Nancy-Université
Université Henri Poincaré

Département de formation doctorale en informatique **École doctorale IAEM Lorraine**
UFR STMIA

Mesures de confiance trame-synchrones et locales en reconnaissance automatique de la parole

THÈSE

présentée et soutenue publiquement le 9 octobre 2007

pour l'obtention du

Doctorat de l'université Henri Poincaré – Nancy Université
(spécialité informatique)

par

Joseph Razik

Composition du jury

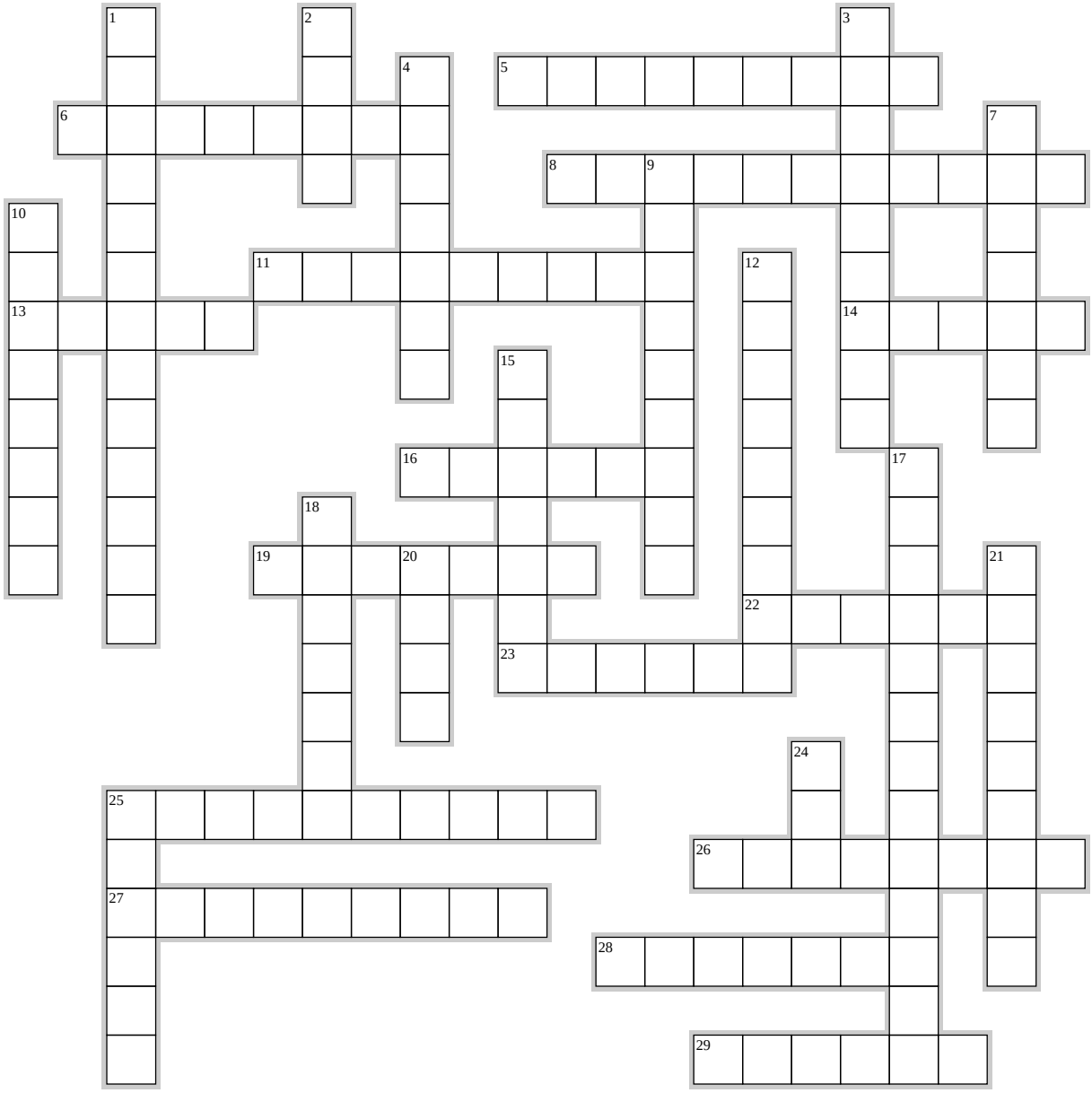
Président :	René Schott	Professeur, UHP–LORIA Nancy
Rapporteurs :	Jean-François Bonastre	Maître de conférences (HDR), LIA–CERI Avignon
	Gérard Chollet	Directeur de recherche CNRS, ENST–TSI Paris
Examinateurs :	Laurent Besacier	Maître de conférences (HDR), CLIPS–IMAG Grenoble
	Jean-Paul Haton	Professeur, UHP–LORIA Nancy (Directeur)
	Odile Mella	Maître de conférences, UHP–LORIA Nancy (Co-directrice)

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Loria

Remerciements

Mes remerciements vont principalement aux personnes dont les noms sont dans la grille ci-dessous.
Je remercie les différentes personnes – cobayes – qui ont bien voulu prendre sur leur temps et participer à l'expérience de compréhension de texte.
Je remercie également tous ceux que j'ai oubliés ainsi que tous ceux qui liront ces quelques lignes et les pages qui suivent.



Horizontal

5. Faire des balades en forêt avec un violon, ce n'est pas pratique, mais pour attraper des truites, ce n'est vraiment pas pratique
6. Mais qu'est ce que c'est que cette histoire de pouet et de canards ?!
8. Un grand dadet qui parle trop fort
11. Le vendredi ce n'est pas la peine de le chercher, c'est golf!
13. Le maître de l'enfumage
14. Consciencieuse et professionnelle, très (trop) impliquée, « attention au surmenage »
16. Paris, la capitale, la tour Eiffel et l'ENST
19. La chartreuse, ça ne vaut pas la mirabelle mais il faut faire avec les moyens du bord
22. Elle pourrait parler plus fort quand elle dit au revoir
23. Femme au volant ...
25. Mister CVS
26. Je n'ai toujours pas les accents, même en HTML!
27. Un peu trop chanceux, même à la xbox
28. It's the Final Countdown
29. Tout le monde ne sort pas indemne d'un stage de « headshot »

Vertical

1. On y danse tous en rond ... peut être que lui aussi
2. Toujours prêt à aider, comme son nom l'indique
3. On a quand même fait de belles parties de ping-pong
4. Mon guide à Pékin et aussi un peu dans la culture chinoise
7. Il est étonnant d'avoir été dans le même petit collège avec une année d'écart puis finalement de ne se connaître qu'une vingtaine d'années plus tard
9. Un jour il oubliera sa tête en plus de son portable, son portefeuille, ses clés ...
10. Petit meusien deviendra Docteur
12. Ca ne sert à rien de regarder dans l'équipe si Vézelise est passée en L1
15. Sans eux, je ne serais pas là
17. Tout un entourage sympathique et accueillant, presque une famille
18. Studio mobile avec vue sur le Loria
20. La vie peut-elle être modélisée par un processus Markovien ?
21. N°46
24. Quelle idée de cuisiner de la viande avec du coca!
25. « Mon... Mon... Mon... Monster Kill »

Remarque

Etant l'auteur des définitions, leur contexte m'est propre et il n'est pas forcément aisé de trouver les noms. Aussi la grille réponse est donnée en annexe A.4.

献给我的漂亮静薇，
她在我秘密的花园长大

A ma jolie fleur tranquille
qui pousse dans mon jardin secret

Table des matières

Introduction générale **xv**

Chapitre 1
Architecture d'un système de reconnaissance automatique de la parole **1**

1.1 Introduction . . . 2
1.2 Architecture d'un système de reconnaissance . . . 2
1.3 Paramétrisation du signal . . . 3
1.3.1 Les paramètres MFCC . . . 4
1.4 Modélisation de la parole – Modélisation acoustique . . . 5
1.4.1 Modèles de Markov cachés . . . 5
1.4.2 Apprentissage des modèles de Markov cachés . . . 6
1.4.2.1 L'algorithme de Baum et Welch . . . 7
1.4.2.2 La méthode "forward" . . . 9
1.4.2.3 La méthode "backward" . . . 9
1.5 Lexique et modèle de langage . . . 10
1.5.1 Lexique . . . 10
1.5.2 Modèle de langage . . . 10
1.6 Principe de fonctionnement d'un moteur de reconnaissance . . . 12
1.6.1 L'algorithme de Viterbi . . . 13
1.6.1.1 Le principe de Viterbi . . . 13
1.6.1.2 Le principe d'optimalité de Bellman . . . 14
1.6.1.3 L'algorithme de Viterbi . . . 14
1.6.1.4 Algorithme de Viterbi dans le cas d'une phrase . . . 15
1.6.2 L'algorithme A^* . . . 16
1.6.3 Résultats de la reconnaissance . . . 18
1.6.3.1 N-meilleures phrases . . . 18
1.6.3.2 Graphe de mots . . . 19
1.6.3.3 Réseau de confusion . . . 20

1.7 Conclusion 21

Chapitre 2
Mesures de confiance

2.1 Introduction 25
2.2 Exemples d'applications et intérêt des mesures de confiance 26
2.2.1 Reconnaissance de la parole : transcription 26
2.2.2 Détection des mots hors vocabulaire 27
2.2.3 Détection de mots clés 27
2.2.4 Dialogue Homme/Machine 29
2.2.5 Apprentissage semi-supervisé 29
2.2.6 Adaptation 31
2.3 Mesures de confiance 31
2.3.1 Critères *non probabilistes* 32
2.3.1.1 Stabilité acoustique 32
2.3.1.2 Densité d'hypothèses 32
2.3.1.3 Dépendance des mots 33
2.3.2 Critères relatifs au modèle de langage 33
2.3.2.1 Le modèle de langage 33
2.3.2.2 Repli du modèle de langage 34
2.3.3 Critères sémantiques et syntaxiques 35
2.3.3.1 Analyse sémantique latente 35
2.3.3.2 Information mutuelle 36
2.3.3.3 Catégorie d'un mot 37
2.3.4 Autres critères empiriques 37
2.3.5 Mesures fondées sur le rapport de vraisemblance 38
2.3.5.1 Tests d'hypothèse 38
2.3.5.2 Rapport de vraisemblance 38
2.3.5.3 Modèle / Anti-Modèle 39
2.3.5.4 Modèle générique 39
2.3.5.5 Modèles compétitifs 40
2.3.6 Mesures fondées sur les probabilité *a posteriori* 41
2.3.6.1 Mesures fondées sur la liste de n-meilleures phrases 42
2.3.6.2 Mesures fondées sur les graphes de mots et l'algorithme de *forward-backward* 42
2.3.6.3 Mesure de confiance du système de reconnaissance *Julius* 44
2.3.6.4 Mesures fondées sur les réseaux de confusion 45

2.3.6.5 Récapitulatif des mesures fondées sur une estimation de la probabilité *a posteriori* . . . 46
2.3.7 Combinaison de mesures de confiance . . . 46
2.3.7.1 Combinaisons de mesures et d'heuristiques . . . 46
2.3.7.2 Combinaison de systèmes de reconnaissance . . . 47
2.4 Méthodes d'évaluation . . . 48
2.4.1 Taux d'égale erreur . . . 48
2.4.2 Taux d'erreur de confiance . . . 50
2.4.3 Entropie croisée normalisée . . . 50
2.4.4 Coefficient de corrélation . . . 51
2.4.5 Rappel / Précision . . . 52
2.4.6 Synthèse . . . 53
2.5 Quelques résultats . . . 53
2.6 Conclusion . . . 54

Chapitre 3
Propositions de nouvelles mesures de confiance

3.1 Objectifs . . . 58
3.1.1 Applications visées . . . 58
3.1.1.1 Transcription d'émissions . . . 58
3.1.1.2 Transcription de cours en salle de classe . . . 59
3.1.1.3 Détection de mots clés . . . 61
3.1.2 Nos mesures de confiance : dans quel but ? comment ? . . . 61
3.1.2.1 Caractéristiques principales de nos mesures de confiance . . . 61
3.1.2.2 Quels types de mesures de confiance ? . . . 62
3.1.2.3 Source d'information pour calculer les mesures . . . 62
3.1.2.4 Mesures de confiance à quel niveau ? . . . 63
3.2 Mesures trame-synchrones . . . 63
3.2.1 Définition des mots concurrents de l'ensemble E . . . 64
3.2.2 Gestion des occurrences multiples . . . 64
3.2.3 Mesure fondée sur la probabilité unigramme . . . 65
3.2.4 Introduction de facteurs d'échelle . . . 66
3.2.5 Mesure fondée sur la probabilité bigramme . . . 66
3.2.6 Mesure fondée sur la probabilité trigramme . . . 67
3.2.7 Implantation . . . 69
3.2.7.1 Construction de l'ensemble $\widehat{E}$ . . . 69

3.2.7.2 Calcul des mesures fondées sur les probabilités unigramme, bigramme et trigramme . . . 69
3.3 Mesures locales . . . 71
3.3.1 Mesures fondées sur la probabilité *a posteriori* . . . 72
3.3.2 Définition des voisinages . . . 73
3.3.3 Introduction d'un facteur de flexibilité η . . . 73
3.4 Homogénéisation de la répartition des valeurs de confiance . . . 74
3.5 Complexité de nos mesures de confiance . . . 77
3.5.1 Mesures trame-synchrones . . . 77
3.5.2 Mesures locales . . . 77
3.6 Conclusion . . . 78

Chapitre 4
Conditions expérimentales

4.1 Introduction . . . 82
4.2 Moteur de reconnaissance : Julius . . . 82
4.2.1 La première passe de *Julius* . . . 83
4.2.2 La deuxième passe de *Julius* . . . 83
4.2.3 Options de compilation . . . 84
4.2.4 Le graphe de mots . . . 84
4.3 Paramétrisation . . . 85
4.4 Modèles acoustiques . . . 85
4.4.1 Monophones . . . 85
4.4.2 Triphones . . . 85
4.5 Lexique et modèle de langage . . . 86
4.5.1 Utilisés conjointement avec les modèles monophones . . . 86
4.5.2 Utilisés conjointement avec les modèles triphones . . . 86
4.6 Corpus de développement et de test . . . 87
4.7 Complexité de nos mesures de confiance . . . 87
4.7.1 Mesures trame-synchrones . . . 88
4.7.2 Mesures locales . . . 88
4.8 Conclusion . . . 88

Chapitre 5
Evaluation des mesures de confiance avec le taux d'EER

5.1 Introduction . . . 90
5.2 Protocole d'évaluation . . . 90

5.3 Mesure de référence – Probabilité *a posteriori* globale . . . 91
5.4 Mesures trame-synchrones . . . 92
5.4.1 Mesure fondée sur la probabilité unigramme . . . 93
5.4.1.1 Gestion des occurrences multiples par sommation . . . 94
5.4.2 Mesure fondée sur la probabilité bigramme . . . 95
5.4.2.1 Gestion des occurrences multiples par maximisation . . . 95
5.4.2.2 Gestion des occurrences multiples par sommation . . . 96
5.4.2.3 Prédécesseur au sens de Viterbi . . . 96
5.4.2.4 Filtrage par les n-meilleures phrases . . . 97
5.4.2.5 Probabilité bigramme seule . . . 98
5.4.2.6 Inclusion/exclusion du mot $\widehat{w}$ dans l'ensemble $\widehat{E}$ . . . 99
5.4.2.7 Probabilité bigramme inverse . . . 100
5.4.2.8 Homogénéisation des valeurs . . . 100
5.4.3 Mesure fondée sur la probabilité trigramme . . . 102
5.4.3.1 Probabilité trigramme inverse . . . 104
5.4.4 Synthèse . . . 105
5.5 Mesures locales . . . 106
5.5.1 Mesure à voisinage symétrique . . . 106
5.5.2 Mesure à voisinage asymétrique . . . 108
5.5.3 Homogénéisation des valeurs de confiance . . . 110
5.5.4 Synthèse . . . 112
5.6 Influence de la taille des mots . . . 113
5.7 Comparaison avec la mesure de confiance intégrée dans le système de reconnaissance Julius . . . 116
5.8 Evaluation sur le corpus de test et conclusion . . . 117

Chapitre 6
Evaluation dans le cadre d'applications spécifiques

6.1 Introduction . . . 122
6.2 Application à la détection de mots clés . . . 122
6.3 Intégration d'une mesure de confiance dans le moteur de reconnaissance . . . 125
6.3.1 Méthodologie . . . 125
6.3.2 Expérimentation . . . 126
6.4 Transcription de cours en salle de classe . . . 129
6.4.1 Présentation du système initial . . . 129
6.4.2 Utilisation de la mesure de confiance . . . 130
6.4.3 Protocole de test . . . 131

6.5 Conclusion 135

Conclusion et perspectives **137**

Annexe A **145**

A.1 Entropie croisée normalisée 145

A.2 Taux d'erreur de confiance 146

A.3 Influence de la taille des mots 146

A.4 Questionnaire pour l'évaluation des transcriptions pour malentendants 149

Glossaire **151**

Bibliographie **153**

Publications personnelles **163**

Liste des figures

1.1 Architecture d'un système de reconnaissance automatique de la parole et des apprentissages nécessaires. . . . 2
1.2 Etapes de calcul des coefficients cepstraux à échelle Mel. . . . 4
1.3 Filtres triangulaires à échelle Mel (20 bandes). . . . 4
1.4 HMM gauche-droite à trois états. . . . 5
1.5 Graphe de Viterbi pour un HMM à 3 états gauche-droite et une séquence de 10 observations 13
1.6 Exemple d'un extrait de graphe afin d'illustrer le principe d'optimalité de Bellman. 14
1.7 Exemple d'un graphe de mots 19
1.8 Second exemple d'un graphe de mots 20
1.9 Exemple d'un réseau de confusion 20

2.1 Etapes de réalisation d'un apprentissage semi-supervisé avec l'utilisation d'une mesure de confiance. . . . 30
2.2 Exemple d'une courbe ROC-DET. L'intersection entre la première bissectrice et la courbe détermine le point EER. . . . 49

3.1 Les 5 positions de la main pour le codage des voyelles phonétiques en Langage Parlé Complété. . . . 59
3.2 Les 8 configurations de doigts pour le codage des consonnes phonétiques en Langage Parlé Complété. . . . 60
3.3 Tête codeuse de synthèse développée au Loria pour le projet LABIAO (le son « pa » en LPC). . . . 60
3.4 Illustration du voisinage pris en compte pour la mesure de confiance symétrique de paramètre de taille x. . . . 73
3.5 Illustration du voisinage pris en compte pour la mesure de confiance asymétrique de paramètre de taille x et y. . . . 73
3.6 Distribution du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique pour tous les mots d'un ensemble de graphes de mots. . . . 75
3.7 Distribution du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique pour les mots faisant partie d'un ensemble de phrases reconnues. . . . 76

5.1 Courbe DET de la mesure de référence fondée sur la probabilité *a posteriori* globale ($\alpha = 0,1$), ($\beta = 1$) et ($\eta = 1$). EER = 22,0% (corpus de développement). . . . 92
5.2 Courbe DET de la mesure de confiance fondée sur la probabilité unigramme ($\alpha = 0,1$), ($\beta = 0,5$) et ($\varepsilon = 0,1$). EER = 37,6% (corpus de développement). . . . 94

