'approcher les problèmes généraux qui se posent plus généralement en classification et en catégorisation dans le choix des distances, des algorithmes et dans l'interprétation qui est faite des classes. Les HMM se révèlent être de bons outils pour extraire des régularités dans des gros corpus de données ayant une dimension temporelle. Deux approches complémentaires sont décrites dans le chapitre 4 : la première permet d'intégrer des connaissances en ajoutant des contraintes sur les HMM au niveau des états, la deuxième utilise les HMM comme outils de catégorisation. Sur ce sujet, nous avons rencontré des préoccupations communes avec des chercheurs de notre équipe qui travaillent sur la classification appliquée à la représentation des connaissances ; ce qui prouve le bien fondé de notre présence concomitante au sein de la même équipe.

En dernier lieu, dans le chapitre 5, nous avons exposé nos réalisations pratiques en reconnaissance de la parole. Différents contrats industriels ont été passés dans ce domaine. Dans chacun des cas, le sujet de l'étude demandée par le contractant correspond à un thème de recherche : reconnaissance dans le bruit ou interaction homme-machine pour l'accès à des centres de renseignement. Nous montrons ainsi notre capacité à nourrir nos travaux de recherche par une implication forte avec le milieu industriel national ou européen.

Enfin, nous avons tenu à tester nos algorithmes sur les corpus de référence en matière de reconnaissance de la parole.

2 Perspectives

Dans ce paragraphe, nous essaierons de répondre à la question : « Que sera devenu notre thème de recherche dans dix ans ? »

Notre projet scientifique vise à faire progresser de façon significative la reconnaissance automatique de la parole au sein de l'équipe RFIA en mettant en œuvre les idées développées qui ont été mes préoccupations pendant dix ans :

- Localisation de mots dans de la parole continue et spontanée ;
- Reconnaissance par îlots de confiance ;
- Définition et utilisation de critères d'optimalité locale ;
- Utilisation de méthodes stochastiques.

Ces méthodes seront utilisées conjointement avec d'autres développées dans notre équipe comme les réseaux neuro-mimétiques et les systèmes à base de connaissances. Compte tenu de l'état d'avancement des travaux dans le monde, cette progression ne peut se faire qu'en comparant nos résultats à ceux déjà obtenus par d'autres équipes à l'aide des corpus de référence français ou américains. Nous continueront donc à implémenter des systèmes de dictée vocale pour les tester au regard de la concurrence.

Les techniques de reconnaissance à optimalité locale ou par îlots de confiance semblent particulièrement adaptées au traitement de la parole spontanée. La conception de systèmes capables de reconnaître ce type de parole constitue un enjeu pour les années à venir et ils apparaîtront dans un de nos thèmes de recherche. Il en va de même pour l'utilisation de parole téléphonique dans les systèmes de reconnaissance. Nous avons toujours le désir de poursuivre notre collaboration avec les laboratoires STL sur ce sujet.

Dans 10 ans, notre contrat CNET sera fini depuis 7 ans. On saura mieux allier le geste à la parole. Les conditions dans lesquelles l'utilisation de la parole accroît le coté convivial d'une interface homme-machine seront plus claires et ce mode d'interaction sera devenu plus naturel.

Les techniques de reconnaissance de la parole ont des débouchés dans d'autres domaines que ceux de la machine à dicter ou de l'accès à des centres de renseignements. Nous avons montré qu'il était possible d'utiliser des méthodes de reconnaissance et de classification à base de modèles stochastiques pour identifier des sources sonores d'origine mécanique ou biologique en fonction de leur signature sonore. Nos travaux sur l'identification des cris de détresse dans les couloirs du métro ou sur la classification de signatures sonores biologiques ou mécaniques en sont des premiers exemples. La constitution de gros corpus de parole implique leur étiquetage en différentes unités : phonèmes ou mots. Dans ce domaine, notre expérience sur *BREF80* nous permet d'envisager l'étiquetage d'autres corpus comme celui du projet européen VODIS ; nous estimons nécessaire de participer à de telles recherches. Voilà pour le coté applicatif.

