

# AUTO-ADAPTATION ET RECONNAISSANCE AUTOMATIQUE DE LA PAROLE

## THÈSE

présentée et soutenue publiquement le ... juin 2010

pour l'obtention du

**Doctorat de l'Université du Maine**  
**(spécialité informatique)**

par

**ANTOINE LAURENT**

### Composition du jury

<i>Président :</i>	Frédéric Bimbot	Professeur	IRISA, Université de Rennes
<i>Rapporteurs :</i>	Frédéric Béchet Régine André-Obrecht	Professeur Professeure	LIF, Université de Marseille IRIT, Université de Toulouse
<i>Examinateurs :</i>	Paul Deléglise Sylvain Meignier	Professeur Maître de Conférences	LIUM, Université du Maine LIUM, Université du Maine
<i>Invité :</i>	Domique Ristori	Gérant	Spécinov, Trélazé, Maine et Loire



## **Remerciements**

Merci...



# Table des matières

<b>Table des figures</b>	<b>ix</b>
--------------------------	-----------

<b>Liste des tableaux</b>	<b>xi</b>
---------------------------	-----------

<b>Introduction : Contexte applicatif</b>	<b>1</b>
1    Contexte . . . . .	2
2    Problématique . . . . .	3
3    Adaptation synchrone / asynchrone . . . . .	4
4    Structure du document . . . . .	5

---

<b>Partie I Transcription d'enregistrements</b>	<b>7</b>
---	----------

---

<b>Chapitre 1</b>	
<b>Méthodes et outils pour la reconnaissance automatique de la parole</b>	
1.1    Introduction . . . . .	10
1.2    Contexte . . . . .	10
1.3    Vue générale d'un système de reconnaissance automatique de la parole . . . . .	11
1.3.1    Paramétrisation du signal . . . . .	12
1.3.2    Dictionnaire de phonétisations . . . . .	14
1.3.3    Les modèles acoustiques . . . . .	15
1.3.4    Les modèles de langage . . . . .	19

*Table des matières*

---

1.3.5	Segmentation en locuteur . . . . .	23
1.3.6	Décodage . . . . .	23
1.3.7	Métrique . . . . .	25
1.3.8	Performances . . . . .	26
1.3.9	Correction des transcriptions automatiques . . . . .	26
1.4	Conclusion . . . . .	28

---

**Partie II Assistance automatique à la transcription manuelle** **29**

---

<b>Chapitre 2</b>	
<b>État de l'art</b>	<b>33</b>

2.1	Introduction . . . . .	34
2.2	Stratégies d'affichage pour l'assistance à la correction de transcriptions . .	34
2.3	Traduction assistée par ordinateur . . . . .	37
2.4	Réordonnancement des hypothèses . . . . .	37
2.5	Conclusion . . . . .	39

<b>Chapitre 3</b>	
<b>Méthode proposée</b>	<b>41</b>

3.1	Réordonnancement automatique des hypothèses de reconnaissance . . . . .	42
3.2	Méthode proposée . . . . .	43
3.2.1	Principe . . . . .	43
3.2.2	Application . . . . .	45
3.2.3	Exemple . . . . .	46
3.3	Modèle cache . . . . .	47
3.4	Mots hors vocabulaire . . . . .	48

---

<b>Chapitre 4</b>	
<b>Expériences et résultats</b>	<b>49</b>

4.1	Corpus & SRAP . . . . .	50
4.2	Métriques . . . . .	50
4.3	Résultats . . . . .	54
4.3.1	Sans utiliser la méthode de réordonnancement automatique . . . . .	54
4.3.2	En utilisant la méthode de réordonnancement automatique . . . . .	54

<b>Chapitre 5</b>	
<b>Conclusion : Assistance automatique à la transcription manuelle</b>	<b>57</b>

---

---

<b>Partie III</b>	<b>Phonétisation automatique</b>	<b>59</b>
-------------------	----------------------------------	-----------

---

<b>Chapitre 6</b>	
<b>État de l'art : méthodes de phonétisation</b>	<b>63</b>

6.1	Introduction . . . . .	64
6.2	Système à base de règles . . . . .	64
6.3	Systèmes guidés par les données . . . . .	65
6.3.1	Prononciation par classifications locales . . . . .	66
6.3.2	Prononciation par analogie . . . . .	67
6.3.3	Utilisation des données acoustiques . . . . .	70
6.4	Conclusion . . . . .	71

<b>Chapitre 7</b>	
<b>Méthode proposée</b>	<b>73</b>

7.1	Introduction . . . . .	74
7.2	Méthodes de G2P utilisées pour construire le dictionnaire initial . . . . .	76
7.2.1	Système à base de règles . . . . .	76
7.2.2	Corpus parallèle ( <i>bitext</i> ) . . . . .	76

*Table des matières*

---

7.2.3	Système à base de modèles à séquences jointes (JSM) . . . . .	77
7.2.4	Utilisation d'un système SMT (Statistical Machine Translation) pour la conversion G2P . . . . .	78
7.3	Extraction de phonétisations à l'aide d'un DAP . . . . .	79
7.4	Filtrage des variantes de phonétisation . . . . .	81
7.4.1	Motivation . . . . .	81
7.4.2	Méthodes . . . . .	81
7.5	Méthode itérative de génération des phonétisations . . . . .	84
7.5.1	Résumé de la méthode . . . . .	84

<b>Chapitre 8</b>
-------------------

<b>Expériences et résultats</b>	<b>87</b>
---------------------------------	-----------

8.1	Expériences . . . . .	88
8.1.1	Corpus . . . . .	88
8.1.2	Modèles acoustiques et linguistiques . . . . .	88
8.1.3	Métrique . . . . .	90
8.2	Résultats . . . . .	91
8.2.1	Nombre de variantes de phonétisation par nom propre . . . . .	91
8.2.2	En utilisant une seule itération globale (alignement / extraction / filtrage) . . . . .	91
8.2.3	En utilisant le processus itératif complet . . . . .	93

<b>Chapitre 9</b>
-------------------

<b>Conclusion : Phonétisation automatique</b>	<b>97</b>
---	-----------

<b>Chapitre 10</b>
--------------------

<b>Conclusion et perspectives</b>	<b>101</b>
-----------------------------------	------------

10.1	Réordonnancement automatique des hypothèses de reconnaissance . . . . .	102
10.1.1	Méthode proposée . . . . .	102
10.1.2	Perspectives . . . . .	102
10.2	Phonétisation automatique des noms propres . . . . .	103
10.2.1	Méthode proposée . . . . .	103
10.2.2	Perspectives . . . . .	103
10.3	Perspectives générales . . . . .	104

---

**Annexe A****Applications pour la transcription manuelle****105**

A.1 Transcriber . . . . .	106
A.2 Praat . . . . .	107
A.3 WinPitch . . . . .	108
A.4 XTrans . . . . .	109
A.5 Conclusion . . . . .	111

**Acronymes****113****Bibliographie personnelle****117****Bibliographie****119****Résumé****132**

*Table des matières*

---

# Table des figures

1.1	Vue d'ensemble d'un système de SRAP . . . . .	13
1.2	Représentation d'un MMC à 5 états . . . . .	16
1.3	Decodage ESTER 2 . . . . .	25
2.1	Extrait de l'article [Nanjo 2006] . . . . .	35
2.2	Extrait de l'article [Cardinal 2007] . . . . .	36
2.3	Exemple de CATS – Extrait de l'article [Rodríguez 2007] . . . . .	38
3.1	Graphe de mots . . . . .	42
3.2	Réseau de confusion . . . . .	43
3.3	Réseau de confusion avec des informations temporelles et la meilleure hypothèse	43
3.4	Réseau de confusion : 2 chemins possibles . . . . .	46
4.1	Méthode de calcul . . . . .	51
4.2	Interface à laquelle pourrait ressembler l'outil d'aide à la correction . . . . .	52
4.3	Calcul du taux d'erreur mot de la méthode de réordonnancement automatique .	53
6.1	Exemple de conversion par classification locale de graphème vers phonème .	66
6.2	Exemple de conversion par analogie de graphème vers phonème . . . . .	67
6.3	Réseau de confusion du système PRONOUNCE [Dedina 1991] . . . . .	68
6.4	Réseau de confusion (extrait de l'article [Marchand 2001]) . . . . .	69
7.1	Principe de base du système (extrait de l'article [Jousse 2008]) . . . . .	74
7.2	Méthode proposée . . . . .	75
7.3	Illustration de l'utilisation du décodage acoustico-phonétique . . . . .	80
7.4	Représentation du processus de sélection non itératif . . . . .	82
7.5	Représentation du processus de filtrage itératif . . . . .	83
7.6	Processus complet d'extraction/filtrage des variantes de prononciation des noms propres . . . . .	84

*Table des figures*

---

8.1	PNER en utilisant chacune des méthodes de G2P (Corpus de test ESTER 1) . . . . .	92
8.2	WER sur le corpus de test sur les segments contenant des noms propres . . . . .	93
8.3	WER utilisant le processus itératif complet (corpus de test ESTER 1, segments contenant des noms propres) . . . . .	93
8.4	PNER en utilisant le processus itératif complet (corpus de test ESTER 1, segments contenant des noms propres) . . . . .	94
8.5	PNER et WER en utilisant le processus itératif complet (corpus de test ESTER 1, segments contenant des noms propres) . . . . .	94
8.6	WER sur l'ensemble des segments sur le corpus de test ESTER 1 . . . . .	96
A.1	Capture d'écran du logiciel Transcriber . . . . .	107
A.2	Capture d'écran du logiciel Praat . . . . .	108
A.3	Capture d'écran du logiciel Winpitch . . . . .	109
A.4	Capture d'écran du logiciel XTrans . . . . .	110

# Liste des tableaux

1.1	<i>Taux d'erreur mot sur le corpus de test ESTER 2</i>	26
1.2	<i>Durée totale de la transcription (durées respectives des corpus : 2h08 et 2h10)</i>	27
1.3	<i>Résultats des différents participants à la campagne ESTER 2</i>	28
4.1	<i>KSR et WER sur le corpus de test ESTER 2 sans aide à la transcription</i>	55
4.2	<i>KSR et WSR sur le corpus de test ESTER 2 avec aide à la transcription</i>	55
6.1	<i>Résumé et comparaison de la précision de différents systèmes de G2P sur des corpus anglophones (Extrait de [Bisani 2008])</i>	72
7.1	<i>Exemple des représentations A, B et C du corpus bitext (phonèmes au format Sampa)</i>	77
8.1	<i>Nombre de variantes de phonétisation</i>	91
8.2	<i>Résumé des résultats obtenus sur le corpus de test d'ESTER 1</i>	95
9.1	<i>Résultats obtenus sur le corpus de test d'ESTER 1</i>	99

*Liste des tableaux*

---

# **Introduction : Contexte applicatif**

## 1 Contexte

La société Spécinov, SSII (Société de Services en Ingénierie Informatique) située à Trélazé dans le Maine et Loire, souhaite réaliser une application d'aide à la gestion de réunions qui intègrerait un Système de Reconnaissance Automatique de la Parole (SRAP). De nombreuses situations nécessitent de garder des traces des réunions, comme leurs enregistrements sonores et leurs transcriptions. Ces transcriptions sont actuellement réalisées manuellement en saisissant dans un logiciel de traitement de texte les mots prononcés par les locuteurs. Une autre méthode consiste à confier le travail de transcription à un sténotypiste. Celui-ci est formé à utiliser un appareil, appelé sténotype, proche d'une machine à écrire permettant de saisir le flux de parole sous forme de caractères spéciaux (sténogrammes) basés sur les syllabes prononcées. Ces sténogrammes sont ensuite transformés automatiquement à l'aide d'un logiciel spécialisé (par exemple TASF+ développé par IBM) en une suite de mots ; ceci avec un nombre d'erreurs non négligeable qui seront ensuite corrigées manuellement. Dans les deux cas, le coût des transcriptions est élevé. Dans le premier cas, la transcription manuelle est faite en plus de 10 heures pour obtenir la transcription d'une heure de réunion. Dans le second cas les sténotypistes ne sont pas nombreux, leur formation est longue et difficile, ce qui justifie le montant élevé de leurs prestations. Bien que les technologies de transcription automatique aient des performances correctes dans des contextes d'utilisation connus et contrôlés (conditions d'enregistrement, vocabulaire du métier), la qualité n'est pas encore suffisante pour permettre une exploitation directe des transcriptions. Pour le français, dans de bonnes conditions d'enregistrement, environ un mot sur dix comporte une erreur dans les résultats des meilleurs systèmes de transcription automatique [Galliano 2005]. Actuellement, il est donc nécessaire d'effectuer des corrections manuelles en écoutant quasi-intégralement l'enregistrement pour corriger les erreurs de transcription. Les travaux présentés dans ce document proposent des outils pour faciliter et accélérer cette phase de validation, inéluctable au vu de la maturité actuelle des systèmes de transcription. La collaboration entre la société Spécinov et le LIUM (Université du Maine), matérialisée par la mise en place de ma thèse dans le cadre d'une convention CIFRE (Conventions Industrielles de Formation par la REcherche), a pour objectif de fournir des outils d'aide à la correction de textes générés automatiquement par un système de reconnaissance de la parole, en intégrant dans le processus de correction le système de transcription automatique lui-même. Ces outils seront utilisables sans formation supplémentaire par un utilisateur maîtrisant les outils standards de l'informatique. L'outil d'aide à la gestion de réunion visé par la société Spécinov intègrera une méthode d'indexation automatique des réunions transcrrites, afin de pouvoir naviguer aisément entre les différents documents disponibles. Un soin particulier devra être apporté à la qualité de la transcription des noms des participants qui semble être un élément discriminant et important

pour cette tâche. En effet, rechercher les interventions d'un participant dans divers documents audio pourrait être l'une des fonctionnalités envisagées.

## 2 Problématique

Les systèmes de reconnaissance de la parole sont développés pour une tâche donnée dans un contexte d'utilisation connu, comme par exemple la transcription d'émissions radiophoniques et télévisées, ou la transcription de conversations téléphoniques. Les performances des systèmes de transcription sont bonnes lorsque deux éléments critiques sont bien maîtrisés : la qualité de la prise de son et la disponibilité d'enregistrements représentatifs du contexte d'utilisation. Ces enregistrements permettent d'estimer les modèles acoustiques et linguistiques inhérents aux systèmes de transcription. L'objectif du projet étant de créer une application d'aide à la transcription de réunion, le premier verrou à lever concerne la maîtrise de la prise de son. Il s'agit de fournir au système des enregistrements de bonne qualité avec peu de bruit, peu d'écho et peu ou pas de parole superposée (locuteurs s'exprimant simultanément). La résolution de ce verrou a été partiellement étudié dans une partie du projet ne concernant pas cette thèse. De plus, l'application d'aide à la transcription devra permettre d'accélérer la phase de post-traitement des textes générés automatiquement par le SRAP. Un second verrou porte sur le développement d'une méthode permettant d'aider l'utilisateur en l'assistant dans l'étape de correction des sorties du SRAP. De plus, l'assistance apportée devra être suffisamment rapide pour pouvoir être intégrée dans une application interactive. Le troisième verrou concerne la phonétisation des noms propres. L'un des objectifs de ces travaux de thèse est de proposer une méthode permettant de transcrire correctement les noms propres, de façon à faciliter l'indexation automatique des réunions transcrites. Tous les mots du vocabulaire du SRAP doivent être phonétisés (déterminer la suite de sons - phonèmes - qui doivent être émis pour prononcer chaque mot) afin de pouvoir apparaître dans le résultat du SRAP. La phonétisation des noms propres est plus difficile à obtenir que celle des noms communs. En effet, un nom propre écrit de la même manière sera prononcé différemment selon l'origine de ce nom et selon l'origine du locuteur. Il s'agira donc de mettre à profit les données disponibles pour proposer une méthode permettant de déterminer les séquences de phonèmes composant chacun des noms propres rencontrés. Enfin, le dernier verrou porte sur la recherche de méthodes permettant au système de s'enrichir d'un point de vue global en capitalisant les transcriptions déjà réalisées.

### **3 Adaptation synchrone / asynchrone**

La production d'une transcription avec un nombre d'erreur réduit permet d'améliorer l'hypothèse initiale fournit par le SRAP, et ainsi permet de diminuer la durée de la phase de post-traitement (correction).

Deux stratégies d'adaptation du SRAP ont été identifiées. La première consiste à intégrer le système dans la phase de post-traitement elle même. Il s'agit d'une méthode d'adaptation synchrone, dans laquelle utilisateur et système collaborent en vu d'obtenir, en un nombre d'actions réduit, une transcription correcte. Chaque action corrective de la part de l'utilisateur entraîne une réévaluation de l'hypothèse du SRAP. Dans ce manuscrit, une méthode d'adaptation synchrone est proposée.

La seconde stratégie identifiée consiste à adapter les différents modèles, de sorte que la solution puisse s'auto-enrichir en capitalisant, au fur et à mesure de ses utilisations, ses bases de connaissance. L'adaptation consiste donc à utiliser les transcriptions déjà corrigées pour diminuer les taux d'erreurs des futurs décodages (méthode d'adaptation asynchrone). Les éléments sur lesquels peuvent porter cette adaptation sont les suivants :

- Le vocabulaire. Chaque communauté utilise un vocabulaire qui lui est propre.
- Les phonétisations. Chaque nouveau mot introduit dans le vocabulaire doit être associé avec une ou plusieurs phonétisations afin d'apparaître dans le résultat du décodage. Concernant l'amélioration du décodage des noms propres, ce point est crucial, car la phonétisation des noms propres est difficile à obtenir. Une méthode, utilisant les données acoustiques à disposition, est proposée dans ce manuscrit. Elle est basée sur la combinaison de différentes techniques de conversions graphèmes-phonèmes avec un système de décodage acoustico-phonétique.
- Le modèle de langage. Ce modèle permet de déterminer les probabilités des différentes séquences de mots pouvant apparaître dans le résultat du décodage. De nombreux travaux traitent de l'adaptation des modèles de langage dans la littérature. Notamment, les auteurs de [Oger 2008] et de [Allauzen 2003] proposent des stratégies basées sur l'utilisation de données provenant d'Internet et l'utilisation de métadonnées pour l'adaptation thématique des lexiques et modèles de langage.
- Les modèles acoustiques. La disponibilité de corpora de textes synchronisés avec leurs enregistrements audio permet d'adapter les modèles acoustiques initiaux aux conditions acoustiques d'utilisation. De nombreuses stratégies sont également présentes dans la littérature. Dans le chapitre 1 il est fait référence à certaines d'entre elles, comme par exemple l'utilisation de techniques de régression linéaires contraintes (CMLLR - Constrained Maximum Likelihood Linear Regression) [Digalakis 1995] ou non (MLLR

- Maximum Likelihood Linear Regression) [Leggetter 1995] ou l'utilisation du critère MAP (Maximum *a posteriori*) [Gauvain 1994].

## 4 Structure du document

Dans le chapitre suivant, le principe de fonctionnement d'un SRAP, et plus spécifiquement celui développé et utilisé au LIUM, sera présenté. Dans ce même chapitre, des expériences montrant l'intérêt d'utiliser un SRAP comme point de départ pour réaliser des transcriptions plutôt que de partir d'un document vide seront exposées. Ces travaux ont été réalisés par Thierry Bazillon (doctorant linguiste au LIUM).

La seconde partie de ce manuscrit porte sur la description d'une méthode d'assistance à la transcription, basée sur le réordonnancement des hypothèses du SRAP en fonction des corrections apportées par l'utilisateur.

La troisième partie de ce manuscrit présente une technique de phonétisation des noms propres, utilisant les données acoustiques à disposition. La méthode se base sur l'utilisation des transcriptions corrigées pour extraire la manière dont les noms propres ont "réellement" été prononcés.

Pour conclure, nous discuterons des différentes perspectives que permettent d'envisager nos travaux.



## **Première partie**

### **Transcription d'enregistrements**



# Chapitre 1

## Méthodes et outils pour la reconnaissance automatique de la parole

### Sommaire

---

<b>1.1</b>	<b>Introduction</b>	<b>10</b>
<b>1.2</b>	<b>Contexte</b>	<b>10</b>
<b>1.3</b>	<b>Vue générale d'un système de reconnaissance automatique de la parole</b>	<b>11</b>
1.3.1	Paramétrisation du signal	12
1.3.2	Dictionnaire de phonétisations	14
1.3.3	Les modèles acoustiques	15
1.3.4	Les modèles de langage	19
1.3.5	Segmentation en locuteur	23
1.3.6	Décodage	23
1.3.7	Métrique	25
1.3.8	Performances	26
1.3.9	Correction des transcriptions automatiques	26
<b>1.4</b>	<b>Conclusion</b>	<b>28</b>

---

## 1.1 Introduction

La société Spécinov souhaite développer une application d'aide à la gestion de réunions, et plus spécifiquement désire apporter aux utilisateur de l'application visée, une technique permettant de les aider dans la phase de rédaction des comptes rendus associés. Les travaux menés par Thierry Bazillon (Doctorant en linguistique au Mans) [Bazillon 2008b, Bazillon 2008a], présentés dans ce chapitre, montrent l'intérêt d'utiliser les sorties du SRAP du LIUM comme base de travail pour réaliser des transcriptions. L'idée consiste à utiliser un SRAP pour obtenir une transcription automatique d'un enregistrement puis d'injecter le résultat du traitement dans un outil d'aide à la transcription. Ce chapitre présente les différentes méthodes et les différents outils mis en place dans le SRAP du LIUM, ainsi que la façon dont les différents modèles qui le composent ont été appris et adaptés.

## 1.2 Contexte

Le LIUM a choisi d'utiliser comme base de son système de reconnaissance automatique de la parole le système Sphinx [Seymore 1998] développé par Carnegie Mellon University (CMU). Sphinx est l'un des plus anciens et des meilleurs décodeurs probabilistes. De plus, il est diffusé sous licence libre depuis 2000. Le projet SPHINX a débuté en 1986 et une première description précise du système a été présentée dans [Lee 1990]. Il s'agissait à l'époque de développer un système de reconnaissance de la parole continue, à grand vocabulaire et indépendant du locuteur. Grand vocabulaire signifiait un vocabulaire contenant au moins 1000 mots. Ce projet était financé par la NSF (National Science Foundation) et la DARPA (Defence Advanced Project Agency).

Les campagnes ESTER ont été un facteur moteur dans le développement d'un système grand vocabulaire basé sur Sphinx au LIUM. La première campagne (ESTER 1) a commencé en 2003 et s'est terminée en janvier 2005. La seconde (ESTER 2) s'est déroulée de 2007 à 2008.

La campagne d'évaluation ESTER 1 a été organisée dans le cadre du projet Technolangue financé par le gouvernement français sous la supervision scientifique de l'AFCP (Association Francophone de la Communication Parlée), de la DGA (Délégation Générale de l'Armement) et de ELDA (Evaluations and Language resources Distribution Agency). Le corpus fourni par la campagne d'évaluation est composé d'environ 100 heures d'émissions radiophoniques transcrrites, enregistrées entre 1998 et 2004 et provenant de six radios : France Inter, France Info, RFI, RTM, France Culture et Radio Classique. Chaque émission a une durée comprise entre 10 minutes et 1 heure. La plupart des émissions contiennent de la parole préparée (reportages,

### *1.3. Vue générale d'un système de reconnaissance automatique de la parole*

---

informations). Des articles du journal “Le Monde” de 1987 à 2003 peuvent être utilisés en plus du corpus transcrit pour apprendre le modèle de langage.

La campagne d'évaluation ESTER 2 se trouve dans la continuité d'ESTER 1. Elle a été organisée par la DGA et l'AFCP entre 2007 et 2008. Cette nouvelle édition de la campagne d'évaluation reprend le corpus qui était déjà fourni pour la campagne ESTER 1, et est étendu pour permettre de couvrir de nouveaux types de données. En particulier, ESTER 2 inclut plus d'émissions mettant en jeu des locuteurs avec des accents étrangers, et des émissions de parole spontanée. En complément des émissions des 6 radios françaises, la campagne inclut des débats et des programmes provenant de la radio Africaine Radio Africa No 1. Les données ajoutées par la campagne ESTER 2 sont les suivantes :

- 100 heures d'émissions radiophoniques transcrrites manuellement, enregistrées entre 1998 et 2006,
- 6 heures pour le développement et 6 heures pour le test enregistrées en 2007 et 2008,
- 40 heures d'émissions radiophoniques africaines avec transcriptions manuelles rapides.

Les ressources textuelles sont elles aussi étendues avec des articles du journal “Le Monde” de 2004-2006 (en complément des articles de 1987 à 2003). Le système du LIUM présenté lors de la campagne ESTER 2 a été le meilleur système libre de SRAP, avec 24,2 % de WER (Word Error Rate - Taux d'erreur mot) sur le corpus de développement et 17,8% sur le corpus de test. Le corpus de développement contient plus de parole dégradée que le corpus de test, ce qui explique la différence de WER entre les résultats obtenus sur ces deux corpus.

## **1.3 Vue générale d'un système de reconnaissance automatique de la parole**

La volonté de la société Spécinov consiste à transcrire principalement des réunions. Uniquement les systèmes de reconnaissance automatique de la parole (SRAP) probabilistes à grand vocabulaire seront décrits. Bien qu'il existe d'autres méthodes, comme les réseaux de neurones par exemple, l'approche basée sur l'utilisation de méthodes statistiques [Rabiner 1986] dominent depuis plus de 20 ans.

Un SRAP a pour objectif de transcrire sous forme textuelle un signal.

A partir d'une séquence d'observations acoustiques  $X$ , l'objectif est de trouver la séquence de mots prononcés par les locuteurs. Il s'agit donc de déterminer la suite de mots  $\hat{W}$  la plus probable parmi l'ensemble des séquences possibles  $W$ . Cela se traduit par la recherche de  $\hat{W}$

maximisant la probabilité d'émission de  $W$  sachant  $X$  correspondant à l'équation suivante :

$$\hat{W} = \arg \max_W P(W|X) \quad (1.1)$$

avec  $P(W|X)$  la probabilité d'émission de  $W$  sachant  $X$ . Après application du théorème de Bayes, cette équation devient :

$$\hat{W} = \arg \max_W \frac{P(X|W)P(W)}{P(X)} \quad (1.2)$$

où  $P(X)$  représente la probabilité d'observation de la séquence acoustique  $X$ , qui peut être considérée comme une valeur constante et donc être retirée de l'équation 1.2. Nous avons donc :

$$\hat{W} = \arg \max_W P(X|W)P(W) \quad (1.3)$$

La probabilité  $P(W)$  est fourni par le modèle de langage, et la probabilité  $P(X|W)$  correspond à la probabilité attribuée par les modèles acoustiques. Schématiquement, dans le cas d'un modèle acoustique dont l'unité de représentation est le phonème, ce dernier va permettre de rechercher les suites de phonèmes<sup>1</sup> les plus probables à partir des observations acoustiques. Les suites de phonèmes sont contraintes par le dictionnaire de phonétisation qui permet d'associer chacun des mots du vocabulaire avec sa (ou ses) représentation(s) phonétique(s). Le modèle de langage va permettre de sélectionner, parmi toutes les séquences de mots possibles, celle qui a la plus grande probabilité d'apparition.

La figure 1.1 présente une vue d'ensemble d'un SRAP probabiliste, avec les trois ressources essentielles à son fonctionnement (modèles acoustiques, dictionnaire de phonétisation et modèle de langage).

### 1.3.1 Paramétrisation du signal

Le signal acoustique présente, dans le domaine temporel, une redondance qui rend obligatoire un traitement préalable à toute tentative de reconnaissance. Le rôle d'un module de paramétrisation est d'extraire du signal des informations caractéristiques et pertinentes, adaptées à la tâche visée. Ces informations sont représentées sous la forme d'une suite discrète de vecteurs, appelés *vecteurs caractéristiques du signal de parole* ou plus simplement, *vecteurs acoustiques*. Le signal de la parole étant variable au cours du temps, l'extraction des vecteurs d'observation est généralement faite sur des fenêtres d'analyse temporelle de faible durée (de quelques dizaines de millisecondes), de telle sorte que le signal puisse être considéré comme

---

<sup>1</sup>Le phonème est l'unité minimale du langage parlé

### 1.3. Vue générale d'un système de reconnaissance automatique de la parole

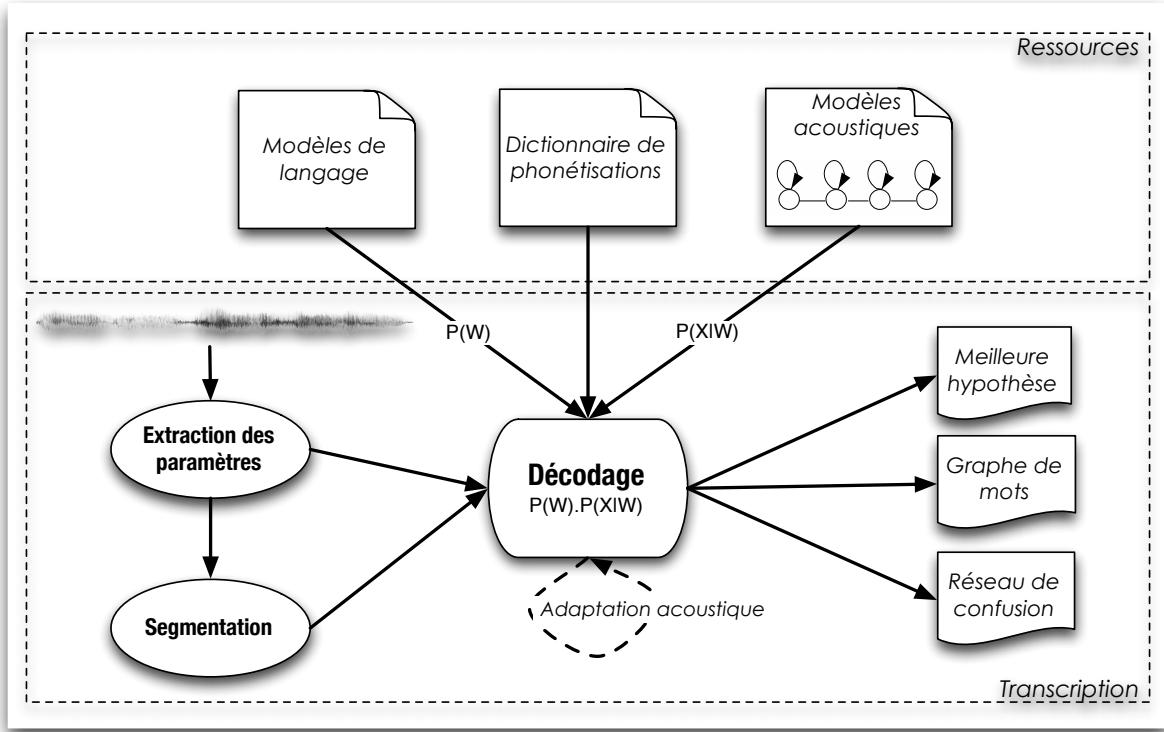


FIG. 1.1 – Vue d’ensemble d’un système de SRAP

stationnaire sur chacune d’entre elles. Une transformée de Fourier est calculée sur chacune de ces fenêtres d’analyse. Le résultat de ces transformées est appelé *spectre à court terme*. La concaténation de l’ensemble des spectres à court terme, par glissement de la fenêtre d’analyse, forme un spectrogramme. Celui-ci représente l’évolution dans le plan temps/fréquence de l’énergie du signal.

Il existe de nombreuses paramétrisations acoustiques [d’Alessandro 1992, Reynolds 1994, Homayounpour 1994, Charlet 1997]. Les paramètres couramment utilisés en reconnaissance automatique de la parole sont les paramètres MFCC (pour "Mel Frequency Cepstral Coefficients") et PLP (pour "Perceptual Linear Prediction"). Pour les paramètres MFCC et PLP, une transformation de type cepstrale est réalisée. La transformation cepstrale est la transformée de Fourier inverse du logarithme de la transformée de Fourier du signal.

Les coefficients MFCC sont des coefficients cepstraux obtenus à partir des énergies d’un banc de filtre à échelle de fréquence Mel [Davis 1980]. L’échelle Mel est linéaire jusqu’à 100 Hz et logarithmique au-delà : c’est une échelle proche du traitement non linéaire de l’oreille humaine.

Les modèles PLP reposent eux aussi sur l’utilisation de la transformation cepstrale, et sont issus d’une analyse en banc de filtre à échelle Bark. Un modèle autorégressif est ensuite estimé

sur la racine cubique des énergies de sortie [Hermansky 1990, Junqua 1990]. Cette échelle réduite tend aussi à se rapprocher de l'oreille humaine, qui possède une bonne résolution spectrale en basse fréquence et mauvaise en haute fréquence. Les coefficients PLP sont légèrement plus robustes que les coefficients MFCC en présence de bruit de fond [Kershaw 1996].

Les vecteurs acoustiques sont généralement complétés de leurs dérivées premières et secondes [Furui 1981, Soong 1988], car elles fournissent des informations sur les changements temporelles du spectre. Des techniques de réduction de dimension comme PCA (Principal Component Analysis) [Pinkowski 1997] ou LDA (Linear Discriminant Analysis) [Katz 2002] peuvent être appliquées aux vecteurs acoustiques pour obtenir des représentations plus compactes et discriminantes.

Les paramètres acoustiques peuvent être normalisés de façon à être rendus plus robustes aux distorsions dues aux canaux de transmissions. La soustraction de la moyenne cepstrale (Cepstral Mean Subtraction - CMS) [Hermansky 1994] est la méthode la plus fréquemment employée en reconnaissance automatique de la parole. Une autre technique de normalisation largement répandue est celle de la normalisation de la longueur de conduit vocal (Vocal Track Length Normalization - VTLN [Cohen 1995]). Cette normalisation repose sur une modification linéaire de l'échelle des fréquences afin de compenser les différences de longueur de conduit vocal entre les locuteurs.

Les paramètres acoustiques utilisés par le LIUM sont au nombre de 39 : il s'agit de descripteurs issus d'une analyse du signal de type PLP et d'un descripteur de l'énergie, ainsi que des dérivées premières et secondes de ces descripteurs.

### **1.3.2 Dictionnaire de phonétisations**

Le dictionnaire de phonétisations est l'élément qui permet de faire le lien entre le niveau acoustique et le niveau lexical lors du décodage. Il est également indispensable durant la phase d'apprentissage des modèles acoustiques. Le système du LIUM utilise comme unité de modélisation le phonème. Le dictionnaire comprend une liste de mots, chacun associé à une ou plusieurs séquences de phonèmes. La création du dictionnaire de phonétisations implique donc de disposer d'un jeu de phonèmes et d'un vocabulaire.

Le décodeur n'est capable que de fournir des suites de mots connus (présents dans son vocabulaire). Ce vocabulaire étant de taille finie, il ne couvre pas tous les mots du langage. Il arrive donc qu'un mot prononcé ne soit pas présent dans le vocabulaire, il est alors désigné sous le terme « mot hors vocabulaire » (OOV – *Out Of Vocabulary*). La présence d'un mot hors vocabulaire durant le processus de transcription automatique engendre plusieurs erreurs. En effet, pour le français sur le corpus ESTER 1 (défini en 1.2), [Dufour 2008] montre que la

### *1.3. Vue générale d'un système de reconnaissance automatique de la parole*

---

présence d'un mot hors vocabulaire propage des erreurs sur les mots qui lui sont proches. En moyenne, parmi les trois mots précédant le mot hors vocabulaire, 42% des mots sont erronés, et parmi les trois mots suivants, 78% des mots sont également en erreur. Le choix du vocabulaire est donc un élément crucial au développement d'un système de SRAP.

Le jeu de phonème utilisé pour représenter les mots du vocabulaire est dépendant de la langue pour laquelle le SRAP est développé. Le nombre de phonèmes varie selon la langue. Les valeurs constatées sont généralement proches de 35 phonèmes pour le français, 45 pour l'anglais et 26 pour l'espagnol. Un SRAP utilise un jeu de phonèmes permettant de représenter l'ensemble des mots de son vocabulaire, et également un ensemble d'unités servant à modéliser des phénomènes acoustiques tels que les silences, la musique, les hésitations, etc.

Le processus de construction du dictionnaire de phonétisations consiste à déterminer le vocabulaire qui sera utilisé par le SRAP, et à obtenir les prononciations des mots correspondants. Le choix de la taille du vocabulaire devra être fait de façon à couvrir au mieux les mots du domaine sur lequel le système sera ensuite utilisé, et devra prendre en compte les limitations logicielles et matérielles pouvant intervenir, comme par exemple le temps de calcul, les contraintes mémoire et la quantité de texte permettant d'apprendre le modèle de langage.

La phonétisation est un problème à part entière qui fait l'objet d'une étude dans la partie III.

Les phonétisations des mots composant le vocabulaire du système du LIUM ont été obtenues en utilisant le dictionnaire de phonétisations BDLEX [De Calmes 1998]. Pour les mots inexistant dans le dictionnaire BDLEX, le système de phonétisation automatique à base de règles LIA\_PHON [Béchet 2001] a été utilisé (voir partie III).

Pour construire le vocabulaire, un modèle unigramme<sup>2</sup> résultant de l'interpolation linéaire<sup>3</sup> des modèles unigrammes des trois sources de données listées dans le paragraphe précédent a été réalisée. Les coefficients mis en jeu dans l'interpolation linéaire ont été optimisés sur le corpus de développement d'ESTER 2.

Les 122000 mots les plus probables ont ensuite été extraits du modèle de langage interpolé. Ces mots constituent le vocabulaire du SRAP du LIUM.