5.3 Distribution du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique sur le corpus de développement pour la mesure bigramme ($\alpha = 0,1$), ($\beta = 0,95$) et ($\varepsilon = 0,1$). 101
5.4 Variation du taux d'EER de la mesure de confiance fondée sur la probabilité trigramme, en fonction du rapport des facteurs d'échelle linguistique et acoustique β/α ($\alpha = 0,1$ et $\varepsilon = 0,1$). 103
5.5 Courbe du taux d'EER de la mesure locale à voisinage symétrique relativement à différentes tailles de voisinage. ($\alpha = 0,1$), ($\beta = 0,95$) et ($\eta = 0,5$) 107
5.6 Taux d'EER de la mesure de confiance locale à voisinage asymétrique à taille de voisinage passé variable et taille de voisinage futur fixe (0, 40, 60, et 84 trames) . 109
5.7 Répartition du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique sur le corpus de développement pour la mesure locale symétrique avec voisinage de 84 trames, ($\alpha = 0,1$), ($\beta = 0,95$) et ($\eta = 0,5$) 111
5.8 Répartition du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique sur le corpus de développement pour la mesure locale asymétrique trame-synchrone prenant en compte tout le voisinage passé depuis le début de la phrase, ($\alpha = 0,1$), ($\beta = 0,95$) et ($\eta = 0,5$) 112
5.9 Evolution du taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure de référence 114
5.10 Evolution du taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure locale symétrique 115
5.11 Evolution du taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure trame-synchrone bigramme directe 115
5.12 Répartition des mots de la reconnaissance pour le corpus de développement selon leur taille en phonèmes 116
5.13 Courbe DET de la mesure de confiance intégrée dans Julius ainsi que celles de la mesure locale symétrique avec voisinage de 84 trames et de la mesure trame-synchrone bigramme inverse. 117

6.1 Evolution du nombre de fausses acceptations et du nombre de bons mots clés restant en fonction du seuil de décision (corpus de développement). 123
6.2 Evolution du nombre de fausses acceptations et du nombre de bons mots clés restant en fonction du seuil de décision (corpus de test). 124
6.3 Distribution du taux de mots corrects en fonction de la valeur moyenne de confiance pour 20 intervalles de taille identique sur le corpus utilisé dans le cadre de la mesure bigramme intégrée dans le moteur de reconnaissance. 127
6.4 Tête codeuse de synthèse développée au Loria pour le projet LABIAO (le son « pa » en LPC). 129

A.1 Evolution des taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure de référence 147
A.2 Evolution des taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure locale symétrique 148
A.3 Evolution des taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure trame-synchrone bigramme directe 148

Liste des tableaux

1.1 Exemple de liste des 5 meilleures phrases issues d'un système de reconnaissance. . 18

2.1 Résultats obtenus par différentes mesures de confiance sur différents corpus. . . . 54

5.1 Taux d'EER de la mesure de référence fondée sur la probabilité *a posteriori* globale calculée sur la phrase complète avec différents facteurs d'échelle et facteur de flexibilité (corpus de développement). 92

5.2 Taux d'EER obtenus par la mesure de confiance unigramme avec différents facteurs d'échelle et de relâchement (corpus de développement). 93

5.3 Taux d'EER des mesures de confiance unigramme avec gestion des occurrences multiples par maximisation et sommation avec différents facteurs d'échelle et $\varepsilon = 0,1$ (corpus de développement). 94

5.4 Taux d'EER obtenus par la mesure de confiance bigramme avec gestion par maximisation pour différents facteurs d'échelle et de relâchement (corpus de développement). 95

5.5 Taux d'EER des mesures de confiance bigramme avec gestion des occurrences multiples par maximisation et sommation avec différents facteurs d'échelle et $\varepsilon = 0,1$ (corpus de développement). 96

5.6 Taux d'EER des mesures de confiance bigramme avec gestion par maximisation et avec précédents temporels directs ou avec précédent au sens de Viterbi avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement). 97

5.7 Taux d'EER de la mesure bigramme avec et sans filtrage des mots précédents par les n-meilleures phrases, $(\alpha = 0,1)$, $(\beta = 0,95)$ (corpus de développement). 98

5.8 Taux d'EER de la mesure de confiance bigramme avec prédécesseurs temporels ou de Viterbi et avec ou sans probabilité unigramme. 99

5.9 Taux d'EER des mesures de confiance bigramme, mesures incluant ou excluant $\widehat{w}$ de l'ensemble $\widehat{E}$ avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement). 99

5.10 Taux d'EER des mesures de confiance fondée sur la probabilité bigramme directe et inverse avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement). . 100

5.11 Taux d'EER des mesures de confiance bigramme avec gestion par maximisation et tous les précédents temporels directs, avec et sans homogénéisation des valeurs de confiance avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement). 102

5.12 Taux d'EER de comparaison de la mesure de confiance fondée sur la probabilité trigramme et de sa version modifiée, $\varepsilon = 0,1$ (corpus de développement). 104

5.13 Taux d'EER des mesures de confiance fondée sur la probabilité trigramme directe et inverse avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement). . 105

5.14 Taux d'EER obtenus par la mesure de confiance locale fondée sur la probabilité *a posteriori* avec un voisinage symétrique de 84 trames, pour différents facteurs d'échelle et de relâchement (corpus de développement). 107
5.15 Synthèse des résultats obtenus par nos mesures de confiance ainsi que par la mesure de référence sur corpus de développement en taux d'EER et sur le corpus de test en taux de fausses alarmes (FA), taux de faux rejets (FR) et de CER. 119

6.1 Liste des 33 mots clés. 122
6.2 Taux d'erreur en mots à la fin de la première passe suivant différentes intégrations de la valeur de confiance. 128
6.3 Taux d'erreur en mots à la fin de la deuxième passe suivant différentes intégrations de la valeur de confiance. 128
6.4 Exemple des valeurs de confiance des mots d'une phrase. 130
6.5 Taux d'erreur en mots sur les parties retranscrites des textes suivant les différentes modalités. 134
6.6 Taux de réponse aux questions des textes selon les différentes modalités. 135

Introduction générale

Au commencement était la parole (Jean 1 :1-5).

Que ce soit sous la forme de grognements comme nos ancêtres primitifs, ou sous une forme plus évoluée et complexe à notre ère, le langage et la communication tiennent une place prépondérante dans la société humaine. Depuis ces temps immémoriaux, la parole a toujours été le support majeur d'expression des êtres humains. Grâce à la voix, les personnes peuvent partager des informations, dialoguer, exprimer des sentiments, etc. Bien que tout être humain soit capable de s'exprimer par la parole depuis des millénaires, les mécanismes associés à la production ou à l'acquisition de la parole sont complexes et ne sont pas encore totalement maîtrisés.

La production d'un son est le résultat d'une combinaison de nombreuses interactions mécaniques et physiologiques qui vont influencer ses caractéristiques acoustiques. Par exemple, un homme, une femme, ou un enfant auront une fréquence fondamentale différente et donc le son émis sera lui même différent. Des caractéristiques plus subtiles sont également transmises via la parole. Par exemple, comment expliquer le fait que nous puissions *savoir* avec une quasi certitude qu'une personne parle au téléphone en souriant ?

Cette complexité au niveau de la production du son amène également de nombreuses difficultés au niveau de l'acquisition du langage et de sa compréhension. Depuis leur plus jeune âge, les enfants écoutent les adultes parler, tentent de reproduire tant bien que mal ce qu'ils entendent mais également essaient de donner un sens à ce flot sonore. Pour cela plusieurs étapes sont nécessaires : segmenter la phrase en mots, les reconnaître, analyser leur signification et finalement comprendre la phrase.

L'intelligence artificielle voudrait pouvoir réaliser ces étapes qu'effectuent quotidiennement les humains, et même les enfants, par le biais d'une machine, d'un processus automatique. Avec les débuts des enregistrements sonores de voix ou de musique, de nombreuses personnes ont tenté d'analyser, de retrouver sur ces traces de voix ce qui avait été prononcé. Trouver le secret qui permet de distinguer un « a » d'un « o ». Les premières expériences peuvent apparaître de nos jours comme très grossières, voire inimaginables, mais une grande partie des connaissancess sur la production et l'analyse de la parole vient de ces études du signal brut.

Puis des personnes de tous horizons, chercheurs ou auteurs, ont commencé à imaginer des applications associées à la reconnaissance de ces sons. Avec l'utilisation grandissante des ordinateurs et l'idée d'une intelligence artificielle, la science fiction a souvent donné des exemples d'application de la reconnaissance vocale par des machines, des ordinateurs.

Par exemple des robots serviteurs commandés à la voix : mécaniciens ou traducteurs dans un univers futuriste comme Star Wars. Ou bien, dans un contexte plus contemporain l'exemple d'une voiture équipée d'un ordinateur capable de dialoguer et de montrer un comportement quasi humain. Cette vision d'une simple voiture qui puisse à la fois comprendre, s'exprimer et même se comporter comme un être humain, laisse dans l'esprit du public l'idée que cette technologie est

presque « existante » alors que cela est encore un objectif difficile à atteindre pour les chercheurs en intelligence artificielle.

Depuis plusieurs années nous pouvons commander oralement un ordinateur, pour des tâches simples, avec un dialogue strictement directif et limité ; des applications de dictée vocale existent ; des applications de type renseignements téléphoniques voient le jour, acceptant divers scénarios de dialogue, plus ou moins flexibles. Toutefois nous sommes encore très loin des rêves des auteurs de science fiction. En effet, les applications citées précédemment existent mais avec des conditions d'utilisation extrêmement restrictives : vocabulaire limité, généralement une seule langue traitée, conditions d'utilisation optimales, importante sensibilité aux bruits, longue phase d'apprentissage, etc.

Actuellement, le but de la recherche en reconnaissance automatique est de considérer toutes ces limitations d'utilisation, de les dépasser et de comprendre de mieux en mieux les mécanismes liés à la parole (production, perception, compréhension).

Parler à une machine et voir celle-ci retranscrire mot pour mot ce qui a été prononcé présente un côté magique et fascinant. Les applications de dictée vocale ou plus généralement de transcription d'un document sonore, ont comme objectif de fournir sous forme de texte la parole contenue dans le signal audio traité. Cette tâche est une des plus difficiles du domaine de la reconnaissance automatique de la parole.

Les premiers systèmes de reconnaissance ne traitaient que des mots isolés, puis, la puissance des ordinateurs augmentant, le traitement de phrases entières a été envisagé. Cependant, la modélisation de la grammaire d'une langue naturelle est difficile voire impossible, car celle-ci a évolué au cours du temps, avec des modifications, des simplifications, des habitudes qui ne suivent pas forcément les règles de la langue. De plus, chaque règle n'a-t-elle pas son exception ?

Ainsi les premiers systèmes traitant des phrases étaient fondés sur une modélisation limitée de la langue de sorte que le locuteur n'avait que peu de choix dans les phrases qu'il était autorisé à prononcer.

A l'heure actuelle, les systèmes de reconnaissance automatique de la parole acceptent des conditions d'utilisation de moins en moins contraintes, plus proche des conditions d'utilisation rencontrées dans la réalité. Cela implique de prendre en compte des phénomènes de plus en plus complexes comme le bruit, les tours de parole, la langue utilisée, la langue maternelle, etc.

Plus les conditions expérimentales sont difficiles et plus le système est à même de faire des erreurs. L'incidence de ces erreurs peut être plus ou moins importante : une erreur d'accord grammatical implique très rarement une mauvaise compréhension, ce qui est souvent le cas pour un mot totalement mal reconnu n'ayant aucun sens avec le contexte.

Plusieurs directions sont envisageables afin d'éviter ces erreurs :

- affiner les différents modèles mis en jeu dans le processus de reconnaissance,
- explorer de nouvelles directions de recherche afin de trouver des modèles plus robustes (paramétrisation, classifieur, etc),
- utiliser d'autres informations afin de corriger ou détecter les erreurs potentiellement commises (débruitage, mesures de confiance).

Concernant ce dernier point, il serait intéressant de définir des indices supplémentaires, autres que le résultat de la reconnaissance, afin d'estimer la qualité de la phrase reconnue, puis de prendre en compte ces indices afin d'effectuer soit des corrections, soit des alertes. Les mesures de confiance remplissent ce rôle. L'objectif d'une mesure de confiance est de pouvoir estimer au mieux la probabilité qu'une phrase ou qu'un des mots reconnus soit juste.

Concevoir des mesures de confiance est une problématique difficile apparue en reconnaissance automatique de la parole depuis une dizaine d'années. En effet, choisir et créer une mesure de

confiance n'est pas chose simple, et quand bien même le cadre théorique nous assure d'une bonne efficacité, les résultats concrets sont souvent insuffisants pour être exploités dans des applications. Toutefois dans plusieurs situations les mesures de confiance apportent réellement une connaissance supplémentaire, notamment pour les tâches d'acceptation/rejet d'hypothèses (dialogues homme/machine, détection de mots clés), de sélections de données (apprentissage semi-supervisé) et plus généralement dans les tâches de transcription.

Dans cette thèse nous nous sommes intéressé aux mesures de confiance dans le cadre des applications de reconnaissance de la parole grand vocabulaire et à flux continu. Nous souhaitions définir des mesures de confiance pouvant être calculées sans attendre que le signal (la phrase) ne soit décodé dans son intégralité par le système de reconnaissance. Les applications visées sont plus particulièrement :

- la transcription d'émissions radiophoniques à la volée dans laquelle nous pourrons mettre en couleur les mots de faible confiance,
- la transcription de cours en salle de classe pour des élèves sourds ou malentendants,
- la détection de mots clés à la volée.

Nous décrirons en détail ces applications dans le chapitre 3 de ce mémoire. Toutefois, une caractéristique importante de ces flux, qui seront décodés à la volée, est qu'ils sont virtuellement sans fin, à l'opposé des documents pré-enregistrés. Le fait que la fin du flux ne soit pas déterminée empêche l'utilisation de méthode ou de calcul nécessitant la connaissance et le traitement du signal dans son intégralité. Or actuellement, bien qu'il existe des systèmes de reconnaissance capables de traiter des flux en direct, peu de mesures de confiance peuvent être calculées dans ces conditions. C'est la raison pour laquelle nous avons décidé de définir de nouvelles mesures de confiance qui sont trame-synchrones ou qui ne nécessitent qu'une partie de la phrase pour pouvoir être estimées. Les mesures trame-synchrones permettent de calculer une valeur de confiance exactement en même temps que le décodage de la phrase est effectué par le moteur de reconnaissance. Les mesures locales que nous définissons utilisent des connaissances futures par rapport au mot dont nous voulons estimer la confiance. Cependant, la partie future est de taille limitée, ce qui implique simplement un court délai avant de pouvoir calculer la valeur de confiance d'un mot.

Ce mémoire débute par une présentation de l'architecture générale des systèmes de reconnaissance actuels dans laquelle nous décrivons plus particulièrement les aspects liés au cadre de notre étude.

Le chapitre 2 est consacré à l'état de l'art. Avant de présenter les principales mesures de confiance introduites en reconnaissance de la parole, nous montrons leur utilité pour certaines applications phares de la reconnaissance de la parole. Enfin nous terminons ce chapitre par une description des principales méthodes d'évaluation des mesures de confiance.

Le chapitre 3 concerne nos travaux. Après une introduction des objectifs de notre étude, notamment en ce qui concerne les applications ciblées, nous présentons les nouvelles mesures de confiance que nous avons définies au cours de cette étude : des mesures trame-synchrones et des mesures locales.

Afin d'évaluer les performances de nos mesures de confiance en situation réelle, nous avons défini des conditions d'expérimentation qui sont détaillées dans le chapitre 4 : le moteur de reconnaissance utilisé, les différentes modélisations acoustiques et linguistiques choisies, ainsi que les corpus de développement et de test.

Dans le chapitre 5, les performances des différentes mesures et de leurs variantes sont évaluées selon un critère indépendant de toute application.

Le chapitre 6, quant à lui, regroupe les expérimentations que nous avons menées sur certaines de nos mesures de confiance dans le cadre de deux applications bien spécifiques : une détection

de mots clés et une expérience qualitative de transcription de cours pour des enfants sourds ou malentendants. Ce chapitre se termine par la donnée de quelques résultats sur l'intégration d'une mesure trame-synchrone dans le processus de décodage du système de reconnaissance. Nous concluons ce mémoire par une discussion de nos travaux et de leurs résultats et par une présentation des perspectives envisageables.

Chapitre 1

Architecture d'un système de reconnaissance automatique de la parole

Sommaire

1.1 Introduction 2
1.2 Architecture d'un système de reconnaissance 2
1.3 Paramétrisation du signal 3
1.3.1 Les paramètres MFCC 4
1.4 Modélisation de la parole – Modélisation acoustique 5
1.4.1 Modèles de Markov cachés 5
1.4.2 Apprentissage des modèles de Markov cachés 6
1.4.2.1 L'algorithme de Baum et Welch 7
1.4.2.2 La méthode "forward" 9
1.4.2.3 La méthode "backward" 9
1.5 Lexique et modèle de langage 10
1.5.1 Lexique 10
1.5.2 Modèle de langage 10
1.6 Principe de fonctionnement d'un moteur de reconnaissance 12
1.6.1 L'algorithme de Viterbi 13
1.6.1.1 Le principe de Viterbi 13
1.6.1.2 Le principe d'optimalité de Bellman 14
1.6.1.3 L'algorithme de Viterbi 14
1.6.1.4 Algorithme de Viterbi dans le cas d'une phrase 15
1.6.2 L'algorithme A^* 16
1.6.3 Résultats de la reconnaissance 18
1.6.3.1 N-meilleures phrases 18
1.6.3.2 Graphe de mots 19
1.6.3.3 Réseau de confusion 20
1.7 Conclusion 21

1.1 Introduction

L'objectif d'un système de reconnaissance automatique de la parole est de transcrire la parole contenue dans un document sonore donné en entrée. La transcription se présente habituellement sous la forme d'une séquence de mots. Un défi actuel est de pouvoir reconnaître de la parole spontanée, utilisant un langage naturel.

Le résultat délivré par le système de reconnaissance est la solution d'un problème combinatoire complexe. Depuis quelques décennies, les systèmes qui permettent d'obtenir les meilleures performances sont fondés sur des modélisations statistiques des sons élémentaires (modélisation acoustique) et du langage (modèle linguistique n-grammes). Le système de reconnaissance fournit alors comme solution la séquence de mots la plus probable correspondant au segment de parole analysé, en général une phrase.

Par ailleurs, le système ou moteur de reconnaissance n'utilise pas directement le signal sonore brut mais effectue un pré-traitement du signal afin d'en extraire des paramètres acoustiques plus robustes et plus discriminants.

Nous allons donc brièvement décrire dans ce chapitre les différents concepts que sont la paramétrisation, la modélisation acoustique et la modélisation linguistique. En revanche nous détaillerons un peu plus l'algorithme d'apprentissage des modèles acoustiques ainsi que le moteur de reconnaissance ; plus particulièrement l'algorithme de reconnaissance et les structures de données associées, nos travaux de recherche étant directement liés à ces derniers éléments.

1.2 Architecture d'un système de reconnaissance

Pour un segment sonore donné en entrée, un système de reconnaissance de la parole délivre une transcription écrite de la parole contenue dans ce segment. La figure 1.1 présente les principales étapes d'un système de reconnaissance. Le processus de reconnaissance nécessite la définition d'une paramétrisation du signal et la fourniture de plusieurs données pré-calculées : les modèles acoustiques, le lexique et les modèles linguistiques. La construction des modèles et du lexique nécessite des apprentissages qui doivent être réalisés au préalable.