Étant par goût et par formation initiale attiré par des problèmes mathématiques d'ordre probabiliste et statistique, je suis intéressé par la résolution de différents problèmes d'estimation pour lesquels peu de solutions informatiques existent en reconnaissance de la parole continue. En d'autres termes, il reste encore beaucoup de théorèmes inutilisés et que l'on pourrait mettre en œuvre pour reconnaître de la parole, particulièrement en estimation à l'aide de critères où les modèles entrent en compétition les uns avec les autres ou bien s'adaptent à de nouvelles données. Bien sûr, on représentera les densités de probabilités à l'aide de réseaux de neurones ou de fonctions radiales. Du moins, on saura pourquoi on ne le fait pas.

A plus long terme, il reste à trouver comment utiliser conjointement des résultats symboliques issus d'une mise en œuvre d'une source de connaissances avec ceux issus de modèles stochastiques maximisant un critère probabiliste. Fixer *a priori* des paramètres que des HMM estiment *a posteriori* a été une approche intéressante. Nous pensons qu'il faut suivre avec attention les travaux faits en fouille de données et en représentation des connaissances car beaucoup de problèmes sont identiques.

Dans les mois à venir, et dans le cadre de la construction de la machine à dicter, nous généraliserons la recherche et l'estimation des phonèmes du français en fonction du contexte. Nous verrons ainsi si les résultats de cette catégorisation sont toujours interprétables par un phonéticien et quelle interaction il est possible d'établir afin de croiser des connaissances linguistiques et phonétiques avec les résultats d'une recherche de triphones et polyphones.

Nous sommes aussi intéressés par les nouvelles méthodes de reconnaissance. Plutôt que d'effectuer la reconnaissance dans la totalité du spectre de la parole, on peut envisager de faire une reconnaissance par bande de fréquences [Bourlard 96] et combiner ensuite les résultats symboliques de chaque bande. Cette nouvelle méthode pose les problèmes du parallélisme des traitements, de l'hybridation des méthodes et de la coexistence des méthodes numériques et symboliques.

L'acquisition d'un robot mobile par notre équipe afin d'expérimenter des algorithmes de planification, a été l'occasion de débuter dans un nouveau thème de recherche pour nous : l'utilisation et l'apprentissage automatique de modèles stochastiques dans la recherche de plans et dans l'identification de balises naturelles. Ce dernier problème consiste à classer un environnement afin que le robot mobile détecte son passage devant un point remarquable. Plusieurs thèses débutent sur ce sujet. Au même titre que la reconnaissance de la parole, nous comptons ajouter la robotique comme terrain d'expérimentations pour l'utilisation de modèles stochastiques.

Et pour conclure ce document sur une note généreuse à destination de tous ceux qui m'ont lu jusque là, il faudrait continuer à diffuser les idées, les algorithmes, les méthodes et les bases de données. Les futurs doctorants seront toujours les bienvenus dans notre thème. Nous répétons ici que nous croyons à la formation par la recherche. On pourrait commencer par organiser une école d'été sur les problèmes de classification et d'apprentissage et leurs implications en reconnaissance des formes.

Bibliographie

[Afify 96] M. Afify, Y. Gong et J.-P. Haton. Estimation of Mixtures of Stochastic Dynamic trajectories: Application to Continuous Speech Recognition. *Computer Speech and Language*, 10:23 –36, 1996.

[Amin 85] A. Amin. *IRAC: un système pour la reconnaissance et pour la compréhension de l'arabe écrit et imprimé.* Thèse de Doctorat, Université de NANCY 1, 1985.

[Amin 89] A Amin et J.-F. Mari. Machine Recognition and Correction of Printed Arabic Text. *IEEE Transactions on Systems, Man, and Cybernetics*, 19(5):1300–1306, septembre-octobre 1989.

[André-Obrecht 88] R. André-Obrecht. A new statistical approcah for the automatic segmentation of continuous speech signals. *IEEE Trans. on Acoustics Speech Signal Processing*, 36(1):29 – 40, 1988.

[Anigbogu 92] J. C. Anigbogu. *Reconnaissance de textes imprimés multifontes à l'aide de modéles stochastiques et métriques.* Thèse de Doctorat, Université de NANCY 1, 1992.

[Anigbogu 95] J.C. Anigbogu et A. Belaid. Hidden Markov Model in Text Recognition . *International Journal of Pattern Recognition and Artificial Intelligence*, 9:925 – 958, 1995.