#### **1.3.3 Les modèles acoustiques**

Les modèles acoustiques probabilistes généralement utilisés pour la reconnaissance de la parole sont basés sur l'utilisation de Modèles de Markov Cachés (MMC) [Rabiner 1986, Rabiner 1989, Charniak 1993]. Les MMC sont des machines à états finis probabilisées. À

---

<sup>2</sup>Un modèle de langage unigramme contient les probabilités d'apparition de chaque mot, indépendamment du contexte dans lequel il se trouve. Ces probabilités sont calculées à partir d'un corpus d'apprentissage.

<sup>3</sup>L'interpolation linéaire consiste à construire un modèle de langage dans lequel la probabilité d'apparition de chaque séquence de mots est calculée en réalisant une combinaison linéaire des probabilités de chacun des modèles interpolés.

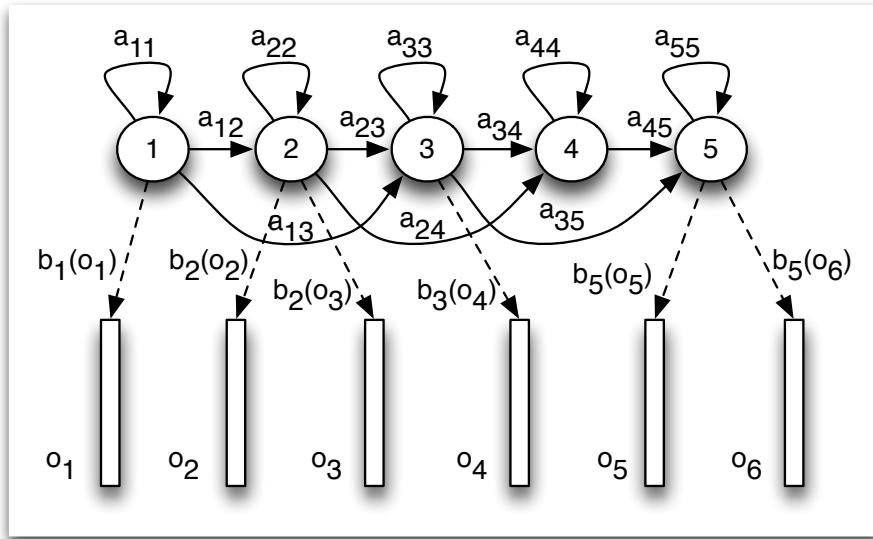


FIG. 1.2 – Représentation d'un MMC à 5 états

chaque état  $i$  sont associées deux types de probabilités :

- Une probabilité d'émission  $b_i(o_t)$  de l'observation  $o_t$  à l'état  $i$  (dans le cas d'un SRAP ces observations sont des vecteurs caractéristiques du signal de la parole, généralement des MFCC - Mel Frequency Cepstral Coefficients - ou des PLP - Perceptual Linear Prediction complétés de leurs dérivées premières et secondes). Il s'agit généralement de mélanges de densités de probabilités gaussiennes, définies par leurs vecteurs de moyennes, leurs matrices de variances diagonales et une pondération associée à chaque densité de probabilité.
- Des probabilités discrètes  $a_{ij}$  définissant les probabilités de passer de l'état  $i$  à l'état  $j$  (avec  $j \geq i$ ).

Généralement la topologie retenue est un MMC à 3 ou 5 états. La figure 1.2 schématise un MMC à 5 états, avec la topologie la plus courante (modèle gauche-droit avec la possibilité de sauter les états 2 et 4). L'unité de modélisation couramment utilisée est le phonème. Pour tenir compte des éventuels liens entre les phonèmes, un MMC est construit pour un phonème donné associé à un contexte gauche et à un contexte droit particulier. Le contexte gauche concerne le phonème qui précède le phonème à modéliser, le contexte droit celui qui lui succède. Ce triplet (gauche, phonème, droit) est appelé triphone, ou phonème en contexte. La position du phonème dans un mot (début, milieu, fin, isolé) est parfois également prise en compte pour affiner la modélisation. Un regroupement des états similaires est alors effectué de façon à réduire la taille du modèle, on parle alors d'états partagés.

**Apprentissage et adaptation des modèles acoustiques** L'apprentissage des modèles acoustiques consiste à estimer les paramètres des chaînes de Markov Cachés à partir d'un ensemble d'enregistrements transcrits. La topologie du modèle (le nombre d'états des modèles et les transitions autorisées entre ces états) est fixée a priori. Ainsi, connaissant une suite d'observations émises par une suite de modèles, il est possible de modifier les paramètres du modèle de manière à rendre plus probable l'émission des observations par le modèle. Il s'agit d'une estimation par le critère du maximum de vraisemblance (Maximum Likelihood Estimation, MLE), obtenu par l'algorithme de Baum-Welch [Baum 1970].

L'estimation par MLE consiste à choisir le jeu de paramètres  $\tilde{\lambda}_{MLE}$  de sorte à rendre maximale la probabilité d'émission des observations  $O$  par le modèle :

$$\tilde{\lambda}_{MLE} = \arg \max_{\lambda} P(O|\lambda) \quad (1.4)$$

Une résolution analytique directe n'est pas possible, mais l'algorithme de Baum-Welch permet une réestimation itérative des paramètres  $a_{ij}$  et  $b_j$  du modèle [Baum 1972, Liporace 1982]. A la suite de la réestimation  $(n + 1)$  des paramètres du modèle  $\lambda_n$ , le nouveau modèle  $\lambda_{n+1}$  vérifie :

$$P(O|\tilde{\lambda}_{n+1}) \geq P(O|\tilde{\lambda}_n) \quad (1.5)$$

La réestimation itérative est basée sur le principe de l'algorithme EM [Dempster 1977]. L'algorithme EM est une méthode permettant de trouver le maximum de vraisemblance des paramètres de modèles probabilistes lorsque le modèle dépend de variables latentes non observables. L'apprentissage étant itératif, les paramètres du modèle initial vont converger vers un jeu de paramètres optimal. Les choix concernant la définition des modèles vont donc avoir une influence sur la qualité de l'apprentissage.

Il existe des techniques d'apprentissage discriminantes tels que MMIE (*Maximal Mutual Information Estimation*) [Valtchev 1997], MPE (*Minimum Phone Error*) [Povey 2002] et MWE (*Minimum Word Error*) [Heigold 2005].

Le principe général du MMI (*Maximal Mutual Information*) est de trouver au sein de différentes classes, quels sont les paramètres qui permettent de singulariser chacune des classes. La méthode couramment utilisée pour déterminer si une composante peut s'associer à une classe est l'information mutuelle entre les valeurs de la composante et les valeurs contenues dans la classe. Cette méthode a ensuite été introduite par [Bahl 1986] afin d'adapter les paramètres de modèles de Markov pour les SRAP. MMIE a ensuite été développée pour les SRAP par [Valtchev 1997]. Cette méthode vise à maximiser les probabilités *a posteriori* des phrases

d'apprentissage. MWE et MPE fonctionnent sur un principe similaire à celui de MMI mais tendent à minimiser respectivement le taux d'erreurs phonèmes et le taux d'erreurs mots.

L'adaptation des modèles acoustiques permet de compenser les différences pouvant exister entre les données d'apprentissage des modèles avec les conditions d'utilisations de ces derniers. Ces différences peuvent être liées à l'environnement dans lequel les données ont été enregistrées ou bien à la façon de parler du locuteur. L'adaptation peut s'effectuer de façon supervisée ou non supervisée. L'adaptation supervisée consiste à utiliser des données transcris pour adapter les modèles acoustiques de base, alors qu'en mode non supervisée, l'adaptation sera réalisée directement sur les données à décoder.

Des modèles acoustiques indépendants du locuteur sont estimés à partir de grands ensembles de données. L'utilisation de la technique de régression linéaires MLLR (*Maximum Likelihood Linear Regression*) permet d'adapter ces modèles de façon à les rendre proches de modèles dépendants du locuteur [Leggetter 1995, Gales 1998]. Il existe deux types de transformations linéaires : le cas non contraint (MLLR) où les transformations sur les moyennes et les variances des Gaussiennes sont décorrélées les unes des autres [Leggetter 1995] et le cas contraint (CMLLR : *Constrained Maximum Likelihood Linear Regression*) [Digalakis 1995]. La technique CMLLR peut être intégrée au processus d'apprentissage (technique SAT – *Speaker Adaptive Training* [Anastasakos 1997]). La transformation SAT-CMLLR peut se formuler comme suit :

$$v' = Av - b \quad (1.6)$$

et

$$\sum' = A \sum A'^T \quad (1.7)$$

où  $v$  et  $v'$  sont les moyennes avant et après transformation,  $\sum$  et  $\sum'$  sont les variances,  $A$  est la matrice de régression et  $b$ , le facteur de décalage. Au moyen de l'algorithme EM, les paramètres  $A$  et  $b$  sont optimisés selon le maximum de vraisemblance sur les données d'adaptation.

La méthode d'estimation du Maximum *a posteriori* (MAP) appliquée en reconnaissance de la parole [Gauvain 1994] permet d'introduire des contraintes probabilistes sur les paramètres des modèles. Le critère MAP est utilisé en mode supervisé, il est appliqué aux modèles ayant fait l'objet d'un apprentissage préalable et pour lesquels on dispose de données *a priori*. Les modèles Markoviens sont toujours estimés avec l'algorithme Baum-Welch mais en maximisant la vraisemblance *a posteriori* au lieu de la vraisemblance des données. Cette technique permet d'obtenir des modèles adaptés à un locuteur particulier, ou à des conditions acoustiques particulières. Cette méthode est utilisée dans l'outil du LIUM pour adapter les modèles acoustiques au type de bande passante (large/étroite) ainsi qu'au genre des locuteurs (homme/femme).

Les modèles acoustiques du système du LIUM ont été appris sur les 80h d'ESTER 1 et sur une partie d'ESTER 2, complétés par 40h d'émissions radiophoniques provenant du projet EPAC<sup>4</sup>. Ce corpus est composé d'environ 191h de données bande large (BL) et de 40h de données bande étroite (BE). Les modèles BL ont été appris avec les 191h de données large bande puis adaptées avec la méthode MAP [Gauvain 1994] pour chacun des genres (homme/femme). Les modèles BE ont été appris sur l'ensemble des données (BL+BE = 231h), puis adaptés aux genres à l'aide de la méthode MAP. Après l'obtention de ces modèles (BL+BE pour homme/femme), une adaptation aux données de BE est réalisée, encore une fois à l'aide de MAP. Le SRAP du LIUM est composé de plusieurs passes de décodage (voir 1.3.6). Les modèles de la première passe sont composés de 6500 états partagés, chaque état étant modélisé par une mixture de 22 gaussiennes. Les modèles utilisés lors des passes deux et trois sont composés de 7500 états partagés, toujours modélisés par une mixture de 22 gaussiennes. Ces modèles ont été estimés grâce à un apprentissage de type SAT combiné à un apprentissage discriminant de type MPE (Minimum Phone Error) [Povey 2002]. Une matrice de transformation CMLLR a été calculée pour chaque locuteur et appliquée sur les paramètres acoustiques de chacun des locuteurs respectifs. La méthode CMLLR pour SAT en seconde passe génère une matrice pleine 39x39 pour chaque locuteur.

### 1.3.4 Les modèles de langage

Le modèle de langage introduit les contraintes linguistiques dans le SRAP. Il modélise les contraintes liées à la langue, afin de guider le décodeur dans sa sélection des hypothèses acoustiques concurrentielles. La probabilité d'apparition de la séquence de mots  $W$  s'exprime de la façon suivante :

$$P(W_1^k) = P(w_1) \prod_{i=2}^k P(w_i|h_i) \quad (1.8)$$

Où  $h_i$  correspond à l'historique du mot  $w_i$ . On a donc  $h_i = w_1, \dots, w_{i-1}$ . Les modèles utilisés en reconnaissance automatique de la parole sont des modèles de langage de type  $n$ -gramme [Jelinek 1976]. Bien qu'ancien, ce type de modèle constitue toujours l'état de l'art.

Les modèles de ce type correspondent à une modélisation stochastique du langage, où l'historique d'un mot est représenté par les  $n - 1$  mots qui le précède :

$$P(W_1^k) = P(w_1) \prod_{i=2}^{n-1} P(w_i|w_1, \dots, w_{i-1}) \prod_{i=n}^k P(w_i|w_{i-n+1}, \dots, w_{i-1}) \quad (1.9)$$

---

<sup>4</sup>EPAC est un projet ANR : Exploration de masse de documents audio pour l'extraction et le traitement de la parole conversationnelle

Les modèles couramment utilisés dans les SRAP sont généralement des modèles d'ordre 3 ou 4 ( $n = 3$  ou  $n = 4$ ). Nous parlons de modèle *trigramme* ou *quadrigramme* (pour  $n = 1$  *unigramme*, pour  $n = 2$  *bigramme*, ...). Dans le cas d'un modèle quadrigramme, l'équation 1.9 s'écrit :

$$P(W_1^k) = P(w_1)P(w_2|w_1)P(w_3|w_1, w_2)\prod_{i=4}^k P(w_i|w_{i-3}w_{i-2}w_{i-1}) \quad (1.10)$$

Bien que ce type de modèle puisse paraître assez simpliste, en ne prenant en compte qu'un historique court, il possède une souplesse en permettant de modéliser des phrases grammaticalement incorrectes. Une qualité fondamentale des modèles  $n - grammes$  est de permettre une couverture totale des phrases pouvant être exprimées dans le langage en attribuant une probabilité aux mots inconnus. Cela permet au système d'être capable de probabiliser des phrases qu'il n'aurait jamais observé dans son corpus d'apprentissage, tout en privilégiant les séquences de mots les plus fréquemment observées. Un lissage des probabilités permet d'attribuer des probabilités non nulles à des séquences jamais observées. Les modèles reposant sur des bases de règles de grammaire formelles seraient facilement mis en défaut, par exemple lors du décodage de parole spontanée, pas toujours correctement grammaticalement formulé.

**Apprentissage des modèles de langage** L'apprentissage des paramètres d'un modèle de langage  $n - gramme$  s'effectue en deux étapes : la première étape consiste à calculer les probabilités d'apparitions des différents  $n - grammes$  vus dans le corpus d'apprentissage, et la seconde étape permet de lisser ces probabilités de façon à pouvoir attribuer des probabilités non nulles à des événements jamais observés. Le principe consiste en l'extraction des probabilités à partir du décompte des suites de mots observés, puis en la redistribution d'une partie de ces probabilités aux événements non observés. La méthode la plus commune pour le calcul des probabilités d'apparitions des  $n - grammes$  est l'estimation par maximum de vraisemblance [Federico 1998] ; la distribution des probabilités du modèle de langage obtenue est celle qui maximise la vraisemblance du corpus d'apprentissage. La probabilité d'apparition d'un  $n - gramme$  est calculé de la sorte :

$$P_{MV}(w_i|h_i) = \frac{n(h_i, w_i)}{n(h_i)} \quad (1.11)$$

Où  $n(x)$  indique la fréquence d'apparition de  $x$ .

Il existe deux grands types de lissage [Chen 1998a, Kneser 1995] : le lissage par interpolation linéaire et le repli. La probabilité d'apparition d'un  $n - gramme$ , dans le cas où l'on

### 1.3. Vue générale d'un système de reconnaissance automatique de la parole

---

utiliserait le lissage par interpolation linéaire, est la combinaison linéaire des probabilités d'apparition d'ordre 1 à  $n$  de ce  $n$ -gramme. La technique de lissage par repli consiste à utiliser, pour les  $n$ -grammes qui n'auraient jamais été observés dans le corpus d'apprentissage, les probabilités issues d'un ordre inférieur ( $n - 1$ , si  $n - 1$  jamais vu :  $n - 2, \dots$ ). La valeur finale de la probabilité proposée par le modèle sera affectée d'un coefficient de repli.

Les données utilisées pour construire les modèles de langage utilisés dans le SRAP du LIUM sont de trois sortes :

- Les transcriptions manuelles d'émissions radiophoniques. Elles correspondent aux données utilisées pour apprendre les modèles acoustiques. Les transcriptions manuelles provenant de conversations issues du corpus PFC [Durand 2002] ont également été utilisées.
- Des articles de journaux : en plus des 19 années d'articles du journal "Le Monde", les articles du journal français "L'Humanité" de 1990 à 2007 et le corpus français Giga Word ont été utilisés.
- Des ressources provenant des sites internet de "L'Internaute", "Libération", "Rue89" et "Afrik.com".

Les modèles ont été appris à l'aide du toolkit SRILM [Stolcke 2002], et en utilisant la technique de lissage dite de Kneser-Ney modifié [Chen 1998a, Kneser 1995] avec interpolation des  $n$ -grams d'ordres inférieurs. Le modèle de langage du LIUM utilisé pour le décodage à grand vocabulaire, comprend 121K 1-grams, 29M de 2-grams, 162M de 3-gram et 376M de 4-grams.

**Adaptation des modèles de langage** Le besoin d'adapter un modèle de langage intervient lorsque le modèle se révèle être en inadéquation avec la tâche, et que les données permettant de construire un modèle pour cette tâche ne sont pas disponibles en quantité suffisante. Cette inadéquation peut être due au fait que les données sur lesquelles a été appris le système ne contiennent pas ou peu de données permettant de modéliser les données de test. Les modèles adaptatifs (ou *Adaptive models*) sont des modèles capables de se spécialiser dynamiquement au cours du processus de reconnaissance automatique de la parole en fonction de ce qui a déjà été reconnu. Dans le cas d'une application grâce à laquelle nous disposerions des corrections des utilisateurs au fur et à mesure du décodage, il serait possible d'utiliser ces données pour fabriquer ce genre de modèle.

Il existe dans la littérature, trois types de modèles adaptatifs.

Le modèle *cache* [Kuhn 1990] part du principe que si un mot vient d'être reconnu, alors sa probabilité d'apparition se voit accrue. Les mots les plus récemment reconnus sont stockés dans

un *cache*. Dans le cas d'une adaptation par interpolation linéaire, la probabilité d'apparition du mot  $w_i$  s'écrit :

$$P(w_i|h_i) = (1 - \lambda)P_{ref}(w_i| h_i) + \lambda P_{cache}(w_i) \quad (1.12)$$

La probabilité  $P_{cache}$  peut être calculée de différentes manières. La probabilité  $P_{cache}(w_i)$  peut par exemple être calculée en additionnant les positions des occurrences du mot  $w_i$  dans  $h_i$  divisé par l'ensemble des positions des mots de  $h_i$ . [Clarkson 1997] propose d'attribuer une probabilité exponentiellement proportionnelle à la distance entre la position actuelle et les apparitions précédentes du mot.

L'utilisation de modèle cache modifie uniquement les probabilités des mots observés récemment. Or, une séquence ou un mot qui vient d'être reconnu, modifie la probabilité d'apparition d'autres séquences ou d'autres mots qui lui sont généralement associés [Rosenfeld 1992]. Ceci revient à modéliser une dépendance entre les mots sur un plus long historique que seulement les  $n - 1$  mots précédents, en faisant l'hypothèse que si une séquence de mots A est fortement corrélée à une séquence de mots B, alors l'apparition de la séquence A modifie la probabilité d'apparition de la séquence B. Le modèle *trigger* est un modèle permettant de prendre en compte ce phénomène. La première étape de sa construction consiste en la sélection des paires de déclenchements (*trigger pairs*). Le critère de détermination des paires qui sont corrélées utilise soit les fréquences de cooccurrences soit l'information mutuelle entre les deux séquences. La prise en compte de ces probabilités se fait soit par interpolation linéaire avec le modèle de référence (comme pour les modèles *cache*), soit en utilisant le maximum d'entropie [Rosenfeld 1996].

Une autre approche consiste à utiliser des mélanges de modèles thématiques. Ces modèles sont appris sur des sous parties du corpus d'apprentissage correspondant à des thèmes ou des sous langages différents. Ces modèles sont ensuite interpolés. Les poids d'interpolation sont généralement calculés en utilisant l'algorithme EM afin de trouver les poids minimisant la perplexité  $PP$  définie par l'équation suivante :

$$PP = 2^{-\frac{1}{n} \sum_{i=1}^n \log_2 P(w_i|h)} \quad (1.13)$$

Où  $n$  représente le nombre de mots du corpus, et  $P(w_i|h)$  la probabilité du mot  $i$  sachant l'historique  $h$ .

Ces stratégies d'adaptation linguistique ne sont pas intégrées dans le système de décodage, mais elles sont utilisées dans la stratégie d'auto-réordonnement présentée dans la partie suivante de ce manuscrit.

### 1.3.5 Segmentation en locuteur

Le LIUM a développé son propre outil de segmentation [Meignier 2010]. Cet outil découpe le signal en terme de locuteur, genre et largeur de bande. Cette étape est cruciale dans un système de reconnaissance automatique de la parole, puisqu'elle permet d'utiliser les paramètres acoustiques adaptés au moment du décodage. L'outil du LIUM repose sur une segmentation acoustique de type GLR (Generalized Likelihood Ratio) [Willsky 1976, Siu 1992] suivi d'un regroupement ascendant de type BIC (le Critère d'Information Bayésien) [Chen 1998b, Gish 1991]. Chaque cluster est modélisé par une gaussienne de covariance pleine. Les deux clusters les plus proches sont regroupés à chaque itération jusqu'à convergence sur le critère d'arrêt. La métrique BIC est utilisée à la fois comme critère d'arrêt et pour sélectionner les clusters à regrouper. Un décodage avec l'algorithme de Viterbi [Viterbi 1967] est ensuite effectué, de façon à ajuster les frontières des segments. Les régions correspondant à des zones de non-parole (musique, bruit, et silence) sont éliminées en utilisant un nouveau décodage Viterbi. Ce décodage utilise 8 GMMs (mixture de Gaussiennes), contenant 64 Gaussiennes diagonales entraînées avec l'algorithme EM appris sur les données d'ESTER 1. Pour terminer, une détection du genre des locuteurs et du type de bande est réalisée, utilisant 4 GMMs avec 64 Gaussiennes diagonales dépendantes du genre et de la bande passante. La segmentation, le regroupement et le décodage utilisent des paramètres acoustiques modélisés par 13 MFCC incluant le descripteur de l'énergie tandis que la détection des zones de paroles, du type de bande et du genre du locuteur est effectuée avec 12 MFCC complétés par 12 dérivés premières.

Le système pour la tâche de suivi et regroupement en locuteur (SRL) utilisé lors de la campagne ESTER 2 correspond au système décrit ci-dessus auquel a été ajouté une dernière classification reposant sur la métrique CE/NCLR (Cross Entropy [Solomonoff 1998] / Normalized Cross Likelihood Ratio [Reynolds 1998, Le 2007]). Cette dernière étape n'est pas nécessaire en transcription, car elle a tendance à regrouper les segments d'un même locuteur sans tenir compte des environnements sonores présents dans les segments (studio, téléphone, fond calme, fond avec la musique du bruit...). L'outil, développé au LIUM par Sylvain Meignier, a terminé premier lors de la campagne ESTER 2 en obtenant un DER (Diarization Error Rate) de 10,8 %.

### 1.3.6 Décodage

Le LIUM a choisi d'utiliser le système Sphinx (CMU) comme base pour son SRAP. Il existe quatre familles de décodeurs Sphinx, disponibles sous licence de type BSD<sup>5</sup> :

---

<sup>5</sup>Ce type de licence permet une utilisation commerciale des décodeurs et n'est pas contaminant, car il n'impose pas une licence particulière aux logiciels dérivés

Sphinx 2 [Huang 1992], Sphinx 3 [Ravishankar 2000], Sphinx 4 [Walker 2004] et Pocket Sphinx [Huggins-daines 2006]. Le LIUM utilise les versions Sphinx 3 et Sphinx 4 du décodeur. Sphinx 3 utilise des modèles de Markov continu. Deux sous-branches majeures du décodeur Sphinx 3 ont longtemps coexisté : un décodeur lent (*flat*) et un décodeur rapide (*lextree*). La différence majeure entre les deux provient de la gestion acoustique inter-mot de l'algorithme de recherche. Dans le décodeur lent, de vrais phonèmes en contexte (triphone + position du phonème dans le mot) sont utilisés en fin de mot, alors que dans la version rapide une approximation de la modélisation du phonème en fin de mot est effectuée qui accélère le traitement, mais dégrade les performances. La version *flat* est dix fois plus lente que la version *lextree* pour une précision de reconnaissance sensiblement meilleure. Le décodeur Sphinx 3 est toujours en développement : les décodeurs *flat* et *lextree* ont par exemple récemment été unifiés au sein d'un seul et même outil, pendant que de nouveaux ajouts permettent d'améliorer encore les performances et la vitesse d'exécution, comme présenté par exemple dans [Chan 2005].

Une description précise des quatre familles de décodeurs Sphinx est disponible dans [Estève 2009].

La stratégie de décodage du système du LIUM comprend cinq passes et utilise Sphinx 3.7 (en mode rapide) et un décodeur graphe développé par le LIUM (passe 3), basé sur Sphinx 4. Elle s'articule comme suit :

Passe 1 La première passe utilise les modèles acoustiques (adaptés à l'aide de MAP) correspondant au genre et à la largeur de bande détectée lors du processus de segmentation.

Passe 2 La meilleure hypothèse de reconnaissance générée par la première passe de décodage est utilisée pour calculer une transformation CMLLR pour chacun des locuteurs. Le décodage généré par cette passe, utilisant les modèles acoustiques SAT combinés à MPE, et utilisant les transformations CMLLR calculées, permet de générer un graphe de mots.

Passe 3 Cette passe a pour but de réestimer le graphe de mots généré à l'issu de la seconde passe. Ce graphe ainsi obtenu contient des approximations quant aux scores acoustiques des phonèmes situés en fin de mots : ces scores n'ont pas été calculés par rapport à leur véritable contexte droit, mais à l'aide d'une approximation. Le fait d'utiliser une approximation plutôt que le véritable contexte droit permet de diminuer les temps de calcul, mais dégrade le taux d'erreur mot final. En réestimant le graphe de mots, il va être possible de corriger ces imprécisions inter mots en utilisant le vrai contexte droit des phonèmes en fin de mots présents dans le graphe. À l'issu de cette passe, un nouveau graphe de mots est généré.

Passe 4 La quatrième passe consiste à recalculer, à l'aide du modèle de langage quadrigramme, les scores linguistiques des mots du graphe issu de la passe 3.

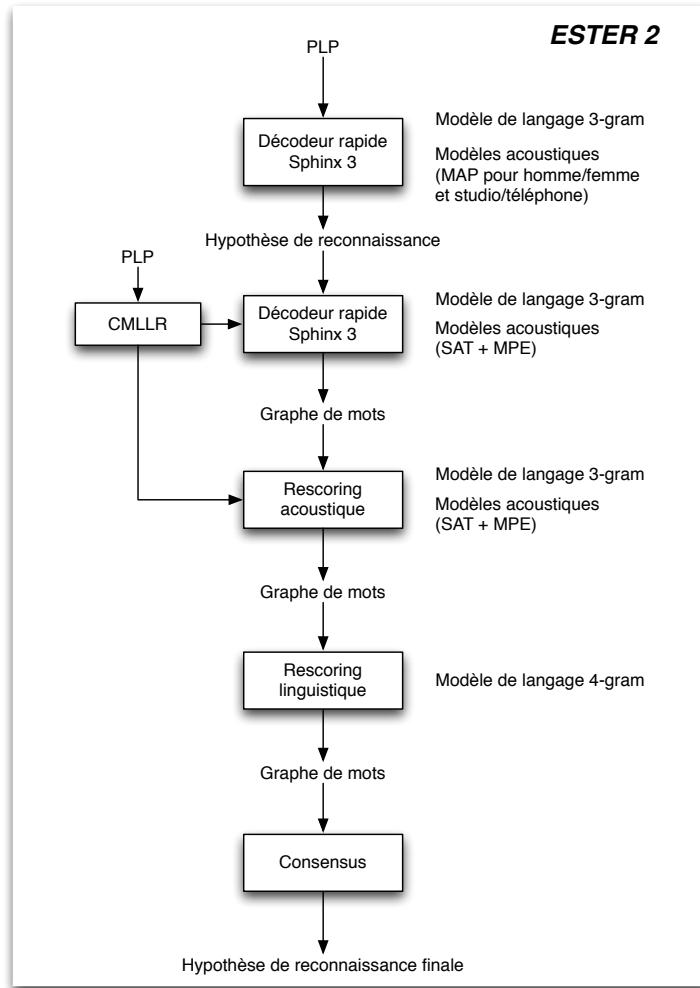


FIG. 1.3 – Decodage ESTER 2

Passe 5 La dernière passe du décodeur consiste à générer un réseau de confusion à partir du graphe de mots. La meilleure hypothèse de reconnaissance est alors extraite du réseau de confusion à l'aide de la méthode de consensus [Mangu 2000].

### 1.3.7 Métrique

Afin d'évaluer les systèmes de reconnaissance automatique de la parole, la métrique couramment utilisée est celle du taux d'erreur mot (WER - Word Error Rate). La meilleure hypothèse de transcription du système est alignée avec la référence, nous obtenons alors trois types d'erreurs. Lorsqu'un mot apparaît dans l'hypothèse de reconnaissance alors qu'il n'y a aucun mot correspondant dans la référence, il s'agit d'une insertion. Si, au contraire, il y avait un mot dans la référence et que ce mot n'apparaît pas dans l'hypothèse, nous parlons de

suppression. Quand un mot de la référence est remplacé par un autre mot lors du décodage, c'est une substitution. Le taux d'erreur est ensuite déterminé par la formule suivante :

$$\text{Taux d'erreur} = \frac{\text{nombre d'insertions} + \text{nombre de suppressions} + \text{nombre de substitutions}}{\text{nombre de mots dans la référence}} \quad (1.14)$$

### 1.3.8 Performances

Sur le corpus de test de la campagne d'évaluation ESTER 2, le système du LIUM a obtenu un WER de 17,8% [Galliano 2009]. Le décodeur du laboratoire à évolué entre la campagne ESTER 1 et ESTER 2.

Afin d'évaluer les apports de la nouvelle version du décodeur (celle présentée dans ce manuscrit et développée pour la campagne ESTER 2), le test ESTER 2 a été décodé avec les systèmes ESTER 1 & 2 en utilisant le même vocabulaire, les mêmes modèles de langages et acoustiques. Le tableau 1.1 montre les résultats, en terme de WER, des deux systèmes sur le décodage du corpus de test ESTER 2.

TAB. 1.1 – *Taux d'erreur mot sur le corpus de test ESTER 2*

SRAP	WER
Ester 1 (avec modèles Ester 1)	29,4%
Ester 2 (avec modèles Ester 1)	24,1%
Ester 2 (avec modèles Ester 2)	17,8%

Le WER chute de plus de 10% en absolu avec la nouvelle version du système. L'article [Deléglise 2009] présente une comparaison détaillée des systèmes développés pour ESTER 1 et pour ESTER 2.

### 1.3.9 Correction des transcriptions automatiques

Le travail de transcription manuelle, aidé par un logiciel avec une interface adaptée, consiste à écouter l'enregistrement sonore, à effectuer une segmentation et à transcrire ce que l'on entend (en respectant certaines normes). Les informations sur le type d'enregistrement (studio, téléphone), le nom des locuteurs, certains phénomènes acoustiques (toux, rire, musique, etc) et toutes les méta-information peuvent être annotés si le logiciel le permet.

Nous présentons en annexe A, 4 outils de transcription manuelle. Ces outils, Transcriber [Barras 1998], Praat [Boersma 2001], Winpitch [Martin 1996] et XTrans, sont principalement utilisés dans des laboratoires de recherche et par les constructeurs de corpus.

### *1.3. Vue générale d'un système de reconnaissance automatique de la parole*

---

Le LIUM a voulu quantifier le gain de temps apporté par l'utilisation d'un système de reconnaissance vocale pour la tâche de transcription de parole. L'idée consiste à utiliser un SRAP pour obtenir une transcription automatique d'un enregistrement audio qui sera injectée dans un outil d'aide à la transcription. Les travaux réalisés consistent à comparer le temps nécessaire à la rédaction de la transcription de façon totalement manuelle avec le temps nécessaire en partant des sorties du SRAP. Thierry Bazillon, doctorant linguiste au LIUM, a réalisé des transcriptions manuelles et des transcriptions semi-automatiques en partant des sorties du système de reconnaissance automatique de la parole du LIUM [Bazillon 2008b, Bazillon 2008a]. Pour ce faire, le logiciel Transcriber (A.1) a été utilisé. 24 segments d'environ 10 minutes chacun ont été extraits des données non transcrisées fournies lors de la campagne ESTER. La moitié de ces segments ont été identifiés comme étant des segments contenant de la parole spontanée (débats et interviews) et l'autre moitié comme contenant des segments de parole préparée (informations). Chacun de ces fichiers a été transcrit manuellement et de façon assistée par le même transcripteur. La deuxième transcription a été effectuée suffisamment longtemps après la première, de façon à ce qu'elle ne soit pas influencée par la mémoire de la première. La tâche consiste à segmenter le signal, à assigner les noms des locuteurs et à transcrire (ou corriger s'il s'agit d'une transcription automatique) les dires de chacun. La segmentation de la transcription peut être réalisée de différentes façons. Dans le cadre de ces expériences, un nouveau segment était créé à chaque respiration du locuteur ou à chaque pause significative. Les tâches de segmentation et de transcription étaient réalisées en parallèle, alors que certains préconisent de commencer par segmenter avant de transcrire.

Un chronométrage de chacune de ces étapes a été effectué.

TAB. 1.2 – *Durée totale de la transcription (durées respectives des corpus : 2h08 et 2h10)*

	Parole préparée	Parole spontanée
Transcription manuelle	17h36	19h33
Transcription assistée	8h31	15h44

Le tableau 1.2 montre que la transcription assistée induit un gain de temps, surtout pour la parole préparée. Sur ce type de parole, le fait d'assister le transcripteur dans sa tâche permet de diviser le temps de traitement par deux. Lorsqu'il s'agit de parole spontanée, ce bénéfice est bien moindre. Cela peut s'expliquer par le fait que le SRAP du LIUM est moins performant sur de la parole spontanée que sur de la parole préparée. Sur les segments de parole préparée, le taux d'erreur mot du SRAP est d'environ 17%, alors qu'il s'élève à 35% sur de la parole spontanée. L'article [Bazillon 2008b] présente de plus amples détails sur les résultats, et la manière dont les segments ont été annotés en parole préparée / parole spontanée.

## 1.4 Conclusion

Dans ce chapitre, nous avons présenté le fonctionnement général d'un système de reconnaissance automatique de la parole probabiliste, puis nous avons présenté le système développé et utilisé au LIUM.

TAB. 1.3 – *Résultats des différents participants à la campagne ESTER 2*

Systèmes	IRISA Rennes	LIA Avignon	LIMSI Orsay	LIUM Le Mans	LORIA Nancy	Vecsys Research Orsay
WER	26,1	26,8	12,1	17,8	26,3	15,1

Sur les évaluations ESTER (1&2), le système ayant obtenu les meilleurs résultats est celui du LIMSI (Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur - Orsay), avec un WER de 11,9% lors de la campagne ESTER 1 et un WER de 12,1% lors de la campagne ESTER 2.

Lors des deux campagnes d'évaluation des systèmes de reconnaissance automatique de la parole, le système du LIUM a été le meilleur système open source, avec un WER de 17,8% lors de la campagne ESTER 2 (voir le tableau 1.3 extrait de [Galliano 2009]) et un WER de 23,6 % lors d'ESTER 1. Les améliorations apportées entre les deux versions du système sont significatives, ce qui montre l'investissement de l'équipe parole du LIUM pour proposer un système à l'état de l'art. Les travaux en cours, concernant l'amélioration du SRAP, consistent à coupler le système du LIUM avec ceux du LIA et de l'IRISA, dans le cadre du projet ANR ASH (Attelage de Systèmes Hétérogènes).

Dans cette partie, les expériences de Thierry Bazillon, montrant l'intérêt d'utiliser un SRAP pour réaliser des transcriptions, ont également été présentées. Dans le cadre des expériences menées au LIUM, l'utilisation du SRAP a permis de diminuer de moitié le temps nécessaire à l'opérateur pour la tâche de transcription de parole préparée, et de 19% pour la transcription de parole spontanée.

## **Deuxième partie**

# **Assistance automatique à la transcription manuelle**



**D**ans la partie précédente de ce manuscrit, l'apport de l'utilisation du SRAP du LIUM pour la transcription de la parole a été montré par le biais de la présentation des expériences réalisées par Thierry Bazillon. Le transcriveur utilise le SRAP pour obtenir une transcription automatique d'un flux de parole. À l'issue du processus de décodage, l'utilisateur importe le résultat dans une application, corrige toutes les erreurs du système et ajoute éventuellement des informations (le nom des locuteurs, le type de bande son, etc.).

La méthode proposée ici a pour but d'aider de façon dynamique l'utilisateur dans cette phase de correction (méthode d'adaptation synchrone). Elle est basée sur une réestimation automatique de l'hypothèse du SRAP, à chaque correction apportée par l'utilisateur. Cette réestimation va générer une nouvelle hypothèse qui sera validée ou corrigée par l'utilisateur.

Dans le contexte du présent travail, cette stratégie a été étudiée afin d'être intégrée dans l'application visée par la société Spécinov, de façon à permettre aux utilisateurs de la solution de parvenir à une transcription correcte en un minimum de temps. La génération d'une nouvelle hypothèse doit donc être la plus rapide possible pour que la méthode puisse être intégrée dans une application interactive.

Cette partie est divisée en 4 chapitres. Tout d'abord, un état de l'art sur les méthodes d'assistance à la transcription est présenté. Après avoir décrit le fonctionnement de la stratégie proposée et exposé les résultats obtenus, nous conclurons sur l'intérêt de la méthode et les aspects qui peuvent être améliorés.