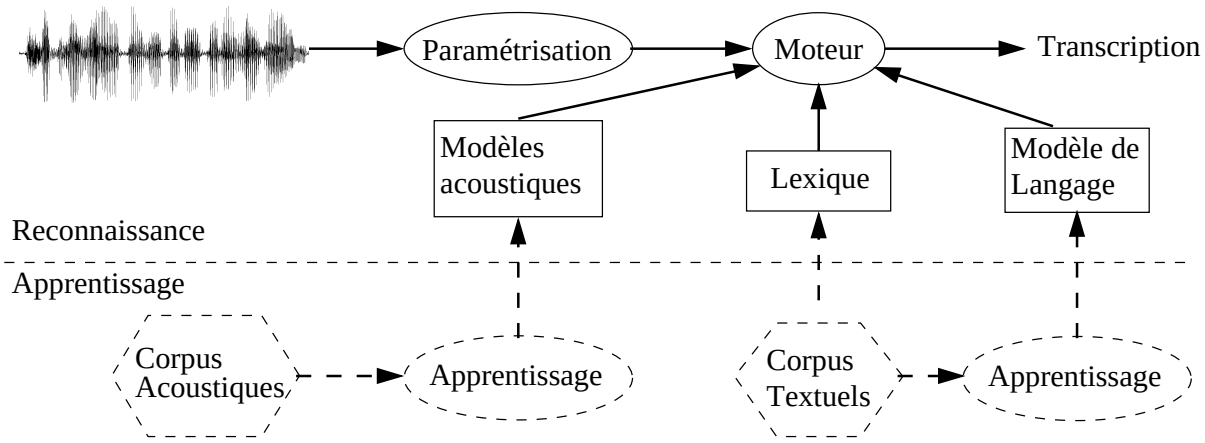


FIG. 1.1 – Architecture d'un système de reconnaissance automatique de la parole et des apprentissages nécessaires.

La première étape consiste à transformer le signal audio brut en paramètres plus robustes et plus discriminants. Ces paramètres vont servir à la fois à la construction des modèles acoustiques et au moteur de reconnaissance. Au cours de cette étape de paramétrisation, le signal sonore est tout d'abord échantillonné. Puis, plusieurs traitements mathématiques sont appliqués sur ces échantillons afin d'obtenir des vecteurs de paramètres, appelés *observations*.

La seconde étape représente le cœur du système : le moteur de reconnaissance. Le moteur utilise les structures de données externes suivantes :

- des modèles acoustiques,
- un lexique (vocabulaire),
- un modèle de langage (grammaire).

Les modèles acoustiques sont la représentation probabiliste d'unités élémentaires de parole : phones, phonèmes, syllabes ou mots.

Le lexique définit l'ensemble des mots qui pourront être reconnus par le système. Un mot qui n'est pas dans ce vocabulaire prédéfini ne pourra jamais figurer dans un résultat du système de reconnaissance.

Le modèle de langage tente de représenter, par le biais de probabilités, des phénomènes syntaxiques de la langue. La modélisation généralement utilisée est construite à partir de l'analyse de séquences de mots dans un corpus textuel. Trois types d'information sont habituellement estimés : les probabilités d'apparition d'un mot (unigrammes), d'une séquence de deux mots successifs (bigrammes) et d'une séquence de trois mots (trigrammes).

Les modélisations acoustiques, linguistiques et le lexique sont construits au préalable à partir d'importants corpus de parole et de texte, distincts de ceux sur lesquels le système sera testé. Une fois ces phases d'apprentissage réalisées, le moteur a toutes les données nécessaires pour effectuer la reconnaissance d'un signal de parole.

Dans les sections suivantes, nous décrivons un peu plus en détails ces différentes étapes préliminaires de paramétrisation, de modélisation ainsi que le moteur de reconnaissance.

1.3 Paramétrisation du signal

La paramétrisation du signal acoustique joue un rôle majeur dans le système de reconnaissance de la parole. Son objectif est de transposer le signal sonore brut dans un domaine plus robuste et plus discriminant. C'est-à-dire que les paramètres devront être les plus indépendants possibles des conditions d'enregistrement, mais aussi permettre de distinguer au maximum les différentes unités élémentaires de parole entre elles.

Par exemple, différentes paramétrisations peuvent être envisagées dans le domaine spectral : le spectre du signal, les formants, les coefficients de codage prédictif linéaire (Linear Predictive Coding –LPC) [Markel 76, Rabiner 78, Hai 03]. D'autres techniques proposent d'ajouter au domaine de paramétrisation des connaissances issues de la psycho-acoustique humaine. C'est notamment le cas de la prédiction linéaire perceptive (Perceptual Linear Prediction – PLP) [Hermansky 90]) ou de la transformation bilinéaire Bark (Bark Bilinear Transform – BBT) [Smith 95], qui toutes deux se basent sur une résolution non linéaire en fréquence à l'aide de l'échelle Bark. La paramétrisation la plus largement répandue en reconnaissance automatique de la parole se situe dans le domaine cepstral et utilise les coefficients cepstraux à échelle Mel encore appelés MFCC [Davis 80]. A la différence des coefficients spectraux, l'interprétation des coefficients MFCC n'est pas simple. Toutefois, ceux-ci demeurent globalement les plus robustes et les plus performants. Cependant, une nouvelle paramétrisation fondée sur les ondelettes semble avoir un fort potentiel

[Deviren 03]. Les ondelettes se placent dans un domaine temps-fréquence alors que les paramétrisations classiques ne contiennent plus d'informations temporelles. Mais la mise en place des ondelettes est difficile car ce type de paramétrisation n'est pas encore suffisamment maîtrisé.

Quelle que soit la paramétrisation, les dérivées d'ordre multiples des paramètres sont également associées aux valeurs statiques afin de tenir compte de la dynamique de la parole. L'évolution des paramètres au cours du temps est souvent une donnée plus importante que les valeurs des paramètres eux-mêmes.

Dans le cadre de nos travaux, notre système sera basé sur une paramétrisation par les cepstres à échelle Mel associés à leurs dérivées premières et secondes.

1.3.1 Les paramètres MFCC

Les principales étapes du calcul des coefficients cepstraux à échelle Mel (Mel Frequency Cepstral Coefficient – MFCC) sont décrites Figure 1.2. Le processus de calcul commence par un découpage du signal en fenêtres recouvrantes, puis les étapes d'obtention des MFCC sont successivement appliquées à chacune de ces fenêtres [Davis 80, Rabiner 93]. Ces étapes sont : une pré-accentuation afin de renforcer les hautes fréquences du spectre, l'utilisation de fenêtre de type Hamming, une transformée de Fourier pour passer dans le domaine spectral, puis un filtrage suivant l'échelle fréquentielle non linéaire Mel du logarithme du spectre et enfin une transformée de Fourier inverse afin de passer dans le domaine cepstral.

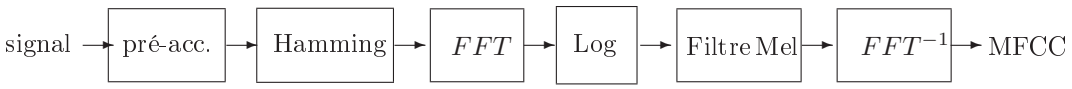


FIG. 1.2 – Etapes de calcul des coefficients cepstraux à échelle Mel.

L'échelle Mel, par rapport à une répartition linéaire en fréquence, tente de reproduire une caractéristique physiologique de l'oreille humaine. Des sons à une fréquence de 100 Hz et 150 Hz sont clairement distincts pour tous mais il nous est quasiment impossible de distinguer un son à 4000 Hz d'un son à 4050 Hz. L'échelle Mel schématise cette perception en définissant une échelle logarithmique de répartition des fréquences. Une représentation d'un banc de vingt filtres Mel est donnée Figure 1.3.

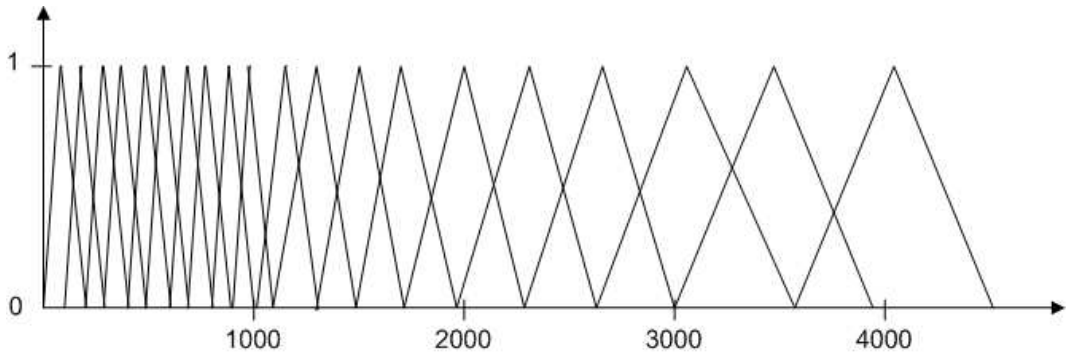


FIG. 1.3 – Filtres triangulaires à échelle Mel (20 bandes).

1.4 Modélisation de la parole – Modélisation acoustique

La modélisation acoustique permet au moteur de reconnaissance de déterminer quelles ont été les unités acoustiques prononcées (phones, phonèmes, syllabes, mots). La littérature propose plusieurs modélisations possibles et parmi les plus fréquentes se trouvent les réseaux de neurones [Robinson 88, Robinson 94, Tebelskis 95], les modèles de Markov cachés et les réseaux Bayesiens [Rabiner 89, Deviren 02]. Les modèles de Markov cachés (Hidden Markov Model - HMM) ont été introduits dans le domaine de la reconnaissance de la parole depuis déjà une trentaine d'années [Baker 75, Jelinek 76], et la majeure partie des modélisations actuelles sont fondées sur ces modèles. Une telle modélisation probabiliste de la parole peut être étendue par l'intermédiaire de structures telles que les HMM multidimensionnels ou encore par les réseaux Bayesiens dont les modèles de Markov cachés sont un cas particulier [Mari 97, Deviren 04].

1.4.1 Modèles de Markov cachés

Un modèle de Markov caché peut être décrit comme un automate probabiliste à N états comportant deux processus : un processus caché de changement d'état et un processus d'émission. Le processus de changement d'état est caché car celui-ci n'est pas observable. Cependant, par le processus d'émission, la transition du modèle dans un état génère une observation. La figure 1.4 représente un modèle de Markov caché à trois états.

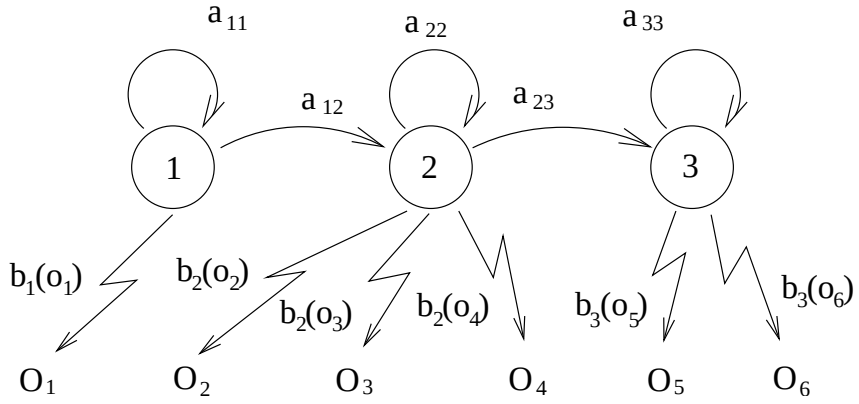


FIG. 1.4 – HMM gauche-droite à trois états.

La réalisation d'un processus de Markov caché se traduit par l'existence d'une séquence $Q = (q_0, \ldots, q_T)$ d'états de l'automate. Le processus d'émission du modèle de Markov caché associe à Q une séquence de T observations $O = (o_1, \ldots, o_T)$. Avant le début du processus, le système se trouve dans un état initial q_0 sans émettre d'observations. Au temps t, le HMM effectue une transition vers l'état q_t et émet l'observation o_t. Un modèle de Markov caché est caractérisé par trois paramètres :

- π_i, les probabilités initiales, c'est-à-dire la probabilité d'être dans l'état i de l'automate au temps 0,
$$\pi_i = P(q_0 = i), \quad \forall i \in \{1, N\}$$
- $A = ((a_{ij}))$, la matrice de transition entre les états de l'automate ; a_{ij} représente la probabilité de transition pour aller de l'état i à l'état j,
$$a_{ij} = P(q_t = j | q_{t-1} = i), \quad \forall i, j \in \{1, N\}^2$$

– $b_i(o_t)$, la distribution des probabilités d'émission de l'observation o_t à l'état i de l'automate,

$$b_i(o_t) = P(o_t|q_t = i), \quad \forall i \in \{1, N\}, \forall t \in \{1, T\}.$$

Pour chaque état, la probabilité d'émission représente la probabilité qu'un état de l'automate ait généré une observation particulière. Cette probabilité d'émission de l'observation est généralement modélisée par une somme pondérée de G fonctions de densité gaussienne $\mathcal{N}(\mu, \Sigma)$ (Gaussian Mixture Model - GMM) d'espérance μ et de matrice de covariance Σ. La probabilité d'observation est alors définie par l'équation suivante :

$$b_i(o_t) = \sum_{k=1}^{G} c_{ik}\ \mathcal{N}(o_t, \mu_{ik}, \Sigma_{ik}), \qquad \sum_{k=1}^{G} c_{ik} = 1 \tag{1.1}$$

chaque gaussienne ayant une densité de probabilité continue égale à

$$\frac{1}{\sqrt{(2\pi)^D det(\Sigma)}} exp(-\frac{1}{2}\ {}^t(o_t - \mu)\Sigma^{-1}(o_t - \mu))$$

pour laquelle o_t représente le vecteur d'observation à D composantes, μ le vecteur moyen de la gaussienne, et Σ la matrice de covariance.

La matrice de transition définit la topologie de l'automate du modèle de Markov caché. Dans l'exemple de la figure 1.4, le modèle à 3 états possède des transitions gauche-droite entre les états ainsi que des transitions de bouclage. Ce type de modélisation est employé pour définir des modèles de phonèmes. Dans ces modèles, les transitions sont gauche-droite (pas de retour arrière possible) pour essayer de représenter l'évolution acoustique d'un phonème au cours du temps : le début du phonème, sa partie centrale et la fin du phonème.

La modélisation HMM peut être appliquée de plusieurs manières pour traiter le cas des sons d'une langue. La plus simple est de faire autant de modèles que de phonèmes. Les modèles représenteront ce qu'on appelle des *monophones*. Cependant, il est possible de choisir une modélisation plus fine en tenant compte du contexte acoustique entourant le phonème. Ainsi, un modèle ne définit plus un phonème mais une réalisation particulière de ce phonème, dépendante du ou des phonèmes le précédant et de celui ou ceux lui succédant. Ce sont les *n-phones*.

Les modèles acoustiques, qu'ils soient monophones ou n-phones, sont appris sur un corpus acoustique contenant des exemples de parole et leur transcription phonétique. Le passage à des modèles n-phones augmente fortement le nombre de modèles à apprendre et en même temps diminue fortement le nombre d'occurrences de chacun d'eux dans le corpus d'apprentissage. Des méthodes ont alors été développées afin de limiter l'impact du manque d'exemples pour l'apprentissage des modèles. Une possibilité consiste à définir un ensemble fixe de gaussiennes qui seront partagées par les différents modèles ou par les états des modèles [Lee 00].

1.4.2 Apprentissage des modèles de Markov cachés

La phase de construction d'un modèle est le point crucial de tout système. L'apprentissage revêt donc une grande importance. Les modèles acoustiques nécessitent un important corpus sonore transcrit de plusieurs centaines d'heures. Une transcription phonétique est associée à chaque échantillon sonore de sorte qu'au final chaque modèle possède des représentants dans le corpus. Une fois l'ensemble des données prêt, la phase d'apprentissage des modèles s'effectue. Nous allons décrire dans cette section l'algorithme communément utilisé dans la phase d'apprentissage des modèles de Markov cachés pour la reconnaissance de la parole.

A partir d'exemples dont nous connaissons à la fois la séquence des modèles et la séquence d'observations engendrée, nous souhaitons déterminer les paramètres définissant les modèles de Markov cachés de chaque unité phonétique. Il nous faut donc estimer pour chaque modèle :

- les probabilités initiales π_i,
- les probabilités de transition a_{ij},
- les probabilités d'émissions $b_i(o_t)$ qui sont caractérisées par :
  - les moyennes μ_i,
  - les matrices de covariances Σ_i,
  - les coefficients du mélange de gaussiennes c_i.

Dans le cadre modèles de Markov cachés, la méthode communément utilisée repose sur le critère du maximum de vraisemblance (Maximum Likelihood – ML). Toutefois, d'autres méthodes ont été développées, par exemple la technique d'apprentissage discriminant fondée sur le critère du maximum d'information mutuelle (Maximum Mutual Information – MMI). Soit $\lambda = (\pi_i, a_{ij}, b_i)$ les paramètres définissant un modèle HMM, nous devons d'après le critère du maximum de vraisemblance, trouver un modèle Λ qui maximise $P(O|\lambda)$.

$$\Lambda = \arg\max_{\lambda} P(O|\lambda)$$

Or, il n'existe pas de méthode directe pour résoudre ce problème de maximisation de Λ.

1.4.2.1 L'algorithme de Baum et Welch

Baum a eu l'idée d'introduire d'autres fonctions redéfinissant le problème de recherche d'un système λ. Puis il a décrit un algorithme permettant l'estimation des nouveaux modèles de manière itérative [Baum 70].

Soit p une fonction positive, et $P(\lambda) = \int p(q, \lambda)\mathrm{d}q$, nous pouvons alors introduire une fonction auxiliaire Q :

$$Q(\lambda, \lambda') = \frac{1}{P(\lambda)} \int p(q, \lambda) \log p(q, \lambda')\mathrm{d}q$$

Baum a démontré d'une part la propriété suivante :

$$Q(\lambda, \lambda') - Q(\lambda, \lambda) \leq \log P(\lambda') - \log P(\lambda)$$

et d'autre part qu'en définissant la fonction T ainsi :

$$T(\lambda) = \arg\max_{\lambda'} Q(\lambda, \lambda')$$

nous avons alors l'inégalité suivante :

$$P(T(\lambda)) \geq P(\lambda).$$

L'algorithme de Baum et Welch consiste à trouver un nouveau modèle λ' qui maximise $Q(\lambda, \lambda')$. Cet algorithme est itératif et commence par un jeu de paramètres arbitraires λ_0. Ensuite, nous cherchons λ_1 qui maximise $Q(\lambda_0, \lambda)$, puis λ_2 qui maximise $Q(\lambda_1, \lambda)$, et ainsi de suite. Nous avons, de plus, la propriété $P(\lambda_2) \geq P(\lambda_1) \geq P(\lambda_0)$.

Il faut trouver maintenant une méthode pour maximiser la fonction Q.

Dans le cas des modèles de Markov considérés, $P(\lambda)$ s'écrit :

$$P(\lambda) = \sum_{q \in \Xi} \pi_{q_0} \prod_{t=1}^{T} a_{q_{t-1} q_t} b_{q_t}(o_t)$$

Ξ étant l'ensemble des chemins possibles pour un HMM gauche-droit.

Dans ce cas $Q(\lambda, \lambda')$ peut se réécrire sous la forme d'une somme de trois termes :

$$Q(\lambda, \lambda') = A(\pi_i) + B(a_{ij}) + C(b_i).$$

Ainsi, maximiser $Q(\lambda, \lambda')$ revient à maximiser les trois termes séparément. Or, la topologie de nos modèles force les valeurs des π_i, car nous commençons obligatoirement par le premier état du modèle. Les π_i sont donc constants. La maximisation des termes B et C conduisent à des formules de ré-estimation pour les probabilités d'observation et de transition. Nous donnons ces formules pour des probabilités d'observation monogaussiennes.

Aussi, pour les probabilités de transition a'_{ij} et pour une loi gaussienne $\mathcal{N}(\mu'_k, \Sigma'_k)$ à l'état k du modèle λ', il faut exprimer les quantités μ'_k, Σ'_k et a'_{ij} en fonction du modèle λ. Ces écritures n'étant pas immédiates, il est nécessaire d'introduire de nouvelles variables γ et ξ, puis α et β.