[Atlan 92] H. Atlan. *L'organisation biologique et la théorie de l'information.* Hermann - Éditeurs des sciences et des arts, 1992.

[Aubert 95] X. Aubert et C. Dugast. Improved Acoustic-Phonetic Modeling in Philips's Dictation System by Handling Liaisons and Multiple Pronunciations. *Proceedings of European Conference on Speech Communication and Technology*, pages 767 – 770, 1995.

[Bahl 88] L. Bahl, L. Brown, P. De Souza et R. Mercer. Acoustic Markov Models Used in the Tangora Speech Recognition System. *Proceedings IEEE-ICASSP*, 1988.

[Baker 74] J. K. Baker. Stochastic Modeling for Automatic Speech Understanding. D.R. Reddy, éditeur, *Speech Recognition*, pages 521 – 542. Academic Press, New York, New-York, 1974.

[Baker 75] J.K. Baker. The Dragon system- An overview. *IEEE Trans. on ASSP*, 23(11):24 – 29, 1975.

[Baker 92] P. De Baker. Rapport de stage de DEA : The Use of Prosody in Speech Recognition Systems, 1992.

[Baum 72] L. E. Baum. An Inequality and Associated Maximization Technique in Statistical Estimation for Probabilistic Functions of Markov Processes. *Inequalities*, 3:1 – 8, 1972.

[Bengio 96] Y. Bengio. *Neural Networks for Speech and Sequence Recognition.* International Thomson Computer Press, 1996.

[Berndt 96] D. J. Berndt. Finding Patterns in Time Series . U. M. Fayyad, G. Piatetsky-Shapiro, P. Smyth et R. Uthurusamy, éditeurs, *Advances in Knowledge Discovery and Data Mining*, pages 229 – 248. AAAI Press / The MIT Press, 1996.

[Beulen 95] K. Beulen, L. Welling et H. Ney. Experiments with Linear Feature Extraction in Speech recognition. *Proceedings of European Conference on Speech Communication and Technology*, pages 1415 – 1418, Madrid, September 1995.

[Binda 94] H. Binda. Rapport de stage de DEA : Modèles mixtes symboliques et stochastiques pour la reconnaissance de la parole, 1994.

[Blaydon 70] C. C. Blaydon, R.L. Kashyap et K.S. Fu. Applications of the Stochastic Approximation Methods . J. M. Mendel et K. S. Fu, éditeurs, *Adaptive, Learning and Pattern Recognition Systems*, pages 357 – 391. Academic Press, 1970.

[Boll 79] S.F. Boll. Suppession of Acoustic Noise in Speech Using Spectral Substraction. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 27:113 – 120, 1979.

[Boulianne 94] G. Boulianne, P. Kenny, M. Lennig, D.O'Shaughhessy et P. Mermelstein. Books on tape as training data for continuous speech recognition. *Speech Communication*, 14:61 – 70, 1994.

[Bourjot 89] C. Bourjot, A. Boyer et J.-F. Mari. Methodology about Assessment of Large Vocabulary Systems. *Proceedings 7th FASE*, Edinburgh (Scotland), août 1989.

[Bourlard 94] H. Bourlard et N. Morgan. *Connectionist Speech Recognition : a Hybrid Approach.* Kluwer academic publishers, 1994.

[Bourlard 96] H. Bourlard et S. Dupont. A New ASR Approach Based on Independent Processing and Recombination of Partial Frequency Bands. *Proceedings of International Conference on Spoken Language Processing*, Philadelphia, 1996.

[Brown 87] P.F. Brown. *The Acoustic-Modeling Problem in Automatic Speech Recognition.* Thèse de Doctorat, C.S.D., Carnegie Mellon University, 1987.

[Buzo 80] A. Buzo, A. H. Gray, R. M. Gray et J. D. Markel. Speech Coding Based upon Vector Quantization. *IEEE Trans. on Acoustics Speech Signal Processing*, 28(5):562 – 574, 1980.

[Calliope 89] Calliope. *La parole et son traitement automatique*. Masson, 1989.