# **Chapitre 2**

## **État de l'art**

### **Sommaire**

---

<b>2.1</b>	<b>Introduction</b>	<b>34</b>
<b>2.2</b>	<b>Stratégies d'affichage pour l'assistance à la correction de transcriptions</b>	<b>34</b>
<b>2.3</b>	<b>Traduction assistée par ordinateur</b>	<b>37</b>
<b>2.4</b>	<b>Réordonnancement des hypothèses</b>	<b>37</b>
<b>2.5</b>	<b>Conclusion</b>	<b>39</b>

## 2.1 Introduction

Nous présentons ici un état de l'art des méthodes d'assistance à la transcription de la parole (*Computer Assisted Transcription of Speech* – CATS). Cet état de l'art est divisé en trois sous-parties. La première présente des projets expérimentaux mettant en œuvre des aides dans des applications d'assistance à la correction de transcriptions. La seconde partie expose des travaux réalisés sur la tâche, relativement proche d'un point de vue applicatif, d'assistance à la traduction (*Computer Assisted Translation* – CAT). La dernière partie de cet état de l'art décrit une méthode de réévaluation des hypothèses du système de reconnaissance automatique de la parole (SRAP) en utilisant les corrections apportées par l'utilisateur.

## 2.2 Stratégies d'affichage pour l'assistance à la correction de transcriptions

Des applications pour la transcription manuelle et/ou assistée sont présentées en annexe A. En mode transcription assistée, ces applications permettent d'aider l'utilisateur à corriger les sorties du système de reconnaissance automatique de la parole, en proposant un affichage de la transcription associé à sa bande son. Les temps de début et de fin des segments sont utilisés pour permettre au transcripteur d'écouter facilement la portion du signal correspondant aux mots reconnus.

Un certain nombre de travaux introduisent des stratégies permettant d'interagir de façon plus prononcée avec l'application [Ainsworth 1992, McNair 1994, Suhm 1996, Suhm 1997]. L'utilisateur peut répéter les zones qui auraient été particulièrement mal transcris, épeler les mots incorrects et remplacer les mots de la meilleure hypothèse de reconnaissance par d'autres mots présents dans une liste déroulante.

Les articles de [Luz 2008] et [Wald 2006] évaluent l'impact qu'ont certaines aides sur les utilisateurs et comment ceux-ci les perçoivent. [Luz 2008] montre que les utilisateurs préfèrent utiliser une application contenant des aides du type « sélectionner différents mots dans une liste déroulante » plutôt que de n'avoir que le fichier son et la transcription à leur disposition. [Wald 2006] a exploré comment les différentes modalités, comme l'utilisation du clavier, de la souris, de raccourcis claviers, de combinaisons de touches peuvent réduire les temps de correction.

Les travaux [Nanjo 2006, Ogata 2005, Ogata 2007, Goto 2007, Cardinal 2007] mettent en œuvre l'utilisation des réseaux de confusion dans l'interface utilisateur. Le système [Nanjo 2006] intègre la possibilité de sélectionner les mots via l'interface présentant les hypothèses du réseau de confusion (voir figure 2.1). Outre l'utilisation du clavier pour saisir les

## 2.2. Stratégies d'affichage pour l'assistance à la correction de transcriptions

mots absents du graphe, il permet de re-prononcer les zones erronées. Sur le corpus utilisé dans ce projet, le meilleur taux d'erreur qui pourrait être obtenu en choisissant toujours le meilleur chemin dans le réseau de confusion (taux oracle) se situe aux alentours de 6%.

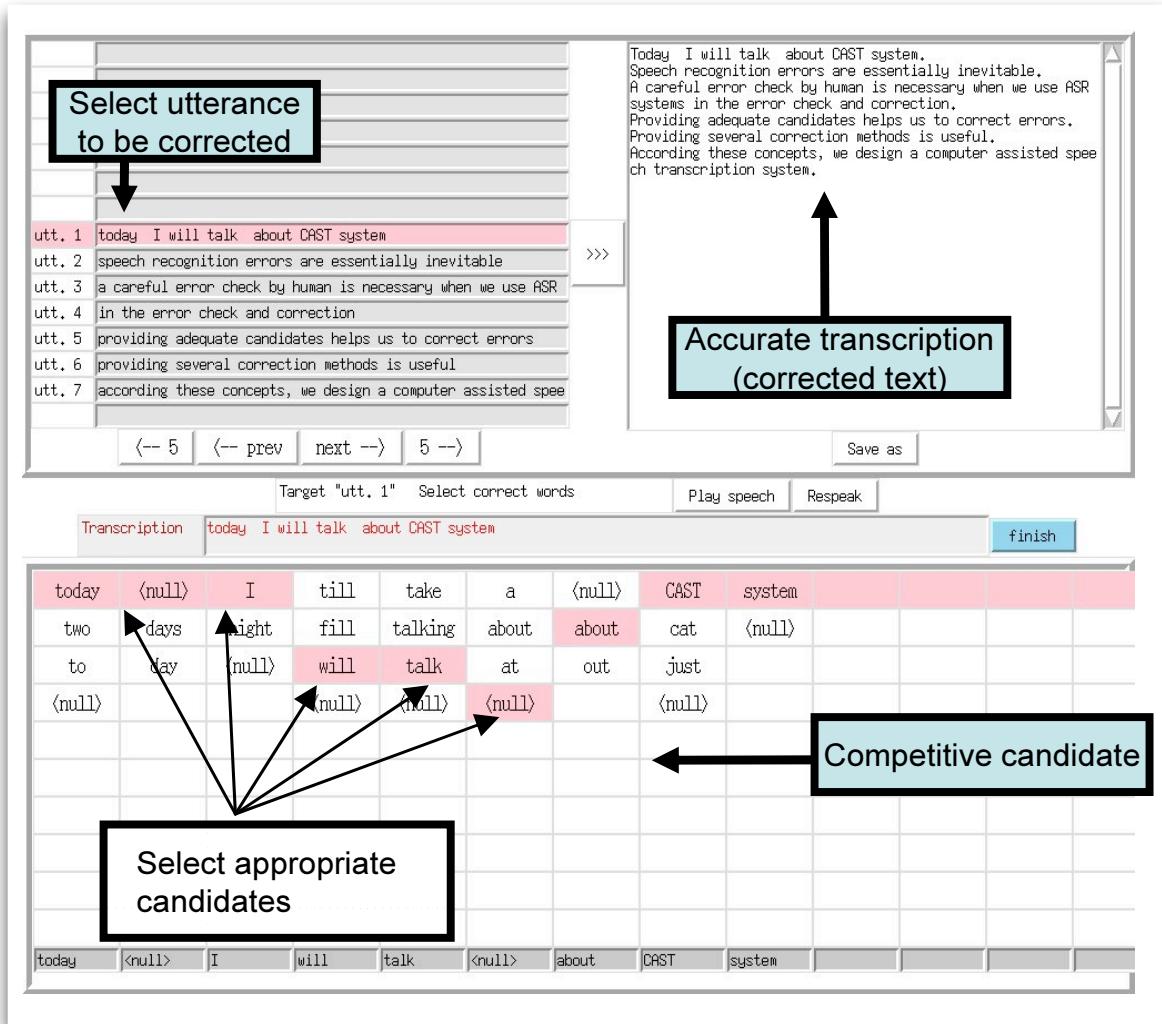


FIG. 2.1 – Extrait de l'article [Nanjo 2006]

Ces projets ont été testés par des utilisateurs, puis évalués en terme de gain de temps. L'utilisation des réseaux de confusion permet de diminuer significativement, dans tous les travaux cités précédemment, les temps de traitement (31% dans l'article [Ogata 2005]).

Des travaux plus spécifiques, sur l'aide à la rédaction de sous-titres d'émissions télévisées, sont également présents dans la littérature. Les travaux de [Imai 2002] et [Bateman 2000] suggèrent de procéder de la façon dite du « perroquet ». Un ou plusieurs opérateurs répètent ce qui est dit durant l'émission. Le fichier son contenant l'enregistrement de l'opérateur est ensuite

## Chapitre 2. État de l'art

fourni à un SRAP, puis la correction de la transcription est réalisée manuellement, ou en utilisant une application d'aide.

[Bateman 2000] et [Cardinal 2007] introduisent l'idée d'utiliser un délai contrôlable pour la phase de correction. Dans l'application de [Cardinal 2007], cela se traduit de la façon suivante :

- Les mots apparaissent un à un, du coin supérieur gauche de l'écran jusqu'au coin inférieur droit.
- Après un certain délai (modifiable), le mot suivant apparaît à droite du mot en cours, ou sur la ligne suivante si le bout de la ligne a été atteint. Cela permet, d'après les auteurs de [Bateman 2000] et [Cardinal 2007] à l'utilisateur de se concentrer sur un seul mot à la fois.
- Une fois le coin inférieur droit atteint par le dernier mot, l'apparition des mots redémarre du coin supérieur gauche.

La figure 2.2 montre l'interface graphique de cette application. Les mots en bleu (de « son » à « gauche ») sont toujours modifiables, alors que ceux en rouge ne peuvent plus, ou pas encore, être édités.

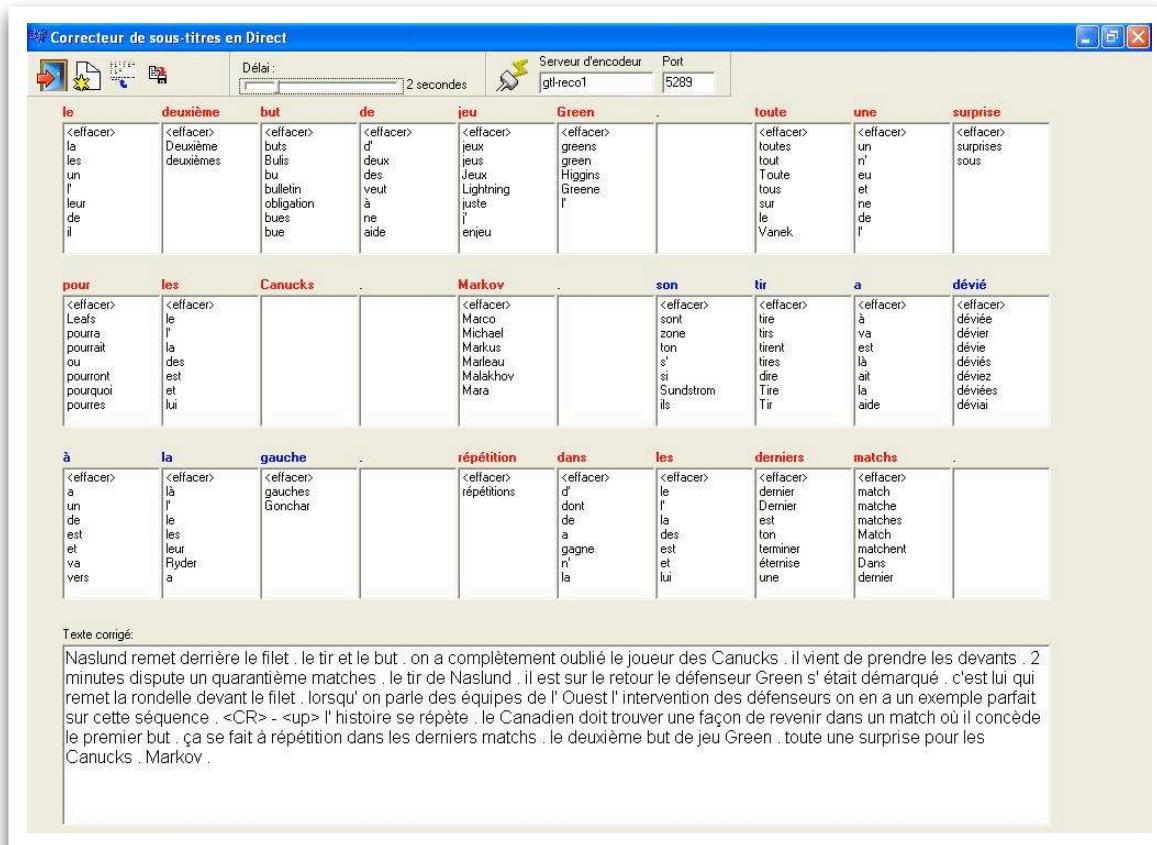


FIG. 2.2 – Extrait de l'article [Cardinal 2007]

L’application [Cardinal 2007], tout comme celles de [Nanjo 2006] et [Ogata 2007], permet de remplacer un mot par un autre qui serait dans la liste des mots alternatifs. Cependant, quelques fois, un mot long peut avoir été remplacé par plusieurs plus petits mots (ou inversement). Les auteurs de [Cardinal 2007] proposent une méthode originale pour prendre en compte ce phénomène. Quand un mot est supprimé par l’utilisateur, les phonèmes qui le composaient sont concaténés à ceux du mot suivant. La séquence de phonèmes résultante est utilisée pour rechercher dans le dictionnaire phonétisé du système les mots qui pourraient correspondre. Ces mots sont ensuite ajoutés automatiquement à la liste de choix du mot suivant.

## 2.3 Traduction assistée par ordinateur

Les méthodes d’assistance à la transcription sont peu nombreuses dans la littérature. D’autre part, de nombreux travaux [Civera 2005, Tomás 2006, Civera 2004, Foster 2002, Foster 1997] ont été réalisés sur la traduction assistée par ordinateur. Le système propose une traduction, l’utilisateur lit cette traduction et corrige le premier mot erroné qu’il rencontre. Le système propose alors une traduction alternative en prenant en compte cette correction. Les auteurs de [Civera 2005] proposent de construire un graphe de mots dans lequel les différentes possibilités de traductions les plus probables sont conservées. À chaque correction de la part de l’utilisateur, ce graphe est réévalué, et une nouvelle traduction est proposée au correcteur. En utilisant cette technique, les auteurs de [Civera 2005] estiment que la charge de travail restant au transcriveur est divisée par quatre. Les auteurs de [Tomás 2006] proposent un décodage interactif de recherche dans les différentes traductions, en fonction des corrections apportées par l’utilisateur. Les travaux de [Foster 2002] présentent, entre autres, des stratégies d’auto-complétion permettant d’accélérer le procédé de correction des traductions en suggérant des mots alternatifs lorsque l’utilisateur commence à remplacer un mot par un autre. Certains travaux [Amengual 2000, Ney 2000, Casacuberta 2004a, Casacuberta 2004b, Vidal 2006] décrivent des méthodes de traduction assistées utilisant en entrée, non pas du texte, mais du langage parlé. L’idée générale consiste à utiliser un modèle de langage combinant un modèle de langage *n-gram* avec les probabilités de traduction de chaque mot.

## 2.4 Réordonnancement des hypothèses

Dans [Rodríguez 2007], les auteurs proposent de relancer le processus de décodage après chaque correction de l’utilisateur.

Le procédé décrit par [Rodríguez 2007] s’articule de la façon suivante :

1. Le processus débute lorsque le SRAP propose une hypothèse de transcription pour le signal d'entrée.
2. Le correcteur humain lit la transcription jusqu'à ce qu'il trouve une erreur. Les mots précédant la zone en erreur sont considérés corrects.
3. L'utilisateur corrige (ou saisit) un ou plusieurs mots.
4. À cette étape, le SRAP prend en compte ce nouveau préfixe correct, composé des mots précédant la zone en erreur et du(des) mot(s) entré(s) par l'utilisateur. Il propose alors une nouvelle hypothèse de reconnaissance.
5. Le processus recommence à l'étape 2, jusqu'à ce que le texte ne contienne plus d'erreurs.

Ce procédé est représenté par la figure 2.3, extraite de l'article [Rodríguez 2007].

	(x)	
ITER-0	(p)	( )
ITER-1	(ŝ)	(Nine extra soul are planned half beam discovered these years)
	(ŝp)	(Nine)
	(c)	(extrasolar)
	(p)	(Nine extrasolar)
ITER-2	(ŝ)	(planets have been discovered these years)
	(ŝp)	(planets have been discovered)
	(c)	(this)
	(p)	(Nine extrasolar planets have been discovered this)
FINAL	(ŝ)	(year)
	(c)	(#)
	(p ≡ t)	(Nine <u>extrasolar</u> planets have been discovered <u>this</u> year)

FIG. 2.3 – Exemple de CATS – Extrait de l'article [Rodríguez 2007]

Les expériences menées dans le cadre de cet article montrent que la quantité de travail devant être fournie par le correcteur humain est diminuée grâce à cette technique. Selon le corpus sur lequel la méthode a été testée (EuTrans [Amengual 2000] et Albayzin Geographic [Días-Verdejo 1998]), respectivement 19% et 14% de mots en moins sont à corriger.

Le procédé de l'étape 4, permettant de proposer une nouvelle hypothèse de reconnaissance en prenant en compte le préfixe validé par l'utilisateur, est basé sur un décodage. Deux modèles de langage sont utilisés. Tout d'abord un modèle spécial est utilisé pour forcer le système à décoder à nouveau le préfixe validé, puis un modèle *n-gram*, combiné avec le premier, est utilisé pour rechercher le nouveau suffixe. L'inconvénient majeur de cette méthode, mettant en œuvre

un nouveau décodage, est qu'elle risquerait d'avoir un impact négatif sur le temps de réaction d'une interface homme-machine.

## **2.5 Conclusion**

Nous avons identifié deux types d'aides pouvant être apportées à l'utilisateur dans la phase d'assistance à la transcription de la parole. Le fait d'apporter des aides visuelles dans l'interface graphique permet de diminuer les temps de traitement de correction de l'utilisateur. De même, une méthode utilisant un décodage itératif prenant en compte les corrections/validations apportées par le transcriveur humain permet d'observer des gains de temps de traitement.



# **Chapitre 3**

## **Méthode proposée**

### **Sommaire**

---

<b>3.1</b>	<b>Réordonnancement automatique des hypothèses de reconnaissance .</b>	<b>42</b>
<b>3.2</b>	<b>Méthode proposée . . . . .</b>	<b>43</b>
3.2.1	Principe . . . . .	43
3.2.2	Application . . . . .	45
3.2.3	Exemple . . . . .	46
<b>3.3</b>	<b>Modèle cache . . . . .</b>	<b>47</b>
<b>3.4</b>	<b>Mots hors vocabulaire . . . . .</b>	<b>48</b>

### 3.1 Réordonnancement automatique des hypothèses de reconnaissance

Dans cette partie nous proposons une méthode, basée sur les mêmes principes que celle de [Rodríguez 2007], de réévaluation de l'hypothèse de reconnaissance du SRAP en fonction des corrections apportées par le correcteur. Nous avons fait le choix de ne pas réaliser à nouveau des décodages, de manière à pouvoir intégrer nos travaux dans une application interactive. Le décodage, en l'état actuel, ne s'effectuant pas en temps réel, l'utilisateur aurait dû attendre la fin de la retranscription du segment en cours de correction, à chaque modification de sa part.

En sortie du SRAP, des treillis de mots sont obtenus. Ils représentent, après élagage, tous les chemins hypothèses développés (voir figure 3.1). Chaque état correspond à un instant, chaque lien représente un mot, associé à une probabilité (qui n'est pas représentée dans le schéma).

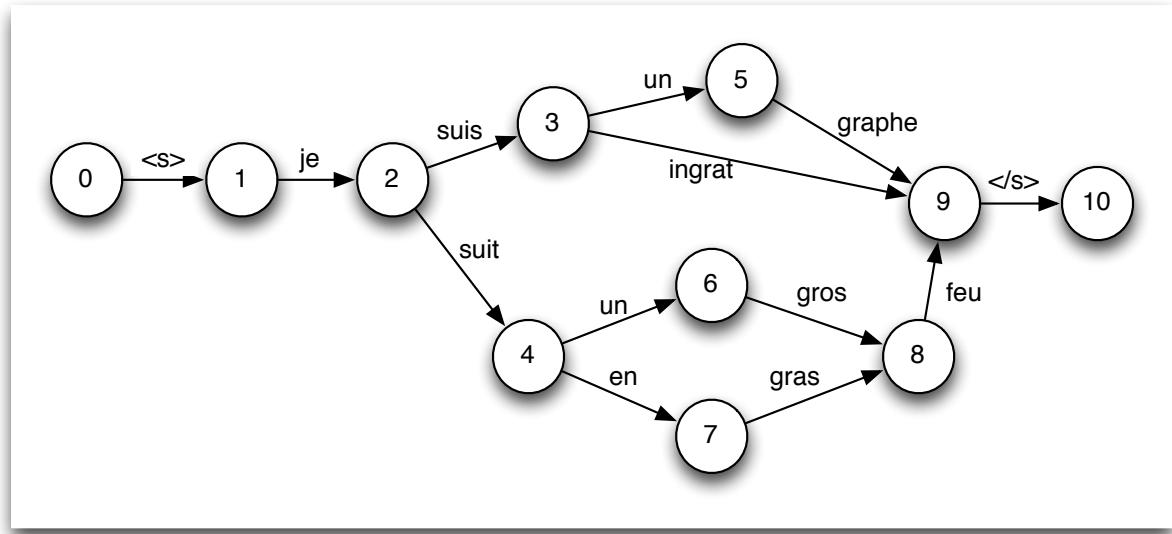


FIG. 3.1 – Graphe de mots

Ce graphe est transformé en réseau de confusion (voir figure 3.2). La transformation consiste à fusionner les mots localement identiques, à regrouper sur des ensembles de confusion communs les mots temporellement proches, et à supprimer les chemins d'hypothèse trop faible [Mangu 2000]. Chaque mot obtient un nouveau score qui est sa probabilité *a posteriori* (obtenue à partir du treillis de mots) divisée par la somme des probabilités *a posteriori* des mots en concurrence avec lui.

Le symbole  $\epsilon$  représente une transition vide (absence de mot).

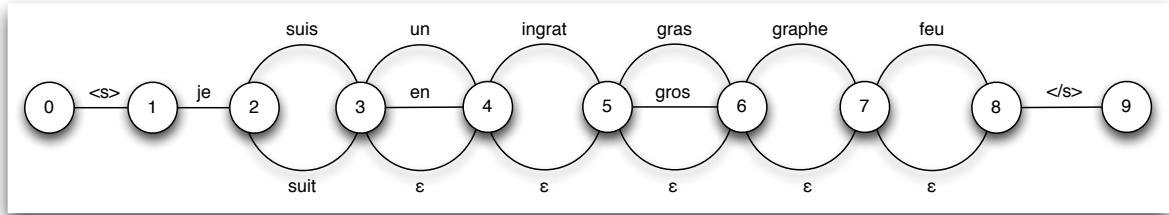


FIG. 3.2 – Réseau de confusion

## 3.2 Méthode proposée

La méthode proposée s'inspire des méthodes développées pour la tâche de CAT. Elle consiste à réévaluer la meilleure hypothèse d'un réseau de confusion en fonction des corrections apportées par l'utilisateur, sans nécessiter un décodage complet.

### 3.2.1 Principe

Les réseaux de confusion sont créés à partir du graphe de mots généré par la passe numéro 4 du système du LIUM développé pour la campagne d'évaluation ESTER 2 (voir chapitre 1.2). Cette méthode de transformation, développée au LIUM, est une adaptation de la méthode de [Mangu 2000]. Les réseaux de confusion contiennent, en plus de l'ensemble des mots en concurrence, des temps approximatifs provenant du graphe de mot.

L'instant de départ d'un état du réseau de confusion correspond au plus petit instant des mots associés à cet état ; le temps de fin correspond à l'instant de fin du dernier de ces mots.

La meilleure hypothèse de reconnaissance est la séquence de mots qui maximise chacune des probabilités *a posteriori* du réseau de confusion (voir figure 3.3). Il sera proposé à l'utilisateur de corriger cette meilleure hypothèse.

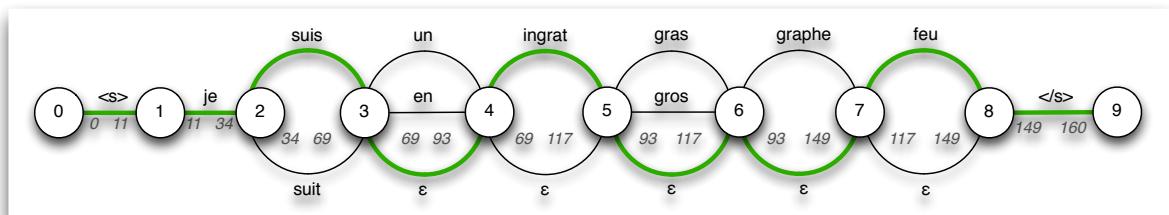


FIG. 3.3 – Réseau de confusion avec des informations temporelles et la meilleure hypothèse

Le chemin en vert, lu de la gauche vers la droite, indique la meilleure hypothèse de reconnaissance. Les chiffres en italique représentent les temps de début et de fin de chacun des états du réseau de confusion.

Comme nous l'avons vu dans le chapitre 1.3, à partir d'une séquence d'observations acoustiques  $X$ , l'objectif du SRAP est de trouver la séquence de mots  $\hat{W}$  la plus probable parmi l'ensemble des séquences possibles  $W$ . Cela se traduit par la recherche de  $\hat{W}$  maximisant la probabilité d'émission de  $W$  sachant  $X$ , correspondant à l'équation suivante :

$$\hat{W} = \arg \max_W P(W|X) \quad (3.1)$$

qui après application du théorème de Bayes et simplification devient :

$$\hat{W} = \arg \max_W P(X|W)P(W) \quad (3.2)$$

où  $P(W)$  est fourni par le modèle de langage et la probabilité  $P(X|W)$  correspond à la probabilité attribuée par le modèle acoustique.

Dans notre cas, la séquence de mots possibles  $W$  est séparée en deux : un préfixe  $p$  qui a été validé et/ou corrigé par l'utilisateur et un suffixe  $s$  à déterminer en fonction du préfixe  $p$ . Nous cherchons donc la séquence de mots  $\hat{s}$ , parmi tous les suffixes possibles  $s$  maximisant l'équation suivante :

$$\hat{s} = \arg \max_s P(s|X, p) \quad (3.3)$$

soit :

$$\hat{s} = \arg \max_s P(X|s, p)P(s|p) \quad (3.4)$$

La méthode présentée ici ne remet pas en cause l'acoustique,  $P(X|s, p)$  est constant.  $P(s|p)$  sera calculé à partir d'une interpolation linéaire entre un modèle de langage quadrigramme et un modèle cache calculé à partir des mots présents dans le préfixe validé. Nous cherchons donc, parmi tous les suffixes candidats  $\tilde{s}$  dans le réseau de confusion, le suffixe  $\hat{s}$  maximisant la probabilité suivante :

$$\hat{s} = \arg \max_{\tilde{s}} ((1 - \lambda)P_{4G}(s|p) + \lambda P_{cache}(s|p)) \quad (3.5)$$

avec  $P_{4G}$  la probabilité attribuée par le modèle quadrigramme classique de notre système et  $P_{cache}$  la probabilité attribuée par le modèle cache. Ce dernier est construit à partir des mots contenus dans le préfixe  $p$ . Il permet de renforcer la probabilité des mots récemment rencontrés ; on suppose ici qu'ils ont une plus forte chance d'apparaître dans le futur (dans  $s$ ).

Plusieurs types de modèle cache ont été testés. Afin d'évaluer son impact sur la technique de réordonnancement, des expériences ont également été menées en ne prenant pas en compte la probabilité  $P_{cache}$  ( $\lambda = 0$ ).

Les meilleurs résultats en terme de perplexité ont été obtenus à partir de la méthode proposée dans [Clarkson 1997]. La probabilité d'apparition du mot  $w_i$  est exponentiellement proportionnelle à la distance entre la position actuelle et les apparitions précédentes du mot  $w_i$  dans l'historique  $h_i$  :

$$P_{cache}(w_i|h_i) = \beta \sum_{j=1}^{i-1} I_{\{w_i=w_j\}} e^{-\alpha(i-j)} \quad (3.6)$$

avec  $\alpha$  le coût du décalage dans le cache,  $I_{w_i=w_j} = 1$  si  $w_i = w_j$  et 0 sinon, et  $\beta$  est une constante de normalisation calculée comme suit :

$$\beta = \frac{1}{\sum_{j=1}^{i-1} e^{-\alpha j}} \quad (3.7)$$

**Exemple** Supposons que nous ayons la phrase suivante : “m1 m2 m3 m4 m5 m6 m1”, composée de 7 mots. La probabilité de voir “m1” apparaître à nouveau en position 8 sera la suivante, d’après (3.6) et (3.7) :

$$P_{cache}(m1|m1, m2, \dots, m6, m1) = \frac{1}{\sum_{j=1}^7 e^{-\alpha j}} (e^{-\alpha(8-1)} + e^{-\alpha(8-7)}) \quad (3.8)$$

### 3.2.2 Application

À partir du réseau de confusion et du préfixe corrigé et/ou validé, la recherche des suffixes candidats dans le réseau de confusion est effectuée de la manière suivante. Soit  $t$  l'instant de fin du dernier mot validé. Le principe va être de rechercher, parmi tous les états suivants du réseau de confusion, ceux ayant un instant de début supérieur ou égal à  $t$ . La recherche est récursive et exhaustive. Pour chacun des états concurrents, nous recherchons à nouveau les états pouvant lui succéder et ainsi de suite. L'utilisation de ces temps, bien qu'approximatifs, permet d'éviter de choisir des séquences de mots dans lesquels certains mots se chevaucheraient. Imaginons que nous ayons un mot qui débute à l'instant  $t + 1$  et se termine en  $t + 3$ . Nous ne pourrons pas avoir, dans la même séquence, un mot qui débuterait en  $t + 1$  et se terminerait en  $t + 2$ .

La méthode proposée ne se déclenche que lorsque l'utilisateur remplace un mot par un autre (erreur de type substitution). Le réordonnancement automatique s'arrête dès que le dernier mot proposé correspond à un mot qui était présent dans l'hypothèse initiale. Si l'utilisateur fait le choix de supprimer un mot (mot incorrect glissé entre deux mots corrects) ou d'en insérer un (mot manquant entre deux mots corrects) plutôt que de faire une correction (substitution), l'hypothèse a été faite que le mot suivant était juste. Le réordonnancement automatique ne s'exécute donc pas sur ce type d'erreurs.

Comme dans [Rodríguez 2007], les mots précédant le mot en erreur sont considérés corrects. La première étape va consister à rechercher à quel état du réseau de confusion le mot venant d'être corrigé peut être rattaché. Si ce mot est présent dans le graphe à l'endroit de la correction, il sera rattaché à l'état correspondant, s'il n'est pas présent, le mot sera ajouté à l'état du mot substitué.

Cette première étape réalisée, toutes les séquences possibles de mots pouvant succéder à l'état sélectionné du graphe sont recherchées, en respectant les indices temporels.

Le score calculé grâce à l'équation (3.5) permet de départager les différentes séquences possibles. Si deux séquences de mots ont le même score (cas exceptionnel), la probabilité *a posteriori* moyenne de la séquence de mots devient l'élément discriminant.

### 3.2.3 Exemple

Pour reprendre les mots contenus dans les graphes de mots précédents, imaginons que la phrase correcte soit “je suis un graphe” et que l'hypothèse retenue par le SRAP soit “je suis ingrat feu” (figure 3.3). L'utilisateur va corriger le mot “ingrat” en le mot “un”. En respectant les indices temporels présents dans le réseau de confusion, les différentes séquences possibles sont alors : “**je suis un** gras/gros feu” ou “**je suis un** graphe”. La figure 3.4 représente les différents chemins possibles.

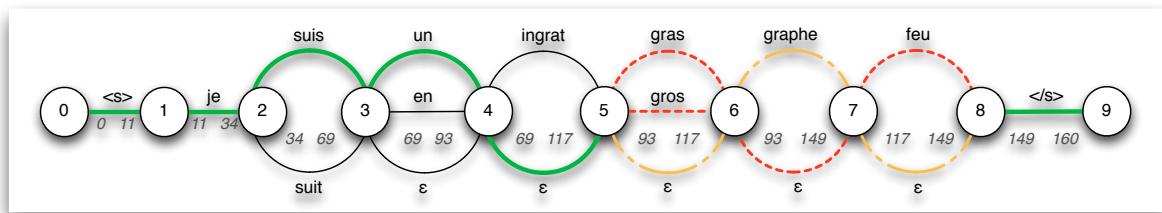


FIG. 3.4 – Réseau de confusion : 2 chemins possibles

La partie du réseau représentée en vert symbolise les zones où il n'y a pas d'erreur. Les arcs oranges et rouges (pointillés) représentent les deux chemins possibles dans le réseau de confusion. Pour faire notre choix parmi les 3 phrases possibles, nous utilisons un modèle de langage interpolé avec un modèle cache. La séquence de mot avec la plus grande probabilité d'apparition dans le modèle sera choisie.

Dans notre exemple, les probabilités des séquences “**je suis un** gros”, “**je suis un** gras” et “**je suis un** graphe” vont être calculées. Imaginons que la séquence retenue soit “je suis un graphe”. Si il reste encore une alternative après le mot “graphe”, la même stratégie est appliquée (la fenêtre de calcul des probabilités *n-gram* glisse d'un mot vers la droite). Par exemple, nous aurons à choisir entre : “**suis un graphe** orienté” et “**suis un graphe** pondéré”.

Dans cet exemple, le simple fait d'avoir corrigé le mot “ingrat” en “un” a permis de corriger toute la phrase.

### 3.3 Modèle cache

À la fin de la correction de chaque segment, les mots qui le composent sont stockés dans un historique servant à construire un modèle cache.

Afin d'apprendre les paramètres  $\lambda$  et  $\alpha$  (équations 3.5 et 3.6) qui représentent respectivement le poids attribué au modèle cache par rapport à celui du modèle *4-gram* et la pénalité affectée à la distance entre deux apparitions d'un mot, un modèle de langage *4-gram* a tout d'abord été estimé sur le corpus d'apprentissage ESTER 1. Ensuite, l'optimisation des poids a été réalisée comme suit :

- Les calculs ont été réalisés sur le corpus de test d'ESTER 1. L'outil prend donc comme paramètres d'entrée le corpus de test d'ESTER 1 et le modèle *4-gram*.
- Chaque segment est traité un par un. Il s'agit de déterminer les probabilités de chacun des mots le composant. Pour cela, nous avons fixé les valeurs de  $\lambda$  et  $\alpha$ . Soit  $P_S$  le total des probabilités des mots du segment provenant du modèle interpolé (exprimé en log base 10) et  $N_S$  le nombre de mots du segment, la perplexité  $PPL_S$  sera exprimée de la façon suivante :

$$PPL_S = 10^{-\frac{P_S}{N_S}} \quad (3.9)$$

- Un fois le traitement du segment terminé, l'ensemble des mots le composant, hormis les mots “outils”, sont insérés dans le modèle cache. Sont considérés comme mots “outils” les mots apparaissant plus de 1000 fois dans le corpus d'apprentissage de ESTER 1. Il s'agit de 93 mots dont “de”, “la”, “le”, “les”, “des”, etc.
- À chaque changement d'émission radiophonique, le modèle cache est réinitialisé. Tous les mots qu'il contient sont supprimés.
- À la fin du traitement du corpus, la perplexité sur l'ensemble des segments est calculée. Elle représente la perplexité du modèle ayant les paramètres  $\lambda$  et  $\alpha$ . Soit  $P_G$  la somme des probabilités de tous les mots du corpus, et  $N_G$  le nombre de mots du corpus, la perplexité globale sur tout le corpus,  $PPL_G$ , sera exprimée de la façon suivante :

$$PPL_G = 10^{-\frac{P_G}{N_G}} \quad (3.10)$$

- Tout le processus est répété jusqu'à obtenir les paramètres  $\lambda$  et  $\alpha$  permettant de minimiser la perplexité globale.

La configuration ayant permis d'obtenir les meilleurs résultats est la suivante :

- Quand le mot  $w$  est présent dans le modèle cache, le coefficient  $\lambda = 0,69$ , sinon  $\lambda = 0$ .
- Le paramètre  $\alpha$  a été fixé à 0,01.

## 3.4 Mots hors vocabulaire

En l'état actuel, la liste de mots pouvant être proposée de façon automatique est limitée aux mots se trouvant dans le réseau de confusion. Si l'utilisateur saisit un mot inconnu du SRAP dans le préfixe, la recherche du suffixe parmi les différents suffixes candidats fera intervenir la probabilité du mot inconnu. Ce mot sera ajouté au modèle cache, mais il n'est pas possible que ce mot nouveau réapparaisse de façon automatique dans la suite des corrections. Pour qu'un mot absent du réseau de confusion puisse ressortir de façon automatique, il faudrait apporter des modifications à deux niveaux. Tout d'abord, il faudrait être capable d'ajouter des nouveaux mots dans le réseau de confusion. Cela pourrait être réalisé en phonétisant les différents mots du réseau de confusion et en repositionnant le mot nouveau sur les états contenant des mots phonétiquement proches (ou identiques). Dès lors, le nouveau mot pourrait apparaître de façon automatique durant la phase de réordonnancement des hypothèses. Si ce mot n'est pas présent dans le modèle *4-gram* du SRAP, cela pourrait introduire un biais en augmentant la probabilité d'apparition de ce nouveau mot auquel serait attribuée la probabilité du mot inconnu. Pour que ce mot retourne une probabilité différente de celle du mot inconnu, il faudrait modifier le modèle *4-gram*. Pour cela, nous pourrions imaginer un modèle de langage à base de classes, contenant une classe pour les mots nouveaux. Chaque mot à l'intérieur de la classe disposerait d'une probabilité en rapport avec sa fréquence d'apparition. Cet ajout dans le modèle de langage, couplé avec l'ajout du mot dans le dictionnaire de phonétisations, permettrait également d'enrichir le vocabulaire du SRAP. Le nouveau mot pourrait alors être présent dans les résultats des futurs décodages, et donc dans les futurs réseaux de confusions. Cela n'a pas été réalisé, faute de temps, dans le cadre de ces travaux de thèse. Ce manque constitue une limite forte qui devra être levée rapidement pour permettre une utilisation optimale, en conditions réelles, de la méthode proposée.