Les probabilités d'observation

Les formules de ré-estimation des probabilités d'observation pour une loi gaussienne $\mathcal{N}(\mu'_k, \Sigma'_k)$ du nouveau modèle λ' sont décrites par les équations suivantes :

$$\begin{aligned}
\mu'_k &= \frac{nb\ de\ fois\ à\ l'état\ k\ et\ observation\ de\ o_t}{nb\ de\ fois\ à\ l'état\ k} \\
&= \frac{\sum_{t=1}^{T} \gamma_t(k) o_t}{\sum_{t=1}^{T} \gamma_t(k)} \\
\Sigma'_k &= \frac{\sum_{t=1}^{T} \gamma_t(j)(o_t - \mu_j)\overline{(o_t - \mu_j)}}{\sum_{t=1}^{T} \gamma_t(j)}
\end{aligned}$$

γ étant la probabilité *a posteriori* de s'être trouvé à l'état i à l'instant t connaissant la séquence d'observations et le modèle :

$$\gamma_t(i) = P(q_t = i | O, \lambda)$$

Les probabilités de transition

Les valeurs des probabilités de transition sont :

$$a'_{ij} = \frac{nb\ de\ transitions\ ij}{nb\ trans.\ sortantes\ de\ i} = \frac{\sum_{t=1}^{T} \xi_t(i,j)}{\sum_{t=1}^{T} \xi_t(i)}$$

ξ étant la probabilité de s'être trouvé à l'état i à l'instant t, et à l'état j à l'instant $t+1$ connaissant la séquence d'observations et λ :

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j | O, \lambda)$$

Hélas, le problème est simplement reporté sur deux nouvelles inconnues, γ et ξ. Il est alors possible d'introduire de nouveau deux variables supplémentaires :

- soit $\alpha_t(i) = P(o_1 \dots o_t, q_t = i|\lambda)$, la probabilité d'observer la séquence $o_1 \dots o_t$ et d'être à l'état i à l'instant t sachant le modèle λ.
- soit également $\beta_t(i) = P(o_{t+1} \dots o_T|q_t = i, \lambda)$, la probabilité d'observer la séquence $o_{t+1} \dots o_T$ sachant λ, et d'être à l'état i au temps t.

Les valeurs de γ et ξ peuvent s'exprimer en fonction de α et β. Nous obtenons alors les équations suivantes :

$$\begin{aligned} \gamma_t(i) &= \frac{\alpha_t(i)\beta_t(i)}{\sum_{i=1}^{N} \alpha_t(i)\beta_t(i)} \\ \xi_t(i,j) &= \frac{\alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)}{\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_t(i)a_{ij}b_j(o_{t+1})\beta_{t+1}(j)} \end{aligned}$$

Par l'introduction de nouvelles inconnues α et β, Baum et Welch reformulent la définition des inconnues γ et ξ. Toutefois, Il est à présent la possibilité de calculer ces nouvelles inconnues à partir des probabilités d'observation et des transitions initiales du modèle λ par les méthodes "forward" et "backward".

1.4.2.2 La méthode "forward"

En effet, il est possible de calculer α par récurrence car chaque étape de calcul au temps t ne nécessite que les observations des temps précédents. Voici la définition de cette récurrence :

- **Initialisation :** $\alpha_1(i) = \pi_i b_i(o_1)$
- **Récurrence :**

$$\alpha_{t+1}(j) = \Big[\sum_{i=1}^{N} \alpha_t(i)a_{ij}\Big] b_j(o_{t+1})$$

De plus, nous avons la propriété suivante :

$$P(O|\lambda) = \sum_{i=1}^{N} P(O, q_T = i|\lambda) = \sum_{i=1}^{N} \alpha_T(i)$$

1.4.2.3 La méthode "backward"

De même que pour α, il est possible de calculer β par récurrence. Par contre, dans le cas de β, les calculs à l'étape de temps t ont besoin des observations des temps suivants. Voici la définition de la récurrence de β :

- **Initialisation :** $\beta_T(i) = 1,\ 1 \leq i \leq N$
- **Récurrence :**

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij}b_j(o_{t+1})\beta_{t+1}(j)$$

Ces résultats amènent une propriété intéressante :

$$P(O|\lambda) = \sum_{i=1}^{N} \alpha_T(i) = \sum_{i=1}^{N} \beta_1(i) = \sum_{i=1}^{N} \alpha_t(i)\beta_t(i).$$

Dans ces deux récurrences relatives aux valeurs α et β, aucune nouvelle inconnue n'a été introduite, et toutes les autres valeurs présentes sont définies, le calcul effectif est donc possible. Il suffit d'utiliser les différentes égalités et inconnues introduites pour obtenir les valeurs de ré-estimation des a_{ij} et des b_i.

Ensuite, nous itérons le processus de calcul du système λ_i, qui sera au moins aussi bon que le système λ_{i-1}. Le nombre d'itérations réalisées est habituellement défini à une valeur fixée de manière arbitraire, ou bien dépend d'un critère d'arrêt exprimant la stabilisation de la convergence des modèles λ_i.

1.5 Lexique et modèle de langage

1.5.1 Lexique

Le système de reconnaissance nécessite la définition de l'ensemble des mots qu'il sera à même de reconnaître. Cet ensemble est dénommé lexique ou vocabulaire. Un mot n'appartenant pas au lexique ne pourra jamais être présent dans la solution fournie par le système de reconnaissance. Une façon classique de construire le lexique consiste à extraire à partir d'un corpus textuel l'ensemble des mots les plus fréquents de ce corpus. Pour les systèmes de reconnaissance dit *grand vocabulaire* la taille du lexique est de l'ordre de plusieurs dizaines de milliers de mots et le corpus textuel de l'ordre de plusieurs millions de mots (respectivement 54747 mots et plus de 200 millions de mots dans le cadre de nos travaux). En plus de définir l'ensemble des mots connus du système, le lexique associe à chaque mot sa décomposition en unités phonétiques. Le lexique doit également tenir compte des multiples prononciations possibles d'un mot causées par des élisions ou des liaisons entre les mots.

1.5.2 Modèle de langage

Les modèles de langage ont pour objectif de représenter les lois qui régissent le comportement de la langue. Ainsi, le modèle de langage aide à déterminer si une phrase est valide ou non par rapport à la langue modélisée. S'inspirer des connaissances linguistiques est la manière la plus intuitive de construire un modèle de langage. Généralement, les connaissances linguistiques s'expriment par des règles décrivant la structure d'une phrase. L'avantage de cette modélisation vient du fait qu'elle décrit précisément les conditions de validité, de construction d'une phrase de la langue. Cependant certains phénomènes restent mal modélisés et de telles constructions grammaticales ne sont pas en adéquation avec la langue orale. En effet, en parole spontanée, les règles de construction des phrases ne sont pas souvent respectées et ainsi ce type de modélisation ne validerait pas souvent la parole spontanée. De plus, l'élaboration de ces modèles nécessite une part importante d'intervention manuelle, ce qui ne facilite guère l'adaptation de la modélisation aux évolutions de la langue ou à une autre langue.

Dans les systèmes de reconnaissance automatique de la parole, la langue est modélisée par une autre catégorie de modèles de langage, les modèles statistiques. Ces modèles sont appris

automatiquement sur des corpus textuels de taille importante (plusieurs millions de mots). La modélisation statistique n-grammes est la plus utilisée. Cette modélisation consiste à estimer, à partir d'un corpus textuel, les probabilités des séquences de n mots. Le principe de base est d'exploiter la fréquence d'apparition de séquences de mots et d'en déduire des estimations des probabilités unigrammes (probabilité d'apparition d'un mot), bigrammes (probabilité d'apparition d'une séquence de deux mots) et plus généralement de n-grammes. Le critère communément utilisé pour l'estimation de ces différentes probabilités est le critère du maximum de vraisemblance [Federico 98].

Considérant l'ensemble des séquences de n mots possibles à partir du lexique, beaucoup n'apparaissent pas dans le corpus d'apprentissage parce qu'elles sont impossibles voire très improbables pour le langage considéré, comme par exemple le bigramme « le maison ». Toutefois un nombre non négligeable d'entre elles sont valides au sens du langage mais peuvent ne pas apparaître dans le corpus d'apprentissage. Aussi, pour laisser une chance à ces séquences d'être reconnues, chaque séquence de n-mots doit avoir une probabilité non nulle. Différentes techniques dites de repli (*backoff*) permettent d'estimer la probabilité de ces séquences, même si celles-ci n'ont jamais été rencontrées dans le corpus [Chen 99].

Soit la séquence de mot w_1, w_2, w_3, nous définissons la probabilité trigramme comme la quantité $p(w_3|w_1, w_2)$. L'algorithme 1.1 décrit le calcul de la probabilité trigramme directe $p(w_3|w_1, w_2)$ dans le cas d'un modèle linguistique intégrant la notion de repli. Les mêmes techniques sont utilisées dans le calcul de la probabilité bigramme $p(w_2|w_1)$ (Algo. 1.2). Dans ces algorithmes :

- $p_n(w_1, \ldots, w_n)$ est l'estimation sur le corpus d'apprentissage de la probabilité n-gramme $p(w_n|w_1 \ldots w_{n-1})$ dans le modèle langage,
- $repli_{n-1}(w_1, \ldots, w_{n-1})$ est la valeur de repli calculée par le modèle de langage pour une séquence de n mots non rencontrée dans le corpus d'apprentissage.

Lorsqu'une séquence de n mots n'est pas modélisée par le modèle de langage (probabilité n-gramme), un premier niveau de repli est effectué en n'utilisant plus que des relations entre au maximum $n-1$ mots. Ce processus peut être appliqué récursivement tant qu'une probabilité m-gramme n'est pas définie dans le modèle de langage.

Algorithme 1.1 :

si *le trigramme* w_1, w_2, w_3 *existe dans le modèle de langage*
alors
  `/* on utilise la valeur donnée par le modèle */`
  $p(w_3|w_1, w_2) = p_3(w_1, w_2, w_3)$
sinon
  `/* on utilise un premier niveau de repli */`
  si *le bigramme* w_1, w_2 *existe* **alors**
    $p(w_3|w_1, w_2) = repli_2(w_1, w_2) * p(w_3|w_2)$
  sinon
    `/* on utilise un deuxième niveau de repli */`
    $p(w_3|w_1, w_2) = p(w_3|w_2)$
  fin
fin

Tout comme il est possible d'ajouter des connaissances psycho-acoustiques au niveau de la pa-

Algorithme 1.2 :

si *le bigramme* w_1, w_2 *existe dans le modèle de langage*
alors
| $p(w_2|w_1) = p_2(w_1, w_2)$
sinon
| $p(w_2|w_1) = repli_1(w_1) * p_1(w_2)$
fin

ramétrisation, il est également possible d'ajouter des connaissances linguistiques à un modèle de langage statistique. Aussi, des travaux ont proposé des modèles utilisant des classes syntaxiques ou sémantiques de mots ou des modèles se basant sur des traits caractéristiques tels que le genre et le nombre des mots [Brown 92, Brill 98, Rosenfel 96, Kuhn 90, Lavecchia 06].

1.6 Principe de fonctionnement d'un moteur de reconnaissance

Nous décrivons dans cette section le principe de fonctionnement d'un moteur de reconnaissance fondé sur une modélisation acoustique stochastique à base de modèles de Markov cachés. Après l'étape de paramétrisation, nous obtenons en entrée du moteur une séquence O de T vecteurs d'observation, $O = (o_1, \ldots, o_T)$. Effectuer la reconnaissance d'une phrase revient à déterminer la séquence de mots $W^* = w_1 \ldots w_n$ qui maximise la probabilité que cette séquence corresponde à la séquence d'observations O. Ce problème s'écrit ainsi :

$$W^* = \arg\max_W P(W|O)$$

Cependant, il est difficile voire impossible de calculer directement la probabilité $P(W|O)$. Toutefois le théorème de Bayes permet de reformuler cette équation ainsi :

$$W^* = \arg\max_W \frac{P(O|W)P(W)}{P(O)} \tag{1.2}$$

Par cette nouvelle formulation, nous obtenons l'expression du problème en fonction de trois autres probabilités :

- $P(O|W)$: la probabilité d'observer la séquence O des observations sachant la séquence de mots W (probabilité acoustique),
- $P(W)$: la probabilité *a priori* de la séquence de mots W (probabilité linguistique),
- $P(O)$: la probabilité de l'observation.

La séquence d'observations O étant fixée, $P(O)$ ne dépend pas de la séquence de mots W étudiée. L'équation 1.2 se simplifie alors en l'équation 1.3 qui ne dépend plus que des probabilités acoustiques et linguistiques :

$$W^* = \arg\max_W P(O|W)P(W) \tag{1.3}$$

Afin de résoudre ce problème, il est donc nécessaire de calculer $P(W)$ et $P(O|W)$ pour toutes les séquences de mots possibles, puis de comparer les scores $P(O|W)P(W)$ entre eux.

Si nous supposons dorénavant que les séquences W sont uniquement limitées à des modèles de Markov cachés M d'un mot, alors :

$$P(O|W) = P(O|M) = \max_{q \in \Xi} \pi_{q_0} \prod_{t=1}^{T} a_{q_{t-1}q_t} b_{q_t}(o_t) \tag{1.4}$$

Ξ étant l'ensemble des séquences d'états possibles et T la longueur de la séquence d'observations. Or, calculer $P(O|M)$ directement en parcourant toutes les séquences d'états possibles pose un problème d'explosion combinatoire. En effet, pour un modèle de Markov caché à N états, la complexité de ce calcul est en $O(T.N^T)$. Il est donc nécessaire d'introduire des méthodes plus astucieuses permettant de résoudre ce problème en gardant une complexité raisonnable. La méthode habituellement utilisée est l'algorithme de Viterbi qui permet de réduire la complexité du calcul en $O(T.N^2)$ [Viterbi 67, Forney 73]. Cet algorithme, ou ses variantes, est au cœur de nombreux systèmes automatiques de reconnaissance de la parole : Julius [Lee 01], HTK [Young 94a], Sphinx-4 [Lamere 03], ESPERE [Fohr 00], SPIRAL [Linares 05], Sirocco [Gravier 02], ISIP [Deshmukh 99]. D'autres algorithme de recherche de meilleur chemin ont également été utilisés tels l'algorithme A^*, des algorithmes à pile ou l'algorithme de programmation dynamique à deux niveaux (Two-Level Dynamic Programming – TLDP) [Agbago 04]. Le système de reconnaissance *Julius*, que nous avons utilisé dans nos expérimentations, est fondé sur un processus de reconnaissance en deux passes : une passe avant utilisant l'algorithme de Viterbi et une passe arrière basée sur l'algorithme A^*.

1.6.1 L'algorithme de Viterbi

1.6.1.1 Le principe de Viterbi

Nous cherchons ici à déterminer la séquence d'états maximisant $P(O|M)$. Le système à résoudre peut se représenter sous la forme d'un graphe à deux dimensions : la séquence d'observations en abscisse, le modèle M en ordonnée.

La Figure 1.5 représente un tel graphe pour un exemple d'une séquence de 10 observations et un modèle de Markov caché à trois états à topologie de transition gauche-droite. Dans ce graphe, un nœud représente un état i du modèle pour une certaine observation o_t avec une valeur associée égale à $b_i(o_t)$. Les arcs correspondent aux transitions d'un état i à un état j (i peut être égal à j) et ont comme valeur associée la probabilité de transition a_{ij} (c.f. section 1.4.1).

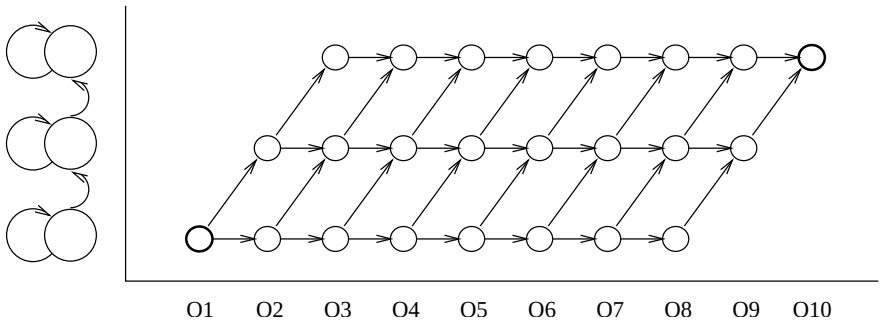


Fig. 1.5 – Graphe de Viterbi pour un HMM à 3 états gauche-droite et une séquence de 10 observations

La solution se présente ainsi sous la forme d'un chemin ayant comme origine un état du modèle de Markov au temps $t = 1$ et comme extrémité un état du modèle au temps $t = T$. Dans l'exemple de la figure 1.5, la topologie du modèle de Markov est gauche-droite, ce qui implique

des contraintes supplémentaires sur les chemins possibles. En effet, le chemin doit forcément débuter dans le premier état du modèle et terminer dans le dernier état de celui-ci. A partir de cette représentation graphique du problème, l'emploi d'un algorithme de recherche du meilleur chemin dans un graphe semble naturel.

L'algorithme de Viterbi permet d'effectuer cette recherche en s'appuyant sur le principe d'optimalité de Bellman.

1.6.1.2 Le principe d'optimalité de Bellman

Le principe d'optimalité de Bellman utilisé en programmation dynamique peut s'appliquer dans le cadre de la recherche de chemin dans un graphe et s'exprime ainsi : imaginons que nous connaissions le chemin optimal pour arriver en A_1, A_2 et A_3 (Figure 1.6).

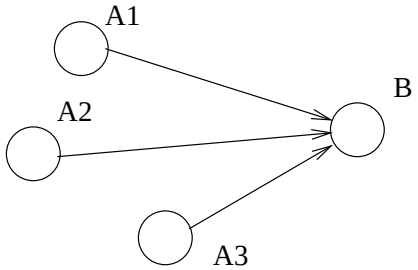


FIG. 1.6 – Exemple d'un extrait de graphe afin d'illustrer le principe d'optimalité de Bellman.

Alors, le chemin optimal pour aller en B, donné par $dcumul(B)$, est :

$$dcumul(B) = \min \begin{cases} dcumul(A_1) + d(A_1, B) \\ dcumul(A_2) + d(A_2, B) \\ dcumul(A_3) + d(A_3, B) \end{cases}$$

$dcumul(A_i)$ étant la distance cumulée pour arriver au point A_i, et $d(A_i, B)$ la distance locale pour aller de A_i à B, $\forall i \in \{1, 3\}$.

1.6.1.3 L'algorithme de Viterbi

En se reportant à la Figure 1.5 et en utilisant le principe d'optimalité de Bellman, nous constatons que les scores cumulés pour chaque état du modèle de Markov au temps t dépendent seulement :

- des scores cumulés pour chaque état au temps précédent $t-1$,
- des probabilités de transition entre les états du temps $t-1$ et ceux du temps t,
- des probabilités d'émission de l'observation o_t par les états du HMM.

Soit $\delta_t(j)$ la probabilité du meilleur chemin qui s'arrête à la trame t à l'état j du HMM, nous obtenons alors la relation suivante :

$$\delta_t(j) = \max_i \Big(\delta_{t-1}(i) \times a_{ij} \times b_j(o_t) \Big) \tag{1.5}$$

L'algorithme de Viterbi est fondé sur cette relation de récurrence qui permet à chaque instant t de connaître la probabilité du meilleur chemin menant à l'état j du modèle. De cette manière,

tous les chemins possibles sont parcourus et le score du meilleur chemin est connu. Nous avons de plus la relation suivante qui lie l'équation 1.5 au problème initial :

$$P(O|M) = \max_i \delta_T(i) \tag{1.6}$$

l'algorithme de Viterbi se décompose ainsi :
- **Initialisation :** $\delta_0(i) = \pi_i$, la probabilité initiale d'être dans un des états du modèle de Markov.
- **Récurrence :** au temps t pour chaque état i du modèle nous calculons $\delta_t(i)$ par l'équation 1.5 qui ne dépend que des δ_{t-1}.
- **Terminaison :** pour chaque état i du modèle, nous cherchons $\delta_T(i)$ maximal. Nous obtenons ainsi $P(O|M)$ (voir Eq. 1.6).