[Chollet 82] G. Chollet et C. Gagnoulet. On the Evaluation of Speech and Data Bases Using a Reference System. *Proceedings IEEE-ICASSP*, volume 3, pages 2026 – 2029, Paris, 1982.

[CLA 96] Actes du séminaire “Aspects de la classification”. J.F. Mari et A. Napoli éditeurs, Rapport de Recherche CRIN 96-R-072, Nancy, France, 1996.

[Cole 91] R. Cole, K. Roginski et M. Fanty. English Alphabet Recognition With Telephone Speech. *Proceedings of European Conference on Speech Communication and Technology*, pages 479 – 482, 1991.

[Cole 92] R. Cole, K. Roginski et M. Fanty. A Telephone Speech Database of Spelled and Spoken Name. *Proceedings of International Conference on Spoken Language Processing*, pages 891–895, Banff, 1992.

[Davis 80] S.B. Davis et P. Mermelstein. Comparaison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences . *IEEE Transactions on Acoutics, Speech and Signal Processing*, 28(4):357 – 366, 1980.

[Dempster 77] A.P. Dempster, N.M. Laird et D.B. Rubin. Maximum-Likelihood From Incomplete Data Via The EM Algorithm. *Journal of Royal Statistic Society, Ser. B (methodological)*, 39:1 – 38, 1977.

[Digalakis 94] V. Digalakis et H. Murveit. Genones : Optimizing the Degree of Mixture Tying in a Large Vocabulary Hidden Markov Model Based Speech Recognize. *Proceedings IEEE-ICASSP*, volume 1, pages 537–540, Adelaide, 1994.

[Dugast 95] C. Dugast, X. Aubert et R. Kneser. The Philips Large Vocabulary Recognition System for American English, French and German. *Proceedings of European Conference on Speech Communication and Technology*, pages 197 – 200, 1995.

[Emptoz 96] H. Emptoz et F. Lebourgeois. Reduction des ensembles d'apprentissage en reconnaissance de formes. 5eme journée du groupe de travail sur l'apprentissage – AFCET AFIA, May 1996.

[Fayyad 96] U. M. Fayyad, G. Piatetsky-Shapiro et P. Smyth. From Data Mining to Knowledge Discovery: An Overview . U. M. Fayyad, G. Piatetsky-Shaouri, P. Smyth et R. Uthurusamy, éditeurs, *Advances in Knowledge Discovery and Data Mining*, pages 1 – 34. AAAI Press / The MIT Press, 1996.

[Ferguson 80] J. D. Ferguson. Variable Duration Models for Speech. *Proc. of the Sym. on the Applications of Hidden Markov Models to Text and Speech*, pages 143 – 179, Princeton, NJ, 1980. IDA-CRD.

[Fohr 89] D. Fohr, N. Carbonell et J.P. Haton. Phonetic decoding of continuous speech with the aphodex expert system. *Proceedings of European Conference on Speech Communication and Technology*, Paris, September 1989.

[Fohr 95] D. Fohr, J.-F. Mari et J.-C. Junqua. Feature Extraction for Recognition over the Telephone. *Project European COST 249*, septembre 1995.

[Fohr 96] D. Fohr, J.-F. Mari et J.-P. Haton. Utilisation de modèles de Markov pour l'étiquetage automatique et la reconnaissance de BREF80. *XXIèmes Journ'ees d'Etude sur la Parole*, pages 339 – 342, Avignon, 1996.

[Forney 73] G.D. Forney. The Viterbi Algorithm. *IEEE Transactions*, 61:268–278, March 1973.

[Fu 70] K. S. Fu. Stochastic automata as models of learning systems. J. M. Mendel et K. S. Fu, éditeurs, *Adaptive, Learning and Pattern Recognition Systems*, pages 393 – 429. Academic Press, 1970.

[Furui 81] S. Furui. Cepstrum Analysis Techniques for Automatic Speaker Verification. *IEEE Transactions on Acoutics, Speech and Signal Processing*, April 1981.

[Gales 94] M.F.J. Gales et S.J>Young. Parallel Model Combination on a Noise Corrupted Ressource Management Task. *Proceedings of International Conference on Spoken Language Processing*, volume 1, pages 255 – 258, Yokohama, 1994.