# **Chapitre 4**

## **Expériences et résultats**

### **Sommaire**

---

<b>4.1</b>	<b>Corpus &amp; SRAP . . . . .</b>	<b>50</b>
<b>4.2</b>	<b>Métriques . . . . .</b>	<b>50</b>
<b>4.3</b>	<b>Résultats . . . . .</b>	<b>54</b>
4.3.1	Sans utiliser la méthode de réordonnancement automatique . . . . .	54
4.3.2	En utilisant la méthode de réordonnancement automatique . . . . .	54

---

Les expériences menées simulent le comportement d'un transcriveur corigeant la meilleure hypothèse du réseau de confusion généré par le SRAP.

## 4.1 Corpus & SRAP

L'optimisation des coefficients du modèle cache a été réalisée sur le corpus ESTER 1. Les expériences ont été effectuées sur le corpus de test d'ESTER 2. Ces deux corpus sont présentés section 1.2. Pour rappel, il s'agit d'émissions radiophoniques francophones.

Les réseaux de confusion sont créés à partir du graphe d'hypothèse des mots généré par le SRAP du LIUM développé lors de la campagne ESTER 2.

Sans l'ajout de traitements particuliers pour les segments provenant de la radio africaine, le taux d'erreur mot (*Word Error Rate – WER*) obtenu avec le SRAP du LIUM sur le corpus de test de la campagne ESTER 2 est de 19,2%. Le taux d'erreur oracle, correspondant au WER minimal pouvant être obtenu en sélectionnant la meilleure hypothèse dans les réseaux de confusion générés par le SRAP, est de 11,9% sur le corpus de test d'ESTER 2.

## 4.2 Métriques

Nous proposons d'évaluer notre méthode selon deux métriques. Le WSR (*Word Stroke Ratio*) [Civera 2004, Cubel 2004, Rodríguez 2007] et le KSR (*Keystroke Saving Rate*) [Wood 1996, Wandmacher 2007, Trost 2005].

Le WSR est une métrique couramment utilisée dans les méthodes de CAT. Le WSR est le nombre de mots à corriger divisé par le nombre total de mots dans la référence. Ce taux va donc être calculé en comptant le nombre de mots erronés à corriger. Dans le cas où l'on n'utilisera pas de méthode d'aide à la transcription, le WSR serait identique au WER classique si l'on considère que chaque mot erroné doit être corrigé. La comparaison entre le WER et le WSR va donc nous donner une mesure du nombre de mots à corriger avec et sans aide à la transcription.

Le KSR a été mis en place dans les systèmes de communication assistés (AAC – *Augmentative and Alternative Communication*) destinés aux handicapés ; il se calcule de la façon suivante :

$$KSR = \left(1 - \frac{k_p}{k_a}\right) * 100 \quad (4.1)$$

où  $k_p$  est le nombre d'actions effectivement réalisés par l'utilisateur lors de la saisie d'un message et  $k_a$  le nombre d'actions qui auraient été nécessaires sans aide à la composition de mots. Ces actions peuvent être des frappes sur un clavier ou des interactions avec un dispositif

particulier mis en place pour la gestion de l'handicap de l'utilisateur : joystick, clignement d'un oeil, etc.

Dans notre cas,  $k_p$  représentera le nombre d'actions réalisées par l'utilisateur pour corriger l'hypothèse du SRAP, en utilisant un clavier, et  $k_a$  le nombre d'actions qui auraient été nécessaires en partant d'une hypothèse vide (ne contenant pas de mots).

Pour calculer le KSR, il est supposé que l'utilisateur appuiera sur le moins de touches possible pour obtenir la transcription corrigée.

Dans la méthode présentée figure 4.1, le système calcule le nombre d'actions de deux manières différentes :

- Combien d'actions sont nécessaires si tous les mots de la zone en erreur sont supprimés puis retapés ?
- Combien d'actions devront être réalisées si le maximum de lettres de l'hypothèse sont conservées et que les actions ne portent que sur les lettres en erreur ?

Un alignement entre l'hypothèse générée par le système de reconnaissance automatique de la parole et la transcription de référence est effectué. Cet alignement sera réalisé au niveau des mots et au niveau des lettres correspondant aux zones en erreur.

La figure 4.1 présente un exemple d'alignement entre une référence et une hypothèse.

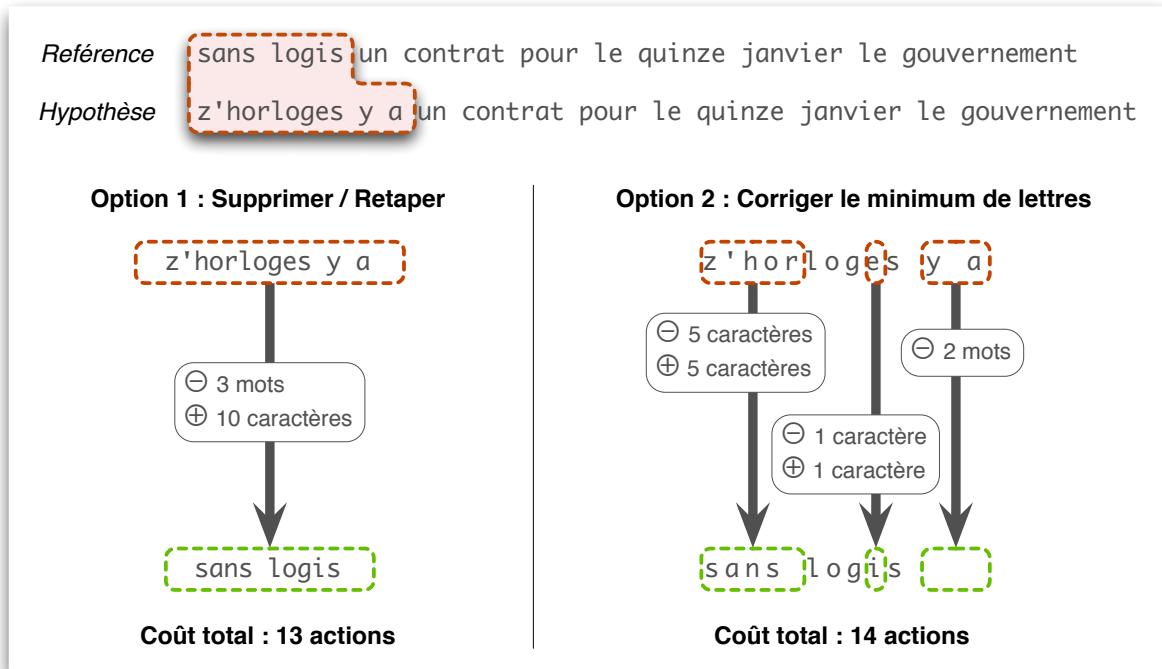


FIG. 4.1 – Méthode de calcul

Dans l'exemple présenté ici, la méthode présentant le coût le moins élevé est celle consistant à supprimer et à retaper les mots en erreur. Les coûts appliqués sont les suivants :

## Chapitre 4. Expériences et résultats

---

- Les coûts de déplacement à l’intérieur du texte de mot en mot ne sont pas pris en compte. L’application d’aide à la correction, comme celle de [Rodríguez 2007], présente les mots un à un lors du procédé de transcription, et chaque mot est corrigé lors de son apparition.
- La suppression d’un mot coûte 1. Nous avons imaginé qu’un raccourci clavier permettrait de supprimer un mot entier.
- L’appui sur une touche du clavier coûte une action.

Dans un premier temps, le nombre d’actions à réaliser par l’utilisateur a été calculé en ne lui proposant que la sortie du système de reconnaissance automatique à corriger, sans autre aide. La possibilité lui a ensuite été donnée de remplacer un mot par un autre par simple sélection d’un mot concurrent dans une liste de mots. Ayant imaginé que la liste des concurrents était toujours visible à l’écran, un coût de un a été attribué pour le remplacement d’un mot de cette manière. L’interface de correction pourrait ressembler à la capture d’écran présentée figure 4.2.

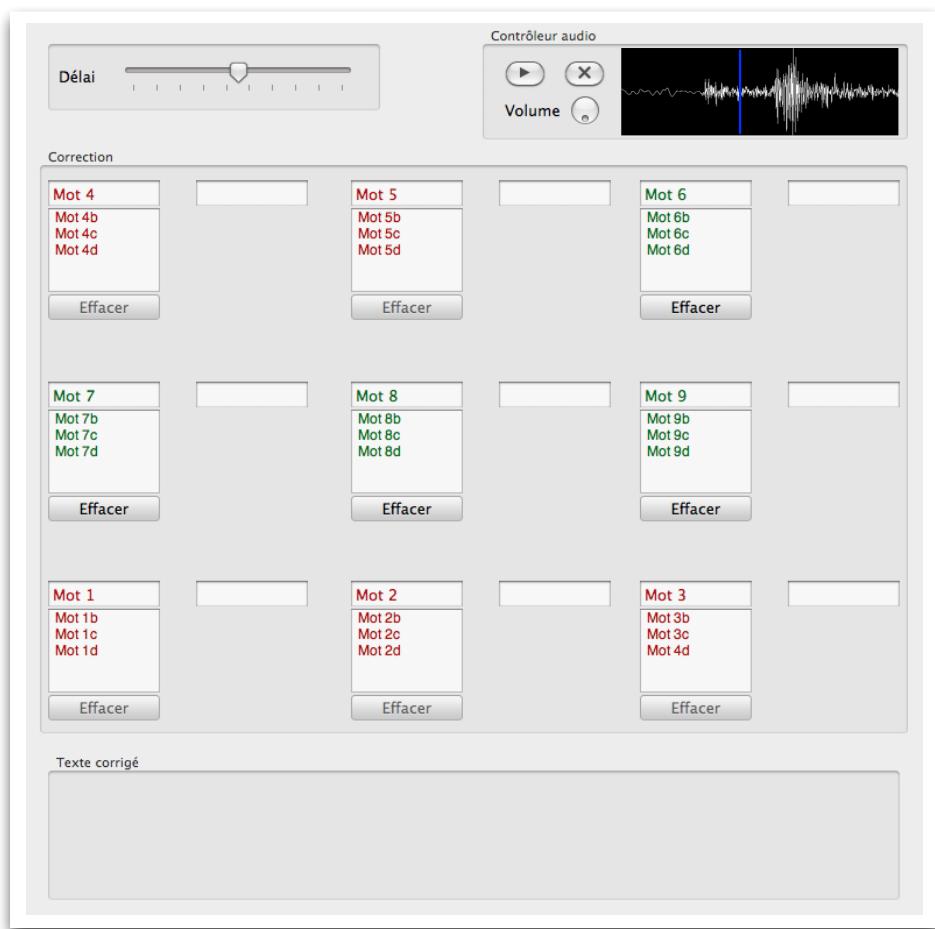


FIG. 4.2 – Interface à laquelle pourrait ressembler l’outil d’aide à la correction

Comme dans [Rodríguez 2007], les mots apparaîtraient un à un à l'écran, en commençant du coin supérieur gauche. Passé un certain délai, la fenêtre de mots éditables se décalerait vers la suite du texte. Les mots en bleu dans l'interface sont modifiables alors que ceux en rouge ne le sont plus. La suppression complète d'un mot pourrait être réalisée à l'aide d'un raccourci clavier. Entre chaque mot, une zone de texte permet de rajouter des mots ayant éventuellement été supprimés lors du décodage.

Le KSR ne prend pas en compte la pénibilité de chaque action pour l'utilisateur. La méthode de réordonnancement a donc été également évaluée en terme de WSR. Le nombre de mots erronés contenus dans chaque segment est comptabilisé jusqu'à rencontrer une erreur de *substitution*. L'erreur de *substitution* est elle aussi comptée puis la méthode automatique se déclenche et propose une nouvelle hypothèse. Le même calcul est alors effectué jusqu'à obtenir une hypothèse identique à la référence (*ie* jusqu'à ce que l'hypothèse ne contienne plus d'erreur). Le WSR est alors égal au nombre total d'erreurs comptabilisées divisé par le nombre de mots de la référence.

La figure 4.3 montre un exemple de calcul de ce WSR.

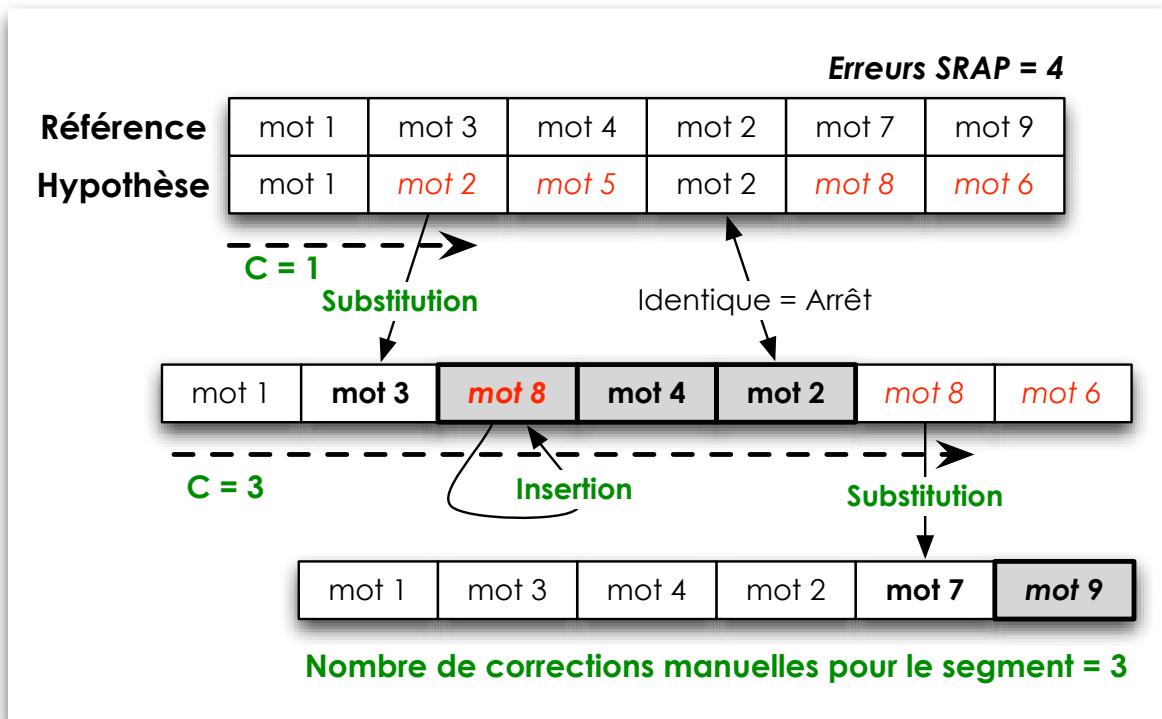


FIG. 4.3 – Calcul du taux d'erreur mot de la méthode de réordonnancement automatique

La figure 4.3 présente un cas où la méthode de réordonnancement permet l'économie de la correction d'un mot. Dans cet exemple, le WER est de  $4/6 = 66,7\%$  et le WSR est de  $3/6 = 50\%$ .

C représente le nombre de corrections manuelles. Le WSR est égal à la somme des corrections manuelles de chaque segment divisé par le nombre de mots de la référence. Les zones grises de la figure représentent les mots ayant été proposés de façon automatique par la méthode de réordonnancement.

## 4.3 Résultats

### 4.3.1 Sans utiliser la méthode de réordonnancement automatique

**Sans les listes déroulantes** Tout d'abord, le nombre d'actions à réaliser pour saisir la référence de transcription manuellement a été comptabilisé. La référence est composée de 435 005 lettres (caractères espace compris). Il faut donc appuyer 435 005 fois sur les touches du clavier pour la saisir dans sa totalité («Méthode manuelle» dans le tableau 4.1). En utilisant les sorties du système de reconnaissance automatique de la parole (ligne SRAP), ce nombre d'actions est de 53 492. L'utilisation du système de reconnaissance automatique de la parole a donc permis de réaliser un gain en terme de *KSR* de 87,7%, pour un taux d'erreur mot de 19,2%.

**Avec les listes déroulantes** L'utilisateur a maintenant la possibilité d'utiliser les hypothèses du réseau de confusion pour remplacer un mot par un autre. Il peut donc remplacer un mot par l'un de ses concurrents en le sélectionnant dans une liste déroulante. Cette liste contient uniquement les mots situés sur le même état que le mot sélectionné. Pour reprendre le réseau de confusion présenté figure 3.4, la liste déroulante située sous le mot “un” ne contiendra que le mot “en”. Nous considérons toujours que l'utilisateur choisit la méthode qui va lui permettre de minimiser le nombre d'actions qu'il a à réaliser. Dans ce cas, le nombre d'actions à effectuer par l'utilisateur diminue et passe à 52 003, soit un *KSR* de 88% (ligne SRAP+liste). Le WER ne change pas étant donné que le nombre de mots erronés reste le même : seule la méthode permettant de remplacer ces mots a évolué.

### 4.3.2 En utilisant la méthode de réordonnancement automatique

**Méthode automatique seule (sans liste déroulante, sans modèle cache)** Les sorties du SRAP associées à la mise en œuvre de la méthode de réordonnancement automatique permettent d'observer un *KSR* de 89,2% (46 795 actions), pour un WSR de 17% (ligne SRAP+auto).

**Méthode automatique avec liste déroulante (sans modèle cache)** L'ajout de la sélection des mots dans une liste déroulante permet de diminuer ce nombre d'actions à 44 732, soit un *KSR* de 89,7% (SRAP+auto+liste).

**Méthode automatique avec liste déroulante et utilisation du modèle cache** Enfin, le système complet (SRAP+auto+liste+cache) utilisant le modèle cache, la technique de réordonnancement automatique et la sélection des mots dans la liste déroulante, permet d'observer un *KSR* de 90,3% (41 992 actions) et un *WSR* de 15,8%.

Les tableaux 4.1 et 4.2 présentent un résumé de tous ces résultats.

TAB. 4.1 – *KSR et WER sur le corpus de test ESTER 2 sans aide à la transcription*

Méthode	Nombre d'actions	KSR	WER
Manuelle	435005	0%	–
SRAP	53492	87,7%	19,2%
SRAP + liste	52003	88,0%	19,2%

TAB. 4.2 – *KSR et WSR sur le corpus de test ESTER 2 avec aide à la transcription*

Méthode	Nombre d'actions	KSR	WSR
SRAP + auto	46795	89,2%	17,0%
SRAP + auto + liste	44732	89,7%	17,0%
SRAP + auto + liste + cache	41992	90,3%	15,8%

En terme de nombre de mots à corriger, la méthode proposée permet d'obtenir un gain d'environ 3,4% points sur le corpus de test d'ESTER 2.

Il est à noter que lorsque le *WSR* diminue, le nombre d'actions à réaliser suit la même tendance. En effet, le nombre de mots à corriger passe de 19,2% à 15,8%, soit un gain de 17,7%. Le nombre d'actions, quant à lui, chute de 53 492 à 41 992, soit un gain de 21,5%. Lorsque l'on compare le nombre d'actions nécessaires à la correction de la sortie du SRAP seul, avec l'utilisation du SRAP complété de la méthode de réordonnancement automatique, là encore, le nombre d'actions clavier et le nombre de mots à corriger diminuent tous les deux : 12,5% de gain en terme de nombre d'actions et 11,4% de gain en terme de *WER/WSR*.



## **Chapitre 5**

# **Conclusion : Assistance automatique à la transcription manuelle**

**N**ous avons présenté dans cette partie une technique de réordonnancement automatique des hypothèses du SRAP. Nous avons également proposé une métrique, basée sur le KSR, visant à déterminer le nombre d'actions requises pour corriger une transcription. Cette technique permet d'observer un gain relatif de 17,7% en terme de WSR et de diminuer le KSR de 21,5% par rapport à l'utilisation seule des sorties du système de reconnaissance automatique de la parole.

Certaines améliorations peuvent encore être apportées à cette méthode, puisqu'elle ne permet pas pour l'instant l'ajout automatique de nouveaux mots (*ie* de mots qui ne seraient pas présents dans le réseau de confusion). Pour réaliser cela, une stratégie a été présentée dans le chapitre 3.4.

La méthode pourrait également être améliorée en propageant les corrections apportées par l'utilisateur. Pour l'instant, la correction d'un mot déclenche le réordonnancement automatique des hypothèses du réseau de confusion pour la fin du segment en cours de correction. Cette correction pourrait avoir un impact sur d'autres segments se trouvant plus éloignés dans la transcription.

Pour évaluer la méthode d'assistance à la transcription, il serait nécessaire de développer une application réelle et d'évaluer le gain de temps apporté par son utilisation. Cette application pourra également prendre en compte les transcriptions corrigées précédentes pour créer des modèles (vocabulaire, modèle de langage, modèles acoustiques) adaptés à l'utilisateur et au type de transcription que ce dernier réalise.

## **Troisième partie**

# **Phonétisation automatique**



**D**ans la partie précédente de ce manuscrit, une technique permettant à un transcripteur humain de collaborer avec un système de reconnaissance automatique de la parole a été proposée. Cette technique a pour but de minimiser le nombre d'actions à effectuer manuellement pour arriver à une transcription sans erreur.

Comme nous l'avons souligné dans l'introduction de ce document, la société Spécinov souhaite pouvoir, via des méthodes d'indexation, retrouver facilement les enregistrements dans lesquels certains noms propres sont mentionnés. De façon à permettre une indexation des documents audio transcrits automatiquement, il est nécessaire de disposer d'un décodage fiable des noms propres. Dans cette partie, nous nous intéressons à la phonétisation automatique des mots, et plus spécifiquement à celle des noms propres.

Le phonème est l'unité minimale du langage parlé. Plusieurs phonèmes prononcés les uns à la suite des autres forment un mot. La phonétisation automatique consiste à déterminer les séquences de phonèmes qui doivent être prononcées pour former chacun des mots d'un vocabulaire. Dans la plupart des langues naturelles, l'association entre les séquences de lettres (graphèmes) et les séquence de phonèmes est souvent ambiguë et dépend du contexte dans lequel se trouvent ces lettres. Beaucoup de langages ont continué à évoluer après que la graphie des mots a été mise en place, donc la correspondance entre la graphie et la prononciation de ces mots a changé au cours du temps. En particulier, les mots étrangers gardent souvent la graphie de leur langue d'origine plutôt que d'être adaptés pour correspondre aux conventions de la langue qui a emprunté ce mot. La phonétisation automatique de texte a d'abord été mise en œuvre dans le cadre d'applications de type texte vers parole (TTS – Text To Speech). Le texte en entrée doit être converti en séquences de phonèmes qui sont ensuite utilisées par un système de synthèse vocale pour produire le signal sonore correspondant. En reconnaissance automatique de la parole, le dictionnaire de phonétisation permet de faire le lien entre le niveau acoustique et le niveau linguistique pendant le processus de décodage.

La méthode de phonétisation automatique proposée est conduite par les données. Cette technique repose sur l'utilisation d'un décodeur acoustico-phonétique sur les zones du signal correspondant à la prononciation des noms propres. Cette phase d'extraction est suivie par une étape de filtrage, visant à éliminer les variantes de phonétisation considérées comme superflues. Ce procédé d'extraction/filtrage est itératif. Les résultats obtenus montrent un gain en terme de taux d'erreur noms propres et en terme de taux d'erreur mots sur les segments contenant des noms propres, tout en n'affectant pas le taux d'erreur mot global. Tout d'abord, un état de l'art sur les stratégies couramment utilisées pour phonétiser automatiquement des mots est présenté. Ensuite, la méthode proposée ainsi que les différentes expérimentations réalisées sont décrites. Pour clore cette partie, les résultats obtenus seront présentés et commentés.



# Chapitre 6

## État de l'art : méthodes de phonétisation

### Sommaire

---

<b>6.1</b>	<b>Introduction</b>	<b>64</b>
<b>6.2</b>	<b>Système à base de règles</b>	<b>64</b>
<b>6.3</b>	<b>Systèmes guidés par les données</b>	<b>65</b>
6.3.1	Prononciation par classifications locales	66
6.3.2	Prononciation par analogie	67
6.3.3	Utilisation des données acoustiques	70
<b>6.4</b>	<b>Conclusion</b>	<b>71</b>

## 6.1 Introduction

Beaucoup de techniques de conversion graphèmes vers phonèmes sont présentes dans la littérature. Les noms attribués à cette tâche sont nombreux : conversion graphème-phonème (grapheme-to-phoneme conversion) [Andersen 1996, Bellegarda 2005], modélisation de la prononciation des phonèmes (phonetic pronunciation modeling) [Riley 1999], traduction lettre vers son (letter-to-sound translation) [Pagel 1998], conversion lettre-phonème (letter-to-phoneme conversion) [Rama 2009], génération des formes de base des phonèmes (phonetic baseform generation) [Bahl 1991, Ramabhadran 1998], transcription phonétique (Phonetic Transcription) [Bisani 2001], traduction texte-phonème (text-to-phoneme mapping) [Suontausta 2000]... Les techniques de conversion graphème-vers-phonème (G2P) sont divisées en trois sous catégories : la recherche dans un dictionnaire phonétisé par un expert humain [Ferrané 1992, De Calmes 1998], l'utilisation de systèmes à base de règles de prononciations [Béchet 2001, Kaplan 1994] et l'utilisation de systèmes guidés par les données [Suontausta 2000, Bisani 2008, Bellegarda 2005, Kienappel 2001, Ma 2001, Black 1998, Yvon 1997, Galescu 2001, Pagel 1998].

Les systèmes que nous appelons systèmes à base de règles sont ceux pour lesquels il n'y a pas besoin d'apprentissage. Les règles de phonétisations sont écrites (ou partiellement écrites) à la main. Certains systèmes [Black 1998, Kienappel 2001, Ma 2001] utilisent des corpus d'apprentissage pour extraire un ensemble de règles.

Bien qu'efficace, la recherche des mots dans un dictionnaire phonétisé par un expert humain présente certains désavantages. Construire ce genre de dictionnaire à la main est fastidieux. De plus, la liste des mots de la langue n'étant pas finie (lieus, noms propres, mots empruntés à une autre langue, etc.), ce genre de dictionnaire a une couverture limitée.

## 6.2 Système à base de règles

Les systèmes à base de règles ont été mis en place pour dépasser la limite de couverture de la simple recherche dans un dictionnaire phonétisé. Ces règles ont pour but d'être capables de proposer une ou plusieurs phonétisations pour un mot uniquement d'après la façon dont ce dernier s'écrit. Le but de ce genre de système est d'encapsuler les régularités du langage à travers un petit nombre de règles reprenant les principes généraux de la langue.

De nombreux systèmes de G2P à base de règles ont été développés. Parmi tous ces systèmes, on trouve notamment, pour ne citer que les Français : le système du LADL (Laboratoire d'Automatique Documentaire et Linguistique) [Laporte 1988], le système TOPH développé à l'ICP (Institut de la Communication Parlée – Grenoble) [Aubergé 1991], les systèmes VARION

développés au LIMSI (Laboratoire d’Informatique pour la Mécanique et les Sciences de l’Ingénieur – Orsay) [Lacheret-Dujour 1990], le système GEPH développé à l’IRIT (Institut de Recherche en Informatique de Toulouse)[Tihoni 1991], les systèmes GRIPHON [Béchet 1995] et LIA\_PHON [Béchet 2001] du LIA (Laboratoire d’Informatique d’Avignon), ...

Tous ces systèmes fonctionnent exclusivement sur la base de règles de réécriture qui permettent de convertir une chaîne de lettres (graphème) en une chaîne de phonèmes, et ce généralement en trois étapes :

- Normalisation orthographique généralisée. Cette première phase de traitement consiste à réécrire les mots de façon à lever certaines ambiguïtés. Généralement, il existe plusieurs jeux de règles qui ne s’appliquent qu’à certaines classes de mots (les noms propres, les sigles, les verbes, ...). Par exemple, le mot Chesnay sera réécrit “ChênaY”.
- Conversion graphème-phonème, par application d’un petit jeu de règles, qui correspond approximativement à la connaissance de l’orthographe d’un enfant apprenant à lire [Yvon 1996]. Ce traitement s’applique indépendamment à chacun des mots à convertir.
- Un ensemble de règles phonologiques est ensuite appliqué. Cette fois, les règles sont appliquées aux phrases entières de façon à traiter, par exemple, les liaisons.

Bien que cette technique permette d’avoir une couverture complète de la langue, elle présente certains désavantages. Chaque langue contient un nombre important d’exceptions qui doivent être capturées par des listes d’exceptions ou par les règles exceptionnelles. Cette solution finit souvent par causer des effets de bord non désirés. Les interdépendances entre les règles peuvent devenir complexes, cela rend le développement et la maintenance de ces bases de règles très compliqué. De plus, concevoir des règles de prononciation demande des aptitudes spécifiques en linguistique. Les règles sont généralement formulées à l’aide d’un automate à états finis [Kaplan 1994].

À cause de ces déficiences, les systèmes à base de règles sont souvent utilisés en tant que complément, mais ne remplacent pas la recherche dans un dictionnaire phonétisé par un expert humain.

## **6.3 Systèmes guidés par les données**

Pour dépasser les limites des systèmes de phonétisation à base de règles, les systèmes guidés par les données ont vu le jour. Ces méthodes de G2P regroupent un large éventail d’algorithmes statistiques pour convertir automatiquement des orthographies en prononciations, en utilisant un corpus assez important, approprié au domaine considéré.

Ces systèmes peuvent être divisés en deux catégories. La technique basée sur des classifications locales (*local classification* ou *top-down approach*) et la technique de prononciation par analogie (*pronunciation by analogy* ou *bottom-up strategy*).

### 6.3.1 Prononciation par classifications locales

Cette stratégie de phonétisation est composée de deux phases. La première consiste à aligner les lettres et les phonèmes du corpus d'apprentissage. Cet alignement est généralement fait de façon à ce que chaque lettre corresponde à un phonème. Dans la pratique, une lettre peut correspondre à 0, 1 ou plusieurs phonèmes, et un phonème peut correspondre à plus d'une lettre. Pour cette raison, le symbole “-” (absence de lettre, absence de phonème) est introduit dans l'alignement, du côté des lettres et des phonèmes (voir figure 6.1).

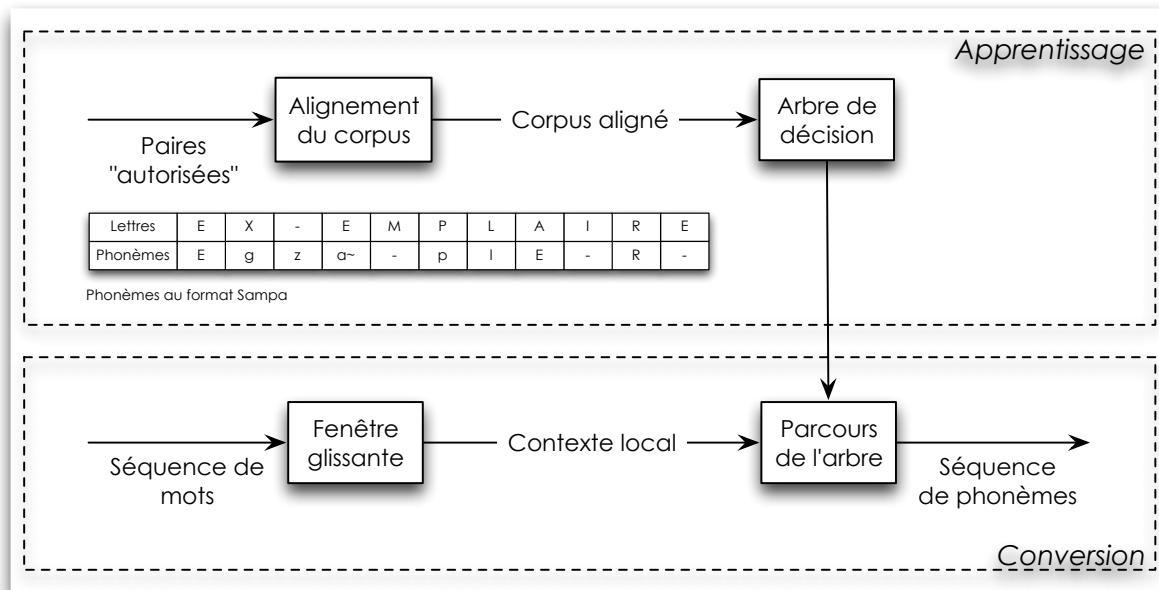


FIG. 6.1 – Exemple de conversion par classification locale de graphème vers phonème

Cette phase d'alignement est généralement faite dans une phase de pré-traitement. Elle est réalisée en utilisant des paires lettres-phonèmes dont la correspondance est autorisée [Black 1998]. Ces paires de correspondances sont appelées *paires “autorisées”* dans la littérature. Elles sont généralement déterminées manuellement [Pagel 1998], bien que des travaux utilisent une version de l'algorithme EM [Galescu 2001, Deligne 1995] pour les obtenir. Ces paires permettent, dans notre exemple figure 6.1, de définir que la lettre “X” correspond aux phonèmes “g z”. La séquence d'entrée est généralement parcourue de gauche à droite. Pour chaque lettre, une séquence de phonèmes (pouvant correspondre à 0, 1 ou

n phonèmes) est choisie parmi les paires de correspondances possibles. Après cette phase d’alignement, une technique d’apprentissage est utilisée pour permettre de prendre des décisions sur les mots non phonétisés. Les techniques les plus souvent utilisées sont les arbres de décisions [Torkkola 1993, Kienappel 2001, Daelemans 1997, Andersen 1996, Pagel 1998, Suontausta 2000, Häkkinen 2003], les réseaux Bayésiens [Ma 2001] et les réseaux de neurones [Sejnowski 1987, McCulloch 1987, Jensen 2000].

Le choix de la séquence de phonème est fait en fonction du contexte de la lettre en cours. Comme cette décision est prise à chaque position, avant de passer à la suivante, cette technique correspond à une classification locale.

### 6.3.2 Prononciation par analogie

Le terme prononciation par analogie (ou ascendante) est approprié pour toutes les techniques de conversion graphème-phonème guidées par les données. Néanmoins, ce terme est généralement utilisé pour les approches décrivant des techniques du type “recherche du plus proche voisin” [Yvon 1997]. Bien que la technique présentée dans la section précédente (6.3.1) donne de bonnes performances pour les mots vérifiant les règles standards de prononciation de la langue, les résultats se dégradent rapidement en présence de mots atypiques (dont la graphie ne permet pas, en utilisant les conventions de la langue, de déterminer la prononciation) [Black 1998].

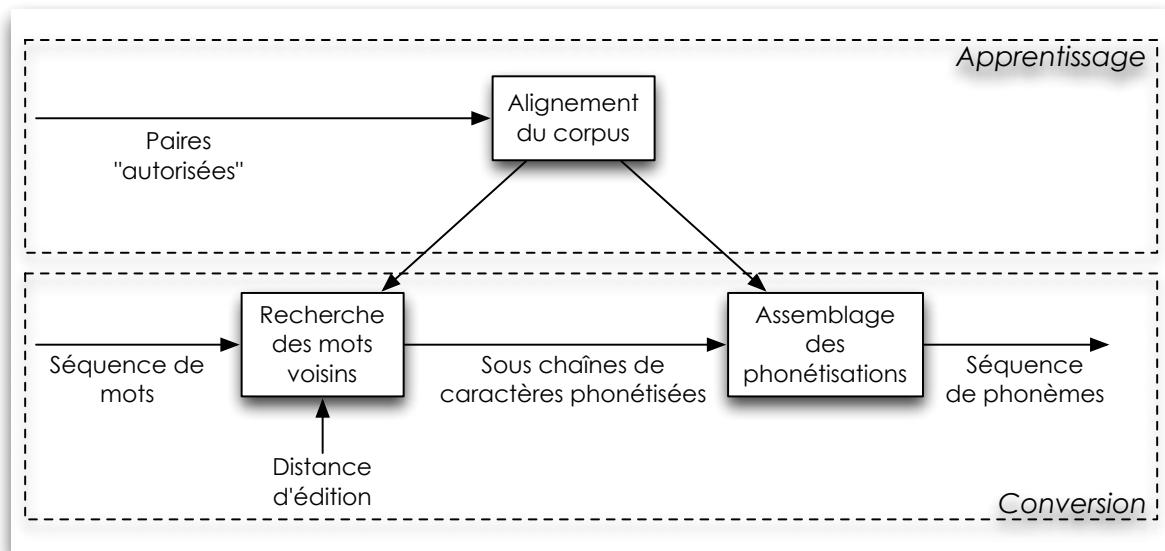


FIG. 6.2 – Exemple de conversion par analogie de graphème vers phonème

En utilisant une mesure de similarité entre les mots, comme par exemple la distance d'édition de Levenshtein [Levenshtein 1966], la technique consiste à extraire des phonétisations partielles des mots appris et à les réutiliser pour phonétiser les mots jamais rencontrés. Les fragments de phonétisations sont ensuite concaténés pour obtenir la phonétisation finale. Cette technique permet de prendre en compte les cas les plus rares, aux dépens d'une potentielle perte de robustesse [Luk 2001, Marchand 2001]. En considérant chaque mot dans son ensemble, l'approche ascendante va au-delà de la classification locale. Par exemple, imaginons que nous ayons dans notre corpus d'apprentissage le mot “Chesne”, prononcé, selon l'alphabet Sampa, “S E n” (et non pas “S E s n @”). Si dans notre test, nous avons le mot “Fresne” à phonétiser, la technique va consister à rechercher si un mot voisin à déjà été traité. Comme la partie du mot “esne” est déjà transcrise, la phonétisation correspondante, et correcte, sera réutilisée.