L'algorithme ne dépendant ainsi que du nombre d'états du modèle et, pour un temps t, que des scores cumulés du temps $t-1$, la complexité devient linéaire par rapport à la longueur de la séquence d'observations. Un deuxième point important de cet algorithme est qu'il est non seulement possible d'estimer $\max_Q P(O|M,Q)$, mais surtout de connaître la meilleure séquence Q. En effet, il suffit de conserver pour chaque meilleur chemin les états par lesquels ce chemin est passé.

1.6.1.4 Algorithme de Viterbi dans le cas d'une phrase

Dans la section précédente, nous avons expliqué l'algorithme de Viterbi dans le cas général d'un modèle de Markov caché M pour calculer $P(O|M)$. Grâce à cet algorithme, pour un modèle donné, nous pouvons déterminer la séquence d'alignement des états du modèle sur la séquence d'observations.

Le principe pour effectuer la reconnaissance d'une phrase est le même. Nous allons en fait construire un méta-modèle de Markov caché dans lequel chaque méta-état représente un mot du lexique. Le méta-modèle est ergodique, toutes les transitions entre méta-états sont possibles et dépendent du modèle de langage. Ainsi, la meilleure séquence de méta-états calculée par l'algorithme de Viterbi correspond à une phrase, la solution du système de reconnaissance.

Trouver la séquence de mot W^* maximisant l'équation 1.3 revient chercher la séquence de mots qui maximise la quantité suivante :

$$\max_{W \in \Xi} \pi_{w_0} \prod_{w_i \in W} P(O|w_i) P(w_i|w_{i-1} \dots w_0) \tag{1.7}$$

Ξ représente l'ensemble des séquences de mots appartenant au lexique qu'il est possible de construire, π_{w_0} est la probabilité initiale du premier mot de la séquence, $P(O|w_i)$, la probabilité acoustique du mot w_i de la séquence W, l'équivalent de la probabilité d'émission, et $P(w_i|w_{i-1} \dots w_0)$ représente la probabilité linguistique n-gramme, jouant le rôle des probabilités de transition dans le modèle de Markov.

De par son processus de construction, chaque mot du graphe est l'extrémité d'un unique chemin partant du début de la phrase. Cette unicité provient de la propriété d'optimalité de Bellman. Aussi, pour chaque mot du graphe, le mot le précédant sur ce chemin est déterminé de façon unique. Nous nous référerons à ce mot sous le terme de *prédécesseur au sens de Viterbi* dans la suite de ce document.

1.6.2 L'algorithme A^*

L'algorithme A^* est également un algorithme de recherche du meilleur chemin dans un graphe [Nilsson 71]. Il est d'ailleurs très souvent utilisé en intelligence artificielle, notamment dans divers jeux impliquant la recherche d'un chemin optimal entre deux points. Certains systèmes de reconnaissance sont fondés sur les deux algorithmes, Viterbi et A^* afin de déterminer la phrase solution. Nous allons d'ailleurs illustrer certains points de l'algorithme A^* par rapport à son implantation dans le système de reconnaissance *Julius*.

Julius effectue la reconnaissance d'une phrase en deux temps :

- une première passe, fondée sur l'algorithme de Viterbi, construit une structure interne (graphe de mots) de façon à limiter l'espace de recherche des solutions,
- une deuxième passe, utilisant l'algorithme A^*, détermine la solution du système à partir de modèles plus fins et en utilisant la structure construite pendant la première passe.

Toutefois, il est important de noter que la première passe s'effectue du début vers la fin de phrase, alors que la seconde passe commence par la fin de la phrase et finit par le début de celle-ci.

L'algorithme A^* dépend de trois données :

- une liste ouverte d'hypothèses contenant des chemins partiels non encore traités, partant de la fin la phrase. Ces chemins pourront éventuellement être prolongés et faire partie de la solution finale,
- une liste fermée d'hypothèses contenant des chemins partiels déjà traités qui ne seront plus à considérer,
- une heuristique estimant la vraisemblance du chemin restant à parcourir pour atteindre le début de la phrase.

Le principe de l'algorithme A^* consiste à prolonger les chemins hypothèses de la liste ouverte, mot après mot, afin d'atteindre le début de la phrase et sélectionner le meilleur d'entre eux.

Nous introduisons quelques notations afin de préciser le fonctionnement de l'algorithme :

- $g(w)$, la vraisemblance cumulée d'un chemin hypothèse de la fin de la phrase jusqu'à un mot w d'instant de début τ et d'instant de fin t, noté $[w, \tau, t]$.
- $t(w, w')$, le score de transition de $[w, \tau, t]$ à un nouveau mot $[w', \tau', t']$ avec $t' = \tau - 1$ (score acoustique et linguistique).
- $h(w')$, le score heuristique de la vraisemblance de la partie de phrase de w' jusqu'au début de la phrase.
- <s> et </s>, les mots clés spéciaux correspondant à des marqueurs, respectivement de début et de fin de phrase.

A chaque hypothèse de chemin partiel commençant de la fin de la phrase correspond un score estimé $st(w)$ qui est la combinaison de la vraisemblance calculée du chemin hypothèse et de la vraisemblance estimée par la fonction heuristique du chemin restant, de w à « <s> ». La vraisemblance totale estimée $st(w)$ est calculée ainsi :

$$st(w) = g(w) + h(w)$$

Dans l'algorithme A^*, toutes les hypothèses de la liste ouverte ne sont pas considérées en même temps. L'hypothèse de chemin partiel de la liste ouverte ayant le score $st(w)$ maximal sera développée en premier. Dans le moteur de reconnaissance de Julius, le score heuristique $h(w)$ est la vraisemblance cumulée calculée pendant la première passe du moteur du chemin reliant le début de la phrase au mot w.

L'algorithme procède ainsi :

Initialisation : Dans le moteur de reconnaissance Julius, l'algorithme A^* est utilisé en deuxième passe, en commençant par la fin de la phrase. L'initialisation consiste alors à considérer le mot clé spécial « </s> » qui marque la fin d'une phrase comme élément de départ et le placer dans la liste ouverte. A ce mot est associé le score $st(</s>)$ représentant la vraisemblance de la phrase solution déterminée par le moteur de reconnaissance à l'issu de la première passe. Ainsi au départ, la liste fermée est vide et la liste ouverte ne contient que « </s> ».

Itération : L'hypothèse de chemin partiel, allant de la fin de la phrase à un mot $[w, \tau, t]$, de la liste ouverte ayant le score $st(w)$ maximal est choisie puis déplacée dans la liste fermée. Puis, un ensemble de mots est déterminé : l'ensemble des mots $[w', \tau', t']$ du graphe de la première passe qui peuvent précéder le mot $[w, \tau, t]$ tels que $t' = \tau - 1$ et tels qu'il n'existe pas d'hypothèses commençant par $[w', \tau', t']$ dans la liste fermée. L'utilisation du graphe de mots construit par le moteur de reconnaissance au cours de la première passe permet de restreindre l'ensemble des mots à considérer. Dans le cas contraire, chaque mot du lexique devrait être considéré comme prédécesseur possible de w.

Pour chaque mot w', le score associé $st(w')$ est calculé ainsi :

$$st(w') = g(w) + t(w, w') + h(w') = g(w') + h(w')$$

Deux cas se présentent suivant le mot w' considéré :

- si pour un mot w' il n'existe aucune hypothèse de chemin commençant par w' dans la liste ouverte, alors une nouvelle hypothèse de chemin est constitué. Cette nouvelle hypothèse est alors ajoutée à la liste ouverte.
- s'il existe un déjà un chemin partiel hypothèse dans la liste ouverte qui commence par le mot w', deux cas se présentent :
  - si le score $st(w')$ du nouveau chemin est inférieur à celui déjà dans la liste, alors la nouvelle hypothèse est simplement ignorée,
  - si le score $st(w')$ du nouveau chemin est meilleur que celui se trouvant déjà dans la liste ouverte, alors l'hypothèse du chemin partiel de la liste ouverte est mise à jour avec comme score $st(w')$ et comme « parent » le mot $[w, \tau, t]$ (et donc son chemin partiel associé).

En effet, afin de pouvoir retrouver le meilleur chemin calculé par l'algorithme, à chaque mot est également associé un mot parent permettant de remonter le chemin jusqu'à la fin de la phrase.

Terminaison : L'algorithme se termine une fois qu'un chemin hypothèse ayant atteint le début de la phrase existe dans la liste fermée. Le meilleur chemin correspond alors à cette hypothèse. Cependant, il peut arriver qu'à une itération, la liste ouverte soit vide. Dans ce cas, il n'y a pas de solution au problème et l'algorithme s'arrête.

La fonction heuristique : le point clé de l'algorithme A^* est la fonction heuristique. Cette fonction permet à l'algorithme de déterminer plus ou moins rapidement le meilleur chemin. Pour que l'algorithme puisse se dérouler correctement, la fonction heuristique doit être admissible, c'est-à-dire qu'elle ne doit pas sous-estimer la vraisemblance du chemin restant à parcourir. Mieux la fonction estimera ce coût et moins il sera nécessaire d'explorer le graphe afin de trouver la solution. Dans Julius, l'algorithme est appliqué en deuxième passe, il est ainsi possible de connaître une estimation de la vraisemblance du chemin restant à parcourir. Julius a donc choisi de définir la fonction heuristique comme étant la vraisemblance du chemin partiel depuis le début de la phrase, valeur issue de la première passe, justifiant ainsi l'utilisation de l'algorithme A^* [Gopalakrishnan 95].

1.6.3 Résultats de la reconnaissance

Généralement, le système de reconnaissance ne fournit qu'une seule et unique solution à un problème de reconnaissance de la parole : la phrase la plus probable. Cependant, dans bien des cas, cette information n'est pas suffisante. En effet, s'il doit y avoir des post-traitements à la reconnaissance, des informations complémentaires sur le résultat sont nécessaires. Par exemple, déterminer la valeur de confiance de chaque mot reconnu est impossible avec uniquement la phrase résultat. Nous présentons ici différentes représentations des résultats du moteur de reconnaissance autres que la seule solution maximale. Ces représentations incluent les n-meilleures phrases, le graphe de mots et le réseau de confusion.

1.6.3.1 N-meilleures phrases

Les n-meilleures phrases sont un sous-ensemble de toutes les phrases qu'il est possible de générer suivant le lexique et les modèles linguistiques. Ce sous-ensemble est déterminé par le moteur de reconnaissance suivant le critère du score de vraisemblance de chacune des phrases hypothèses. L'algorithme qui permet d'extraire la liste des n-meilleures phrases est très semblable à l'algorithme de Viterbi [Schwartz 90].

Généralement, le moteur de reconnaissance effectue plusieurs passes pour déterminer le résultat final. Une première passe rapide avec des modèles moins précis permet de restreindre l'espace de recherche de solution. Les autres passes successives introduisent des modèles plus précis ou des informations supplémentaires, par exemple, de nature sémantique. Un exemple d'utilisation de la structure des n-meilleures phrases consiste à extraire de la première passe l'ensemble des n-meilleures phrases puis, à partir de cet ensemble restreint d'hypothèses, effectuer une deuxième passe de ré-estimation des scores avec des modèles plus précis.

Un élément de la structure des n-meilleures phrases se présente sous la forme d'une séquence de mots. Généralement, l'information importante concerne la position des mots dans la séquence. Cependant, il est possible de conserver d'autres types d'information : l'instant de début et de fin des mots, les probabilités acoustiques et linguistiques, etc.

Un point négatif des n-meilleures phrases réside dans la distribution des phrases dans cet ensemble. Dans l'exemple table 1.1, nous pouvons remarquer que la plupart des phrases ne diffèrent entre elles que d'un mot. En effet, des phrases ayant des vraisemblances proches sont en général et logiquement très semblables. Le système aura donc tendance à fournir des phrases très similaires. Ainsi, pour une phrase longue, il faut considérer un nombre n de plus en plus important de séquences pour obtenir une variété suffisante de phrases.

TAB. 1.1 – Exemple de liste des 5 meilleures phrases issues d'un système de reconnaissance.

```
phrase 1: elles approcheront vingt-quatre vingt-cinq degrés sur le massif central
phrase 2: elles se approcheront vingt-quatre vingt-cinq degrés sur le massif central
phrase 3: et ils approcheront vingt-quatre vingt-cinq degrés sur le massif central
phrase 4: -elles approcheront vingt-quatre vingt-cinq degrés sur le massif central
phrase 5: elles le approcheront vingt-quatre vingt-cinq degrés sur le massif central
```

1.6.3.2 Graphe de mots

La structure de graphe de mots permet de représenter de manière plus précise et plus complète les informations issues généralement de la première passe [Ney 94, Ortmanns 97]. Le moteur de reconnaissance, du fait de l'algorithme de Viterbi, génère en interne un graphe de mots contenant les mots hypothèses conservés par le moteur de reconnaissance. Un graphe de mots inclut les multiples chemins possibles qui vont du début à la fin de la phrase. Les informations stockées sont, entre autres, le score acoustique des mots et leurs instants de début et de fin. De par son processus de construction, chaque mot du graphe est l'extrémité d'un unique chemin partant du début de la phrase. Cette unicité provient de la propriété d'optimalité de Bellman. Aussi, pour chaque mot du graphe, l'information du mot prédécesseur au sens de Viterbi est conservée ainsi que la vraisemblance cumulée depuis le début de la phrase de cet unique chemin.

La figure 1.7 présente un exemple de graphe de mots associé à la phrase « je mange ici vers midi ». Les flèches illustrent les liaisons de précédence entre les mots du graphe. L'axe horizontal correspond à l'axe temporel.

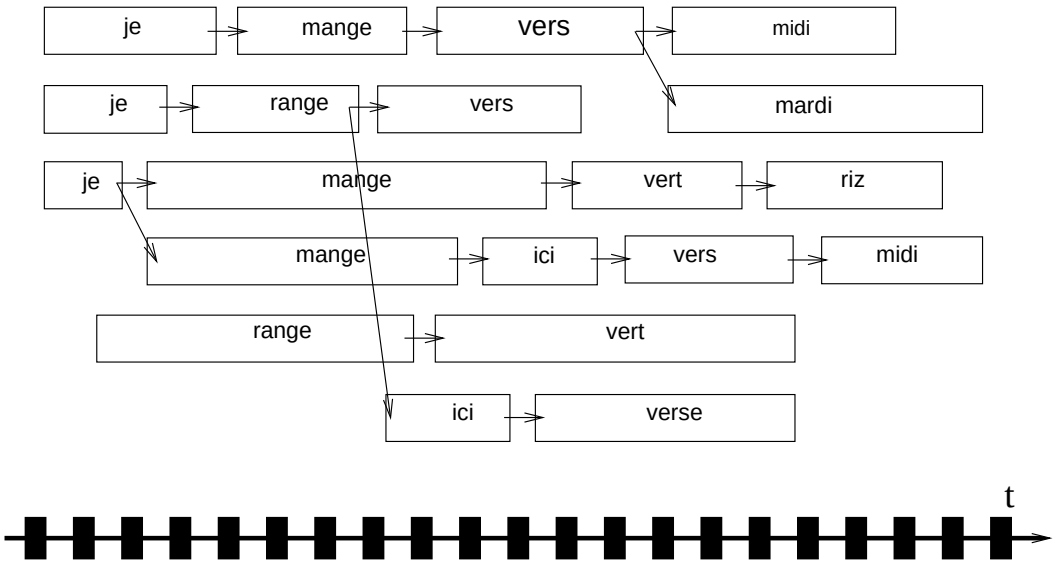


Fig. 1.7 – Exemple d'un graphe de mots

Le graphe de mots est bien plus efficace dans la représentation des informations que la structure des n-meilleures phrases. Par exemple, pour une phrase de 10 mots ayant pour chaque mot un choix possible entre 2 hypothèses alternatives, il faudrait $2^{10} = 1024$ phrases dans la structure des n-meilleures phrases pour représenter toutes les alternatives possibles. Or, avec une structure de graphe de mots, garder en mémoire seulement 20 mots suffit. Le graphe de mots est également plus efficace du point de vue calculatoire. En effet, avec la structure des n-meilleures phrases, le traitement de phrases alternatives très similaires vont nécessiter d'effectuer plusieurs fois des calculs identiques. Par contre, avec un graphe de mots, les calculs pour les parties communes des alternatives ne seront réalisés qu'une seule fois. Ainsi, les données et les calculs peuvent être mieux partagés, et donc l'utilisation d'un graphe de mots est finalement moins coûteuse en place et en temps de calcul.

1.6.3.3 Réseau de confusion

Le réseau de confusion est une simplification du graphe de mots. Les mots n'y sont plus localisés suivant leur instant de début et de fin, mais suivant leur position dans la phrase/séquence.

Le résultat est un graphe d'alignements multiples avec différentes hypothèses (parfois nulles) à chaque position possible d'un mot. La figure 1.8 présente un graphe de mots standard associé à la phrase « je mange vers midi » alors que la figure 1.9 présente son équivalent sous forme de réseau de confusion. On peut noter les éléments « <s> » et « </s> » qui correspondent respectivement aux marqueurs de début et de fin de phrase. L'introduction d'un élément spécial « - », représentant une position optionnelle dans la phrase, permet d'indiquer la possibilité de passer directement au mot suivant pour former une phrase sans ajouter une nouvelle hypothèse de mot.

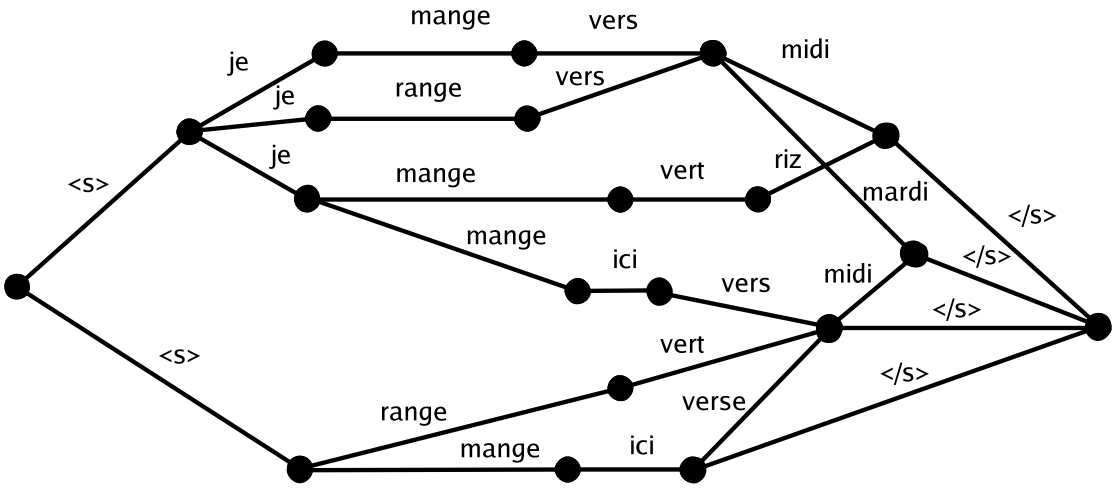


FIG. 1.8 – Second exemple d'un graphe de mots

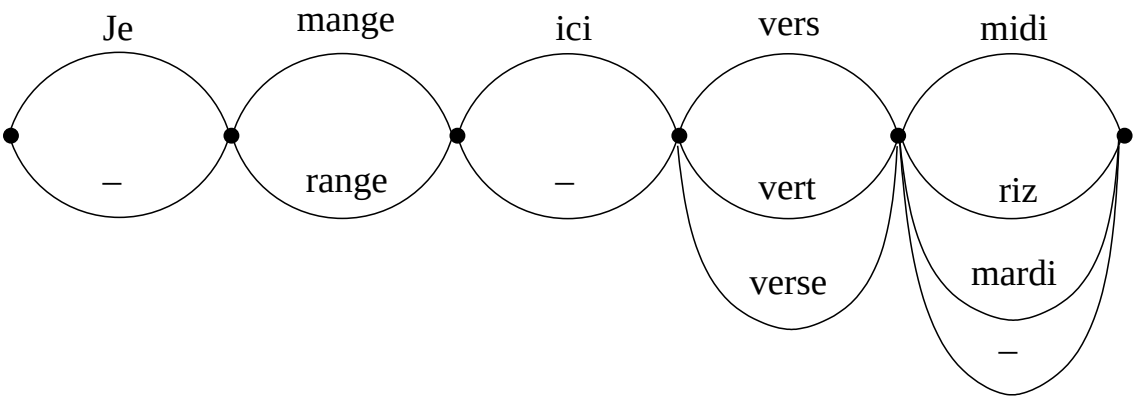


FIG. 1.9 – Exemple d'un réseau de confusion

La création d'un réseau de confusion est un point délicat et difficile. En effet, se pose alors des problèmes de sélections, éliminations, regroupements d'hypothèses qui peuvent conduire à l'apparition de *trous* dans le continuum temporel qu'est censé représenté le graphe de mots.