[Garofolo 93] J.S. Garofolo, L.F. Lamel, W.M. Fisher, J.G. Fiscus, D.S. Pallet et N.L. Dahlgren. The DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus CDROM , 1993.

[Gauvain 90] J.L. Gauvain, L.F. Lamel et M. Eskénazi. Design Considerations and Text Selection for BREF, a Large French Read-speech Corpus. *Proceedings International Conference on Spoken Language Processing*, Kobe, Japan, 1990.

[Gauvain 91] J.L. Gauvain, L.F. Lamel et M. Eskénazi. Bref, a large vocabulary spoken corpus for french. *Proceedings of European Conference on Speech Communication and Technology*, pages 505 –508, Genova, Italy, 1991.

[Gong 94a] Y. Gong, J.-P. Haton et J.-F. Mari. Issues in Acoustic Modeling of Speech for Automatic Speech Recognition. H. Niemann, R. de Mori et G. Hanrieder, éditeurs, *Progress and Prospects of Speech Research and Technology*, pages 34–44. Infix, 1994.

[Gong 94b] Y. Gong et J.P. Haton. Stochastic Trajectory Modelling for Speech Recognition . *Proceedings IEEE-ICASSP*, volume 1, pages 57 – 60, Adelaide, 1994.

[Haeb-Umbach 93] R. Haeb-Umbach, D. Geller et H. Ney. Improvements in Connected Digit Recognition Using Linear Discriminant Analysis and Mixture Densities. *Proceedings IEEE-ICASSP*, pages 239 – 242, 1993.

[Haton 74] J.-P. Haton. *Contribution à l'analyse, la paramétrisation et la reconnaissance automatique de la parole.* Thèse de Doctorat, Université de NANCY 1, 1974.

[Haton 87] J.-P. Haton, N. Carbonell, D. Fohr, J.-F. Mari et A. Kriouile. Interaction Between Stochastic Modeling and Knowledge-Based Techniques in Acoustic-Phonetic Decoding of Speech. *Proceedings IEEE-ICASSP 87*, Dallas, avril 1987.

[Haton 95] J.-P. Haton. Les modèles neuronnaux et hybrides en reconnaissance automatique de la parole : états des recherches. H. Méloni, éditeur, *Actes Ecole Thématique sur les Fondements et Perspectives en Traitement Automatique de la Parole*, pages 139 – 153. Université d'Avignon et des pays de Vaucluse, Marseille, Juillet 1995.

[Huang 90] X. D. Huang, Y. Ariki et M.A. Jack. *Hidden Markov Models for Speech Recognition.* Edinburgh University Press, 1990.

[Hughey 96] R. Hughey et A. Krogh. Hidden Markov Models for Sequence Analysis: Extension and Analysis of the Basic Method. *Computer Applications in the Biosciences*, 12(2):95 – 107, April 1996.

[Huttenlocher 84] D.P. Huttenlocher et V.W. Zue. A Model for Lexiical Acces Based on Partial Phonetic Information. *Proceedings IEEE-ICASSP*, page 26.4, 1984.

[Hwang 92] M.-Y. Hwang et X. Huang. Subphonetic Modeling With Markov States – Senone. *Proceedings IEEE-ICASSP*, pages I–33 – I–36, 1992.

[Jelinek 76] F. Jelinek. Continuous Speech Recognition by Statistical Methods. *IEEE Trans. on ASSP*, 64(4):532 – 556, April 1976.

[Jouvet 93] D. Jouvet, M.N. Lokbani et J. Monné. Application of the N-Best Solutions Algorithm to Speaker-Independent Spelling Recognition over the Telephone. *Proceedings of European Conference on Speech Communication and Technology*, pages 2081 – 2084, 1993.

[Jouvet 94] D. Jouvet, J. Monne, K. Bartkova et C. Mokbel. Structure de modèles de Markov et coefficients acoustiques, 10-11 mars 1994.

[Junqua 94] J-C. Junqua, B. Mak et B. Reaves. A Robust Algorithm for Word Boundary Detection in the Presence of Noise. *IEEE Transactions on Speech and Audio Processing*, 2(3):406 – 412, july 1994.