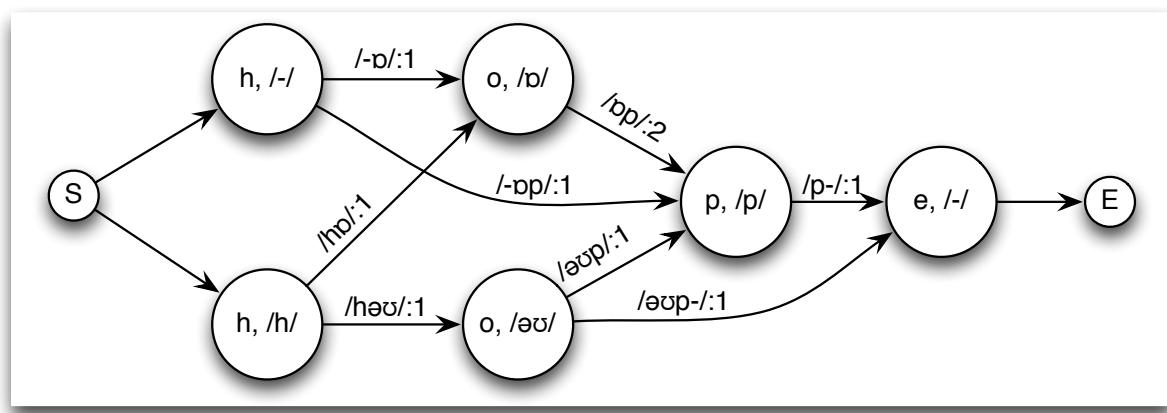


FIG. 6.3 – Réseau de confusion du système PRONOUNCE [Dedina 1991]

La méthode proposée par [Dedina 1991] (voir figure 6.3) scrute l'ensemble des mots du lexique et construit un réseau de confusion : chaque nœud représente un phonème candidat et chaque arc représente une prononciation possible. [Marchand 2001] étend cette méthode en attribuant des scores de fréquences d'apparitions sur les arcs (voir figure 6.4).

La méthode décrite dans [Bagshaw 1998] utilise un ensemble de correspondances graphèmes-phonèmes réalisé manuellement, et introduit des règles dépendantes du contexte entre ces unités. Un réseau de confusion des différents segments en compétition est ensuite construit. Les scores de transitions entre les différents états sont basés sur les poids ayant été attribués par les différentes règles activées et par les pénalités attribuées pour la violation de certaines règles. La transcription finale est obtenue par une recherche globale à travers ce réseau de confusion.

Comme nous l'avons vu plus haut, cette stratégie de phonétisation par analogie, en récupérant des phonétisations partielles et en les assemblant, permet de mieux gérer les cas les plus

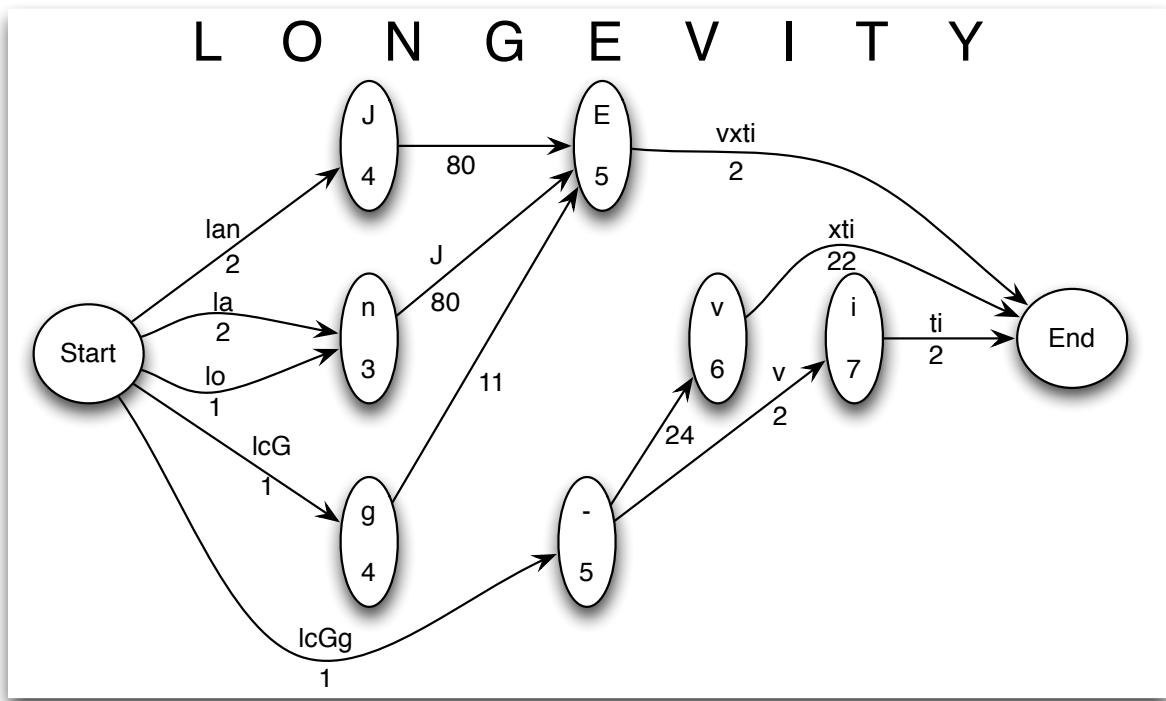


FIG. 6.4 – Réseau de confusion (extrait de l'article [Marchand 2001])

rares [Damper 1999]. Cependant, la phase d’alignement reste problématique, puisque l’alignement est effectué en utilisant une liste de paires autorisées dépendante de la langue. Les travaux de [Galescu 2001] et [Bisani 2008] proposent une méthode basée sur l’utilisation de modèles n-grams à séquences jointes (*n-gram joint sequence models*), qui permet de relâcher certaines de ces contraintes, tout en augmentant grandement les données nécessaires à l’apprentissage et le temps d’apprentissage des modèles.

[Bellegarda 2005] utilise l’analyse sémantique latente pour définir une mesure de similarité globale entre les mots. Pour transcrire un mot, tout d’abord l’ensemble des entrées lexicales similaires est recherché. Ensuite, chacune des séquences de la liste des similarités est alignée, et pour chaque position de l’alignement, le phonème le plus fréquent est choisi.

Pour le cas particulier des noms propres, une étude sur la génération dynamique des déformations plausibles des formes canoniques de noms propres est proposée dans [Béchet 2002]. Cette étude a été menée dans le cadre d’une application d’annuaire téléphonique développée par France Télécom R&D. La méthode utilisée consiste à réévaluer les *n* meilleures hypothèses de reconnaissance générées lors d’une passe de décodage dans laquelle les déformations dépendent de la nature des hypothèses en compétition.

### 6.3.3 Utilisation des données acoustiques

[Byrne 1998, Bisani 2001, Sloboda 1995, Bahl 1993, Haeb-Umbach 1995, Mokbel 1999, Svendsen 1995, Deligne 2003, Deligne 2001, Rose 1997, Ramabhadran 1998, Wu 1999] proposent des méthodes de phonétisation basées sur l'utilisation des données acoustiques à disposition.

[Bahl 1993] et [Svendsen 1995] utilisent l'algorithme A\* pour trouver la meilleure phonétisation à partir de plusieurs réalisations acoustiques des mots. L'algorithme A\* [Pearl 1984] est un algorithme de recherche asynchrone du meilleur chemin dans un graphe. Pour cela, il utilise une fonction d'évaluation  $F(n)$  pour chaque nœud exploré. Cette valeur représente une estimation du coût du meilleur chemin passant par le nœud  $n$ . Elle est obtenue en sommant le coût réel  $r(n)$  du chemin du début du graphe à  $n$ , et une estimation du coût  $h(n)$  (appelée sonde) du chemin restant à parcourir (du nœud  $n$  jusqu'à la fin du graphe).

[Bahl 1993, Haeb-Umbach 1995] utilise une fonction heuristique pour la détermination de ce coût  $h(n)$ , visant à trouver la phonétisation maximisant la vraisemblance à partir de plusieurs réalisations acoustiques du mot à phonétiser (voir l'équation 6.3 dans laquelle le coût  $h(n)$  est estimé par  $\varphi_w^*$ ).

Supposons que nous ayons  $U$  réalisations acoustiques  $x_1, \dots, x_U$  d'un mot  $w$ , et que notre ensemble de phonèmes soit noté  $\Phi$ , alors la vraisemblance  $V$  du modèle de prononciation  $\pi$ , étant donné le modèle acoustique  $\Theta$  est :

$$V(\pi) = \prod_{u=1}^U \sum_{\varphi \in \Phi^*} p(x_u | \varphi; \Theta) p(\varphi | w; \pi) \quad (6.1)$$

Nous pouvons supposer que pour chaque mot  $w$  il y a une meilleure phonétisation notée  $\varphi_w^*$  :

$$p(\varphi | w) = \begin{cases} 1 & \text{si } \varphi = \varphi_w^* \\ 0 & \text{sinon} \end{cases} \quad (6.2)$$

La phonétisation maximisant la vraisemblance de  $w$  peut donc être formulée comme suit :

$$\varphi_w^* = \arg \max_{\varphi} \prod_{u=1}^U p(x_u | \varphi; \Theta) \quad (6.3)$$

Cette méthode, visant à rechercher la phonétisation maximisant la vraisemblance de  $w$ , part du postulat qu'il n'existe qu'une phonétisation par mot. [Svendsen 1995] a amélioré cette approche en calculant un coût  $h(n)$  à partir du meilleur chemin de chacune des réalisations acoustiques. Malheureusement, cette heuristique est trop optimiste dans des conditions où il y a une

grande variabilité entre les différentes réalisations acoustiques. [Wu 1999] propose une méthode pour éliminer ce problème en introduisant une stratégie de présélection, visant à restreindre la recherche à un réseau de phonèmes construit à partir d'heuristiques. [Mokbel 1999] propose deux critères de décision pour sélectionner les  $k$  meilleures transcriptions phonétiques. Le premier de ces critères s'appuie sur la fréquence d'apparition de cette transcription, et la seconde sur la maximisation de la vraisemblance (équation 6.3). La méthode offrant les meilleurs résultats est celle basée sur la maximisation de la vraisemblance. Pour chaque réalisation acoustique, la liste des n-best (avec  $n=50$ ) est construite et contrainte par la maximisation de vraisemblance de l'union de ces listes. [Sloboda 1995] utilise le deuxième critère, à savoir la sélection des  $k$  plus fréquentes phonétisations extraites.

Les auteurs de [Bisani 2001] extraient les 100 mots les plus fréquents et phonétisent ces mots en utilisant leurs réalisations acoustiques. Ces phonétisations sont ensuite évaluées en terme de PER (Phoneme Error Rate).

[Deligne 2003, Deligne 2001] met en place une méthode basée sur l'utilisation d'un décodage acoustico-phonétique pour l'ajout de nouvelles phonétisations dans le vocabulaire personnalisé d'un utilisateur. Pour cela, ce dernier doit répéter une ou deux fois chacun des mots qu'il veut ajouter à son lexique.

[Rose 1997, Ramabhadran 1998] présentent un décodeur acoustico-phonétique sensiblement identique, nécessitant de la part de l'utilisateur la prononciation successive de mots à phonétiser. Chaque utilisateur doit prononcer douze noms propres différents, et doit téléphoner dix fois à partir de téléphones différents (portable, fixe) et dans un environnement acoustique différent (hall, cafétéria, ...). Le décodage est basé sur la combinaison de modèles acoustiques indépendants du locuteur avec un modèle représentant les transitions entre les phonèmes (modèle de langage ne contenant que des phonèmes).

Les travaux présentés dans [Galescu 2001] sont basés sur l'utilisation d'un modèle bidirectionnel à base de n-grams à séquences jointes. Ce modèle peut être utilisé pour déterminer la suite de phonèmes d'un mot à partir de sa graphie ou à partir d'une représentation acoustique de ce dernier.

## 6.4 Conclusion

Dans cette partie, un ensemble de techniques visant à obtenir les phonétisations des mots ont été présentées. Chaque stratégie présente des avantages et des inconvénients. Les auteurs de [Bisani 2008] ont comparé certains des travaux présentés dans cet état de l'art sur des corpus anglophones (voir tableau 6.1).

TAB. 6.1 – Résumé et comparaison de la précision de différents systèmes de G2P sur des corpus anglophones (Extrait de [Bisani 2008])

Corpus	Auteur	Type de G2P	PER [%]	WER [%]
OALD	[Pagel 1998] avec POS	Classifications locales	6,03	21,87
	[Pagel 1998] sans POS	Classifications locales		23,34
	= [Bisani 2008]	Analogie	$3,54 \pm 0,19$	$17,49 \pm 0,78$
NETtalk 15k	[Andersen 1996] [Bisani 2008]	Classifications locales Analogie	8,26 $\pm 0,32$	47,0 $33,67 \pm 1,10$
NETtalk 18k	[Torkkola 1993]	Classifications locales	9,2	
	[Yvon 1996]	Analogie		36,04
	[Galescu 2001]	Analogie	9,00	36,07
	[Galescu 2001]	Acoustique	10,03	41,87
	[Bisani 2008]	Analogie	$7,83 \pm 0,16$	$31,79 \pm 0,54$
NETtalk 19k	[Marchand 2001]	Analogie		34,5
	[Bisani 2008]	Analogie	$7,66 \pm 0,31$	$31,00 \pm 1,09$
CMUDict	[Galescu 2001]	Analogie	7,0	28,5
	[Galescu 2001]	Acoustique	11,5	49,7
	= [Bisani 2008]	Analogie	$5,88 \pm 0,18$	$24,53 \pm 0,65$

Les lignes marquées avec “=” utilisent exactement les mêmes données pour l’apprentissage et le test. Les autres lignes utilisent des reproductions fidèles. “±” indique un intervalle de confiance de 90%.

Le système proposé par [Bisani 2008] (basé sur une approche par analogie – voir 6.3.2) montre de très bons résultats sur l’ensemble des corpus anglophones testés.

# Chapitre 7

## Méthode proposée

### Sommaire

---

<b>7.1</b>	<b>Introduction</b>	<b>74</b>
<b>7.2</b>	<b>Méthodes de G2P utilisées pour construire le dictionnaire initial</b>	<b>76</b>
7.2.1	Système à base de règles	76
7.2.2	Corpus parallèle ( <i>bitext</i> )	76
7.2.3	Système à base de modèles à séquences jointes (JSM)	77
7.2.4	Utilisation d'un système SMT (Statistical Machine Translation) pour la conversion G2P	78
<b>7.3</b>	<b>Extraction de phonétisations à l'aide d'un DAP</b>	<b>79</b>
<b>7.4</b>	<b>Filtrage des variantes de phonétisation</b>	<b>81</b>
7.4.1	Motivation	81
7.4.2	Méthodes	81
<b>7.5</b>	<b>Méthode itérative de génération des phonétisations</b>	<b>84</b>
7.5.1	Résumé de la méthode	84

---

## 7.1 Introduction

Les systèmes de reconnaissance vocale à grand vocabulaire ont des performances correctes dans des contextes d'utilisation connus et contrôlés. Cependant, les noms propres sont fréquemment des mots hors vocabulaire et leur reconnaissance est généralement considérée comme une tâche difficile.

De nombreuses situations nécessitent de transcrire correctement les noms propres. Il est généralement intéressant de savoir qui parle et quand, en particulier lors de tâches comme l'indexation multimédia, la recherche documentaire, la transcription et le compte-rendu de réunions.

Le SRAP du LIUM inclut, dans son processus de décodage, une étape de segmentation en tours de parole puis une étape de classification, permettant de regrouper tous les segments d'un même locuteur. Les travaux de [Jousse 2008], basés sur le principe proposé par [Canseco-Rodriguez 2005], s'appuient sur les transcriptions (manuelles pour l'instant) pour créer un arbre de classification sémantique couplé avec un système de détection d'entités nommées. Cette méthode est bien adaptée aux enregistrements radiotélévisés où le passage de parole est généralement accompagné de l'annonce sur le nom du locuteur suivant. La figure 7.1, extraite de l'article [Jousse 2008] présente le principe de base du système.

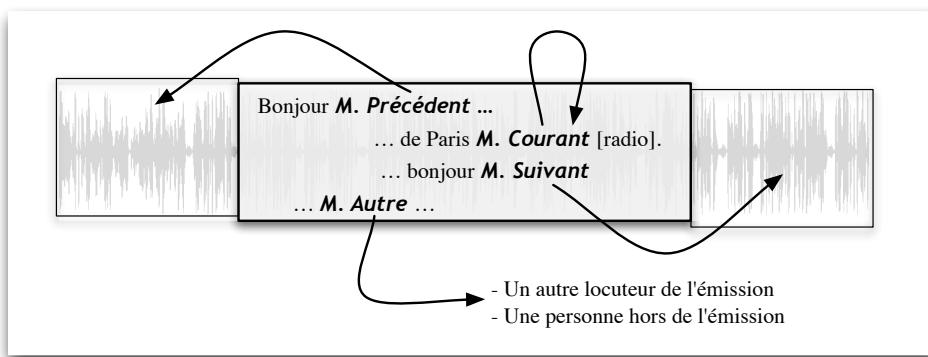


FIG. 7.1 – Principe de base du système (extrait de l'article [Jousse 2008])

Lorsque ces travaux seront appliqués sur les sorties automatiques du système de reconnaissance de la parole, il faudra que les noms propres soient bien décodés par le système. C'est une des applications possibles des travaux présentés dans cette partie. La prononciation des noms propres est moins normalisée que la prononciation des autres mots. Une suite de lettres, dans un mot quelconque, sera généralement prononcée de la même manière, quel que soit le mot dans lequel cette suite de lettres apparaîtra. Ce n'est pas le cas pour les noms propres.

Dans les transcriptions manuelles à notre disposition, les mots ne sont pas alignés avec le signal, les temps de début et de fin de chaque mot ne sont pas disponibles. Ces frontières sont

déterminées à l'aide d'un alignement forcé de chaque mot dans le signal. Quand les occurrences des noms propres sont isolés, un Décodage Acoustico Phonétique (DAP) est réalisé pour générer un ensemble de transcriptions phonétiques. Ce jeu de phonétisations contient un nombre très important de variantes de phonétisations. Nous proposons une méthode de filtrage visant à éliminer les variantes qui semblent ne pas être utiles au décodage et qui pourraient générer des confusions avec d'autres mots phonétiquement proches.

Au cours de l'avancement de nos travaux, nous avons constaté que le dictionnaire servant à réaliser la phase d'alignement forcé des mots sur le signal sonore, avait une influence importante sur nos résultats. En effet, l'utilisation d'un dictionnaire mal phonétisé pose des problèmes au niveau de la détection des frontières des mots à aligner. Trois systèmes de G2P différents ont été comparés pour réaliser cette phase d'initialisation. La méthode est itérative : le meilleur dictionnaire filtré de chaque passe est réutilisé pour réaliser à nouveau la phase d'alignement, et le procédé redémarre. Le processus s'arrête quand le dictionnaire filtré de la passe précédente est strictement identique au nouveau dictionnaire filtré (voir figure 7.2)

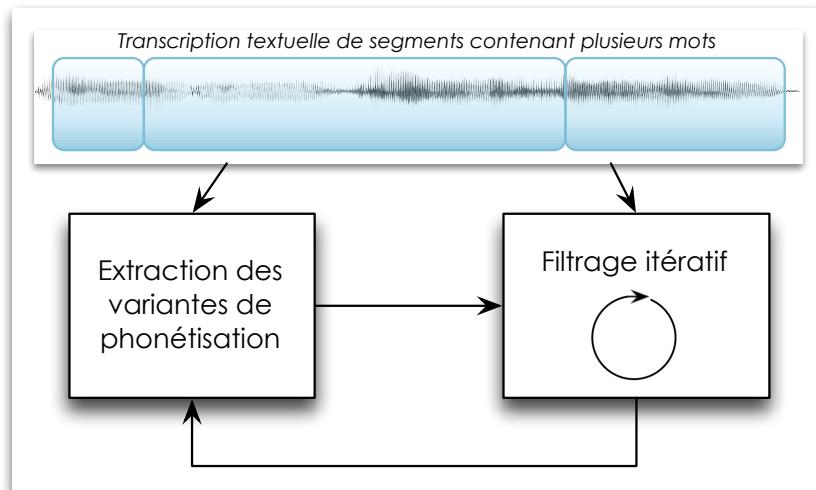


FIG. 7.2 – Méthode proposée

Les nouvelles phonétisations générées seront évaluées en terme de taux d'erreur mots (Word Error Rate — WER), et en terme de taux d'erreur noms propres (Proper Noun Error Rate — PNER). Ces taux seront calculés sur le corpus de la campagne d'évaluation ESTER 1 [Galliano 2005].

Tout d'abord, les différents systèmes de conversion G2P que nous avons utilisés pour réaliser le dictionnaire d'alignement vont être présentés. Ensuite la méthode d'extraction et de filtrage sera décrite, puis les résultats obtenus seront présentés.

## 7.2 Méthodes de G2P utilisées pour construire le dictionnaire initial

Trois systèmes différents de phonétisation ont été comparés pour la détection des frontières des noms propres. Une détection erronée de ces frontières génère des phonèmes incorrects en début et/ou fin des phonétisations des noms propres.

### 7.2.1 Système à base de règles

Tout d'abord, le système de conversion G2P à base de règles LIA\_PHON [Béchet 2001] a été testé. Il utilise la graphie des mots pour déterminer les suites de phonèmes correspondantes. L'un des atouts de ce système est d'être capable de phonétiser des mots sans nécessiter le signal sonore correspondant (voir la partie 6.2).

LIA\_PHON a participé à la campagne de tests des phonétiseurs français connue sous le nom d'ARC B3 (Action de Recherche Concertée B3). Les phonétisations générées par le système ont été comparées à des textes phonétisés par des experts. Le taux d'erreur phonème a été calculé sur la même base que le taux d'erreur mot utilisé en reconnaissance automatique de la parole. Sur 86938 phonèmes, 99,3% des transcriptions phonétiques générées par LIA\_PHON ont été identifiées comme correctes. Cependant, les résultats présentés dans [Béchet 2001] montrent une répartition non uniforme des erreurs selon les classes de mots. 25,6% des erreurs générées par LIA\_PHON sont issues de la phonétisation des noms propres qui ne représentent pourtant que 5,8% des mots du corpus de test.

En effet, la phonétisation des noms propres présente un haut niveau de variabilité difficile à prédire. Par exemple, dans le corpus de développement ESTER, le prénom du chanteur « Joey Starr » est prononcé de quatre manières différentes (« dZoe », « dZoj », « Zoe », ou « Zoj » en format Sampa), bien que tous les locuteurs parlent français et connaissent ce chanteur. Cette variabilité illustre la très grande difficulté pour générer l'ensemble des règles permettant de produire l'ensemble des variantes de phonétisation.

Idéalement, le système devrait être capable de détecter à la fois l'origine du nom propre et la façon dont les gens, en fonction de leurs origines socio-culturelles, pourraient prononcer ce nom. Malheureusement, ces deux tâches sont très complexes, voire impossibles, car le système ne dispose pas d'informations *a priori* sur l'origine du locuteur.

### 7.2.2 Corpus parallèle (*bitext*)

Afin d'apprendre les modèles nécessaires au fonctionnement des deux autres méthodes de phonétisation utilisées, il est indispensable de disposer d'un corpus parallèle, également

nommé *bitext*. Ce corpus associe, dans notre cas, une séquence de lettres avec une séquence de phonèmes. Le tableau 7.1 montre les trois représentations de ce corpus qui ont été testées. Dans la représentation A, une séquence de lettres correspond à un mot. Une même séquence de lettres (même mot) peut être associé à plusieurs séquences de phonèmes (variantes de phonétisation). Dans les représentations B et C, les séquences de lettres correspondent à un groupe de mots. Ces groupes de mots sont les plus longues séquences de mots, observées dans le corpus d'apprentissage, se situant entre deux phonèmes représentant des phénomènes acoustiques correspondant à autre chose que de la parole (musique, silence, rire, ...). En effet, l'hypothèse a été faite que l'influence de la prononciation d'un mot sur celle de son voisin est négligeable quand ces derniers sont séparés par ce type de phénomène. Dans la représentation C, un symbole a été ajouté pour marquer la limite de chaque mot.

TAB. 7.1 – Exemple des représentations A, B et C du corpus bitext (phonèmes au format Sampa)

Rep.	Graphemes	Phonemes
A	d e s j e u n e s f i l l e s	d E Z 9 n f i j
B	d e s j e u n e s f i l l e s	d E Z 9 n f i j
C	d e s # j e u n e s # f i l l e s #	d E # Z 9 n # f i j #

Les représentations B et C permettent de prendre en compte les règles phonologiques comme, par exemple, les liaisons. La représentation C permet de différencier les influences *inter* et *intra* mots. Dans les représentations B et C, les séquences de phonèmes sont obtenues en réalisant un alignement forcé utilisant les modèles acoustiques et le dictionnaire de référence.

### 7.2.3 Système à base de modèles à séquences jointes (JSM)

Le système proposé par [Bisani 2008] a été présenté comme étant l'un des meilleurs système de phonétisation automatique (voir la partie 6.4). Il s'agit d'un système à l'état de l'art dont une implémentation logicielle est disponible sous licence Open Source.

Cette méthode repose sur une approche de phonétisation par analogie guidée par les données (voir la partie 6.3.2) utilisant des modèles à séquences jointes. L'idée générale de cette méthode repose sur le fait que, ayant déjà observé plusieurs "morceaux" d'un mot à phonétiser, il devrait être possible, en joignant les phonétisations des "morceaux" de mots correspondants, de générer la phonétisation de ce nouveau mot.

Ce type de modèles présente l'intérêt de permettre de générer des phonétisations, même pour les formes atypiques de mots rarement rencontrés. Les résultats présentés dans le tableau

6.1 montrent que cette stratégie est à l'état de l'art, puisqu'elle rivalise avec la plupart des autres méthodes présentées. La phase d'entraînement nécessite en entrée un corpus parallèle associant des séquences de lettres avec des séquences de phonèmes. La représentation qui a été retenue pour construire les modèles est la représentation A. L'apprentissage des modèles en utilisant les représentations B et C a été abandonnée, du fait du temps de traitement (plusieurs semaines). L'idée consiste à apprendre des modèles successivement, en commençant par un modèle *1-gram*, dont l'ordre des *n-gram* augmente à chaque itération, jusqu'à converger vers un modèle ne permettant plus de progresser sur le corpus de cross-validation. Dans notre cas, le modèle final retenu est un modèle *6-gram*.

Le dernier modèle est alors utilisé pour transcrire des mots qui n'étaient pas dans le dictionnaire ayant servi à l'apprentissage.

Cette méthode de phonétisation sera appelée JSM dans la suite de ce manuscrit.

#### **7.2.4 Utilisation d'un système SMT (Statistical Machine Translation) pour la conversion G2P**

Nous avons proposé une méthode dans [Laurent 2009] basée sur l'utilisation de SMT (Statistical Machine Translation) pour convertir des graphèmes vers des phonèmes. Ce travail a été réalisé en collaboration avec Paul Deléglise. Nous avons montré que l'utilisation de la méthode permettait de diminuer la taille des données d'apprentissage nécessaire à la construction du dictionnaire de phonétisation, tout en obtenant des résultats proches de ceux constatés en utilisant l'ensemble des données à disposition. Les systèmes de SMT (Statistical Machine Translation) sont généralement utilisés pour convertir une séquence de mots d'un langage source vers une séquence de mots d'un langage cible. Nous l'avons utilisé pour réécrire une séquence de lettres en une séquence de phonèmes. La méthode a été développée et testée avec les données provenant de la campagne d'évaluation ESTER 2 (2008) [AFCP 2008] (voir partie 1.2). Pour la phase d'apprentissage d'un SMT, il est nécessaire de disposer d'un corpus bitext. Généralement ce corpus associe des phrases du langage source avec, en parallèle, les phrases correspondantes en langage cible. Dans le cas de l'utilisation d'un SMT pour la conversion G2P, un bitext va associer une séquence de lettres avec une séquence de phonèmes. Plusieurs représentations de ce corpus ont été testées (voir 7.2.2). Celle ayant permis d'observer les meilleurs résultats est la représentation C .

Généralement, l'optimisation des modèles de traduction est réalisée en maximisant le score BLEU [Papineni 2002]. Les paramètres du modèle sont adaptés jusqu'à obtenir le meilleur score BLEU possible. Nous avons proposé une méthode basée sur la minimisation de la distance

d'édition de Levenshtein [Levenshtein 1966], qui dans notre cas donne de meilleurs résultats qu'une optimisation à l'aide du score BLEU.

À la fin de chaque étape d'itération, les trois meilleures hypothèses de phonétisation sont générées pour chaque exemple d'entraînement (séquence de lettres) avec le modèle de traduction courant. La somme des distances d'éditions de Levenshtein normalisées  $S_{nd}$  est calculé entre les phonétisations proposées et les phonétisations de référence.

$$S_{nd} = \sum_{t \in T} \log\left(1 - \frac{d_t}{l_t}\right) \quad (7.1)$$

où  $d_t$  est la distance d'édition de Levenshtein de la phonétisation  $t$ ,  $l_t$  est la longueur de la phonétisation de référence correspondant à  $t$ , et  $T$  est l'ensemble des phonétisations générées.

Les paramètres du modèle de traduction sont ajustés jusqu'à obtenir la plus petite somme  $S_{nd}$ .

Les résultats obtenus en utilisant cette méthode montrent un gain en terme de taux d'erreur mot (WER) par rapport à l'utilisation de JSM. Sur le corpus de développement d'ESTER 2, SMT, utilisé pour phonétiser l'ensemble du vocabulaire du SRAP du LIUM, donne un taux d'erreur mot de 27,1% contre 27,7% pour JSM utilisé dans les mêmes conditions.

## 7.3 Extraction de phonétisations à l'aide d'un DAP

Un système de décodage acoustico-phonétique (DAP) permet de générer la suite de phonèmes la plus probable correspondant à un signal sonore. Pour obtenir les phonétisations automatiques des noms propres, les portions du signal correspondant aux mots à phonétiser sont extraites automatiquement grâce aux transcriptions manuelles. Ces portions sont ensuite décodées en utilisant le système de DAP. Les noms propres qui sont présents plusieurs fois dans le corpus peuvent donc être phonétisés différemment, permettant ainsi d'obtenir des variantes de prononciations.

Les auteurs de [Bisani 2001] indiquent qu'un décodage sans contrainte linguistique ne permet pas d'obtenir un décodage phonétique fiable. Nous sommes arrivés expérimentalement à la même conclusion.

L'utilisation d'un modèle de langage sur les suites de phonèmes permet de guider le système de reconnaissance vocale en minimisant le risque de voir apparaître des séquences de phonèmes très peu probables. Le décodage a été contraint en utilisant des triphones à états partagés et un modèle de langage 3-gram ne contenant que des phonèmes. Ce modèle de langage a été appris à partir du dictionnaire *expert* BDLEX [De Calmes 1998]. Ce dictionnaire ne contient pas de nom propre.

Le système de décodage acoustico-phonétique est très proche d'un système de décodage classique utilisé en reconnaissance de la parole, mis à part que son lexique et son modèle de langage ne contiennent que des phonèmes à la place de mots.

La première étape consiste à isoler les portions du signal qui correspondent aux noms propres en utilisant la transcription manuelle à notre disposition. Les mots considérés comme étant des noms propres sont annotés dans la transcription manuelle. Les mots de la transcription manuelle ne sont pas alignés sur le signal sonore : les instants de début et de fin de chaque mot ne sont pas disponibles. Seuls les instants de début et de fin de chaque segment sont à notre disposition. Les frontières temporelles de chaque mot des segments sont déterminées en les alignant sur le signal à l'aide d'un système d'alignement forcé (voir figure 7.3). Cette phase d'alignement est réalisée entre les mots de la référence et le signal sonore correspondant.

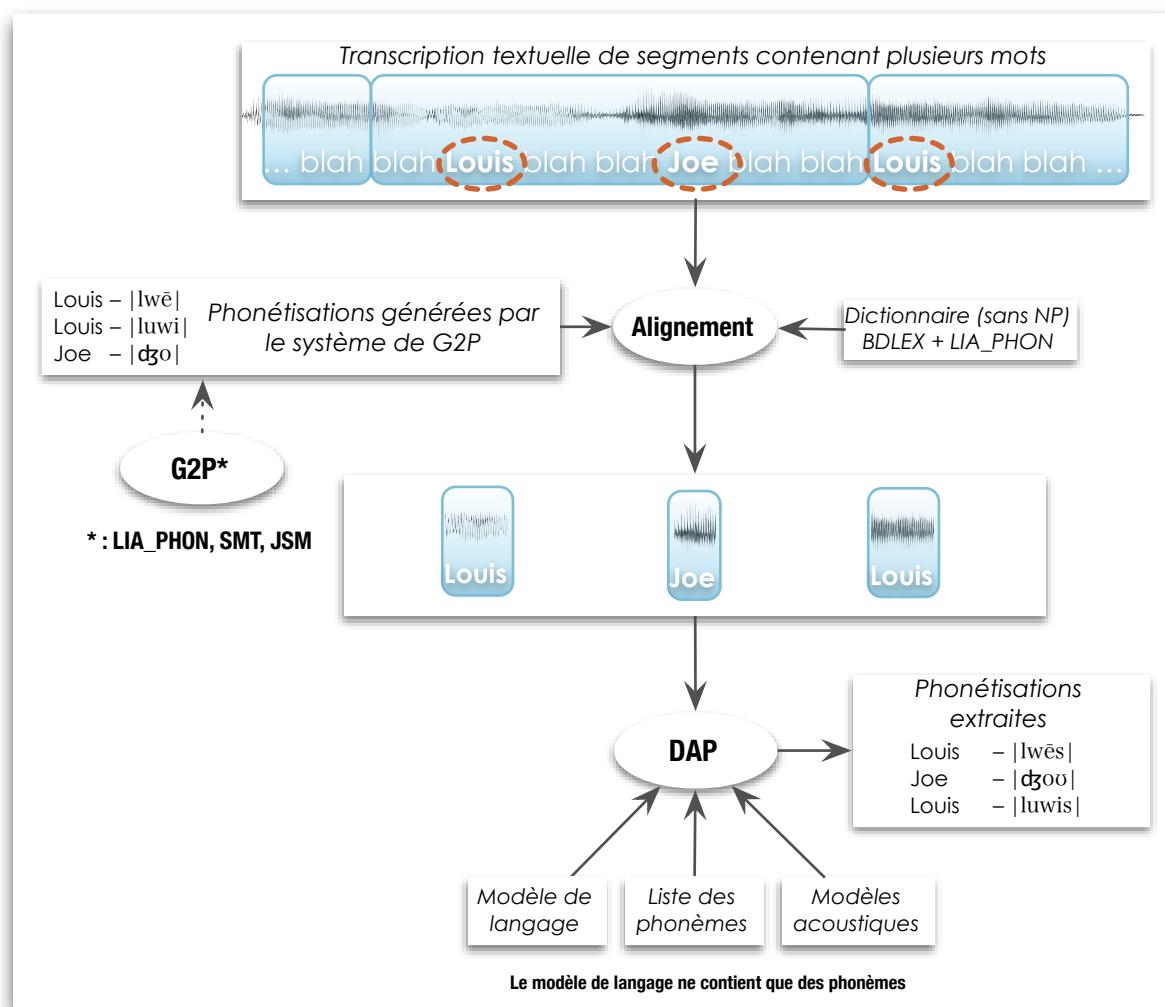


FIG. 7.3 – Illustration de l'utilisation du décodage acoustico-phonétique

La phonétisation des noms propres utilisés pour cette tâche est obtenue en utilisant différentes méthodes de phonétisation automatique. LIA\_PHON, SMT et JSM ont été testés pour la réalisation de cette étape.

Afin de réaliser l'alignement du signal sur la référence, nous avons besoin de disposer de tous les mots de la référence. Les mots n'étant pas présents dans le dictionnaire BDLEX, et n'étant pas des noms propres, ont été phonétisés automatiquement à l'aide de LIA\_PHON (dictionnaire de référence de notre SRAP) dans tous les cas, de façon à ne pas introduire de biais dans l'évaluation de la méthode.

Les transcriptions phonétiques des noms propres générées en utilisant chacune de ces méthodes peuvent être erronées. Par conséquent, la détection des frontières pouvant être imprécise, des phonèmes erronés sont susceptibles d'être générés en début et/ou fin de nom propre, ce qui introduit une cause d'erreur lors de l'utilisation ultérieure de cette variante en transcription.

## 7.4 Filtrage des variantes de phonétisation

### 7.4.1 Motivation

Le système à base de DAP extrait un nombre important de variantes de phonétisation par nom propre. Environ 20000 variantes de phonétisations sont extraites pour environ 29000 occurrences des noms propres dans le corpus d'apprentissage. Le nombre de variantes extrait est reporté dans le tableau 8.1. Ce nombre important de variantes augmente le risque de décoder des noms propres à la place d'autres mots phonétiquement proches. Comme le nombre d'occurrences des autres catégories de mots est normalement largement supérieur au nombre d'occurrences des noms propres, il y a un risque d'augmenter le taux d'erreur mot (WER) global, tout en observant une diminution du taux d'erreur nom propre (PNER).

Pour minimiser ce risque, nous filtrons les variantes de phonétisation extraites en ne gardant que celles qui semblent les plus appropriées.

### 7.4.2 Méthodes

Dans un premier temps, la méthode de filtrage “sélection” a été mise en place.