Cependant un algorithme a été proposé afin de réaliser ces opérations tout en conservant un ordre total et la consistance du graphe obtenu [Mangu 00].

1.7 Conclusion

Dans ce chapitre, nous avons rappelé les bases d'un système automatique de reconnaissance de la parole, et plus particulièrement certaines parties correspondant à nos propres conditions d'expérimentation et au système de reconnaissance utilisé pour celles-ci : paramétrisation MFCC, modélisation linguistique n-gramme, modélisation acoustique par Modèles de Makov cachés (HMM). Nous avons présenté l'algorithme de Baum et Welch permettant l'apprentissage des modèles acoustiques fondés sur des HMMs. L'algorithme de Viterbi, utilisé au sein du moteur de reconnaissance a été aussi décrit car celui-ci joue un rôle majeur dans les systèmes de reconnaissance modernes, mais également parce que nos travaux vont interagir avec cet algorithme. Nous avons également présenté des structures de données, issues du système de reconnaissance en vue de post-traitements, qui nous donnent accès à plus d'informations que la seule phrase reconnue : les n-meilleures phrases et les graphes de mots. Comme nous le verrons plus loin, les graphes de mots sont les structures à partir desquelles nous définirons nos mesures de confiance.

Chapitre 2

Mesures de confiance

Sommaire

2.1 Introduction 25
2.2 Exemples d'applications et intérêt des mesures de confiance 26
2.2.1 Reconnaissance de la parole : transcription 26
2.2.2 Détection des mots hors vocabulaire 27
2.2.3 Détection de mots clés 27
2.2.4 Dialogue Homme/Machine 29
2.2.5 Apprentissage semi-supervisé 29
2.2.6 Adaptation 31
2.3 Mesures de confiance 31
2.3.1 Critères *non probabilistes* 32
2.3.1.1 Stabilité acoustique 32
2.3.1.2 Densité d'hypothèses 32
2.3.1.3 Dépendance des mots 33
2.3.2 Critères relatifs au modèle de langage 33
2.3.2.1 Le modèle de langage 33
2.3.2.2 Repli du modèle de langage 34
2.3.3 Critères sémantiques et syntaxiques 35
2.3.3.1 Analyse sémantique latente 35
2.3.3.2 Information mutuelle 36
2.3.3.3 Catégorie d'un mot 37
2.3.4 Autres critères empiriques 37
2.3.5 Mesures fondées sur le rapport de vraisemblance 38
2.3.5.1 Tests d'hypothèse 38
2.3.5.2 Rapport de vraisemblance 38
2.3.5.3 Modèle / Anti-Modèle 39
2.3.5.4 Modèle générique 39
2.3.5.5 Modèles compétitifs 40
2.3.6 Mesures fondées sur les probabilité *a posteriori* 41
2.3.6.1 Mesures fondées sur la liste de n-meilleures phrases 42
2.3.6.2 Mesures fondées sur les graphes de mots et l'algorithme de *forward-backward* 42
2.3.6.3 Mesure de confiance du système de reconnaissance *Julius* . . . 44
2.3.6.4 Mesures fondées sur les réseaux de confusion 45

2.3.6.5 Récapitulatif des mesures fondées sur une estimation de la probabilité *a posteriori* 46
2.3.7 Combinaison de mesures de confiance 46
2.3.7.1 Combinaisons de mesures et d'heuristiques 46
2.3.7.2 Combinaison de systèmes de reconnaissance 47
2.4 Méthodes d'évaluation 48
2.4.1 Taux d'égale erreur 48
2.4.2 Taux d'erreur de confiance 50
2.4.3 Entropie croisée normalisée 50
2.4.4 Coefficient de corrélation 51
2.4.5 Rappel / Précision 52
2.4.6 Synthèse 53
2.5 Quelques résultats 53
2.6 Conclusion 54

2.1 Introduction

Une mesure de confiance est un indice dont le but est d'estimer la qualité du résultat d'une application. Cet indice représente en quelque sorte la confiance que nous pouvons avoir dans la décision prise, par exemple, par un système de reconnaissance quand celui-ci fournit son résultat. Une mesure de confiance idéale affectera la valeur 0 à un résultat faux et la valeur 1 à un résultat juste. Toutefois, il est rare que les valeurs calculées par la mesure soient aussi franches et, cette mesure estimant souvent une probabilité, tout l'intervalle $[0, 1]$ est utilisé. Aussi, un seuil de décision est-il habituellement associé à l'utilisation d'une mesure de confiance. Ce seuil délimite une frontière entre les résultats jugés corrects et ceux jugés incorrects par la mesure. Il est nécessaire au préalable de déterminer la valeur optimale du seuil sur un corpus de développement. Cependant, cette classification n'est nullement obligatoire et la valeur de confiance peut être utilisée directement par un autre processus, sans classification binaire.

En reconnaissance automatique de la parole, quel que soit ce qui a été prononcé ou émis, le système de reconnaissance délivre toujours une solution qui, selon lui, est la plus probable au vu des connaissances à sa disposition (modèles acoustiques et linguistiques). En effet, à partir des différentes modélisations et du lexique, la solution déterminée par le moteur de reconnaissance est l'hypothèse qui obtient le score maximal. Mais ce score ne reflète aucunement la qualité du résultat : le système détermine toujours des solutions possibles et une seule est de score maximal. Ainsi, le système de reconnaissance peut délivrer un résultat comportant des erreurs, mais relativement au système, c'est la meilleure solution.

Plusieurs facteurs peuvent être à l'origine des erreurs commises par le système :

- les conditions d'utilisation difficiles : si les conditions environnementales du document sonore traité sont très différentes des conditions d'apprentissage, les modèles ne sont plus adaptés et le système ne peut plus discriminer suffisamment les sons entre eux (par exemple en présence de bruit ou de fond musical) ;
- le lexique fixé : comme nous l'avons décrit dans le chapitre précédent, les systèmes de reconnaissance de parole ont besoin de la définition d'un lexique qui détermine l'ensemble des mots qui peuvent faire partie du résultat. Aussi grand que soit ce lexique, il y aura toujours des mots, des noms qui n'y appartiennent pas. De tels mots ne pourront alors jamais apparaître dans une solution proposée par le système et seront inévitablement sources d'erreurs ;
- la modélisation statistique : afin de pouvoir traiter des applications nécessitant un grand vocabulaire (plusieurs dizaines de milliers de mots), les modélisations généralement adoptées sont statistiques. Ces modèles étant non déterministes, ils représentent un comportement plus général mais sont en contre-partie moins précis.

Quel que soit le niveau considéré par le système (phrase, mot, phonème, etc), l'introduction d'une mesure de confiance en reconnaissance de la parole permet alors d'estimer la qualité du résultat délivré par le système. La valeur de confiance est ainsi un indice qui peut aider à localiser les erreurs présentes dans la solution déterminée par le système.

Dans cette section, nous allons tout d'abord présenter quelques applications habituellement rencontrées dans le domaine de la reconnaissance automatique de la parole, ainsi que l'intérêt que peut apporter l'utilisation d'une mesure de confiance.

Puis, dans une deuxième partie, nous présenterons un état de l'art des mesures de confiance en reconnaissance de la parole. Dans une dernière partie, nous introduirons les méthodes généralement employées afin d'évaluer les mesures de confiance. Bien qu'il soit difficile de comparer les performances des différentes mesures de confiance entre elles, car souvent les corpus utilisés

sont distincts, nous résumerons à la fin de ce chapitre des résultats concernant quelques mesures de confiance.

2.2 Exemples d'applications et intérêt des mesures de confiance

2.2.1 Reconnaissance de la parole : transcription

La transcription consiste, à partir du signal acoustique d'un document sonore, à fournir son contenu sous forme textuelle. Les documents sonores considérés peuvent être des émissions radiophoniques ou audiovisuelles, des réunions, des conférences, mais également des cours en salle de classe pour les élèves sourds ou malentendants. La transcription n'a pas comme objectif la compréhension du contenu du document traité. De plus, contrairement à la dictée vocale, il n'y a aucune interaction entre l'utilisateur et le processus de transcription. Aussi, cette application fait face à plusieurs difficultés :

- le lexique doit être suffisamment large pour tenir compte d'une utilisation en vocabulaire ouvert,
- présence de nombreux locuteurs très différents les uns des autres,
- environnement sonore varié : passage studio/téléphone, interventions extérieures, dialogue simultané, hésitations, reprises, bruits.

Toutes ces difficultés font de la transcription une tâche ardue qui nécessite des techniques de modélisation acoustique et linguistique robustes. Malgré ces difficultés, des systèmes atteignent des taux d'erreurs en mots de l'ordre de 10%, que ce soit en anglais ou en français [Nguyen 05, Gauvain 05].

Toutefois, dans certaines situations, il n'est pas possible de mettre en place tout un système complet performant, et des concessions doivent être faites. C'est le cas par exemple pour la mise en place d'applications dans des appareils mobiles (téléphones, PDA) dont les capacités de calcul et de mémoire sont bien moindres que celle d'un ordinateur de bureau. De même, quand les conditions d'utilisation sont différentes de celles d'apprentissage, le système de transcription génère plus d'erreurs. De plus, la limitation intrinsèque du lexique est très contraignante à cause des mots hors vocabulaire qui peuvent apparaître souvent (principalement des noms propres).

Les mesures de confiance peuvent alors être vues comme un indice complémentaire au résultat de la transcription et permettent la mise en place de système de post-traitements des erreurs potentielles indiquées par la mesure.

Dans le cas de la transcription d'émissions radiophoniques, la mesure pourrait servir à mettre en exergue les mots ayant une faible valeur de confiance. Par exemple, l'utilisation d'une couleur (rouge) différente de celle du reste du texte (noire) est une façon de pointer des mots peu sûrs. Ensuite, une intervention humaine *a posteriori* pourrait ne vérifier, et au besoin corriger, que les mots colorés de la transcription.

Sans vouloir corriger la transcription, la mise en valeur des mots peu sûrs peut permettre d'améliorer la compréhension par un lecteur n'ayant pas accès à l'acoustique. En effet, si dans une phrase quelques mots sont erronés, ceux-ci peuvent rendre totalement incompréhensible la phrase et perturber le lecteur. L'indication d'un mot potentiellement faux peut mettre en éveil le lecteur afin de conserver ou retrouver plus facilement le sens de la phrase d'origine.

Contrairement à ces deux cas où les mots de faible confiance sont mis en valeur, il est possible d'utiliser des mesures de confiance afin d'améliorer le résultat de la reconnaissance et ainsi la transcription. Il n'y a plus de post-traitements du résultat selon la mesure de confiance. L'objectif consiste à intégrer l'indice que représente cette mesure au cœur du processus de décision du

moteur de reconnaissance. La valeur de confiance pourrait être utilisée directement ou via une fonction de transformation afin de modifier la vraisemblance des hypothèses considérées pendant la phase de reconnaissance. Le but étant alors de diminuer le taux d'erreur en mots du système. Dans ce cas, il n'y a plus de classification des mots de la transcription.

Dans le chapitre 3, nous présenterons plus en détail les applications liées à la transcription sur lesquelles nous avons focalisé notre étude.

2.2.2 Détection des mots hors vocabulaire

Quelle que soit l'application visée, les systèmes de reconnaissance automatique de la parole sont tributaires d'un lexique de taille limitée. Ce lexique définit l'ensemble des mots pouvant faire partie de l'espace de recherche et du résultat du moteur de reconnaissance. A fortiori, un mot ne faisant pas partie de ce vocabulaire ne pourra jamais apparaître dans un résultat du système. Or, dans des applications vocales ne limitant pas le vocabulaire des utilisateurs, il n'est pas rare de rencontrer des mots qui ne fassent pas partie du lexique. Par exemple dans le cas des émissions radiophoniques ou des dialogues en parole spontanée, le système de reconnaissance ne peut imposer aucune contrainte sur le contenu. Même en utilisant un système grand vocabulaire, des mots seront hors vocabulaire. Les noms propres en sont un exemple fréquent. Il est impossible au système de reconnaissance de connaître tous les noms propres et dans le cas d'émissions radiophoniques, l'actualité changeant rapidement, de nouveaux noms propres apparaissent tous les jours. Aussi, ces mots hors vocabulaire entraîneront forcément des erreurs à la reconnaissance.

Plusieurs travaux de recherche se sont focalisés sur la détection de ces mots hors vocabulaire. Dans la plupart de ces travaux, cette détection est réalisée à l'aide d'une mesure évaluant la probabilité qu'un mot soit ou non hors vocabulaire. Ces mesures sont généralement assimilables à des mesures de confiance ou sont définies en tant que telles.

Ainsi, T. Schaaf [Schaaf 01] s'est principalement concentré sur les noms propres hors vocabulaire dans des applications d'annuaires téléphoniques, de gestion d'agendas et des systèmes de réservations de voyages. La taille du lexique était limitée à une dizaine de milliers de mots. Le principal objectif traité était de limiter l'influence du mot hors vocabulaire sur la reconnaissance des mots à son voisinage. Sun et al. [Sun 03] ont également introduit des mesures de confiance pour la détection de noms propres hors vocabulaire dans des dialogues téléphoniques. De même, mais en condition acoustique large bande, Jitsuhiro et al. [Jitsuhiro 98] ont utilisé une mesure de confiance au niveau des phonèmes afin de rejeter les mots hors vocabulaire.

Dans leurs travaux, Decadt et al. [Decadt 01, Decadt 02] ont utilisé un processus de détection de mots hors vocabulaire fondé sur une mesure de confiance pour des applications de transcription de bulletins d'information. La taille du lexique est de l'ordre de 40 000 mots. Dans leurs travaux, les mots hors vocabulaire détectés sont ensuite reconnus à l'aide d'un phonétiseur, puis une conversion de ces suites de phonèmes en graphèmes est effectuée afin de tenter d'écrire le mot.

2.2.3 Détection de mots clés

Le terme *détection de mots clés* couvre les applications dont le but est de rechercher des mots bien spécifiques dans un document sonore. Les documents sonores peuvent être des fichiers préenregistrés ou par exemple une radio en flux continu. Pour chaque mot clé détecté, une alarme est envoyée à une application tierce qui gérera cet évènement. Un système de détection de mots clés est fondé sur la donnée d'une liste limitée de mots qui doit être préalablement définie en coordination avec les besoins de l'application. La taille de cette liste peut dépasser plusieurs centaines de mots.

Ce type d'application est développé par exemple par des sociétés prestataires qui doivent retranscrire toutes les émissions ou tous les débats évoquant les noms de leurs clients. Mus par un intérêt différent, les réseaux d'écoute à grande échelle peuvent tirer partie d'une telle application de détection de mots clés afin de filtrer uniquement les communications contenant des mots clés précis. Les applications de détection de mots clés peuvent également trouver leur place dans des systèmes à commande vocale autorisant un dialogue en parole spontanée [Wilcox 91, Foote 95]. Gorin et al. [Gorin 97] ont utilisé la détection de certains mots clés afin de déterminer la sémantique du dialogue ou du document sonore, pour des applications d'indexation, de classement par le contenu ou de reconnaissance de thème. Pour ces applications, les mots clés détectés doivent être valides afin de ne pas induire d'erreur de sémantique.

Dans la littérature, deux méthodes sont habituellement utilisées pour la détection de mots clés. Une première méthode consiste à associer à chaque mot de la liste un modèle et un anti-modèle [Sukkar 96, BenAyed 03]. L'anti-modèle (*garbage model*) représente la modélisation de n'importe quel mot possible, excepté le mot clé considéré. Le modèle de chaque mot est recherché dans le signal. Le décodage de la phrase entière n'est pas nécessaire car la recherche s'effectue localement et progressivement dans le signal. Les portions traitées où aucun mot clé n'a été trouvé sont simplement rejetées. La décision finale de retenir ou non un mot clé potentiel se fait par *comparaison* avec l'anti-modèle du mot. Les systèmes fondés sur cette méthode ont l'avantage d'être peu sensibles à des problèmes tels que des hésitations ou des reprises dans le dialogue que l'on retrouve souvent en parole spontanée [Wilpon 90].

Une deuxième méthode propose d'effectuer une reconnaissance grand vocabulaire du signal afin d'obtenir une transcription du document sonore, puis de rechercher les mots clés dans le document textuel obtenu. Cette méthode présente l'avantage de faciliter la recherche des mots clés et de diminuer les cas de détection d'un mot clé qui n'est en fait qu'une sous-partie de mot : par exemple le mot clé « action » est contenu dans le mot « satisfaction ». Toutefois, cette méthode nécessite la mise en place d'un système automatique de reconnaissance de la parole grand vocabulaire [Rose 95a].

Quelle que soit la méthode de recherche employée, celle-ci est habituellement couplée avec une phase de décision : accepter ou rejeter l'hypothèse de mot clé trouvée. Cette opération de décision implique principalement deux types d'erreur :

- fausse acceptation : un mot clé détecté a été accepté à tort.
- faux rejet : un mot clé détecté a été rejeté à tort.

Suivant l'application visée, un des deux types d'erreur peut être plus important que l'autre. Par exemple, si l'application consiste à détecter des mots clés critiques tels que des alarmes, des cris ou des appels au secours, il faut minimiser le taux de faux rejet quitte à déclencher des procédures d'alarmes à tort. Par contre, pour des applications qui veulent seulement être certaines qu'un mot clé a bien été prononcé, le plus important est d'obtenir un taux de fausses acceptations faible. Toutefois, les deux taux de faux rejets et de fausses acceptations sont liés et généralement un point de fonctionnement doit être déterminé afin de trouver un compromis entre ces deux taux d'erreur.

Afin de diminuer ces taux d'erreurs, plusieurs travaux ont introduit l'utilisation de mesures de confiance. Par rapport à un seuil déterminé et à une valeur de confiance calculée, une hypothèse de mot clé est acceptée ou rejetée. Ferrer et al. [Ferrer 01] ont défini une mesure de confiance à partir d'une combinaison linéaire entre le score d'une hypothèse de mot clé et un score représentant la distance entre le mot clé et une boucle de phonèmes. La liste de mots clés comportait 18 entrées avec un système de reconnaissance utilisant un vocabulaire de 700 mots. Sur un corpus téléphonique de demandes d'informations concernant des cartes de crédit (SWITCHBOARD) et avec un système de reconnaissance utilisant un vocabulaire de 5000 mots,

Weintraub [Weintraub 95, Weintraub 97] a également introduit une mesure de confiance fondée sur un rapport de vraisemblance afin de diminuer les taux de fausses acceptations.