[Junqua 95a] J.-C. Junqua, D. Fohr, J.-F. Mari, T. H. Applebaum et B. H. Hanson. Time derivatives, Cepstral Normalization, and Parameter Filtering for Continuously Spelled Names over the Telephone. *Proceedings 4th European Conference on Speech Communication and Technology*, volume 1, pages 1385–1388, Madrid (Spain), septembre 1995.

[Junqua 95b] J.-C. Junqua, S. Valente, D. Fohr et J.-F. Mari. An N-Best Strategy, Dynamic Grammars and Selectively Trained Neural Networks for Real-Time Recognition of Continuously Spelled Names over the Telephone. *Proceedings IEEE-ICASSP*, pages 852–855, Detroit (Michigan, USA), mai 1995.

[Junqua 95c] J.C. Junqua, S. Valente, D. Fohr et J.-F. Mari. An N-Best Strategy, Dynamic Grammars and Selectively Trained Neural Networks for Real-Time Recognition of Continuously Spelled Names over the Telephone. *Proc. ICASSP*, pages 852 – 855, Detroit, May 1995.

[Kenny 91] P. Kenny, S. Parthasarathy, V.N. Gupta, M. Lennig, P. Mermelstein et D. O'Shaughnessy. Energy, Duration and Markov Models. *Proceedings of European Conference on Speech Communication and Technology*, volume 2, pages 655 – 658, September 1991.

[Kriouile 90a] A. Kriouile. *La reconnaissance automatique de la parole et les modèles de Markov cachés : modèles du second ordre et distatnce de Viterbi à optimalité locale*. Thèse de Doctorat, Université de NANCY 1, 1990.

[Kriouile 90b] A. Kriouile, J.-F. Mari et J.-P. Haton. L'algorithme de Viterbi-Bloc pour la reconnaissance de la parole continue. *XVIIIèmes journées d'études sur la parole*, pages 207 – 210, Montréal (Quebec), Canada, mai 1990. J.E.P.

[Kriouile 90c] A. Kriouile, J.-F. Mari et J.-P. Haton. Some Improvements in Speech Recognition Algorithms based on HMM. *Proceedings IEEE ICASSP'90 (International Conference Acoustics Speech and Signal Processing)*, pages 545 – 548, Albuquerque, avril 1990.

[Kundu 88] A. Kundu, Y. He et P. Bahl. Recognition of Handwritten Word: First and Second Order Hidden Markov Model Based Approach. *Proc. of IEEE Conference on Computer Vision and Pattern Recognition*, Ann Arbor, Michigan, June 1988.

[Kundu 89] A. Kundu, Y. He et P. Bahl. Recognition of Handwritten Word: First and SAecond Order Hidden Markov Model Based Approach. *Pattern Recognition*, 22(3):283 – 297, 1989.

[Lacouture 95] R. Lacouture. *Algorithmes de recherche lexicale*. Thèse de Doctorat, Université de Mac Hill, 1995.

[Lamel 93] L.F. Lamel et J.L. Gauvain. High Performance Speaker-Independent Phone Recognition using CDHMM. *Proceedings of European Conference on Speech Communication and Technology*, pages 121 – 124, 1993.

[Lamel 95] L. Lamel, M. Adda-Decker et J.L. Gauvain. Issues in Large Vocabulary, Multilingual Speech Recognition. *Proceedings of European Conference on Speech Communication and Technology*, pages 185 – 188, 1995.

[Lebart 95] L. Lebart, A. Morineau et M. Pinon. *Statistique exploratoire multidimensionnelle*. Dunod, 1995.

[Lee 89] K.-F. Lee et H.-W. Hon. Speaker-Independent Phone Recognition Using Hidden Markov Models. *IEEE Transactions on Acoutics, Speech and Signal Processing*, 37(11):1641 – 1648, November 1989.

[Leonard 84] R. G. Leonard. A Database for Speaker Independent Digit Recognition. *Proceedings IEEE-ICASSP*, pages 42.11.1 – 42.11.4, San Diego, CA, March 1984.

[Levinson 86] S. E. Levinson. Continuously Variable Duration Hidden Markov Models for Automatic Speech Recognition. *Computer Speech and Language*, 1:29 – 45, 1986.

[Li 95] H. Li, J.P. Haton et Y. Gong. On MMI Learning of Gaussian Mixture for Speaker Models . *Proceedings of European Conference on Speech Communication and Technology*, pages 363 – 366, 1995.