**Sélection** Pour chaque variante de phonétisation de chaque nom propre, un dictionnaire temporaire est construit. Ce dictionnaire contient uniquement la forme de phonétisation du nom propre à évaluer et l'ensemble des autres mots qui ne sont pas des noms propres. Les phrases qui contiennent ce nom propre sont transcris en utilisant le dictionnaire temporaire. La phonétisation du nom propre est considérée comme valide si le nom propre apparaît au moins

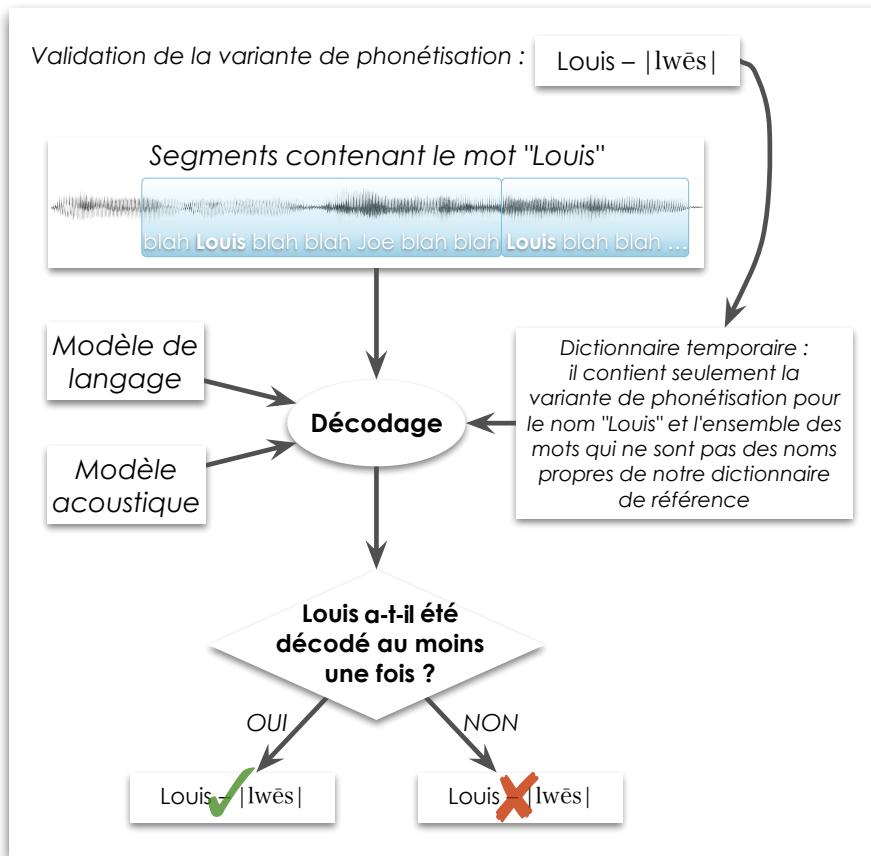


FIG. 7.4 – Représentation du processus de sélection non itératif

une fois dans le résultat de la transcription des phrases. Dans ce processus, les autres mots du dictionnaire temporaire jouent le rôle d'un modèle de rejet quand nous essayons de reconnaître le nom propre évalué. Ce processus est illustré figure 7.4.

Ce procédé possède un inconvénient : il est très coûteux en terme de temps de calcul. Pour un nom propre possédant  $v$  variantes, et présent dans  $s$  segments, il faut décoder l'équivalent de  $v \times s$  segments pour valider ou non l'ensemble des variantes de phonétisations de ce nom propre.

Ce procédé a donc été abandonné au profit du filtrage itératif. Ce dernier permet d'obtenir un important gain en terme de temps de calcul et un léger gain en terme de WER et PNER par rapport à cette méthode non itérative.

**Filtrage itératif** Tout le corpus d'entraînement est décodé en utilisant un dictionnaire de phonétisation des noms propres et le dictionnaire de phonétisation des autres mots. Les phonétisations n'ayant jamais permis de décoder le nom propre correspondant au bon endroit sont

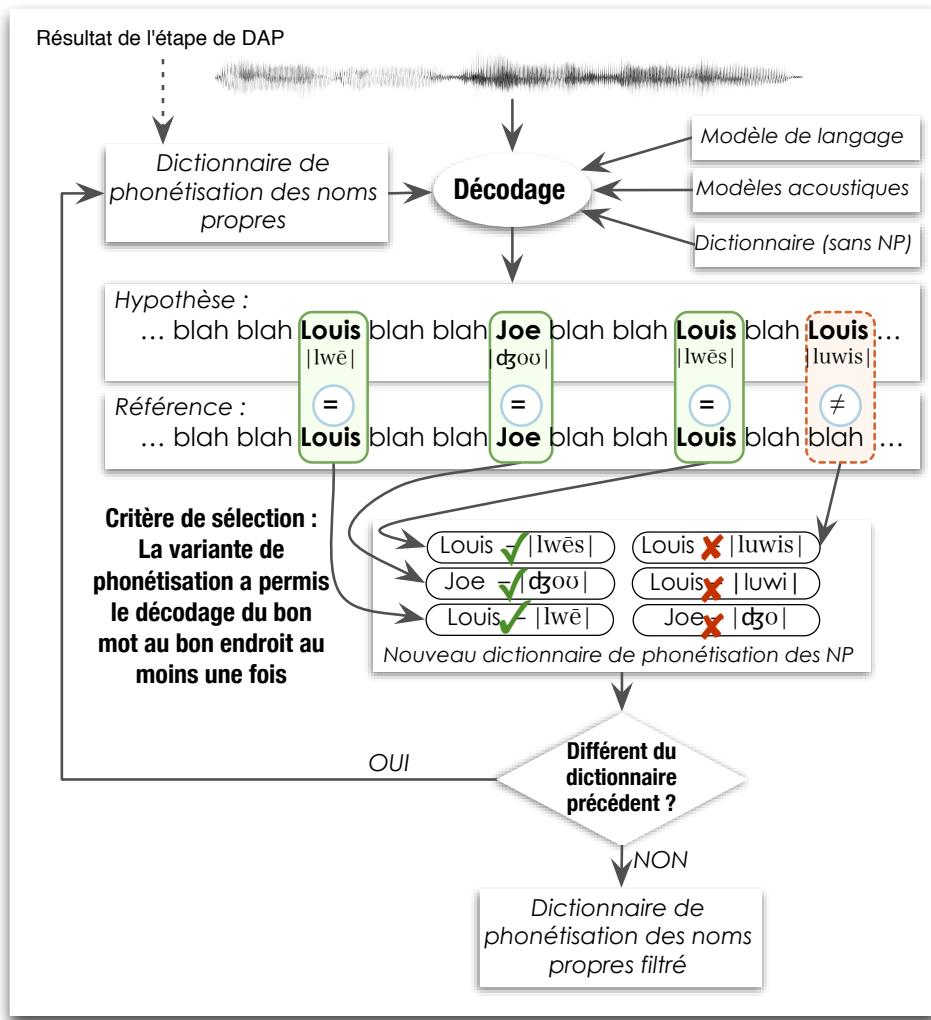


FIG. 7.5 – Représentation du processus de filtrage itératif

enlevées du dictionnaire de phonétisation des noms propres. Le processus est ensuite répété : le corpus est redécodé en utilisant le dictionnaire modifié, qui est ensuite filtré en utilisant le résultat du décodage. Le processus entier de décodage/filtrage est répété jusqu'à ce que plus aucune variante de phonétisation des noms propres ne soit supprimée du dictionnaire. Ce processus est illustré figure 7.5, en utilisant le même exemple que dans la figure 7.3.

## 7.5 Méthode itérative de génération des phonétisations

Nous voulons affiner itérativement le dictionnaire utilisé pour réaliser la phase d’alignement forcé entre le signal et la référence textuelle. Le dictionnaire issu de l’étape de filtrage est réutilisé pour réaliser à nouveau la phase d’alignement, première étape du processus d’extraction (voir figure 7.6).

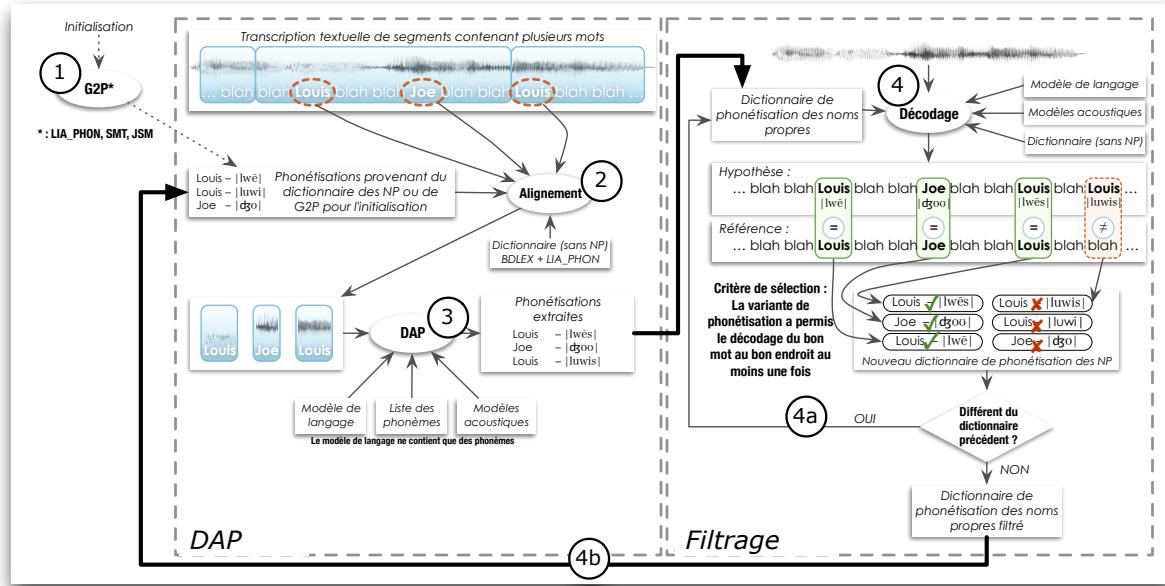


FIG. 7.6 – Processus complet d’extraction/filtrage des variantes de prononciation des noms propres

Le processus entier (alignement/DAP/filtrage) est répété jusqu’à avoir deux fois le même dictionnaire filtré en fin de processus (*i.e.* lorsque dans une même itération, le dictionnaire utilisé pour réaliser l’alignement forcé est le même que celui issu de l’étape de filtrage).

### 7.5.1 Résumé de la méthode

La méthode est donc réalisée en 4 étapes :

- 1 La première étape consiste à phonétiser, en utilisant les trois méthodes de G2P (LIA\_PHON, SMT, JSM), l’ensemble des mots annotés comme étant des noms propres dans notre corpus,
- 2 Le dictionnaire des noms propres (1) ainsi que le dictionnaire des autres mots (dictionnaire de référence : BDLEX + LIA\_PHON) sont utilisés pour réaliser l’alignement des noms propres sur le signal, de façon à déterminer les temps de début et de fin de chacun de ces mots,

- 3 Les parties du signal qui correspondent aux noms propres sont décodées en mode acoustico-phonétique (le vocabulaire et le modèle de language ne contiennent que des phonèmes), le résultat du décodage de chacune de ces parties du signal est considéré comme une phonétisation possible du nom propre en question. Un nouveau dictionnaire des noms propres (2) contenant chacune de ces phonétisations est créé.
- 4 Ce nouveau dictionnaire des noms propres (2) est ensuite filtré afin d'éliminer les phonétisations superflues. Le dictionnaire des noms propres (2) ainsi que le dictionnaire des autres mots sont utilisés pour décoder le corpus. Les phonétisations ayant permis de décoder le bon nom propre au bon endroit au moins une fois sont ajoutées au dictionnaire des noms propres filtré (3). Si les dictionnaires (2) et (3) sont différents :
  - 4a Le dictionnaire filtré (3) est utilisé comme nouveau dictionnaire des noms propres (2), le processus reprend à l'étape 4.
  - Si non :
  - 4b Si les dictionnaires (1) et (3) sont différents : le dictionnaire filtré (3) est utilisé comme nouveau dictionnaire des noms propres (1) et le processus redémarre à l'étape 2.
  - 4c Si les dictionnaires (1) et (3) sont identiques : le processus est terminé.

La méthode proposée ici nécessite de disposer d'un système de conversion graphème vers phonème pour l'étape d'initialisation (étape 1). Les résultats présentés dans la partie suivante mettent en avant l'influence de la méthode de phonétisation employée dans cette étape.



# Chapitre 8

## Expériences et résultats

### Sommaire

---

<b>8.1</b>	<b>Expériences</b>	<b>88</b>
8.1.1	Corpus	88
8.1.2	Modèles acoustiques et linguistiques	88
8.1.3	Métrique	90
<b>8.2</b>	<b>Résultats</b>	<b>91</b>
8.2.1	Nombre de variantes de phonétisation par nom propre	91
8.2.2	En utilisant une seule itération globale (alignement / extraction / filtre) . . . . .	91
8.2.3	En utilisant le processus itératif complet	93

---

**L**es travaux expérimentaux ont débuté en 2007, au début de cette thèse. La méthode de sélection des variantes de phonétisations que nous avions proposée en premier lieu était très coûteuse en terme de temps de calcul. Pour réaliser ces expériences, nous avons donc parallélisé les calculs, utilisé toutes les ressources que nous avions à notre disposition, et réduit la phase de décodage à son strict minimum. Ce décodage a donc été réalisé en une seule passe et utilise la segmentation de référence.

## 8.1 Expériences

### 8.1.1 Corpus

Les expériences ont été menées sur le corpus ESTER 1 (voir partie 1.2). Le corpus d'apprentissage est composé de 81 heures de données enregistrées depuis 4 radios : France Inter, France Info, RFI et RTM. Le corpus de développement est composé de 12,5 heures de données provenant des mêmes radios. Le corpus de test, utilisé pour évaluer la méthode, contient 10 heures d'émissions radiophoniques provenant de 6 radios : France Inter, France Info, RFI, RTM, France Culture et Radio Classique.

Pour extraire et filtrer les variantes de phonétisations, nous avons utilisé le corpus d'apprentissage plus le corpus de développement (environ 93 heures). SMT et JSM ont été entraînés sur le corpus d'apprentissage d'ESTER 1. La méthode SMT a été développée avec le corpus ESTER 2 en 2009, mais a été réapprise sur ESTER 1 pour ne pas induire de biais dans les expériences présentées ici. Chaque corpus était fourni annoté manuellement avec les entités nommées. Les mots considérés comme étant des noms propres sont ceux étiquetés en tant que “personne humaine”<sup>6</sup>.

### 8.1.2 Modèles acoustiques et linguistiques

Les modèles utilisés pour le décodage ont été appris sur les données de la campagne d'évaluation ESTER 1 [Galliano 2005].

**Modèles acoustiques** Les paramètres acoustiques utilisés dans les MMC sont des paramètres cepstraux MFCC. 13 MFCC sont extraits par trames de 25 ms avec recouvrement de 10 ms. Les vecteurs caractéristiques de chaque trame sont complétés avec les dérivées primaires et secondes des MFCC, générant des vecteurs acoustiques de 39 paramètres.

---

<sup>6</sup>Le guide d'annotation d'ESTER est disponible à l'adresse suivante : <http://trans.sourceforge.net/en/transguidFR.php>

Les modèles acoustiques ont été appris sur environ 81h d'émissions radiophoniques transcrrites manuellement. Ce corpus est composé de 73h de données large bande et de 8h de données bande étroite. Ces modèles acoustiques permettent de modéliser 35 phonèmes et 5 sortes de phonèmes représentant des évènements acoustiques (hésitation, musique, bruit, inspiration, silence) différents. Ils sont composés de 5500 états partagés, chaque état étant modélisé par un mélange de 22 gaussiennes diagonales. Le décodage utilise des états partagés tri-phone (en contexte) avec prise en compte de la position du phonème dans le mot (début, fin, milieu ou isolé).

**Modèles de langage** Les modèles de langage qui ont servi lors de la campagne d'évaluation ESTER 1 ont été appris sur trois ensembles de données différents :

- Les transcriptions manuelles des 93,5 heures d'émissions radiophoniques fournies par ESTER (81h du corpus d'apprentissage + 12,5h du corpus de développement). Ces transcriptions sont composées d'environ 1,35 million de mots, dont 34000 différents.
- Des articles du journal “Le Monde” datant de 2003, comprenant environ 19 millions de mots, dont 220000 différents.
- Des articles du journal “Le Monde” de 1987 à 2002, comprenant environ 300 millions de mots.

Trois modèles *3-gram* ont été appris. Un modèle pour les 81h du corpus d'apprentissage d'ESTER 1 ( $M_1$ ), puis un modèle *3-gram* pour chacune des deux autres sources de données provenant du journal “Le Monde” ( $M_2$  et  $M_3$ ). Les valeurs des coefficients d'interpolation linéaire  $\alpha$  et  $\beta$  (voir 8.1) à appliquer pour obtenir un seul modèle trigramme minimisant la perplexité sur les 12,5h restantes (corpus de développement) ont ensuite été recherchées. La probabilité d'apparition du mot  $w_i$  sachant l'historique  $h_i$  dans le modèle de langage interpolé est donné par la formule suivante :

$$P(w_i|h_i) = (1 - \alpha - \beta)P_{M_1}(w_i|h_i) + \alpha P_{M_2}(w_i|h_i) + \beta P_{M_3}(w_i|h_i) \quad (8.1)$$

où  $(\alpha + \beta) \leq 1$  et  $\alpha \leq 1$  et  $\beta \leq 1$  et  $\alpha \geq 0$  et  $\beta \geq 0$ .

Le modèle de langage inclut tous les noms propres présents dans le corpus de développement.

**Vocabulaire** L'ensemble des 34000 mots présents dans les transcriptions manuelles a été incorporé dans le vocabulaire. Les mots apparaissant plus de 10 fois dans les articles du journal “Le Monde” de 2003 (environ 19000 mots) ont également été conservés. Les mots les plus fréquents provenant du journal “Le Monde” de 1987 à 2002 ont été ajoutés pour atteindre la

Le dictionnaire de référence du système contient les phonétisations extraites du dictionnaire *expert BDLEX*, complété, pour les mots non présents dans ce dictionnaire, par l'utilisation de *LIA\_PHON*.

Dans les expériences présentées dans cette partie, tous les dictionnaires contiennent les mêmes noms propres, seules leurs phonétisations varient. L'apprentissage et le filtrage des nouvelles phonétisations a été réalisé sur le corpus d'apprentissage plus le corpus de développement d'ESTER 1. Aucun des 3348 noms propres présents dans notre corpus d'apprentissage (corpus d'apprentissage + corpus de développement d'ESTER 1) n'est présent dans le dictionnaire BDLEX ce qui veut dire que l'ensemble des noms propres du dictionnaire de référence a été phonétisé avec *LIA\_PHON*.

Les résultats présentant dans cette partie l'utilisation du système de G2P *LIA\_PHON* pour générer les phonétisations des noms propres sont donc identiques à ceux du système de référence.

### 8.1.3 Métrique

Nous proposons d'évaluer la qualité des phonétisations des noms propres créés en terme de taux d'erreur mot (Word Error Rate – WER) et de taux d'erreur nom propre (PNER – Proper Noun Error Rate). Le PNER est calculé de la même manière que le taux d'erreur mot classique utilisé en Reconnaissance Automatique de la Parole à la différence qu'il n'est pas appliqué à l'ensemble des mots, mais uniquement aux noms propres :

$$PNER = \frac{I + S + E}{N} \quad (8.2)$$

avec  $I$  le nombre d'insertions erronées de noms propres,  $S$  le nombre de substitutions de noms propres par un autre mot (ou un autre nom propre),  $E$  le nombre d'élisions de noms propres (c'est-à-dire le nombre de noms propres "supprimés" dans la transcription) et  $N$  le nombre total de noms propres.

Le WER permet d'évaluer l'impact du dictionnaire sur l'ensemble du corpus de test, alors que le PNER permet d'évaluer la qualité de la détection des noms propres.

## 8.2 Résultats

### 8.2.1 Nombre de variantes de phonétisation par nom propre

Le tableau 8.1 présente le nombre de phonétisations générées pour les noms propres présents dans les corpus de développement et d'apprentissage, pour chaque système de transcription phonétique. Les corpus de développement et d'apprentissage d'ESTER 1 contiennent 3348 noms propres distincts, apparaissant 28866 fois.

La colonne “généré” montre le nombre de variantes de phonétisations générées par chacun des systèmes de G2P. La colonne “extrait” correspond au nombre de phonétisations qui ont été extraites du signal sonore à l'aide du DAP en utilisant chacun des systèmes de G2P pour la phase d'alignement forcé. La colonne “filtré” contient le nombre de prononciations présentes dans chacun des dictionnaires filtrés à l'issu de la première itération d'alignement/DAP/filtrage. Quand toutes les itérations de filtrage (alignement/DAP/filtrage-alignement/DAP/filtrage-...) sont réalisées, le nombre de variantes du dictionnaire filtré final décroît légèrement (de 6776 à 6502 pour LIA\_PHON, de 7065 à 6802 pour SMT et de 6876 à 6708 pour JSM).

TAB. 8.1 – *Nombre de variantes de phonétisation*

Dictionnaire	Généré	Extrait	Filtré
LIA_PHON	4364	20218	6776
SMT	7031	20184	7065
JSM	3626	20008	6876

SMT génère plus de variantes que les autres méthodes, cependant le nombre de variantes extraites par le DAP reste proche quelque soit le système de G2P utilisé. Cela s'explique par le fait que le dictionnaire généré n'est utilisé que lors de la phase d'alignement permettant de déterminer les frontières des noms propres.

Le DAP extrait en moyenne 4,34 fois le nombre de variantes de phonétisations générées par les différents systèmes de G2P. La méthode de filtrage permet toujours de garder environ 7000 variantes de phonétisations sur les 20000 variantes extraites. (*i.e.* la méthode de filtrage permet d'éliminer environ 13000 variantes de phonétisations quelque soit la méthode de G2P utilisée pour initialiser le système).

### 8.2.2 En utilisant une seule itération globale (alignement / extraction / filtrage)

Les résultats présentés dans les figures 8.1, 8.2 et 8.6 montrent l'intérêt de l'étape de filtrage. En effet, les phonétisations extraites à l'aide du DAP permettent d'observer un gain en terme

de PNER et de WER sur les segments contenant des noms propres (figures 8.1 et 8.2), mais dégradent le taux d'erreur mot global du système (figure 8.6). De plus, les gains obtenus en utilisant les dictionnaires extraits ne sont pas aussi importants que ceux observés en utilisant les dictionnaires après filtrage. L'extraction des phonétisations à l'aide du DAP génère un très grand nombre de variantes de phonétisations. Ce nombre élevé permet de décoder un nombre plus important de noms propres correctement, mais génère également du bruit, faisant privilégier au décodeur certains noms propres par rapport à d'autres. Certains noms propres extraits ont une prononciation proche d'autres mots du vocabulaire. Le système fait donc des erreurs en décodant des noms propres à la place de mots n'en étant pas.

Pour chacun des trois systèmes de phonétisation automatique, la phase de filtrage est réalisée en trois itérations. Le dictionnaire généré à l'issu de la quatrième itération de filtrage est identique à celui de la troisième, ce qui signale la fin du processus de filtrage. Dans cette partie, nous comparons l'utilisation directe des méthodes de G2P utilisées avec la méthode proposée.

La figure 8.1 permet de comparer les PNER de chaque méthode de G2P avec le PNER obtenu en utilisant la méthode de filtrage sur le corpus de test d'ESTER 1.

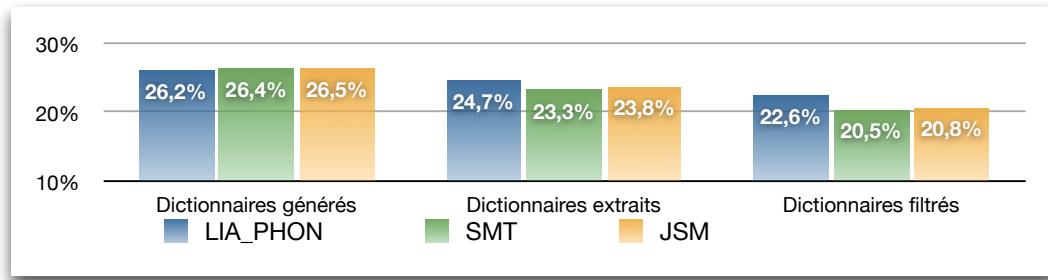


FIG. 8.1 – PNER en utilisant chacune des méthodes de G2P (Corpus de test ESTER 1)

Ces résultats montrent que l'utilisation de notre méthode, initialisée avec tous les systèmes de G2P testés, permet d'obtenir des gains significatifs en terme de PNER. Comme nous pouvons le remarquer, la méthode de DAP utilisant le système basé sur SMT pour initialiser le dictionnaire de phonétisation est celle qui donne le meilleur PNER. Le PNER obtenu en utilisant directement les phonétisations générées par LIA\_PHON (référence) est de 26,2%. L'utilisation de SMT dans le processus d'extraction/filtrage proposé permet d'obtenir un PNER de 20,5% (soit un gain de 5,7 points).

La figure 8.2 permet de comparer les résultats de LIA\_PHON (référence) avec SMT et JSM en terme de WER calculé sur l'ensemble des segments contenant des noms propres. Nous avons décidé de présenter l'impact de ces différents dictionnaires sur le décodage des segments contenant des noms propres, puisque c'est sur ce type de segments qu'il est le plus significatif.

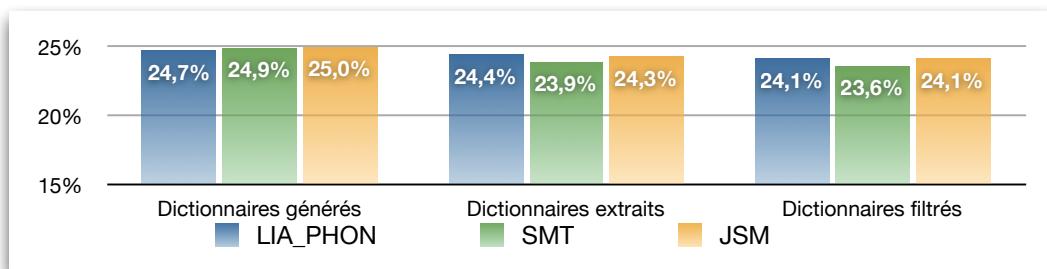


FIG. 8.2 – WER sur le corpus de test sur les segments contenant des noms propres

SMT, bien que n'étant pas le dictionnaire permettant d'observer le plus faible WER utilisé directement, permet d'obtenir les meilleurs WER et PNER lorsqu'il sert de dictionnaire d'alignement au processus de DAP.

LIA\_PHON permet d'observer un WER d'environ 24,7% lors du décodage des segments contenant des noms propres. L'utilisation de SMT dans le processus d'extraction/filtrage permet d'obtenir un WER d'environ 23,6% (soit un gain de 1,1 point).

### 8.2.3 En utilisant le processus itératif complet

Le processus itératif complet est réalisé en trois itérations avec les trois systèmes de G2P testés : le dictionnaire filtré itérativement lors de la quatrième itération globale est identique à celui de la troisième itération globale. Les figures 8.3 et 8.4 montrent les résultats obtenus en utilisant le processus itératif complet avec les systèmes de G2P LIA\_PHON, SMT et JSM.

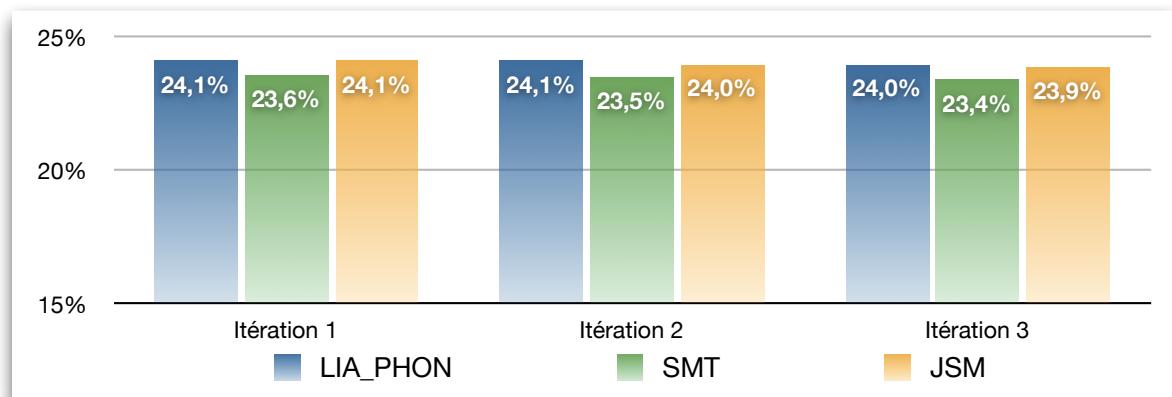


FIG. 8.3 – WER utilisant le processus itératif complet (corpus de test ESTER 1, segments contenant des noms propres)

Les WER et PNER sont calculés sur les segments contenant des noms propres. Nous pouvons observer un faible gain entre l'utilisation des dictionnaires issus de la première passe de

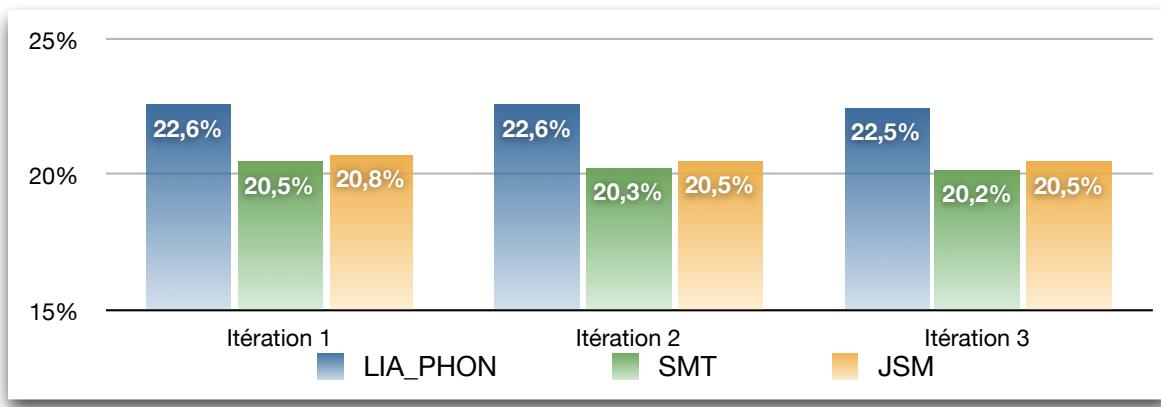


FIG. 8.4 – PNER en utilisant le processus itératif complet (corpus de test ESTER 1, segments contenant des noms propres)

DAP et ceux issus de la dernière. En utilisant LIA\_PHON pour initialiser notre méthode, le WER est descendu de 24,1% à 24% et le PNER est descendu de 22,6% à 22,5%.

SMT permet d'observer un gain de 0,2% en terme de WER et un gain de 0,3% en terme de PNER. L'utilisation de JSM donne, en terme de WER, des résultats semblables à ceux observés avec LIA\_PHON (24,1% à 23,9%). Le PNER décroît légèrement entre l'itération 1 et l'itération 2 (-0,3 point) puis reste constant ensuite.

La figure 8.5 présente les résultats obtenus en utilisant le dictionnaire de référence (BDLEX + LIA\_PHON) en comparaison avec le meilleur des dictionnaires construits grâce à la méthode (processus de DAP initialisé avec le dictionnaire SMT). Les gains sont significatifs (-1,3 point de WER et -6 points de PNER).

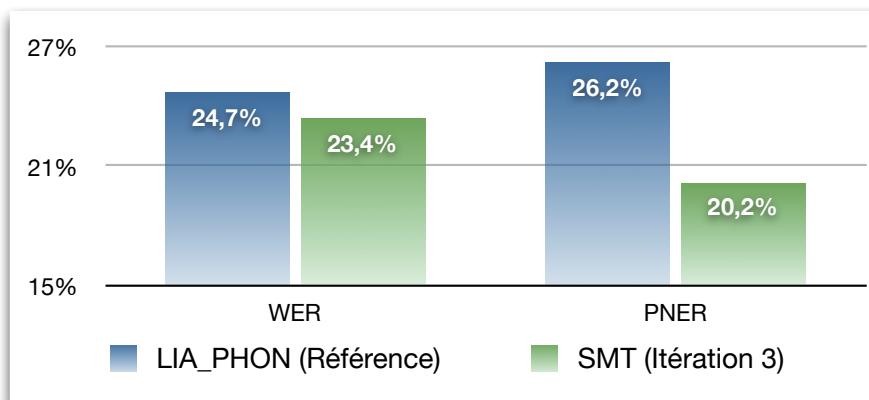


FIG. 8.5 – PNER et WER en utilisant le processus itératif complet (corpus de test ESTER 1, segments contenant des noms propres)

Les résultats (voir tableau 8.2) des trois systèmes de G2P utilisés directement sont sensiblement identiques (24,7%, 24,9% et 25,0% en terme de WER sur le corpus de test pour, respectivement, LIA\_PHON, SMT et JSM, et 26,2%, 26,4% et 26,5% en terme de PNER). Pourtant les gains en utilisant la méthode basée sur SMT sont meilleurs dès la première itération d'extraction et de filtrage des phonétisations.

TAB. 8.2 – Résumé des résultats obtenus sur le corpus de test d'ESTER 1

Dictionnaire	WER (segment contenant NP)	PNER
<b>LIA_PHON (ref)</b>	<b>24,7%</b>	<b>26,2%</b>
SMT	24,9%	26,4%
JSM	25%	26,5%
Première itération de filtrage		
LIA_PHON	24,1%	22,6%
SMT	23,6%	20,5%
JSM	24,1%	20,8%
Seconde itération de filtrage		
LIA_PHON	24,1%	22,6%
SMT	23,5%	20,3%
JSM	24%	20,5%
Troisième itération de filtrage		
LIA_PHON	24%	22,5%
SMT	23,4%	20,2%
JSM	23,9%	20,5%

Cela peut s'expliquer par le fait que SMT est la méthode de G2P qui génère, dans notre cas, le plus de variantes de phonétisation. Ce nombre élevé de variantes permet un meilleur alignement des phrases de la référence sur le signal sonore, tout en dégradant légèrement les WER (24,9% contre 24,7% pour LIA\_PHON) et PNER (26,4% contre 26,2% pour LIA\_PHON) lorsque SMT est utilisé seul.

La figure 8.6 présente les WER sur l'ensemble des segments en utilisant directement les dictionnaires générés par les différentes méthodes de G2P et en utilisant les meilleurs dictionnaires extraits et itérativement filtrés par notre méthode.

Les taux sont pratiquement identiques. Nous pouvons néanmoins noter un très léger gain (0,1 point) de WER en utilisant le dictionnaire généré par notre méthode avec SMT.

Les phonétisations présentent dans le dictionnaire final ne correspondent pas forcément aux phonétisations qu'un expert humain proposerait. Certaines ne paraissent pas correctes lorsque nous les lisons, mais elles prennent en compte les distorsions dues aux imperfections des modèles acoustiques. Par exemple, l'une de phonétisations du nom du président de l'Autorité palestinienne Mahmoud Abbas est la suivante : "a a b d a e s" alors que ce nom est prononcé "a

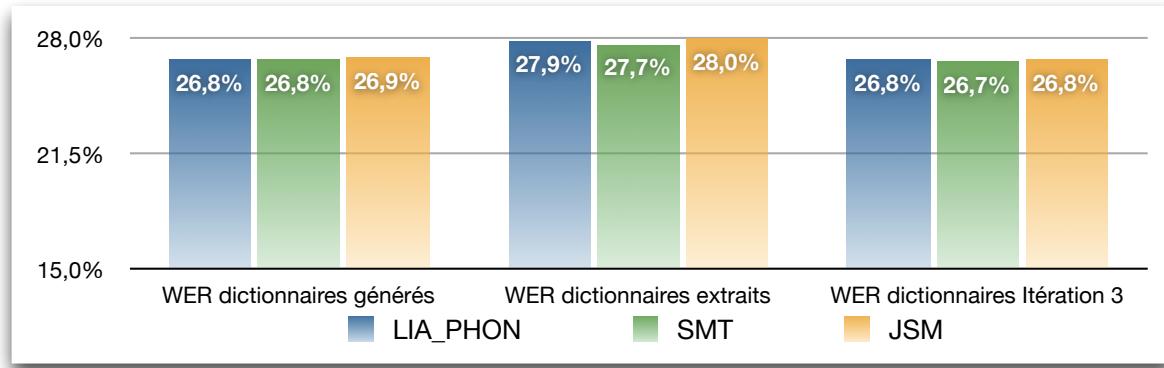


FIG. 8.6 – WER sur l’ensemble des segments sur le corpus de test ESTER 1

R k b a s" (format Sampa) dans les données audio. Les mots phonétisés par cette approche ne doivent pas être réutilisés avec la phonétisation proposée dans le cas où de nouveaux modèles acoustiques seraient estimés : les phonétisations sont dépendantes de la modélisation acoustique utilisée lors de l’alignement phonèmes/signal.

## **Chapitre 9**

### **Conclusion : Phonétisation automatique**

**U**ne méthode itérative, utilisant les informations acoustiques disponibles pour phonétiser automatiquement les noms propres a été proposée. Notre choix s'est porté sur la mise en place d'une méthode basée sur l'acoustique de façon à extraire les prononciations utilisées par les locuteurs.