2.2.4 Dialogue Homme/Machine

L'interaction entre l'homme et la machine n'a fait qu'augmenter au fil des années. Au départ, cette interaction se limitait à un dialogue directif permettant de donner des ordres à l'aide d'un programme de reconnaissance au vocabulaire minimaliste et très contraint. La communication ne se réalisait que dans un sens. Toutefois, certaines applications comme par exemple l'interrogation d'horaires de train nécessitent un dialogue, même rudimentaire.

La mise en place d'un dialogue oral est une tâche complexe. L'application doit pouvoir non seulement reconnaître ce qu'a dit l'utilisateur, mais également détecter des erreurs, hésitations, reprises. Plus le système laisse de liberté à l'utilisateur dans le dialogue et plus la mise en place d'un tel système sera complexe et devra gérer des cas particuliers.

Il est impossible de définir un système de dialogue homme/machine qui puisse répondre à n'importe quelle demande de l'utilisateur. Il sera toujours nécessaire de restreindre l'application de dialogue à un domaine bien précis [Williams 04].

L'acceptation par l'application d'une réponse erronée peut amener des conséquences gênantes pour l'usager et laisser un sentiment de méfiance envers celle-ci. Des mesures de confiance ont ainsi été développées dans le cadre de la vérification de phrases dans les systèmes de dialogue. Par exemple, San-Segundo et al. [San-Segundo 01] ont développé des mesures de confiance, pour un système de dialogue, à plusieurs niveaux : mots, phrases et concepts (ville de départ, d'arrivée, etc). Il est ainsi possible de rejeter des mots mal reconnus ou des phrases hors contexte par rapport à l'application. De plus, si un mot correspondant à un concept tel qu'une ville d'arrivée a une faible valeur de confiance, le système de dialogue pourra reposer la question ou demander une confirmation. De même Hazen et al. [Hazen 02] ont introduit des mesures de confiance aux niveaux des phones, des mots et des phrases dans leur système d'interrogation météorologique. Les mesures de confiance permettent ainsi au système d'être sûr des mots importants et porteur de sens pour l'application et de guider le dialogue en conséquence.

2.2.5 Apprentissage semi-supervisé

Dans les systèmes de reconnaissance de la parole, la modélisation acoustique joue un rôle très important. Plus les modèles sont précis, meilleur sera le résultat de la reconnaissance. Habituellement, l'apprentissage des modèles acoustiques s'effectue à partir d'un corpus contenant à la fois des exemples acoustiques et leur transcription. Cette transcription du corpus, faite de façon manuelle, est très longue et fastidieuse. Cette manière de construire les modèles acoustiques s'appelle un apprentissage supervisé.

Pour obtenir des modèles acoustiques précis, le corpus doit être d'une taille importante. De nos jours, la taille du corpus s'échelonne entre quelques dizaines d'heures et plusieurs milliers d'heures. Or, étiqueter manuellement une telle quantité de données n'est pas une solution réaliste. La technique alors utilisée consiste à apprendre une première ébauche des modèles sur un corpus plus petit, étiqueté manuellement, puis, à réaliser un étiquetage automatique du corpus complet à l'aide de ces modèles imparfaits. Ensuite, un nouvel apprentissage des modèles est réalisé sur le corpus complet. Cet apprentissage, appelé semi-supervisé, consiste donc à effectuer une reconnaissance avec des modèles grossiers afin d'augmenter la taille du corpus d'apprentissage et d'affiner les modèles acoustiques.

La phase de ré-apprentissage peut se faire de deux manières :

– ré-apprendre tous les modèles depuis le début mais sur le corpus complet,

– effectuer une ré-estimation des modèles à partir des nouvelles données du corpus.

L'apprentissage des modèles est habituellement fondé sur l'algorithme de *maximisation de l'espérance* (Expectation Maximisation – EM). Généralement, plusieurs itérations successives du processus sont effectuées en utilisant à chaque fois les modèles ré-estimés. Ces itérations améliorent la qualité de la modélisation acoustique [Gunawardana 03, Lamel 02].

Cependant, si la taille des données initiales est très réduite, les premiers modèles conduisent à un taux d'erreur important lors de l'étiquetage automatique du corpus. Accepter sans restriction les résultats de cette phonétisation en vue du ré-apprentissage des modèles pose un problème. Le risque de dégrader les modèles serait trop élevé par rapport à l'amélioration espérée. Un autre problème apparaît au niveau de l'algorithme EM. En effet, si la modélisation et les données réelles sont trop éloignées (conditions environnementales), les modèles se détériorent [Cohen 04].

Aussi, plusieurs travaux de recherche présentent-ils des méthodes afin d'augmenter la taille du corpus d'apprentissage initial par sélection de nouveaux exemples parmi un corpus non étiqueté manuellement. Une partie de ces travaux repose sur l'introduction de mesures de confiance afin d'effectuer cette phase de sélection [Kemp 98, Zavaliagkos 98, Kemp 99, Wessel 05, Ma 07]. La mesure de confiance joue un rôle d'outil de validation des résultats de la phonétisation ou de la reconnaissance. Par exemple, les phrases reconnues ayant une valeur de confiance supérieure à un seuil seront ajoutées au corpus d'apprentissage alors que les autres phrases jugées non fiables seront tout simplement rejetées. Ainsi, grâce à cet effet de sélection, les nouveaux modèles seront plus pertinents car appris sur des exemples correctement étiquetés. Il faut toutefois surveiller le problème de la spécialisation des modèles au détriment de la généralisation de ceux-ci.

Le critère de sélection des nouveaux exemples peut varier suivant l'application visée. Par exemple, si l'objectif est l'augmentation du corpus d'apprentissage des modèles acoustiques, certaines erreurs d'accord dans la transcription automatique n'influent pas forcément sur la prononciation. Rejeter des mots ayant la même prononciation alors que l'objectif est justement acoustique est trop strict. La mesure de confiance devra se situer au niveau de la transcription phonétique plutôt qu'au niveau des mots. Ainsi par exemple, reconnaître « petits bateaux » à la place de « petit bateau » ne doit pas être considéré comme une erreur puisque la suite de phonèmes est la même. En revanche, si l'objectif est d'augmenter un corpus d'apprentissage d'un modèle de langage, la mesure de confiance doit effectuer une sélection plus stricte au niveau des mots car les accords sont primordiaux.

La figure 2.1 résume les étapes de création des modèles acoustiques avec ré-estimation et en utilisant une mesure de confiance.

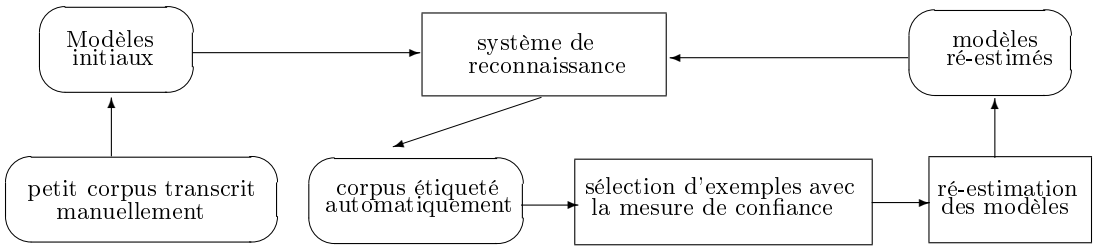


FIG. 2.1 – Etapes de réalisation d'un apprentissage semi-supervisé avec l'utilisation d'une mesure de confiance.

2.2.6 Adaptation

Les modèles acoustiques utilisés dans les systèmes de reconnaissance de la parole sont généralement appris sur des corpus de taille importante. Les modèles obtenus sont ainsi génériques et peuvent permettre la reconnaissance indépendamment du locuteur et des variations des conditions d'enregistrement etc. Cependant, si nous observons les résultats de reconnaissance locuteur par locuteur, nous pouvons remarquer une disparité des performances. En effet, bien que ce soit leur objectif, il est impossible pour les modèles acoustiques génériques de représenter uniformément tous les locuteurs. Pour améliorer les performances du système de reconnaissance il est nécessaire d'adapter les modèles acoustiques aux caractéristiques vocales d'un locuteur particulier.

Les deux techniques d'adaptation communément employées dans la littérature englobent les transformations linéaires et les adaptations bayésiennes des modèles. La famille des adaptations par transformation linéaire est fondée sur la définition d'une fonction linéaire exprimant les nouveaux modèles en fonctions des paramètres des modèles initiaux. La technique originelle dite d'adaptation par régression linéaire avec maximisation de la vraisemblance (Maximum Likelihood Linear Regression – MLLR) a été introduite par Leggetter et al. [Leggetter 95]. La famille des adaptations bayésiennes prend en compte la distribution *a priori* des modèles acoustiques et effectue une adaptation suivant le critère du *maximum a posteriori* (MAP) [Lee 91, Gauvain 94].

Le système débute la reconnaissance avec des modèles génériques, puis, au fur à mesure de la progression de la reconnaissance, les modèles sont adaptés par rapport aux nouvelles données reconnues. La phase d'adaptation nécessite d'itérer le processus afin d'obtenir une convergence suffisante des modèles. Un problème similaire à celui présenté dans la section précédente apparaît : peu de données étant disponibles pour adapter les modèles, les erreurs de reconnaissance peuvent avoir un impact important et détériorer les modèles acoustiques adaptés.

Plusieurs travaux ont proposé d'utiliser des mesures de confiance afin de sélectionner les données qui serviront à l'adaptation au locuteur [Anastasakos 98, Ngyen 99, Wallhoff 00, Pitz 00]. La sélection peut s'effectuer à différents niveaux : phonèmes, mots, phrases.

2.3 Mesures de confiance

Les mesures de confiance ont pour but de donner un aspect qualitatif aux résultats d'une application. En effet, les systèmes de reconnaissance sont généralement fondés sur la vraisemblance des solutions afin de déterminer la meilleure séquence, mais la vraisemblance ne permet pas, à elle seule, de donner une idée de la qualité de la solution. C'est pourquoi il est nécessaire de trouver d'autres indices, d'autres critères afin d'estimer la fiabilité de la solution.

Nous pouvons regrouper les sources d'indices en deux catégories :
- celles provenant du système de reconnaissance, généralement d'ordre probabilistes,
- celles provenant de sources de connaissances externes, souvent plus heuristiques.

Le choix d'une mesure de confiance est lié à l'application visée. Quand un étiquetage du résultat entre *correct* ou *incorrect* est suffisant, la plupart des mesures de confiance que l'on peut définir conviennent. Elles dépendent d'un seuil qui délimitera la frontière entre les résultats considérés comme *justes* de ceux considérés comme *faux*. Par contre, si l'application nécessite une évaluation plus précise, sans séparation des résultats en deux classes, les mesures de confiance fondées sur des critères heuristiques sont rarement adaptées.

Dans cette section, nous présentons un panel de mesures de confiance communément rencontrées. Plusieurs directions ont été étudiées dans divers travaux de recherche. Par exemple

l'analyse du comportement du système de reconnaissance est à l'origine de mesures fondées sur la stabilité acoustique, le comportement du repli du modèle de langage, le nombre de trames des mots, etc [Finke 96, Bansal 98, Wessel 01, Moreno 01, Uhrik 97, Jouvet 99]. L'apport de connaissances extérieures telles que la sémantique ou les catégories des mots a également conduit à la construction de mesures de confiance [Bellagarda 98, Cox 00, Guo 04, Pao 98].

Cependant les deux courants majeurs optent pour une analyse plus probabiliste des résultats du système de reconnaissance. Le premier courant aborde le problème d'un point de vue *test d'hypothèse*. Ce test statistique permet de déterminer si une hypothèse peut être acceptée ou rejetée. Ce test est à l'origine des mesures fondées sur un rapport de vraisemblance [Rose 95b, Sukkar 96, Rahim 96, Ramesh 98, Young 94b, Rahim 95, Weintraub 95, Cox 96, Setlur 96, Weintraub 97] etc.

Le second courant tente d'estimer la probabilité *a posteriori* du résultat (ou bien des mots constituant la phrase résultat) [Rueber 97, Jeanrenaud 93, Kemp 97]. Cette probabilité représente le meilleur critère exploitable à partir du système de reconnaissance. D'ailleurs, les mesures fondées sur la probabilité *a posteriori* sont celles qui donnent les meilleurs résultats [Jiang 05].

Chacune de ces mesures apportant son lot de connaissances et de pouvoir de décision, des travaux ont étudié la combinaison de plusieurs mesures de confiance ou de critères d'analyse [Finke 96, Cox 96, Schaaf 97, Gillick 97, Chase 97, Bansal 98, Kamppari 00, Duchateau 02a, Guo 04].

Nous allons présenter dans les paragraphes qui suivent les principales mesures de confiance de l'état de l'art en essayant de les classer selon le critère sur lequel elles sont fondées. Nous terminerons par les mesures relevant des deux courants majeurs cités et en détaillant la méthode proposée par F. Wessel puisque l'une de nos mesures s'inspire de celle-ci. Enfin, nous présenterons quelques méthodes de fusion des mesures de confiance.

2.3.1 Critères *non probabilistes*

2.3.1.1 Stabilité acoustique

Le critère de stabilité acoustique repose sur un principe assez simple : si un mot apparaît souvent à la même position dans un ensemble de plusieurs phrases hypothèses, alors ce mot doit être certainement correct [Finke 96, Qiu 96]. Un ensemble de phrases hypothèses est généré à partir du système de reconnaissance en utilisant différents jeux de poids pour les scores des modèles acoustiques et linguistiques. Ensuite, chaque phrase est alignée par rapport à la meilleure hypothèse à l'aide d'un algorithme de programmation dynamique. Puis, pour chaque mot de la meilleure phrase, le critère de confiance est calculé ainsi : le nombre d'occurrences du mot à la même position dans toutes les hypothèses alignées est normalisé par le nombre total d'hypothèses [Schaaf 97, Kemp 97, Wessel 01].

2.3.1.2 Densité d'hypothèses

Le critère de densité d'hypothèses vient de l'analyse du comportement du nombre d'hypothèses actives au cours de la reconnaissance. La majorité des systèmes de reconnaissance de la parole effectue un élagage des hypothèses pour ne conserver et ne prendre en compte qu'un nombre plus restreint de solutions candidates. Ainsi, à un instant précis du déroulement de la reconnaissance, il n'y a qu'un nombre limité d'hypothèses qui seront conservées pour la suite du décodage. Habituellement, l'élagage dépend de deux critères :

– une limite maximale fixe d'hypothèses actives,

– une limite qui est une fonction du score de la meilleure hypothèse.

Cette dernière limitation influe sur le nombre d'hypothèses qui peut ainsi être bien inférieur à la limite maximale. Le critère de la densité d'hypothèses exploite le nombre ainsi limité d'hypothèses restantes. Plus le score de vraisemblance de la meilleure phrase hypothèse se détachera des autres et plus le nombre d'hypothèses actives sera réduit du fait de l'élagage. Inversement, si beaucoup d'hypothèses ont un score similaire, un grand nombre d'entre elles seront actives et cela signifiera que la meilleure hypothèse est peu fiable [Cox 96, Kemp 97].

D'autres critères similaires ont été introduits et prennent en compte par exemple le nombre moyen d'hypothèses au début et à la fin du mot [Moreno 01]. Il est également possible de compter le nombre de séquences où le mot analysé apparaît à la bonne position parmi les n-meilleures phrases [Gillick 97].

2.3.1.3 Dépendance des mots

Bansal et Ravishankar [Bansal 98] proposent d'étudier la stabilité de la solution fournie par le système de reconnaissance en perturbant le décodage de la phrase. Deux axes sont étudiés : la *dépendance à la vraisemblance* et la *dépendance au voisinage*.

La dépendance à la vraisemblance consiste à analyser, pour un mot w de la phrase, la variation de la vraisemblance globale entre :

– la solution initiale sans contraintes supplémentaires,
– la meilleure solution avec comme contrainte l'exclusion des occurrences du mot w comme hypothèse possible.

Ainsi, si la vraisemblance de la phrase varie faiblement suivant la présence ou non du mot w dans la phrase, une faible confiance peut lui être attribuée. Par contre, si l'absence du mot w influe de manière importante sur la vraisemblance de la solution, c'est que w joue un rôle majeur dans la phrase et doit avoir une confiance forte.

La dépendance au voisinage consiste à contraindre, pour un mot w, les mots de la solution apparaissant à son voisinage. Pendant la phase de décodage de la phrase, les mots voisins de w sont exclus de la meilleure solution, ce qui modifie le résultat final. Cependant, si le mot w figure toujours parmi la solution trouvée, cela malgré l'exclusion de ses meilleurs voisins, alors ce mot doit être correct.

2.3.2 Critères relatifs au modèle de langage

2.3.2.1 Le modèle de langage

Les modèles de langage statistiques en eux-mêmes permettent de donner un score de confiance au niveau des mots pour le résultat d'une reconnaissance. L'idée que « si un mot w fait partie du résultat du système de reconnaissance et que ce mot, ou une séquence de mots contenant w, a une fréquence d'apparition élevée alors w est certainement correct » apparaît simple et logique. Duchateau et al. [Duchateau 02a] ont étudié des mesures de confiance fondées sur :

– un modèle de langage *forward*,
– un modèle de langage *backward*,
– une combinaison des deux modèles de langage *forward* et *backward*.

Si nous considérons le cas de probabilités linguistiques trigrammes et la séquence de mots $w_1w_2w_3$, le modèle de langage *forward* correspond à la probabilité $P(w_3|w_1w_2)$, les mots sont pris dans le sens de l'écriture et donc un mot dépend des mots le précédant. Le modèle de langage

backward correspond à la probabilité $P(w_1|w_2w_3)$, les mots sont pris de la fin de la phrase vers le début. La probabilité d'un mot w_1 dépend donc des mots qui le suivent dans la phrase.

Ces probabilités linguistiques ont été combinées avec d'autres critères tels que le score du modèle acoustique et la largeur du faisceau de recherche [Duchateau 02a, Weintraub 97]. A partir d'un corpus de nouvelles du *Wall Street Journal* lues et d'un lexique de 20 000 mots, Duchateau et al. notent que la connaissance du modèle de langage *backward*, en sus du modèle *forward*, permet d'améliorer de façon notable les performances des mesures de confiance. Ce résultat s'explique principalement par l'augmentation de la taille du contexte qui est pris en compte. Ils montrent également que l'utilisation conjointe de deux modèles n-gramme, l'un *forward* et l'autre *backward*, est plus pertinent que l'utilisation d'un seul modèle de langage *forward* $(2n-1)$-gramme.

2.3.2.2 Repli du modèle de langage

Bien que le modèle de langage à lui seul puisse être considéré comme une mesure de confiance, l'information la plus communément utilisée est le nombre de fois où le système a été obligé de faire appel à un repli du modèle de langage (*backoff*). Le repli est la méthode qui permet d'estimer la probabilité linguistique d'un évènement non rencontré au cours de l'apprentissage du modèle de langage (cf. 1.5.2 p.11). Le fait qu'une séquence de mots n'ait jamais été rencontrée dans le corpus d'apprentissage peut en effet amener un doute sur la fiabilité de cette séquence, même si celle-ci fait partie du meilleur chemin. Plus le nombre de replis nécessaires dans l'estimation de la probabilité n-gramme d'une séquence de mots est important, plus cette estimation est grossière [Ravishankar 96]. Différents travaux ont exploité cette information. Par exemple, en entrée d'un réseau de neurones, Weintraub et al.[Weintraub 97] introduisent non seulement les probabilités trigrammes *forward* et *backward*, mais également les niveaux de repli utilisés dans le modèle de langage pour calculer ces probabilités sur un corpus de conversations téléphoniques.