[Lim 78] J.S. Lim. Evaluation of a Correlation Substraction Method for Enhancing Speech Degraded by Additive White Noise. *IEEE Trans. on Acoustics, Speech and Signal Processing*, 26, 1978.

[Ljolje 94a] A. Ljolje. High Accuracy Phone Recognition Using Context Clustering And Quasi-triphonic Models. *Computer Speech and Language*, 8:129 – 151, 1994.

[Ljolje 94b] A. Ljolje. The Importance of Cepstral Parameter Correlations in Speech Recognition. *Computer Speech and Language*, 8:223 – 232, 1994.

[Lokbani 93] M.N. Lokbani, D. Jouvet et J. Monné. Segmental Post-Processing of the N-Best Solutions in a Speech Recognition System. *Proceedings of European Conference on Speech Communication and Technology*, pages 811 – 814, 1993.

[Lowerre 76] B.T. Lowerre. *The Harpy Speech Recognition System*. Thèse de Doctorat, C.S.D., Carnegie Mellon University, 1976.

[Mari 79] J. F. Mari. *Contribution à l'analyse syntaxique et à la recherche lexicale en reconnaissance du discours continu*. Thèse de Doctorat, Université de NANCY 1, 1979.

[Mari 81] J. F. Mari. Reconnaissance du discours continu par ilots de confiance. *R.A.I.R.O.*, 15(2):167 – 196, 1981.

[Mari 84] J.-F. Mari et J.-P. Haton. Some Experiments in Automatic Recognition of a Thousand Word Vocabulary. *proceedings IEEE ICASSP*, page 26.6, San Diego (USA), mars 1984.

[Mari 85a] J.-F. Mari. Reconnaissance de mots enchaînés à l'aide de modèles markoviens discrets. *Actes Congrès AFCET Reconnaissance des Formes et Intelligence Artificielle*, pages 859 –867, Grenoble, novembre 1985.

[Mari 85b] J.-F. Mari et S. Roucos. Speaker Independent Connected Digit Recognition Using Hidden Markov Models. Academic Press, éditeur, *Speech Technology*, pages 22 – 24, New York (USA), avril 1985.

[Mari 94a] J.-F. Mari, D. Fohr, Y. Anglade et J.-C. Junqua. Hidden Markov Models and Selectively Trained Neural Networks for Connected Confusable Word Recognition. *Proceedings International Conference on Spoken Language Processing*, page S26.11, Yokohama (Japan), septembre 1994.

[Mari 94b] J.-F. Mari et J.-P. Haton. Automatic Word Recognition Based on Second-Order Hidden Markov Models. *Proceedings International Conference on Spoken Language Processing*, page S07.17, Yokohama (Japan), septembre 1994.

[Mari 96] J.-F. Mari, D. Fohr et J.-C. Junqua. A Second-Order HMM for High Performance Word and Phoneme-Based Continuous Speech Recognition. *Proceedings IEEE-ICASSP*, Atlanta, 1996.

[Mauuary 93] L. Mauuary et J. Monné. Speech/non-Speech Detection From Voice Response Systems. *Proceedings of European Conference on Speech Communication and Technology*, pages 1093 – 1100, 1993.

[Moudenc 95] T. Moudenc, D. Jouvet et J. Monné. Improving Recognition Performances on Field Data with a A-Priori Segmentatin of the Speech Signal. *Proceedings of European Conference on Speech Communication and Technology*, pages 1479 – 1282, 1995.

[Normandin 94] Y. Normandin, R. Cardinand et R. De Mori. High-Performance Connected Digit Recognition Using Maximum Mutual Information Estimation . *IEEE Transactions on Speech and Audio Processing*, 2(2):299–311, April 1994.

[Normandin 95] Y. Normandin. Optimal Splitting Of Gaussian Mixture Components With MMIE Training. *Proceedings IEEE-ICASSP*, pages 449–452, Detroit, 1995.

[Paliwal 92] K.K. Paliwal. Dimensionality Reduction of Enhanced Feature Set for the HMM-Bases Speech Recognizer. *Digital Signal Processing*, 2:157 – 173, 1992.