Les noms propres étant souvent prononcés d'une manière ne répondant pas aux conventions classiques de prononciation de la langue, l'utilisation de méthodes basées sur des règles de prononciation est moins efficace pour cette classe de mots, et en particulier pour les noms propres étrangers.

Par rapport aux méthodes de phonétisation utilisant des données acoustiques, présentées dans l'état de l'art de cette partie, la stratégie a été appliquée à la seule classe des noms propres. Les données acoustiques disponibles sont utilisées sans nécessiter d'actions de la part de l'utilisateur (répéter). La méthode de filtrage itérative proposée permet de retirer les phonétisations superflues. Les résultats présentés montrent l'intérêt de cette phase dans le processus. La méthode a été évaluée en terme de taux d'erreur mots (WER) et de taux d'erreur noms propres (PNER), alors que la plupart des méthodes présentées dans l'état de l'art évalue la qualité des phonétisations produites en terme de taux d'erreur phonèmes (PER). L'utilisation des phonétisations extraites puis filtrées permet d'observer des gains en terme de WER et PNER. Pour obtenir le meilleur dictionnaire de phonétisations, il a été nécessaire de décoder 16 fois (4 itérations complètes x 4 itérations de filtrage) l'ensemble des segments du corpus d'apprentissage contenant des noms propres. Le temps de calcul est donc important, pour un gain pouvant sembler faible. Pour ce qui est de la catégorie des noms propres, cette méthode reste néanmoins intéressante, puisqu'elle permettra d'indexer les réunions en fonction des noms des personnes citées. La méthode proposée utilise le corpus d'ESTER 1 annoté et un alignement forcé réalisé dans une étape d'initialisation, pour extraire les zones du signal correspondant aux noms propres. Un dictionnaire de phonétisation est nécessaire pour réaliser cette première étape. Trois méthodes de G2P ont été testées pour cette phase d'initialisation. Il s'agit du système à base de règles de prononciations LIA\_PHON [Béchet 2001], d'une méthode basée sur l'utilisation de Statistical Machine Translation (SMT) [Laurent 2009], et de l'utilisation de la méthode à base de modèles à séquences jointes (JSM) proposée dans [Bisani 2008]. Une fois l'alignement réalisé, les parties du signal correspondant aux noms propres sont décodées en mode acoustico-phonétique. Le résultat de chaque décodage est considéré comme une phonétisation possible du nom propre en question. Les phonétisations extraites sont ensuite filtrées de façon à éliminer les phonétisations susceptibles de perturber le SRAP. Le dictionnaire filtré est utilisé pour réinitialiser le processus (phase d'alignement), et le procédé complet est répété jusqu'à l'obtention de deux dictionnaires filtrés identiques.

Les résultats sont présentés dans le tableau 9.1.

TAB. 9.1 – *Résultats obtenus sur le corpus de test d'ESTER 1*

Dictionnaire	WER (segment contenant NP)	PNER
<b>Référence</b>	<b>24,7%</b>	<b>26,2%</b>
Méthode initialisée avec LIA_PHON	24% (-0,7)	22,5% (-3,7)
Méthode initialisée avec SMT	23,4% ( <b>-1,3</b> )	20,2% ( <b>-6</b> )
Méthode initialisée avec JSM	23,9% (-0,8)	20,5% (-5,7)

L'utilisation des dictionnaires de noms propres résultant du processus complet de phonétisation permet d'observer des gains en terme de PNER et de WER sur le corpus de test d'ESTER 1. Les meilleurs résultats ont été obtenus en utilisant le système de G2P basé sur SMT pour initialiser le processus. Le WER sur les segments contenant des noms propres a baissé de 1,3 point (de 24,7% à 23,4%) et le PNER de 6 points (de 26,2% à 20,2%). Le WER calculé sur l'ensemble du corpus de test d'ESTER 1 a été très légèrement amélioré (de 26,8% à 26,7%).



# **Chapitre 10**

## **Conclusion et perspectives**

## **10.1 Réordonnancement automatique des hypothèses de re-connaissance**

### **10.1.1 Méthode proposée**

Dans ce manuscrit, une stratégie permettant à un transcriveur humain de collaborer avec une méthode de réordonnancement automatique des hypothèses proposées par un système de reconnaissance automatique de la parole (SRAP) a été présentée. Les travaux de [Bazillon 2008b] ont montré l'intérêt d'utiliser un SRAP pour améliorer la productivité des transcriveurs. Le processus de transcription est réalisé en deux étapes : le SRAP produit une hypothèse de transcription qui est ensuite corrigée manuellement. Le SRAP du LIUM génère, en plus de la meilleure hypothèse de transcription, un réseau de confusion. Ce type de graphe conserve les hypothèses du graphe de mots les plus probables. Le principe de la méthode proposée consiste à faire en sorte que chaque action corrective soit suivie d'une réorganisation des hypothèses contenues dans le réseau de confusion, de façon à proposer une transcription alternative à l'utilisateur. Cette méthode permet d'observer des gains en terme de nombre d'actions nécessaires au correcteur (KSR) pour parvenir à une transcription ne contenant plus d'erreurs, ainsi qu'en terme de taux de correction sur les mots (WSR). Néanmoins, ces métriques présentent une limite, qui est la prise en compte de la pénibilité de la tâche pour l'utilisateur.

### **10.1.2 Perspectives**

Il va falloir, dans un premier temps, développer une application d'assistance à la correction des transcriptions, puis évaluer, dans des conditions réelles, les gains apportés par la méthode. Un autre aspect consistera à étendre la fenêtre de réordonnancement. Pour l'instant, une correction de la part de l'utilisateur permet une réévaluation du segment sur lequel portait cette correction. Il faudrait être en mesure de propager la correction plus loin dans la transcription, sans nécessiter d'action de la part de l'utilisateur. Afin d'améliorer la méthode de réordonnancement automatique, il faudrait mettre en œuvre une stratégie permettant d'ajouter de nouveaux mots dans le réseau de confusion à partir duquel les nouvelles hypothèses sont formulées, et, plus généralement, au vocabulaire du SRAP. Nous avons proposé une stratégie locale d'adaptation de la transcription et il faudrait trouver un moyen de profiter globalement des corrections des transcriptions pour améliorer le système de façon plus générale, d'un point de vue acoustique et linguistique. Pour ce qui est de la partie linguistique, les travaux de [Allauzen 2003] et de [Oger 2008] contiennent un certain nombre de pistes. Ces derniers proposent des stratégies basées sur l'utilisation de données provenant d'Internet et l'utilisation de métadonnées pour

l’adaptation thématique des lexiques et modèles de langage. Concernant l’application visée par la société Spécinov, il s’agira de mettre en place des techniques permettant de concevoir des modèles propres au vocabulaire et à la façon de parler de ses utilisateurs. Pour ce qui est des techniques d’adaptation acoustique, nous avons présenté un certain nombre d’entre elles dans la première partie de ce manuscrit ; il faudrait maintenant trouver des stratégies globales pour les mettre en oeuvre. Ces dernières devront pouvoir s’appliquer sans nécessiter de contrôles humains, de façon automatique et transparente pour les utilisateurs, n’ayant a priori pas de connaissances sur les principes de fonctionnement du système.

## 10.2 Phonétisation automatique des noms propres

### 10.2.1 Méthode proposée

Une méthode de phonétisation automatique basée sur l’utilisation d’un décodeur acoustico-phonétique couplé avec une technique de filtrage permettant d’éliminer les variantes de phonétisation superflues a été proposée. Cette méthode présente l’intérêt de s’appuyer sur les données acoustiques acquises pour proposer une phonétisation proche de celle réellement utilisée par les locuteurs. Appliquée aux noms propres, la méthode proposée permet d’observer, d’une part, des gains en terme de taux d’erreur mot sur les segments les contenant, et d’autre part, une diminution du taux d’erreur sur les noms propres eux-mêmes. La société Spécinov, qui a financé ce doctorat, souhaite développer des applications intégrant le système de reconnaissance automatique de la parole du LIUM. Elle souhaite pouvoir proposer à ses clients une application d’aide à la création de comptes-rendus de réunions. Le fait d’être capable de décoder les noms propres est quelque chose d’important, puisque couplé avec un système d’indexation cela permet de trouver de façon pertinente les réunions dans lesquelles le nom de M. X a été prononcé, sans que la correction manuelle n’ait nécessairement été réalisée.

### 10.2.2 Perspectives

Cette méthode pourrait être généralisée à l’ensemble des mots du vocabulaire. Une expérience très rapide, effectuée il y a un an environ, avec uniquement l’utilisation de LIA\_PHON pour la phase d’alignement, n’a pas montré de bons résultats. Cette piste reste tout du moins intéressante, car elle pourrait être appliquée aux mots dont le décodage pose le plus souvent problème. Il pourrait être imaginé une application qui auto-évalue les taux de reconnaissance de chacun des mots présents dans les transcriptions déjà corrigées et qui déclenche la méthode afin d’extraire automatiquement de nouvelles phonétisations pour les mots fréquemment erronés.

Maintenant que la phase de filtrage des variantes de phonétisations extraites est réalisée en un temps raisonnable, il faudrait réaliser à nouveau toutes les expériences en utilisant le nouveau SRAP du LIUM.

Puisque la méthode permet une amélioration du taux d'erreur noms propres, elle pourrait être également utilisée dans les travaux de [Jousse 2008] qui traitent de l'identification nommée du locuteur.

## **10.3 Perspectives générales**

Dans ce manuscrit, une méthode de phonétisation et une méthode d'aide à la correction de transcriptions sont proposées. Ces deux stratégies vont pouvoir être intégrées dans une application commerciale aidant à la gestion des comptes-rendus de réunions. La société Spécinov souhaite développer une application permettant de traiter les réunions téléphoniques et les réunions classiques. Pour ce qui est des réunions classiques, il va falloir trouver des solutions pour instrumentaliser au mieux les salles de réunion ou procéder à des enregistrement en utilisant les microphones des ordinateurs portables présents dans la salle (comme dans CALO [Voss 2007]). Pour ce qui est des réunions téléphoniques, certains ajustements du SRAP sont à prévoir. De façon à intégrer le vocabulaire de la société ayant acheté le produit, il sera proposé aux utilisateurs d'importer leurs documents existants dans l'application, à partir desquels des modèles pourront être appris et interpolés avec ceux du système. Disposant de retours réels d'utilisation, il sera possible de mettre en place des méthodes d'adaptations plus globales. Il sera nécessaire d'adapter le vocabulaire du système aux utilisateurs et de prendre en compte l'environnement acoustique dans lequel l'application sera utilisée. De façon à pouvoir produire des synthèses des réunions, il faudra également mettre en place des méthodes de résumé automatique. Dans ce domaine, le LIA s'est illustré en développant un moteur de résumé multi-documents à partir de la fusion de cinq systèmes de recherche de passages et de résumé automatique générique [Favre 2006] dont le système Neo-cortex [Boudin 2007]. Il s'agira également de mettre en place un système d'indexation automatique permettant de rechercher des informations dans les documents transcrits.

# **Annexe A**

## **Applications pour la transcription manuelle**

### **Sommaire**

---

<b>A.1 Transcriber . . . . .</b>	<b>106</b>
<b>A.2 Praat . . . . .</b>	<b>107</b>
<b>A.3 WinPitch . . . . .</b>	<b>108</b>
<b>A.4 XTrans . . . . .</b>	<b>109</b>
<b>A.5 Conclusion . . . . .</b>	<b>111</b>

---

**L**E travail de transcription manuelle, aidé par un logiciel avec une interface adaptée, consiste à écouter l'enregistrement sonore, à effectuer une segmentation et à transcrire ce que l'on entend (en respectant certaines normes). Les informations sur le type d'enregistrement (studio, téléphone), le nom des locuteurs, certains phénomènes acoustiques (toux, rire, musique, etc.) et toutes les méta-information peuvent être annotées si le logiciel le permet. Dans cette annexe, nous présentons 4 outils de transcription manuelle de la parole.

## A.1 Transcriber

Le développement de Transcriber [Barras 1998] a été financé par la DGA (Délégation Générale pour l'Armement) et LDC (Linguistic Data Consortium). Ce logiciel a été utilisé pour transcrire les corpus ESTER 1 et ESTER 2 ainsi que pour la transcription de nombreux corpus du catalogue LDC.

Transcriber est un logiciel optimisé pour la transcription et l'annotation de corpus volumineux. Il propose quatre niveaux d'annotation (texte, locuteur, thème et bruits de fond). Il offre une gestion des locuteurs permettant d'indiquer, en plus de leur identité, des informations telles que leur sexe, le degré de spontanéité (parole préparée ou spontanée), le canal d'expression, la qualité de l'enregistrement, etc. Par ailleurs, un nombre important de balises est intégré pour représenter les événements sonores (bruit, respiration, toux, reniflement, etc), les prononciations particulières ou encore des particularités lexicales. L'utilisateur peut créer ses propres balises. Il offre la possibilité d'aligner le signal audio avec la transcription, et présente un affichage permettant d'accéder directement aux segments du fichier son que l'on souhaite écouter. Ce programme est gratuit<sup>7</sup>, open source, ergonomique, simple d'accès et sait traiter de nombreux formats, en entrée comme en sortie. Il peut gérer des fichiers audio de plusieurs heures.

Le gros inconvénient de Transcriber concerne la parole superposée qui est traitée de façon trop simplifiée pour pouvoir rendre compte de ce phénomène majeur de la langue parlée [Bazillon 2008b]. Transcriber ne permet pas de faire de traitements prosodiques.

Transcriber intègre un “manuel du transcripteur”, indépendant du manuel utilisateur, qui passe en revue de nombreux aspects de la langue orale en proposant à chaque fois un codage. Cela permet aux utilisateurs de réaliser rapidement des transcriptions complètes et unifiées. Dans la pratique, des conventions diverses sont nées au fil des projets ou des groupes de recherche qui ont vu le jour. Des initiatives telles que le LDC<sup>8</sup> ou la Text Encoding Initiative (TEI)<sup>9</sup> proposent également des conventions pour la transcription de la parole.

---

<sup>7</sup>Téléchargeable à l'adresse : <http://trans.sourceforge.net>

<sup>8</sup>[www.ldc.upenn.edu/Projects/MDE/Guidelines/SimpleMDE\\_V6.2.pdf](http://www.ldc.upenn.edu/Projects/MDE/Guidelines/SimpleMDE_V6.2.pdf)

<sup>9</sup>[www.tei-c.org](http://www.tei-c.org)

La figure A.1 représente une capture d'écran de l'interface de Transcriber.

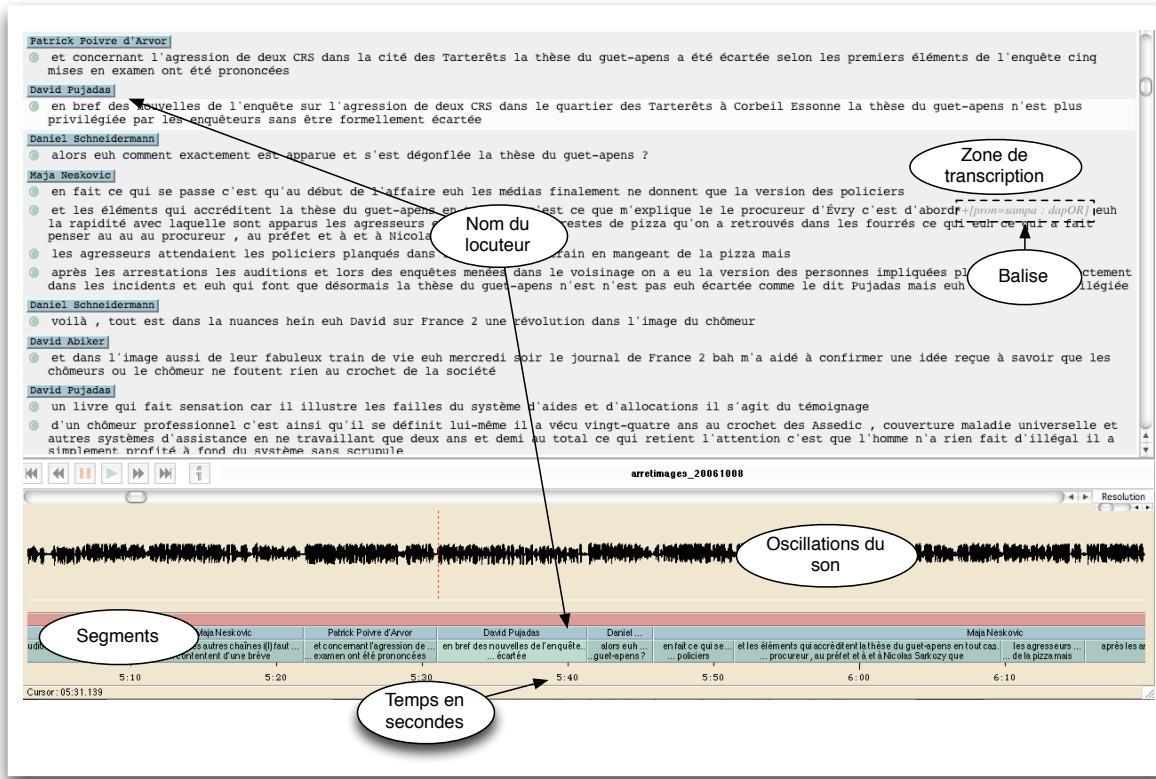


FIG. A.1 – Capture d'écran du logiciel Transcriber

## A.2 Praat

Praat a été conçu par Paul Boersma et David Weening, de l'université d'Amsterdam. Il est également distribué gratuitement<sup>10</sup>. Ce logiciel [Boersma 2001] permet entre autres choses de segmenter et de transcrire des fichiers audio, d'effectuer des analyses phonétiques et acoustiques, de manipuler le signal sonore, etc. Le logiciel, outre les traitements prosodiques standard, permet d'éditer des annotations indépendantes. Il est donc possible de disposer, pour une même zone de la transcription, d'une annotation orthographique, d'une annotation phonétique, etc. Cela permet également de gérer la parole superposée, en créant une annotation par locuteur. Praat intègre également un langage de script, ce qui permet de développer ses propres modules ou d'utiliser des modules complémentaires. Il est donc beaucoup plus complet, mais aussi beaucoup moins évident à utiliser que Transcriber. La figure A.2 montre l'interface du logiciel. Il faut naviguer entre plusieurs fenêtres, ce qui rend l'utilisation du logiciel complexe.

<sup>10</sup>et peut être téléchargé à l'adresse : <http://www.fon.hum.uva.nl/praat/>

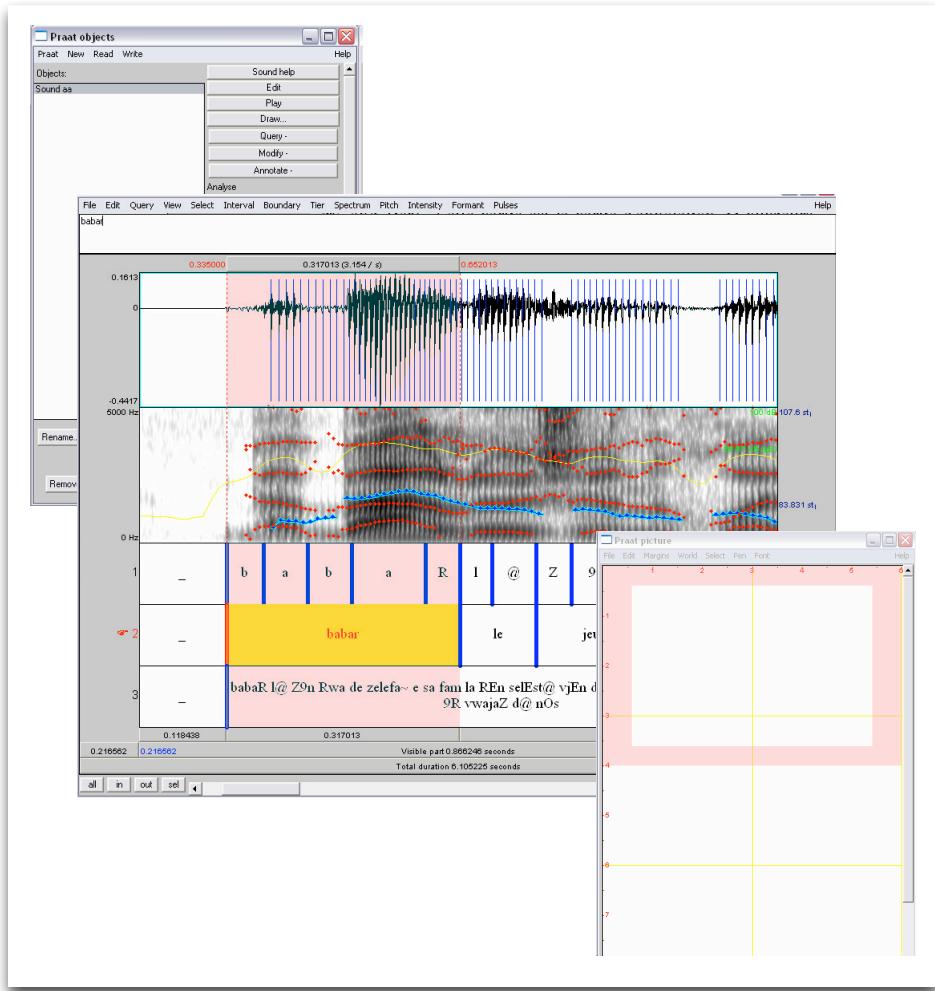


FIG. A.2 – Capture d'écran du logiciel Praat

Le logiciel dispose d'une fenêtre principale et de fenêtres annexes affichant le résultat des divers traitements, comme la stylisation de la fréquence fondamentale ou le tracé du spectre. Praat n'offre qu'une gestion minimale des locuteurs en ne permettant d'indiquer que leurs noms. En résumé, ce logiciel est bien moins efficace que Transcriber pour le traitement de corpus volumineux et est généralement privilégié pour des tâches spécifiques, comme pour l'analyse de la prosodie.

### A.3 WinPitch

Winpitch a été développé par Philippe Martin, de l'université de Paris 7. Ce logiciel [Martin 1996] ne fonctionne que sous Windows et n'est pas distribué librement.

Comme ses deux “concurrents”, Winpitch permet de transcrire orthographiquement et de segmenter le signal sonore. Il dispose en plus d’un système d’aide à la segmentation, et permet de styliser la fréquence fondamentale, de dessiner le spectre et de re-synthétiser le signal de parole. Cette re-synthèse permet de vérifier que la stylisation de la fréquence fondamentale est réaliste. Il est également capable de traiter les fichiers audio et vidéo, et permet une synchronisation entre l’image, le signal et la transcription.

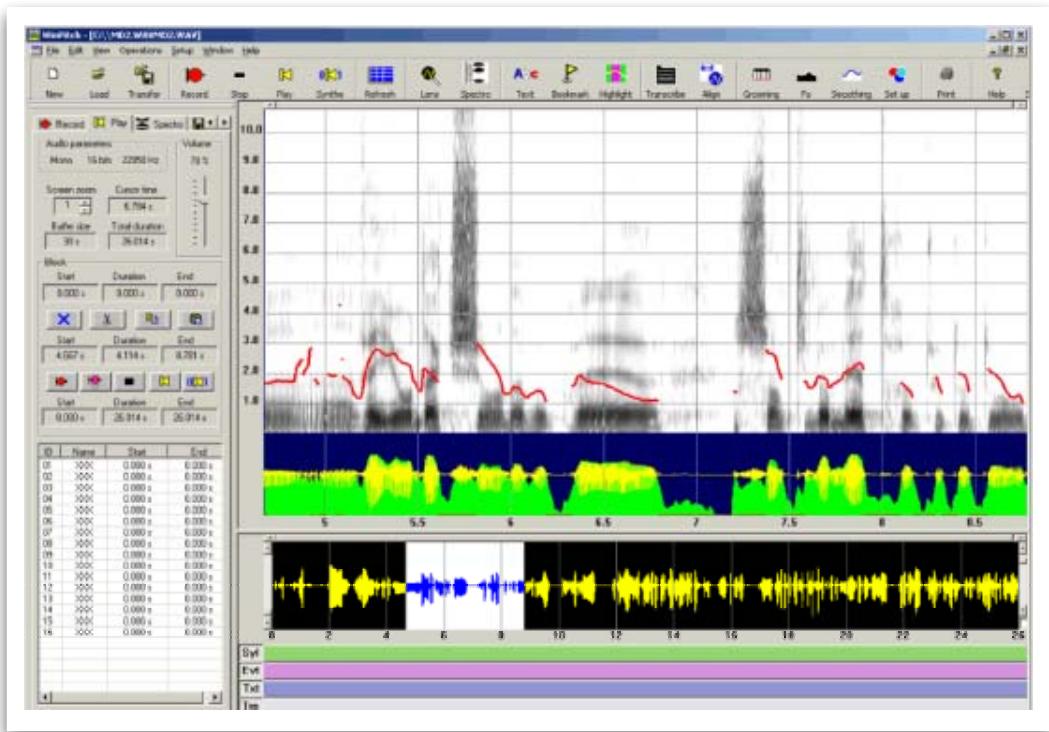


FIG. A.3 – Capture d’écran du logiciel Winpitch

Winpitch propose un grand nombre de fonctionnalités, et son interface surchargée le rend difficile d'accès. La figure A.3, issue du site officiel du logiciel<sup>11</sup>, montre une capture d'écran de WinPitch.

## A.4 XTrans

XTrans est l'outil de transcription nouvelle génération de LDC. Comme son prédecesseur Transcriber, il est distribué gratuitement<sup>12</sup>. XTrans intègre un système permettant de transcrire la parole superposée. Il est possible d'avoir plusieurs locuteurs qui parlent sur un même canal,

<sup>11</sup>[www.winpitch.com](http://www.winpitch.com)

<sup>12</sup><http://www.ldc.upenn.edu/tools/XTrans/downloads/>

## Annexe A. Applications pour la transcription manuelle

ou bien d'avoir différents canaux pour représenter les différents locuteurs. Le logiciel utilise le concept de *Virtual Speaker Channels* (VSC). Chaque VSC correspond à un locuteur. Plusieurs VSC peuvent être associés à un seul canal audio si plusieurs locuteurs sont enregistrés sur ce canal. Un VSC peut être associé à plusieurs canaux audio dans un même fichier son ou à plusieurs fichiers sons (un locuteur - un canal, un locuteur - plusieurs canaux, plusieurs locuteurs - plusieurs canaux). XTrans offre également la possibilité de transcrire du texte dont la lecture se fait de la droite vers la gauche, comme l'arabe ou l'hébreu. Le logiciel inclut aussi des fonctions d'annotations dans l'outil de transcription. L'outil d'annotation MDE (Metadata Extraction) permet de représenter les disfluences, les bruits de fond, les marqueurs de conversations et des unités sémantiques sur les transcriptions existantes. Les transcriptions réalisées à l'aide du logiciel Transcriber peuvent être importées dans XTrans. De la même façon, les transcriptions réalisées avec XTrans peuvent être exportées au format Transcriber. La figure A.4 présente une capture d'écran de XTrans utilisé pour transcrire un flux de parole en arabe, dans lequel interviennent sept locuteurs.

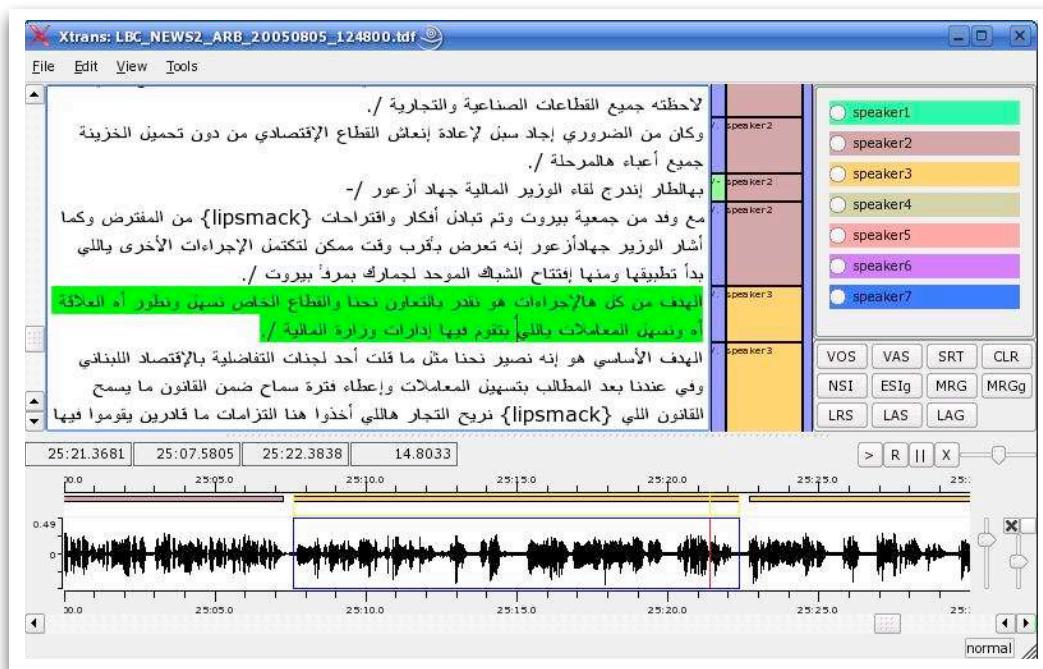


FIG. A.4 – Capture d'écran du logiciel XTrans

## A.5 Conclusion

Dans cette annexe, nous avons présenté des applications d'aide à la transcription manuelle, avec chacune des avantages et des inconvénients.

Les applications Winpitch et Praat offrent de très nombreuses fonctionnalités. Cependant, toutes deux présentent des interfaces surchargées, en particulier si l'on souhaite uniquement les utiliser pour réaliser des transcriptions. De plus, Winpitch ne fonctionne que sous Windows et n'est pas disponible gratuitement. Transcriber est bien adapté à la tâche de transcription. Ce logiciel est simple d'utilisation et offre de nombreuses possibilités d'ajout de méta-information. XTrans est le successeur de Transcriber. En plus des fonctions de ce dernier, il permet de gérer la parole superposée.

*Annexe A. Applications pour la transcription manuelle*

---

## **Acronymes**

## *Acronymes*

---

- AAC** Augmentative and Alternative Communication
- AFCP** Association Francophone de la Communication Parlée
- ANR** Agence Nationale de la Recherche
- ARC** Actions de Recherche Concertées
- ASH** Attelage de Systèmes Hétérogènes
- BE** Bande Étroite
- BIC** Bayesian Information Criterion
- BL** Bande Large
- BLEU** BiLingual Evaluation Understudy
- BSD** Berkeley Software Distribution
- CAT** Computer Assisted Translation
- CATS** Computer Assisted Transcription of Speech
- CE** Cross Entropy
- CIFRE** Conventions Industrielles de Formation par la REcherche
- CMLLR** Constrained Maximum Likelihood Linear Regression
- CMS** Cepstral Mean Substraction
- CMU** Canergie Mellon University
- DAP** Décodage Acoustico Phonétique
- DARPA** Defence Advanced Project Agency
- DER** Diarization Error Rate
- DGA** Délégation Générale de l'Armement
- ELDA** Evaluations and Language resources Distribution Agency
- EM** Expectation-Maximization
- EPAC** ExPloration de masse de documents Audio pour l'extraction et le traitement de la parole Conversationnelle
- ESTER** Évaluation des Systèmes de Transcription enrichie d'Émissions Radiophoniques
- G2P** Grapheme to Phoneme

- 
- GLR** Generalized Likelihood Ratio
- GMM** Gaussian Mixture Models
- ICP** Institut de la Communication Parlée – Grenoble
- IRISA** Institut de Recherche en Informatique et Systèmes Aléatoires – Rennes
- IRIT** Institut de Recherche en Informatique de Toulouse
- JSM** Joint Sequence Models
- KSR** Keystroke Saving Rate
- LADL** Laboratoire d’Automatique Documentaire et Linguistique
- LDA** Linear Discriminant Analysis
- LDC** Linguistic Data Consortium
- LIA** Laboratoire Informatique d’Avignon
- LIF** Laboratoire d’Informatique Fondamentale – Marseille
- LIMSI** Laboratoire d’Informatique pour la Mécanique et les Sciences de l’Ingénieur
- LORIA** Laboratoire Lorrain de Recherche en Informatique et ses Applications
- LIUM** Laboratoire d’Informatique de l’Université du Maine
- MAP** Maximum *A Posteriori*
- MDE** Metadata Extraction
- MFCC** Mel Frequency Cepstral Coefficients
- MLE** Maximum Likelihood Estimation
- MLLR** Maximum Likelihood Linear Regression
- MMC** Modèle de Markov Caché
- MMI** Maximal Mutual Information
- MMIE** Maximal Mutual Information Estimation
- MPE** Minimum Phone Error
- MWE** Minimum Word Error
- NCLR** Normalized Cross Likelihood Ratio
- NP** Nom propre

## *Acronymes*

---

**NSF** National Science Foundation

**OALD** Oxford Advanced Learner's Dictionary

**OOV** Out Of Vocabulary

**PCA** Principal Component Analysis

**PER** Phoneme Error Rate

**PFC** Phonologie du Français Contemporain

**PLP** Perceptual Linear Prediction

**PNER** Proper Noun Error Rate

**POS** Part Of Speech

**PPL** PerPLexité

**RAP** Reconnaissance Automatique de la Parole

**RFI** Radio France Internationale

**RTM** Radio Télévision Marocaine

**SAT** Speaker Adaptive Training

**SMT** Statistical Machine Translation

**SRAP** Système de Reconnaissance automatique de la Parole

**SRILM** Stanford Research Institute Language Modeling

**SRL** Suivi et Regroupement en Locuteur

**SSII** Société de Services en Ingénierie Informatique

**TEI** Text Encoding Initiative

**TTS** Text To Speech

**VSC** Virtual Speaker Channels

**VTLN** Vocal Track Length Normalization

**WER** Word Error Rate

**WSR** Word Stroke Ratio

## **Bibliographie personnelle**

## Conférences d'audience internationale

[Estève 2010] Yannick Estève, Paul Deléglise, Sylvain Meignier, Simon Petit-Renaud, Holger Schwenk, Loic Barrault, Fethi Bougares, Richard Dufour, Vincent Jousse, Antoine Laurent et Anthony Rousseau (2010), Some recent research work at LIUM based on the use of CMU Sphinx, dans Proceeding of CMU SPUD Workshop, Dallas, USA, Mars 2010

[Laurent 2009] Laurent A., Merlin T., Meignier S., Estève Y. et Deléglise P., Iterative filtering of phonetic transcriptions of proper nouns, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 2009)*, pages 4265–4268, Taïpeï, Taïwan, Avril 2009.

[Laurent 2009] Laurent A., Deléglise P. et Meignier S., Grapheme to phoneme conversion using an SMT system, dans Proceedings of International Conference on Spoken Language Processing (ISCA, Interspeech 2009), pages 708–711, Brighton, Angleterre, Septembre 2009.

[Laurent 2008] Laurent A., Merlin T., Meignier S., Estève Y. et Deléglise P., Combined systems for automatic phonetic transcription of proper nouns, dans *Language Evaluation and Resources Conference (LREC 2008)*, Marrakech, Maroc, Mai 2008.

## Conférences d'audience nationale

[Laurent 2010] Laurent A., Meignier S. et Deléglise P., Réordonnancement automatique d'hypothèses pour l'assistance à la transcription de la parole, dans *XXXe Journées d'Etudes sur la Parole (JEP-10)*, Mons, Belgique, 2010.

[Laurent 2008] Laurent A., Merlin T., Meignier S., Estève Y. et Deléglise P., Combinaison de systèmes pour la phonétisation automatique de noms propres, dans *XXIXe Journées d'Etudes sur la Parole (JEP-08)*, Avignon, France, 2008.

[Laurent 2007] Laurent A., Autoadaptation d'un système de reconnaissance vocale à la tache de transcription automatique de réunions, dans *Rencontre des Jeunes Chercheurs en Parole (RJCP'07)*, Paris, France, 2007.

# Bibliographie

- [AFCP 2008] AFCP, Évaluation des systèmes de transcription enrichie d'émissions radiophoniques, plan d'évaluation d'ESTER phases 1 et 2, Octobre 2008.
- [Ainsworth 1992] Ainsworth W. A. et Pratt S. R., Feedback strategies for error correction in speech recognition systems, dans *International Journal of Man-Machine Studies*, volume 36, pages 833–842, Juin 1992.
- [Allauzen 2003] Allauzen A., *Modélisation linguistique pour l'indexation automatique de documents audiovisuels*, Thèse de doctorat, Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur (LIMSI), Paris XI, 2003.
- [Amengual 2000] Amengual J. C., Benedí J. M., Castaño A., Castellanos A., Jiménez V. M., Llorens D., Marzal A., Pastor M., Prat F., Vidal E. et Vilar J., The EuTrans-I speech translation system, dans *Machine Translation*, volume 14, pages 941–951, 2000.
- [Anastasakos 1997] Anastasakos T., McDonough J. et Makhoul J., Speaker adaptive training : A maximum likelihood approach to speaker normalization, dans *ICASSP '97 : Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '97)*, volume 2, pages 1043–1046, Munich, Allemagne, Avril 1997.
- [Andersen 1996] Andersen O., Kuhn R., Lazaridès A., Dalsgaard P., Haas J. et Nöth E., Comparison of two tree-structured approaches for grapheme-to-phoneme conversion, dans *Proceedings of International Conference on Spoken Language Processing (ICSLP 96)*, volume 3, pages 1700–1703, Philadelphia, PA, USA, Octobre 1996.
- [Aubergé 1991] Aubergé V., *La synthèse de la parole : “des règles au lexique”*, Thèse de doctorat, Université Stendhal – Grenoble, 1991.
- [Bagshaw 1998] Bagshaw P., Phonemic transcription by analogy in text-to-speech synthesis : novel word pronunciation and lexicon compression, dans *Computer Speech and Language*, volume 16, pages 119–142, 1998.
- [Bahl 1986] Bahl L., Brown P., de Souza P. et Mercer R., Minimum exact word error training, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 86)*, volume 11, pages 49–52, Tokyo, Japon, Avril 1986.
- [Bahl 1993] Bahl L. R., Brown P. F., de Souza P. V., Mercer R. L. et Picheny M., A method for the construction of acoustic Markov models for words, dans *IEEE Transactions on Speech and Audio Processing*, volume 1, pages 443–452, Octobre 1993.
- [Bahl 1991] Bahl L. R., Das S., deSouza P. V., Epstein M., Mercer R. L., Merialdo B., Nahamoo D., Picheny M. A. et Powell J., Automatic phonetic baseform determination, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 91)*, volume 1, pages 173–176, Toronto, Canada, Avril 1991.