[Pican 96] N. Pican, D. Fohr et J.-F. Mari. HMMs and OWE Neural Network for Continuous Speech Recognition. *Proceedings International Conference on Spoken Language Processing*, Philadelphie, 1996.

[Poritz 88] A.B. Poritz. Hidden Markov Models: A Guided Tour. *Proceedings IEEE-ICASSP*, pages 7 – 13, New York, 1988.

[Pérennou 92] G. Pérennou, D. Cotto, M. De Calmes, I. Ferrané et J.M. Pecatte. Le projet BDLEX de base de données lexicale du français écrit et parlé, 1992.

[Rabiner 85] L. R. Rabiner, B.-H. Juang, S. E. Levinson et M. M. Sondhi. Recognition of Isolated Digits Using Hidden Markov Models With Continuous Mixture Densities. *ATT Technical Journal*, 64(6):1211 – 1234, July-August 1985.

[Rabiner 95] L. Rabiner et B. Juang. *Fundamentals of Speech Recognition*. Prentice Hall, 1995.

[Reddy 76] D.R. Reddy. Speech Recognition by Machine: A Review. *IEEE Trans. on ASSP*, 64(4):501 – 531, April 1976.

[Russell 87] M. J. Russell et A. Cook. Experimental Evaluation of Duration Modelling Techniques For Automatic Speech Recognition. *Proceedings IEEE-ICASSP*, pages 2376 – 2379, Dallas, 1887.

[Russell 88] M. J. Russell. Statistical modelling of state duration correlations in hidden Markov Models. *7th FASE Symposium*, Edinburgh, August 1888.

[Russell 85] M.J. Russell et R. K. Moore. Explicit Modelling of State Occupancy in Hidden Markov Models for Automatic Speech Recognition . *Proceedings IEEE-ICASSP*, volume 1, pages 5 – 8, March 1985.

[Saon 96] G. Saon et A. Belaid. Recognition of Unconstrained Handwritten Words Using Markov Random Fields and HMMs . *Vth. Workshop on Frontiers in Hand-Writing Recognition*, Univ. of Essex, England, 1996.

[Saporta 90] G. Saporta. *Théories et méthodes de la statistique*. Publications de l'institut français du pétrole, 1990.

[Schwartz 90] R. Schwartz et Y. L. Chow. The N-Best Algorithm: And Efficient and Exact Procedure for Finding the N Most Likely Sentence Hypotheses. *Proceedings IEEE-ICASSP*, pages 81 – 84, Albuquerque, April 1990.

[Soong 88] F. K. Soong et A. E. Rosenberg. On the Use of Instantaneous and Transitional Spectral Information in Speaker Recognition. *IEEE Transactions on Acoutics, Speech and Signal Processing*, 1988.

[Suaudeau 93] N. Suaudeau et R. André-Obrecht. Sound Duration Modelling and time variable Speaking rate in a Speech Recognition System. *Proceedings of European Conference on Speech Communication and Technology*, pages 307 – 310, 1993.

[Thiébaut 96] E. Thiébaut, J.-F. Mari, J.-P. Haton, Y. Gong et D. Fohr. Utilisation d'un système de reconnaissance de la parole pour accéder à W3 . *XXIèmes Journ'ees d'Etude sur la Parole*, pages 429 – 432, Avignon, 1996.

[Tou 74] J. T. Tou et R. Gonzales. *Pattern Recognition Principles*. Addison-Wesley, 1974.

[Wassner 96] H. Wassner et G. Chollet. Une nouvelle estimation de coefficients cepstraux pour la reconnaissance automatique de la parole. *XXIèmes Journ'ees d'Etude sur la Parole*, pages 259 – 262, Avignon, 1996.

[Wilpon 93] J. G. Wilpon, C.-H. Lee et L. R. Rabiner. Connected Digit Recognition Based on Improved Acoustic Resolution. *Computer Speech and Language*, 7:15 – 26, 1993.

[Young 93] S.J. Young. The HTK HMM Toolkit: Design and Philosophy. *Technical report CUED/F-INFENG/TR.152*, 1993.

[Young 94] S.J. Young et P.C. Woodland. State Clustering in Hidden Markov Model-Based Continuous Speech Recognition. *Computer Speech and Language*, 8:369 – 383, 1994.