## Bibliographie

---

- [Barras 1998] Barras C., Geoffrois E., Zhibiao W. et Mark L., Transcriber : a free tool for segmenting, labeling and transcribing speech, dans *Language Evaluation and Resources Conference (LREC 1998)*, volume 2, pages 1373–1376, Grenade, Espagne, 1998.
- [Bateman 2000] Bateman A., Hewitt J., Ariyaeenia A., Sivakumaran P. et Lambourne A., The Quest for The Last 5% : Interfaces for Correcting Real-Time Speech-Generated Subtitles, dans *Proceedings of the 2000 Conference on Human Factors in Computing Systems (CHI2000)*, pages 129–130, La Hague, Hollande, Avril 2000.
- [Baum 1970] Baum L., A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains, dans *Annals of Mathematical Statistics*, volume 41, pages 164–171, 1970.
- [Baum 1972] Baum L., An inequality and associated maximization technique in statistical estimation of probabilistic functions of markov processes, dans *Inequalities*, volume 3, pages 1–8, 1972.
- [Bazillon 2008a] Bazillon T., Estève Y. et Luzzati D., Manual vs. assisted transcription of prepared and spontaneous speech, dans *Language Evaluation and Resources Conference (LREC 2008)*, Marrakech, Maroc, Mai 2008a.
- [Bazillon 2008b] Bazillon T., Jousse V., Béchet F., Estève Y., Linarès G. et Luzzati D., La parole spontanée : transcription et traitement, dans *Traitement Automatique des Langues*, volume 49, pages 47–76, 2008b.
- [Béchet 2001] Béchet F., LIA\_PHON : un système complet de phonétisation de textes, dans *Traitement Automatique des Langues*, volume 42, pages 47–67, 2001.
- [Béchet 1995] Béchet F., Derderian S. et El-Bèze M., Conversion graphèmes-phonèmes automatique : le système GRIPHON, dans *Actes des 15èmes journées internationales IA'95*, pages 767–770, Montpellier , France, Juin 1995.
- [Béchet 2002] Béchet F., de Mori R. et Subsol G., Dynamic generation of proper name pronunciations for directory assistance, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 92)*, volume 1, pages 745–748, San Francisco, USA, Mars 2002.
- [Bellegarda 2005] Bellegarda J. R., Unsupervised, language-independent grapheme-to-phoneme conversion by latent analogy, dans *Speech Communication*, volume 46, pages 140–152, 2005.
- [Bisani 2001] Bisani M. et Ney H., Breadth-first for finding the optimal phonetic transcription from multiple utterances, dans *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2001)*, volume 2, pages 1429–1432, Aalborg, Danemark, 2001.
- [Bisani 2008] Bisani M. et Ney H., Joint-sequence models for grapheme-to-phoneme conversion, dans *Speech Communication*, volume 50, pages 434–451, 2008.
- [Black 1998] Black A. W., Lenzo K. et Pagel V., Issues in building general letter to sound rules, dans *In Proceedings of the 3rd International Workshop on Speech Synthesis, ESCA*, pages 77–80, Australie, 1998.
- [Boersma 2001] Boersma P., Praat, a system for doing phonetics by computer, dans *Glot International*, volume 5, pages 341–345, 2001.

- 
- [Boudin 2007] Boudin F. et Torres-Moreno J.-M., NEO-CORTEX : a performant user-oriented multi document summarization system, dans *Conference on Intelligent Text Processing and Computational Linguistics (CICLing 2007)*, pages 551–562, Mexico, Mexique, Février 2007.
- [Byrne 1998] Byrne W., Finke M., Khudanpur S., McDonough J., Nock H., Riley M., Saraciar M., Wooters C. et Zavaliagkos G., Pronunciation modeling using a hand-labelled corpus for conversational speech recognition, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 98)*, volume 1, pages 313–316, Seattle, USA, Mai 1998.
- [Canseco-Rodriguez 2005] Canseco-Rodriguez L., Lamel L. et Gauvain J.-L., A comparative study using manual and automatic transcriptions for diarization, dans *Automatic Speech Recognition and Understanding (IEEE, ASRU 2005)*, volume 1, pages 415–419, Porto Rico, USA, Novembre 2005.
- [Cardinal 2007] Cardinal P., Boulianne G., Comeau M. et Boisvert M., Real-Time Correction of Closed-Captions, dans *Proceedings of Association for Computational Linguistics*, pages 113–116, Prague, République Tchèque, Juin 2007.
- [Casacuberta 2004a] Casacuberta F., Ney H., Och F., Vidal E., Vilar J., Barrachina S., García-Varea I., Llorens D., Martínez C., Molau S., Nevado F., Pastor M., Picó D., Sanchis A. et Tillmann C., Some approaches to statistical and finite-state speech-to-speech translation, dans *Computer Speech and Language*, volume 18, pages 25–47, 2004a.
- [Casacuberta 2004b] Casacuberta F., Vidal E., Sanchis A. et Vilar J., Pattern recognition approaches for speech-to-speech translation, dans *Cybernetic and Systems : an International Journal*, volume 35, pages 3–17, 2004b.
- [Chan 2005] Chan A., Ravishankar M. et Rudnicky A., On improvements of CI-based GMM selection, dans *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2005)*, pages 565–568, Lisbonne, Portugal, Septembre 2005.
- [Charlet 1997] Charlet D., *Authentification vocale par téléphone en mode dépendant du texte*, Thèse de doctorat, École Nationale Supérieure des Télécommunications (ENST), Paris, 1997.
- [Charniak 1993] Charniak E., *Statistical Language Learning*, MIT Press, Cambridge, Massachusetts, 1993.
- [Chen 1998a] Chen J. S. F. and Goodman, An empirical study of smoothing techniques for language modeling, Rapport technique TR-10-98, Center for Research in Computing Technology (Harvard University), 1998a.
- [Chen 1998b] Chen S. et Gopalakrishnan P., Speaker, environment and channel change detection and clustering via the bayesian information criterion, dans *DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, USA, Février 1998b.
- [Civera 2005] Civera J., Vilar J., Cubel E., Lagarda A., Barrachina S., Casacuberta F. et Vidal E., A novel approach to computer assisted translation based on finite-state transducers, dans *Proceedings of Finite-State Methods and Natural Language Processing (FSMNLP)*, volume 4002, pages 32–42, Helsinki, Finlande, Septembre 2005.

## Bibliographie

---

- [Civera 2004] Civera J., Vilar J., Cubel E., Lagarda A., Barrachina S., Casacuberta F., Vidal E., Picó D. et González J., A syntactic pattern recognition approach to computer assisted translation, dans *International Workshop on Structural and Syntactic Pattern Recognition (SSPR)*, volume 3138, pages 207–215, Lisbonne, Portugal, Août 2004.
- [Clarkson 1997] Clarkson P. et Robinson A. J., Language model adaptation using mixtures and an exponentially decaying cache, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 97)*, volume 2, pages 799–802, Munich, Allemagne, Avril 1997.
- [Cohen 1995] Cohen J., Kamm T. et Andreou A., Vocal tract normalization in speech recognition : compensating for systematic speaker variability, dans *Journal of Acoustic Society of America*, volume 97, pages 3246–3247, Mai 1995.
- [Cubel 2004] Cubel E., Civera J., Vilar J., Lagarda A., Barrachina S., Vidal E., Casacuberta F., Picó D., González J. et Rodríguez L., A syntactic pattern recognition approach to computer assisted translation, dans *Proceedings of the 16th European Conference on Artificial Intelligence (ECAI04)*, pages 586–590, Valencia, Espagne, 2004.
- [Daelemans 1997] Daelemans W. et van den Bosch A., Language-independent data-oriented grapheme-to-phoneme conversion, dans *Progress in Speech Synthesis*, pages 77–90, 1997.
- [d'Alessandro 1992] d'Alessandro C. et Demars C., Représentations temps-fréquence du signal de parole, dans *Traitements du signal*, volume 9, pages 153–173, 1992.
- [Damper 1999] Damper R., Marchand Y., Adamson M. et Gustafson K., Evaluating the pronunciation component of text-to-speech systems for English : A performance comparison of different approaches, dans *Computer Speech and Language*, volume 13, pages 155–176, 1999.
- [Davis 1980] Davis S. B. et Mermelstein P., Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences, dans *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume 28, pages 357–366, Août 1980.
- [De Calmes 1998] De Calmes M. et Perennou G., BDLEX : a lexicon for spoken and written French, dans *Language Evaluation and Resources Conference (LREC 1998)*, pages 1129–1136, Grenade, Espagne, Mai 1998.
- [Dedina 1991] Dedina M. et Nusbaum H., Pronounce : a program for pronunciation by analogy, dans *Computer Speech and Language*, volume 5, pages 55–63, 1991.
- [Deligne 1995] Deligne S. et Bimbot F., Language modeling by variable length sequences : theoretical formulation and evaluation of multigrams, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 95)*, volume 1, pages 169–172, Detroit, USA, Mai 1995.
- [Deligne 2001] Deligne S., Maison B. et Gopinath R., Automatic generation and selection of multiple pronunciations for dynamic vocabularies, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 2001)*, volume 1, pages 565–568, Salt Lake City, USA, Mai 2001.
- [Deligne 2003] Deligne S. et Mangu L., On the use of lattices for the automatic generation of pronunciations, dans *Proceedings of International Conference on Acoustics Speech and*

---

*Signal Processing (IEEE, ICASSP 2003)*, volume 1, pages 204–207, Hong-Kong, Chine, Avril 2003.

- [Deléglise 2009] Deléglise P., Estève Y. et Meignier S., Improvements to the LIUM French ASR system based on CMU Sphinx : what helps to significantly reduce the word error rate ?, dans *Proceedings of International Conference on Spoken Language Processing (ISCA, Interspeech 2009)*, pages 2123–2126, Brighton, Angleterre, Septembre 2009.
- [Dempster 1977] Dempster A., N. L. et Rubin D., Maximum likelihood from incomplete data via the em algorithm, dans *Journal of the Royal Statistical Society*, volume 39, pages 1–38, 1977.
- [Digalakis 1995] Digalakis V., Rtschev D., Neumeyer L. et Sa E., Speaker adaptation using constrained estimation of gaussian mixtures, dans *IEEE Transactions on Speech and Audio Processing*, volume 3, pages 357–366, 1995.
- [Dufour 2008] Dufour R., From prepared speech to spontaneous speech recognition system : a comparative study applied to French language, dans *IEEE/ACM CSTST Student Workshop*, volume 1, pages 595–599, Cergy, France, Octobre 2008.
- [Durand 2002] Durand J., Lacks B. et Lyche C., La phonologie du français contemporain : usages, variétés et structure, dans *Romanistische Korpuslinguistik - Korpora und gesprochene Sprache/Romance Corpus Linguistics - Corpora and Spoken Language*, pages 93–106, 2002.
- [Días-Verdejo 1998] Días-Verdejo J., Peinado A., Rubio A., Segarra E. et Prieto N., Albayzin : A task oriented spanish speech corpus, dans *Language Evaluation and Resources Conference (LREC 1998)*, volume 1, pages 497–501, Grenade, Espagne, Mai 1998.
- [Estève 2009] Estève Y., Traitement automatique de la parole : contributions, dans *Habilitation à diriger des recherches*, Laboratoire d’Informatique de l’Université du Maine (LIUM) - Le Mans, France., 2009.
- [Favre 2006] Favre B., Béchet F., Bellot P., Boudin F., El-Bèze M., Gillard L., Lapalme G. et Torres-Moreno J.-M., The LIA-Thales summarization system at DUC-2006, dans *Document Understanding Conference Workshop, HLT-NAACL'06, New York (USA)*, 2006.
- [Federico 1998] Federico M. et De Mori R., Language modelling, *R. De Mori, Ed. Spoken Dialogues with Computers, chapter 7, Academic Press*, pages 204–210, 1998.
- [Ferrané 1992] Ferrané I., De Calmes M., Pecatte J. et Perennou G., Besoins lexicaux à la lumière de l’analyse statistique du corpus de textes du projet BREF : le lexique BDLEX du français écrit et oral, dans *Proceedings of Association for Computational Linguistics*, volume 4, pages 1203–1208, Nantes, France, Juillet 1992.
- [Foster 2002] Foster G., *Text Prediction for Translators*, Thèse de doctorat, Université de Montréal, Canada., 2002.
- [Foster 1997] Foster G., Isabelle P. et Plamondon P., Target-text mediated interactive machine translation, dans *Machine Translation*, volume 12, pages 175–194, 1997.
- [Furui 1981] Furui S., Cepstral analysis technique for automatic speaker verification, dans *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume 292, pages 254–272, Avril 1981.

## Bibliographie

---

- [Gales 1998] Gales M. J. F., Maximum likelihood linear transformations for HMM-based speech recognition, dans *Computer Speech and Language*, volume 12, pages 75–98, 1998.
- [Galescu 2001] Galescu L. et Allen J. F., Bi-directional conversion between graphemes and phonemes using a joint n-gram model, dans *In Proceedings of 4th ISCA Tutorial and Research Workshop on Speech Synthesis*, Perthshire, Écosse, Août 2001.
- [Galliano 2005] Galliano S., Geoffrois E., Mostefa D., Choukri K., Bonastre J. F. et Gravier G., The ESTER phase II evaluation campaign for the rich transcription of French broadcast news, dans *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2005)*, volume 1, pages 1149–1152, Lisbonne, Portugal, Septembre 2005.
- [Galliano 2009] Galliano S., Gravier G. et Chaubard L., The ESTER 2 Evaluation Campaign for the Rich Transcription of French Radio Broadcasts, dans *Proceedings of International Conference on Spoken Language Processing (ISCA, Interspeech 2009)*, volume 1, pages 2583–2586, Brighton, Angleterre, Septembre 2009.
- [Gauvain 1994] Gauvain J.-L. et Lee C. H., Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains, dans *IEEE Transactions on Speech and Audio Processing*, volume 22, pages 291–298, Avril 1994.
- [Gish 1991] Gish H., Siu M.-H. et Rohlicek R., Segregation of speakers for speech recognition and speaker identification, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 91)*, volume 2, pages 873–877, Toronto, Canada, Mai 1991.
- [Goto 2007] Goto M., Ogata J. et Eto K., PodCastle : A Web 2.0 Approach to Speech Recognition Research, *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2007)*, 1 :2397–2400, Août 2007.
- [Haeb-Umbach 1995] Haeb-Umbach R., Beyerlein P. et Thelen E., Automatic transcription of unknown words in a speech recognition system, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 95)*, volume 1, pages 840–843, Detroit, USA, Mai 1995.
- [Heigold 2005] Heigold G., Macherey W., Schulter R. et Ney H., Minimum exact word error training, dans *Automatic Speech Recognition and Understanding (IEEE, ASRU 2005)*, pages 186–190, San Juan, Puerto Rico, Novembre 2005.
- [Hermansky 1990] Hermansky H., Perceptual linear predictive (PLP) analysis of speech, dans *Journal of the Acoustical Society of America*, volume 87, pages 1738–1752, 1990.
- [Hermansky 1994] Hermansky H. et Morgan N., RASTA processing of speech, dans *IEEE Transactions on Speech and Audio Processing*, volume 2, pages 578–589, 1994.
- [Homayounpour 1994] Homayounpour M. M. et Chollet G., Performance comparison of some relevant spectral representations for speaker verification, dans *Workshop on Automatic Speaker Recognition, Identification, Verification*, pages 27–30, 1994.
- [Huang 1992] Huang X., Alleva F., wuen Hon H., yuh Hwang M. et Rosenfeld R., The SPHINX-II Speech Recognition System : An Overview, dans *Computer Speech and Language*, volume 7, pages 137–148, 1992.

- 
- [Huggins-daines 2006] Huggins-daines D., Kumar M., Chan A., Black A. W., Ravishankar M. et Rudnicky A. I., Pocketsphinx : A free, real-time continuous speech recognition system for hand-held devices, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 2006)*, volume 1, pages 185–188, Toulouse, France, Mai 2006.
- [Häkkinen 2003] Häkkinen J., Suonsausta J., Riis S. et Jensen K., Assessing text-to-phoneme mapping strategies in speaker independent isolated word recognition, dans *Speech Communication*, volume 41, pages 455–467, 2003.
- [Imai 2002] Imai T., Matsui A., Homma S., Kobayakawa T., Kazuo O., Sato S. et Ando A., Speech Recognition with a respeak method for subtiling live broadcast, dans *Proceedings of International Conference on Spoken Language Processing (ISCA, ICSLP 2002)*, pages 1757–1760, Denver, USA, Septembre 2002.
- [Jelinek 1976] Jelinek F., Continuous speech recognition by statistical methods, dans *Proceedings of the IEEE*, volume 4, pages 532–556, 1976.
- [Jensen 2000] Jensen K. et Riis S., Self-organizing letter code-book for text-to-phoneme neural network model, dans *Proceedings of International Conference on Spoken Language Processing (ISCA, ICSLP 2000)*, volume 3, pages 318–321, Beijing, Chine, Octobre 2000.
- [Jousse 2008] Jousse V., Jacquin C., Meignier S., Estève Y. et Daille B., Étude pour l'amélioration d'un système d'identification nommée du locuteur, dans *XXVIIe Journées d'Études sur la Parole (JEP-08)*, Avignon, France, Juin 2008.
- [Junqua 1990] Junqua J.-C., Utilisation d'un modèle d'audition et de connaissances phonétiques en reconnaissance automatique de la parole, dans *Traitement du signal*, volume 7, pages 275–284, 1990.
- [Kaplan 1994] Kaplan R. M. et Kay M., Regular models of phonological rule systems, dans *Association for Computational Linguistics*, volume 20, pages 331–378, 1994.
- [Katz 2002] Katz M., Meier H.-G., Dolfini H. et Klakow D., Robustness of linear discriminant analysis in automatic speech recognition, dans *International Conference on Pattern Recognition (ICPR'02)*, volume 3, pages 371–374, Québec, Canada, Août 2002.
- [Kershaw 1996] Kershaw D., Robinson A. J. et Renals S. J., The 1995 abbot hybrid connectionist-hmm large-vocabulary recognition system, dans *Advanced Research Projects Agency Speech Recognition Workshop*, pages 93–98, 1996.
- [Kienappel 2001] Kienappel A. K. et Kneser R., Designing very compact decision trees for grapheme-to-phoneme transcription, dans *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2001)*, volume 1, pages 1911–1914, Aalborg, Danemark, Septembre 2001.
- [Kneser 1995] Kneser R. et Ney H., Improved backing-off for m-gram language modeling, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 95)*, volume 1, pages 181–184, Detroit, USA, Mai 1995.
- [Kuhn 1990] Kuhn R. et De Mori R., A cache-based natural language method for speech recognition, dans *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume 12, pages 570–582, 1990.

## Bibliographie

---

- [Lacheret-Dujour 1990] Lacheret-Dujour A., *Contribution à une analyse de la variabilité phonologique pour le traitement automatique de la parole continue multi-locuteur*, Thèse de doctorat, Université Paris 7, 1990.
- [Laporte 1988] Laporte E., *Méthodes Algorithmiques et Lexicales de Phonétisation de Textes*, Thèse de doctorat, Université Paris 7, 1988.
- [Laurent 2009] Laurent A., Deléglise P. et Meignier S., Grapheme to phoneme conversion using an SMT system, dans *Proceedings of International Conference on Spoken Language Processing (ISCA, Interspeech 2009)*, pages 708–711, Brighton, Angleterre, Septembre 2009.
- [Le 2007] Le V.-B., Mella O. et Fohr D., Speaker diarization using normalized cross likelihood ratio, dans *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2007)*, volume 1, pages 1869–1872, Antwerp, Belgique, Août 2007.
- [Lee 1990] Lee K.-F., Hon H.-W. et Reddy R., An Overview of the SPHINX Speech Recognition System, dans *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume 38, pages 35–45, Janvier 1990.
- [Leggetter 1995] Leggetter C. J. et Woodland P. C., Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models, dans *Computer Speech & Language*, volume 9, pages 171–185, Avril 1995.
- [Levenshtein 1966] Levenshtein V. I., Binary codes capable of correcting deletions, insertions, and reversals, dans *Soviet Physics Doklady*, volume 10, pages 707–710, 1966.
- [Liporace 1982] Liporace L., Maximum likelihood estimation for multivariate observations of Markov sources, dans *IEEE Transactions on Information Theory*, volume 28, pages 729–734, 1982.
- [Luk 2001] Luk R. W. P. et Damper R. I., English letter-phoneme conversion by stochastic transducers, dans *Data-Driven Techniques in Speech Synthesis*, pages 91–123, 2001.
- [Luz 2008] Luz S., Masoodian M., Rogers B. et Deering C., Interface design strategies for computer-assisted speech transcription, dans *Proceedings of OZCHI'08, Australasian Conference on Computer-Human Interaction*, volume 287, pages 203–210, Cairns, Australie, Décembre 2008.
- [Ma 2001] Ma C. et Randolph M. A., An approach to automatic phonetic baseform generation based on bayesian networks, dans *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2001)*, volume 1, pages 1453–1456, Aalborg, Danemark, Septembre 2001.
- [Mangu 2000] Mangu H., Brill E. et Stolcke A., Finding consensus in speech recognition : Word error minimization and other applications of confusion networks, dans *Computer Speech & Language*, volume 14, pages 373–400, 2000.
- [Marchand 2001] Marchand Y. et Damper R. I., A multi-strategy approach to improving pronunciation by analogy, dans *Association for Computational Linguistics*, volume 26, pages 195–219, Toulouse, France, Juillet 2001.

- 
- [Martin 1996] Martin P., Winpitch : un logiciel d'analyse temps réel de la fréquence fondamentale fonctionnant sous Windows, dans *XXIe Journées d'Étude sur la Parole*, pages 224–227, Avignon, France, Juin 1996.
- [McCulloch 1987] McCulloch N., Bedworth M. et Bridle J., NETspeak - a re-implementation of NETtalk, dans *Computer Speech Language*, volume 2, pages 289–302, 1987.
- [McNair 1994] McNair A. et Waibel A., Improving recognizer acceptance through robust, natural speech repair, dans *Proceedings of International Conference on Spoken Language Processing (ICSLP 94)*, volume 1, pages 1299–1302, Yokohama, Japon, Septembre 1994.
- [Meignier 2010] Meignier S. et Merlin T., LIUM SpkDiarization : an open source toolkit for diarization, dans *CMU SPUD Workshop*, Dallas, USA, Mars 2010.
- [Mokbel 1999] Mokbel H. et Jouvet D., Derivation of the optimal set of phonetic transcriptions for a word from its acoustic realizations, dans *Speech Communication*, volume 29, pages 49–64, 1999.
- [Nanjo 2006] Nanjo H., Akita Y. et Kawahara T., Computer assisted speech transcription system for efficient speech archive, dans *Proceedings of WESPAC IX, the 9th Western Pacific Acoustic Conference*, Séoul, Corée du Sud, Juin 2006.
- [Ney 2000] Ney H., Nießen S., Och F., Sawaf H., Tillmann C. et Vogel S., Algorithms for statistical translation of spoken language, dans *IEEE Transactions on Speech and Audio Processing*, volume 8, pages 24–36, 2000.
- [Ogata 2005] Ogata J. et Goto M., Speech repair : Quick error correction just by using selection operation for speech input interfaces, dans *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2005)*, volume 1, pages 133–136, Lisbonne, Portugal, Septembre 2005.
- [Ogata 2007] Ogata J., Goto M. et Eto K., Automatic Transcription for a Web 2.0 Service to Search Podcasts, dans *Proceedings of International Conference on Speech Communication and Technology (ISCA, Interspeech 2007)*, volume 1, pages 2617–2620, Antwerp, Belgique, Août 2007.
- [Oger 2008] Oger S., Linarès G., Béchet F. et Nocera P., On-demand new word learning using the world wide web, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 2008)*, volume 1, pages 4305–4308, Las Vegas, USA, Avril 2008.
- [Pagel 1998] Pagel V., Lenzo K. et Black A. W., Letter to sound rules for accented lexicon compression, dans *Proceedings of International Conference on Spoken Language Processing (ICSLP 98)*, pages 2015–2018, Sydney, Australie, Décembre 1998.
- [Papineni 2002] Papineni K., Roukos S., Ward T. et Zhu W.-J., BLEU : a method for automatic evaluation of machine translation, dans *Association for Computational Linguistics*, pages 311–318, Philadelphia, USA, Juillet 2002.
- [Pearl 1984] Pearl J., Heuristics : Intelligent search strategies for computer problem solving, dans *The Addison-Wesley series in artificial intelligence*, 1984.
- [Pinkowski 1997] Pinkowski B., Principal component analysis of speech spectrogram images, dans *Pattern Recognition*, volume 30, pages 777–787, 1997.

## Bibliographie

---

- [Povey 2002] Povey D. et Woodland P., Minimum phone error and i-smoothing for improved discriminative training, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 2002)*, volume 1, pages 105–108, Orlando, USA, Mai 2002.
- [Rabiner 1989] Rabiner L. R., A tutorial on hidden Markov models and selected applications in speech recognition, dans *IEEE Transactions on Speech and Audio Processing*, volume 77, pages 257–286, 1989.
- [Rabiner 1986] Rabiner L. R. et Juang B. H., An introduction to hidden Markov models, dans *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume 3, pages 4–16, Janvier 1986.
- [Rama 2009] Rama T., Singh A. K. et Kolachina S., Modeling letter-to-phoneme conversion as a phrase based statistical machine translation problem with minimum error rate training, dans *Proceeding of North American Chapter of the Association for Computational Linguistics - Human Language Technologies (NAACL HLT) 2009 conference*, volume 1, pages 90–95, Boulder, USA, 2009.
- [Ramabhadran 1998] Ramabhadran B., Bahl L., deSouza P. et Padmanabhan M., Acoustics-only based automatic phonetic baseform generation, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 98)*, volume 1, pages 309–312, Seattle, USA, Mai 1998.
- [Ravishankar 2000] Ravishankar M., Singh R., Raj B. et Stern R. M., The 1999 CMU 10x real time broadcast news transcription system, dans *Proceedings of the DARPA workshop on Automatic Transcription of Broadcast News*, Washington DC, USA, Mai 2000.
- [Reynolds 1994] Reynolds D. A., Experimental evaluation of features for robust speaker identification, dans *IEEE Transactions on Speech and Audio Processing*, volume 2, pages 639–643, 1994.
- [Reynolds 1998] Reynolds D. A., Singer E., Carlson B. A., O’Leary G. C., McLaughlin J. J. et Zissman M. A., Blind clustering of speech utterances based on speaker and language characteristics, dans *Proceedings of International Conference on Spoken Language Processing (ICSLP 98)*, volume 7, pages 3193–3196, Sydney, Australie, Décembre 1998.
- [Riley 1999] Riley M., Byrne W., Finke M., Khudanpur S., Ljolje A., McDonough J., Nock H., Saraclar M., Wooters C. et Zavaliagkos G., Stochastic pronunciation modelling from hand-labelled phonetic corpora, dans *Speech Communication*, volume 29, pages 209–224, 1999.
- [Rodríguez 2007] Rodríguez L., Casacuberta F. et Vidal E., Computer Assisted Transcription of Speech, dans *Lecture Notes In Computer Science, Proceedings of the 3rd Iberian conference on Pattern Recognition and Image Analysis, Part I*, volume 4477, pages 241–248, Girona, Espagne, Juin 2007.
- [Rose 1997] Rose R. C. et Lleida E., Speech Recognition using Automatically Derived Baseforms, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 97)*, volume 2, pages 1271–1274, Munich, Allemagne, Avril 1997.

- 
- [Rosenfeld 1996] Rosenfeld R., A maximum entropy approach to adaptative statistical language modeling, dans *Computer Speech and Language*, volume 10, pages 187–228, 1996.
- [Rosenfeld 1992] Rosenfeld R. et Huang X., Improvements in stochastic language modeling, dans *HLT '91 : Proceedings of the workshop on Speech and Natural Language*, pages 107–111, Harriman, USA, Février 1992.
- [Sejnowski 1987] Sejnowski T. J. et Rosenberg C. R., Parallel networks that learn to pronounce English text, dans *Complex Systems*, volume 1, pages 145–168, 1987.
- [Seymore 1998] Seymore K., Stanley C., Doh S., Eskenazi M., Gouvea E., Raj B., Ravishankar M., Rosenfeld R., Siegler M., Stern R. et Thayer E., The 1997 CMU Sphinx-3 English broadcast news transcription system, dans *DARPA Broadcast News Transcription and Understanding Workshop*, Lansdowne, USA, Février 1998.
- [Siu 1992] Siu M.-H., Rohlicek R. et Gish H., An unsupervised, sequential learning algorithm for segmentation of speech waveforms with multi speakers, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 92)*, volume 2, pages 189–192, San Fransisco, USA, Mars 1992.
- [Sloboda 1995] Sloboda T., Dictionary learning : performance through consistency, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 95)*, volume 1, pages 453–456, Detroit, USA, Mai 1995.
- [Solomonoff 1998] Solomonoff A., Mielke A., Schmidt M. et Gish H., Clustering speakers by their voices, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 98)*, volume 2, pages 557–560, Seattle, USA, Mai 1998.
- [Soong 1988] Soong F. K. et Rosenberg A. E., On the use of instantaneous and transitional spectral information in speaker recognition, dans *IEEE Transactions on Acoustics, Speech, and Signal Processing*, volume 36, pages 871–879, Juin 1988.
- [Stolcke 2002] Stolcke A., SRILM-an extensible language modeling toolkit, dans *Proceedings of International Conference on Spoken Language Processing (ISCA, ICSLP 2002)*, volume 2, pages 901–904, Denver, USA, Septembre 2002.
- [Suhm 1997] Suhm B., Empirical evaluation of interactive multimodal error correction, dans *IEEE Workshop on Speech recognition and understanding*, pages 583–590, Santa Barbara, USA, Décembre 1997.
- [Suhm 1996] Suhm B., Myers B. et Waibel A., Interactive recovery from speech recognition errors in speech user interface, dans *Proceedings of International Conference on Spoken Language Processing (ICSLP 96)*, volume 2, pages 865–868, Philadelphia, USA, Octobre 1996.
- [Suontausta 2000] Suontausta J. et Häkkinen J., Decision tree based text-to-phoneme mapping for speech recognition, dans *Proceedings of International Conference on Spoken Language Processing (ISCA, ICSLP 2000)*, volume 2, pages 831–834, Beijin, Chine, Octobre 2000.

## Bibliographie

---

- [Svendsen 1995] Svendsen T., Soong F. K. et Purnhagen H., Optimizing baseforms for HMM-based speech recognition, dans *Proceedings of European Conference on Speech Communication and Technology (ISCA, Eurospeech 95)*, pages 783–786, Madrid, Espagne, Septembre 1995.
- [Tihoni 1991] Tihoni J. et Pérennou G., Phonotypical transcription through the GEPH expert system, dans *Proceedings of European Conference on Speech Communication and Technology (ISCA, Eurospeech 1991)*, volume 1, pages 767–770, Genova, Italie, Septembre 1991.
- [Tomás 2006] Tomás J. et Casacuberta F., Statistical phrase-based models for interactive computer-assisted translation, dans *Proceedings of Coling/Association for Computational Linguistics*, pages 835–841, Sydney, Australie, Juillet 2006.
- [Torkkola 1993] Torkkola K., An efficient way to learn English grapheme-to-phoneme rules automatically, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 93)*, volume 2, pages 199–202, Minneapolis, USA, Avril 1993.
- [Trost 2005] Trost H., Matiasek J. et Baroni M., The language component of the fasty text prediction system, dans *Applied Artificial Intelligence*, volume 19, pages 743–781, 2005.
- [Valtchev 1997] Valtchev V., Odell J. J., Woodland P. C. et Young S. J., MMIE training of large vocabulary recognition systems, dans *Speech Communication*, volume 22, pages 303–314, 1997.
- [Vidal 2006] Vidal E., Casacuberta F., Rodríguez L., Civera J. et Martínez C., Computer-assisted translation using speech recognition, dans *IEEE Transaction on Audio, Speech and Language Processing*, volume 14, pages 941–951, 2006.
- [Viterbi 1967] Viterbi A., Error bounds for convolutional codes and an asymptotically optimum decoding algorithm, dans *IEEE Transactions on Information Theory*, volume 13, pages 260–269, 1967.
- [Voss 2007] Voss L. et Ehlen P., The CALO Meeting Assistant, dans *Human Language Technologies : the Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT-2007)*, pages 17–18, Rochester, USA, Avril 2007.
- [Wald 2006] Wald M., Creating Accessible Educational Multimedia through Editing Automatic Speech Recognition Captioning in Real Time, dans *International Journal of Interactive Technology and Smart Education : Smarter Use of Technology in Education*, volume 3, pages 131–142, 2006.
- [Walker 2004] Walker W., Lamere P., Kwok P., Raj B., Singh R., Gouvea E., Wolf P. et Woelfel J., Sphinx-4 : A flexible open source framework for speech recognition, Rapport technique TR-2004-1391, Sun Microsystems Laboratories, Novembre 2004.
- [Wandmacher 2007] Wandmacher T. et Antoine J.-Y., Modèle adaptatif pour la prédiction de mots, dans *Traitements Automatiques des Langues*, volume 48, pages 71–95, 2007.
- [Willsky 1976] Willsky A. et Jones H., A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems, dans *IEEE Transactions on Automatic Control*, volume 21, pages 108–112, 1976.

- 
- [Wood 1996] Wood M. E. et Lewis E., Windmill - the use of a parsing algorithm to produce predictions for disabled persons, dans *Proceedings of the 1996 Autumn Conference on Speech and Hearing*, volume 18, pages 315–322, Bowness-on-Windermere, Angleterre, Novembre 1996.
- [Wu 1999] Wu J. et Gupta V., Application of simultaneous decoding algorithms to automatic transcription of known and unknown words, dans *Proceedings of International Conference on Acoustics Speech and Signal Processing (IEEE, ICASSP 99)*, volume 2, pages 589–592, Phoenix, USA, Mars 1999.
- [Yvon 1996] Yvon F., *Prononcer par analogie : motivation, formalisation et évaluation*, Thèse de doctorat, Ecole Nationale Supérieure des Télécommunications (ENST), Paris, 1996.
- [Yvon 1997] Yvon F., Paradigmatic cascades : a linguistically sound model of pronunciation by analogy, dans *In Proceedings of 35th Annual Meeting of the Association for Computational Linguistics*, pages 428–435, Madrid, Espagne, Juillet 1997.

## Résumé

Cette thèse a été réalisée dans le cadre d'une convention CIFRE avec la société Spécinov, désireuse d'intégrer, dans de futures applications commerciales, le Système de Reconnaissance Automatique de la Parole (SRAP) du LIUM.

L'objectif de ce travail de thèse consiste à proposer des techniques permettant d'aider l'utilisateur dans la phase de correction des sorties du SRAP. L'outil d'aide à la gestion de réunion visé intègrera une méthode d'indexation automatique des réunions transcrrites, afin de pouvoir naviguer aisément entre les différents documents disponibles. La qualité de la transcription des noms propres semble être un élément discriminant et important pour cette tâche. En effet, rechercher les interventions d'un participant dans divers documents audio pourrait être l'une des fonctionnalités envisagées.

La première partie de cette thèse présente une méthode d'assistance à la transcription automatique de la parole. Le transcripteur humain dispose de la meilleure hypothèse fournie par le SRAP, et, à chaque correction de sa part, le système propose une nouvelle hypothèse prenant en compte cette correction. Cette dernière est obtenue à partir d'une réévaluation des réseaux de confusion générés par le SRAP faisant intervenir un score linguistique provenant de l'interpolation linéaire entre un modèle de langage 4-gram et un modèle cache. Les expériences, menées sur le corpus ESTER 2, montrent de bons résultats. L'utilisation de la méthode de réordonnancement permet d'observer un gain absolu de 3,4% (19,2% à 15,8%) en terme de nombre de mots à corriger (WSR - Word Stroke Ratio).

Afin de diminuer le taux d'erreur sur les noms propres, une méthode de phonétisation itérative utilisant les données acoustiques à disposition est proposée dans ce manuscrit. Cette méthode nécessite, dans un processus d'initialisation, l'utilisation d'un convertisseur graphème-phonème. Pour cette tâche, trois techniques ont été évaluées : le système de phonétisations automatique à base de règles LIA\_PHON [Béchet 2001], une méthode de phonétisation utilisant SMT [Laurent 2009], et une méthode à base de modèles à séquences jointes (JSM) [Bisani 2008]. L'utilisation de SMT couplée avec la méthode de phonétisation proposée permet d'observer des gains en terme de taux d'erreur mot (WER) et en terme de taux d'erreur noms propres (PNER).

**Mots-clés:** Reconnaissance automatique de la parole, phonétisation automatique, correction automatique, réseaux de confusion