A partir de l'idée que plus le calcul de la probabilité linguistique d'un n-gramme fait appel à différents niveaux de replis, plus la fiabilité de cette probabilité est faible, C. Uhrik [Uhrik 97] a défini des mesures de confiance uniquement basées sur ces niveaux de repli. Il a ainsi affecté une valeur de confiance au niveau des mots suivant le degré de repli rencontré dans le calcul de la probabilité trigramme de ce mot (*conf1*). La valeur de confiance est choisie de manière arbitraire, répartie entre 0 et 1 suivant les niveaux de replis possibles. Par exemple, un mot, dont le trigramme a bien été estimé dans le corpus d'apprentissage, aura une valeur de confiance 1. Si le trigramme n'existe pas, mais que les bigrammes impliqués dans le repli sont présents dans le modèle de langage, alors la valeur de confiance sera de 0,8. Ainsi de suite, plus le nombre de replis nécessaires est important et plus la valeur de confiance se rapproche de 0, avec une borne inférieure de 0.1 pour les mots jamais rencontrés dans le corpus d'apprentissage du modèle de langage. Il introduit également une seconde mesure de confiance *conf2* pour un mot w en fonction des valeurs de confiance *conf1* des mots présents dans la séquence n-gramme de w. Puis, une mesure au niveau de la phrase elle-même est définie par la moyenne des valeurs de confiance *conf2* des mots de la séquence. L'objectif de cette mesure était d'accepter ou rejeter des phrases entières dans un contexte de comptes rendus d'examens médicaux.

Dans [Mauclair 06], les auteurs définissent une mesure de confiance également fondée sur l'analyse du repli du modèle de langage. A chaque mot de la phrase solution du système de reconnaissance est associé l'ensemble des séquences des n-gramme du mot considéré. Par exemple si la phrase reconnue contient la séquence « ... demain il fera ... », alors l'ensemble associé au mot *fera* est {demain il fera ; il fera, fera, $\emptyset$}. Dans cet ensemble, certaines séquences n-grammes ont été rencontrées dans un corpus de développement qui a servi à la fois à apprendre les modèles acoustiques et linguistiques. La valeur de l'ordre du n-gramme rencontré le plus élevé est affectée

au mot analysé (*fera*), c'est-à-dire que si la séquence « demain il fera » a été rencontrée dans le corpus de développement alors la valeur 3 sera affectée au mot *fera*. A partir de ces valeurs calculées pour chaque mot w d'une phrase, un triplet est associé à w exprimant l'ordre du n-gramme de w par rapport au mot précédent et suivant. Par exemple, si le mot *demain* possède un n-gramme d'ordre 2, *il* et *fera* un n-gramme d'ordre 3, le triplet associé à *il* est défini par $\{-;3;=\}$. Ce triplet représente une sorte de critère d'homogénéité de l'ordre des n-grammes des mots par rapport à leurs voisins. Ces triplets forment des classes d'équivalence. La reconnaissance automatique d'un corpus de développement permet de calculer un taux d'erreur en mots moyen pour chacune de ces classes. Lors du test, à chaque mot w du corpus de test correspond une unique classe d'un triplet particulier. La mesure de confiance du mot w est alors définie comme étant le taux d'erreur en mots moyen de sa classe. Les auteurs ont évalué cette mesure de confiance ainsi que la fusion de celle-ci avec une mesure acoustique sur un corpus de bulletins d'informations radiophoniques. L'objectif de cette mesure de confiance était d'augmenter la taille du corpus d'apprentissage des modèles acoustiques avec des séquences de mots reconnus ayant une valeur de confiance élevée (cf. l'apprentissage semi-supervisé).

2.3.3 Critères sémantiques et syntaxiques

Les trois mesures de confiance présentées dans cette section sont fondées sur des données de plus haut niveau, différentes des modélisations acoustiques et linguistiques utilisées dans le moteur de reconnaissance. Ces informations extérieures au système peuvent venir, par exemple, de l'analyse de la similarité sémantique entre les mots, de l'étude de l'appartenance catégorielle des mots, ou encore du calcul de l'information mutuelle entre les mots.

2.3.3.1 Analyse sémantique latente

L'analyse sémantique latente (Latent Semantic Analysis – LSA) est une technique largement employée dans le domaine de la recherche d'information. Son application dans le domaine de la parole s'est faite peu après au niveau des modèles linguistiques par le biais des travaux de [Bellagarda 98].

L'analyse LSA permet de définir une corrélation entre les mots qui apparaissent en même temps dans un document (un article par exemple) et indiquant ainsi que ces mots sont sémantiquement liés [Landauer 97, Cox 00]. Une matrice des co-occurrences mots/documents est construite sur un corpus d'apprentissage contenant un ensemble de documents. A chaque document correspond un ensemble de mots y appartenant, et, à chaque mot correspond un ensemble de documents dans lesquels il apparaît. Cette matrice très creuse peut voir sa dimension grandement réduite en appliquant une décomposition en valeurs singulières (SVD). Dans cet espace de dimension réduite, la propriété de la méthode LSA implique que deux vecteurs de mots proches dans cet espace seront des mots sémantiquement similaires. De ces informations, il est possible de construire une matrice de similarité sémantique entre les mots puis d'estimer la vraisemblance de deux mots co-occurrents dans la même phrase.

Dans leurs travaux, Cox et al. [Cox 02] ont utilisé un corpus de journaux lus, *Wall Street Journal*, avec un vocabulaire de 20 000 mots. Ils ont défini un document comme étant le texte correspondant à un sujet d'information et ont construit une matrice mot/document W. La matrice de co-occurrence est de largeur 19685 et de longueur 19396 suivant respectivement le nombre de mots distincts et le nombre de documents dans le corpus. Cette matrice a été réduite par la méthode SVD à une taille de vecteur de dimension 100. Un score sémantique, $S(w_i, w_j)$, entre deux mots w_i et w_j est défini par le cosinus entre les deux vecteurs de W associés à chacun des

mots. Cependant, tous les mots ne sont pas pris en compte. En effet certains mots fonctionnels apparaissent dans pratiquement toutes les phrases et ont donc un fort score de similarité sémantique avec tous les autres mots sans vraiment porter de sens. La solution envisagée pour résoudre ce problème consiste à fixer un seuil préalablement déterminé séparant les mots pour lesquels une mesure de confiance sera définie par un score de similarité.

Différentes mesures fondées sur cette valeur de similarité sémantique ont été testées directement en tant que mesure de confiance et obtiennent des performances relativement identiques. Toutefois, ces performances ne permettent pas de séparer clairement ni les mots correctement décodés ni les mots incorects. L'avantage de cette mesure est d'être indépendante du système de reconnaissance : les valeurs de la mesure sont apprises sur le même corpus que les modèles statistiques du système et sans utiliser de données issues de la reconnaissance. Cox et al. ont aussi combiné leur mesure de confiance sémantique avec une mesure de confiance fondée sur le critère de stabilité acoustique à partir de la liste des n-meilleures phrases. Cette combinaison permet d'obtenir une mesure de confiance qui identifie un petit nombre de mots correctement décodés avec une très grande confiance.

2.3.3.2 Information mutuelle

En analyse sémantique latente, une mesure de similarité sémantique est définie à partir de l'analyse de la co-occurrence des mots dans un ensemble de documents. La connaissance du nombre de co-occurrences $N(x, y)$ permet également de définir la probabilité jointe de co-occurrence de deux mots x et y dans n'importe quel document.

$$P(x, y) = \frac{N(x, y)}{\sum_{x,y} N(x, y)} \tag{2.1}$$

Il est alors possible de calculer les probabilités marginales des mots x et y par ces deux équations :

$$P(x) = \sum_{y} P(x, y) \tag{2.2}$$

$$P(y) = \sum_{x} P(x, y) \tag{2.3}$$

L'information mutuelle entre deux mots x et y peut donc par définition être calculée ainsi :

$$IM = \log \frac{P(x, y)}{P(x)P(y)} \tag{2.4}$$

Les auteurs de [Guo 04] définissent alors la mesure de confiance d'un mot x comme étant la moyenne des informations mutuelles entre x et tous les mots y de la phrase solution du système de reconnaissance. Cependant, comme pour l'apprentissage d'un modèle de langage statistique n-gramme, beaucoup de paires de mots x, y ne sont jamais rencontrées ensemble dans le corpus d'apprentissage. Pour pallier le manque de fiabilité de l'estimation de l'information mutuelle de ces paires, une technique de lissage a été employée. En considérant la phrase solution comme un document, les auteurs ont mené des expériences sur deux corpus :

– un corpus de dialogues téléphoniques SWITCHBOARD. La matrice d'information mutuelle est apprise sur l'ensemble du corpus (250 000 phrases). Les tests sont effectués par validation croisée.

– un corpus de journaux chinois lus, dans un contexte grand vocabulaire (65 000 mots). La matrice d'information mutuelle est apprise à partir d'environ 80 000 phrases du corpus d'apprentissage.

La mesure seule et une combinaison linéaire de celle-ci avec une mesure fondée sur la probabilité *a posteriori* ont été évaluées suivant le critère du taux d'égale erreur EER. Ils concluent que l'information mutuelle est plus pertinente que l'analyse sémantique latente. Toutefois, les performances de ces mesures fondées sur des informations sémantiques seules sont inférieures à des mesures plus liées au système de reconnaissance (41,4% d'EER pour la mesure fondée sur l'information mutuelle contre 24,4% pour la probabilité *a posteriori*).

2.3.3.3 Catégorie d'un mot

Dans [Pao 98], les mots sont classés dans des catégories sémantiques. Pour chaque classe sémantique un poids est défini en fonction de l'importance de la classe par rapport à l'application visée, par exemple une application d'interrogation de serveur météorologique dans le cas de [Pao 98] (59 000 phrases). Dans cette application, les noms de lieux ont un poids important tandis que les mots fonctionnels comme « merci » ont un poids bien plus faible. Le poids sémantique d'une phrase est la somme des poids des classes des mots de la phrase. Deux mesures de confiance sont définies par : le poids sémantique de la meilleure phrase, et la distance sémantique entre les trois meilleures hypothèses de la liste des n-meilleures phrases. L'objectif est d'accepter ou rejeter des phrases entières suivant la valeur de confiance de celles-ci. Ces mesures de confiance sont combinées à d'autres paramètres tels que le score linguistique, le score acoustique par phone, etc. De plus, une analyse discriminante linéaire de Fisher et une analyse par arbres de régression ont été utilisées afin de sélectionner un ensemble des meilleurs paramètres de confiance.

[Stemmer 02] introduit également une mesure de confiance fondée sur une classification en catégorie des mots dans le cadre de dialogues entre personnes issus du corpus VERMOBIL (20 000 phrases). Le nombre de catégories est important (environ 160) et les probabilités d'appartenance d'un mot à une catégorie sont apprises sur un corpus étiqueté manuellement.

2.3.4 Autres critères empiriques

Bien d'autres mesures de confiance ont été définies à partir d'idées plus ou moins simples pouvant fournir une information qui aiderait à décider de la justesse d'un phonème, d'un mot ou d'une phrase.

Généralement, ces critères non statistiques ne sont pas utilisés seuls mais sont combinés en un vecteur d'indices qui pourra servir à l'apprentissage d'un modèle.

Ces critères concernent par exemple :

– l'étude de la durée des phonèmes dans le mot [Cox 96, Jouvet 99, Vergyri 00],
– l'analyse de la vitesse d'élocution [Schaaf 97, Hernández-Abrego 00],
– le nombre de phonèmes dans le mot [Weintraub 97],
– la prosodie de la phrase [Jouvet 99],
– le nombre de phonèmes en commun entre le mot hypothèse et une boucle de phonèmes non contrainte [Chase 97],
– le nombre de fois que le mot a été rencontré dans le corpus d'apprentissage des modèles acoustiques [Chase 97],
– le rapport signal à bruit du mot, le minimum et le maximum du rapport signal à bruit des trames du mot [Schaaf 97],

– la différence entre le score acoustique du mot et celui du meilleur score [Gillick 97].

2.3.5 Mesures fondées sur le rapport de vraisemblance

Les mesures de confiances fondées sur le rapport de vraisemblance sont une extension des tests d'hypothèse. La notion de test d'hypothèse est importante pour des situations qui nécessitent l'acceptation ou le rejet d'un évènement. Ces critères de décision peuvent ainsi être dérivés en mesures de confiance pour des applications s'appuyant sur une séparation binaire des résultats. Nous allons commencer par présenter les tests d'hypothèse puis la façon dont ceux-ci sont étendus aux mesures de confiance.

2.3.5.1 Tests d'hypothèse

Les tests statistiques d'hypothèse sont des critères de décision concernant un état binaire d'un évènement par rapport à un seuil défini. Dans le cas de la reconnaissance de la parole, nous considérons le résultat produit par le système. Deux hypothèses H_0 et H_1 sont alors définies par :

– l'hypothèse nulle H_0 : le résultat du système de reconnaissance est correct.
– l'hypothèse alternative H_1 : le résultat du système est incorrect.

Le taux de reconnaissance global d'un système de reconnaissance étant généralement supérieur à 50%, nous supposons que le résultat de la reconnaissance est correct et allons tester l'hypothèse H_0.

Deux types d'erreur sont définis :

– erreur de première espèce : « faux » rejet de H_0 (appelée aussi erreur de type I).
– erreur de deuxième espèce : « fausse » acceptation de l'hypothèse H_0 (appelée également erreur de type II).

Tester l'hypothèse H_0 *versus* l'hypothèse H_1, c'est déterminer si nous devons accepter ou rejeter H_0. Le lemme de Neyman-Pearson énonce alors que la solution optimale du test d'hypothèse est fondée sur un rapport de vraisemblance et un seuil τ suivant la relation suivante :

$$LR = \frac{P(X|H_0)}{P(X|H_1)} \tag{2.5}$$

X représente le résultat du système de reconnaissance. Si $LR \geq \tau$, alors l'hypothèse H_0 est acceptée, sinon elle est rejetée. En faisant varier τ, il est possible d'influencer le nombre d'erreurs de première et deuxième espèce afin de favoriser un des deux types d'erreur.

2.3.5.2 Rapport de vraisemblance

Le rapport de vraisemblance de l'équation 2.5 sert de base afin de définir des mesures de confiance. Pour cela, il faut interpréter les hypothèses H_0 et H_1 suivant la modélisation utilisée dans la reconnaissance de la parole. Soient O une séquence d'observations correspondant à un signal de parole, M le modèle reconnu et $\widetilde{M}$ le modèle alternatif. Les hypothèses H_0 et H_1 s'expriment maintenant sous la forme suivante :

– H_0 : le modèle M a généré la séquence d'observations O.
– H_1 : le modèle alternatif $\widetilde{M}$ a généré la séquence d'observations O.

L'équation 2.5 s'exprime alors ainsi :

$$LR = \frac{P(O|M)}{P(O|\widetilde{M})} \tag{2.6}$$

La problématique du rapport de vraisemblance se concentre dans la modélisation de l'hypothèse alternative $\widetilde{M}$. Trois principales stratégies ont été décrites dans la littérature : la création d'un anti-modèle ou d'un modèle générique et l'utilisation des hypothèses concurrentes.

2.3.5.3 Modèle / Anti-Modèle

La méthode la plus communément employée consiste à entraîner un anti-modèle $\overline{M}$ spécifique pour chaque modèle M [Rahim 95, Rose 95b, Sukkar 96, Rahim 97, Moreau 00]. L'anti-modèle $\overline{M}$ est appris à partir de tous les éléments du corpus qui n'ont pas servi à engendrer le modèle M. Ainsi le système a, par exemple, pour chaque entité phonétique son modèle M et son anti-modèle $\overline{M}$. Les hypothèses H_0 et H_1 s'expriment maintenant sous la forme suivante :

- H_0 : le modèle M a généré l'observation O.
- H_1 : l'anti-modèle $\overline{M}$ a généré l'observation O.

Par exemple, Moreau et al. utilisent un rapport de vraisemblance modèle/anti-modèle afin de rejeter des noms ou des phrases dans le cadre de l'interrogation d'un répertoire téléphonique avec un vocabulaire spécifique de 2004 noms. Les résultats sont analysés du point de vue des taux de faux rejets et de fausses acceptations.

Habituellement, l'équation 2.6 n'est pas utilisée directement mais subit une transformation logarithmique. Nous obtenons ainsi dans le cas des anti-modèles l'équation suivante :

$$LLR = \log \frac{P(O|M)}{P(O|\overline{M})} \tag{2.7}$$

Le résultat du logarithme du rapport de vraisemblance peut être utilisé en tant que mesure de confiance au niveau des phonèmes [Sukkar 96, Ramesh 98]. En ce qui concerne les mots, plusieurs possibilités ont été étudiées : soit directement en travaillant avec une modélisation des mots, soit en moyennant les rapports de vraisemblance des phonèmes constituant les mots. Dans leurs travaux, Falavigna et al. [Falavigna 02] ont introduit une telle mesure de confiance dans le cadre d'une application d'acceptation/rejet avec trois sortes de corpus : des noms propres ou des noms de ville (1781 au total), des conversations téléphoniques de type SWITCHBOARD, des dialogues homme-homme de longueur courte (39 mots) en réponse à la question « Comment puis-je vous aider ? ». L'analyse a été faite suivant le taux d'égale erreur entre les faux rejets et les fausses acceptations.

D'autres travaux ont introduit une fonction de transformation monotone, par exemple sigmoïdale, afin de normaliser le rapport de vraisemblance dans l'intervalle $[0, 1]$ [Garcia-Mateo 99]. Dans leurs travaux, Garcia et al ont évalué les mesures de confiance qu'ils ont définies dans un cadre de reconnaissance de mots isolés (des noms propres au téléphone) à l'aide du critère du taux d'égale erreur.

2.3.5.4 Modèle générique

Une autre façon de générer un modèle alternatif $\widetilde{M}$ consiste à définir un modèle générique M' qui représente n'importe quelle entité [Kamppari 00, Mengusoglu 03, Fabian 05]. Une entité peut représenter un mot, un phonème ou une phrase. Par exemple, si nous désirons calculer le

rapport de vraisemblance entre des mots, dans ce cas, le modèle M' représente le modèle moyen de tous les mots du vocabulaire. Le modèle M' sera appris sur l'ensemble de toutes les entités du corpus. Le rapport de vraisemblance s'exprime alors ainsi :

$$LR = \frac{P(O|M)}{P(O|M')} \tag{2.8}$$

Cette méthode d'estimation de la confiance a été étudiée dans [Fabian 05] au niveau des états des modèles de Markov afin d'effectuer un élagage dynamique du faisceau de recherche au cours de la phase de décodage du moteur de reconnaissance. La valeur de confiance détermine la largeur du faisceau de recherche. Les auteurs ont évalué l'impact de l'intégration de cette mesure de confiance suivant le critère du taux d'erreur en mots du système de reconnaissance, ainsi que suivant un critère de facteur de temps gagné. Le cadre de l'expérience était des phrases (1000) issues de dialogue de réservation en allemand (VERMOBIL), avec un lexique de taille réduite (5343 mots).

Une seconde façon d'estimer $P(O|M')$ consiste à utiliser pour M' une boucle de phonèmes sans contraintes linguistiques. De cette manière, le modèle M' représente une suite de phonèmes dont le score acoustique est maximal pour chaque observation de O.

Ce rapport entre la vraisemblance d'un modèle M et un modèle générique M' représente en quelque sorte l'écart entre le modèle M et un modèle générique. Cette méthode est par exemple utilisée pour la détection de mots ou de phrases hors vocabulaire [Young 94b, Sukkar 96]. Pour Sun et al. [Sun 03] l'application consistait en la détection de mots hors vocabulaire (noms propres) pour des dialogues téléphoniques. L'analyse a été menée sur l'évolution des taux de faux rejets et de fausses alarmes.

D'autres travaux ont défini un modèle alternatif comme une combinaison d'un anti-modèle et d'un modèle générique [Lleida 96, Setlur 96]. Pour ces méthodes, à chaque modèle est associé un modèle alternatif, comme dans le cas de la définition d'anti-modèles.

2.3.5.5 Modèles compétitifs

Les deux méthodes précédentes nécessitent l'apprentissage de nouveaux modèles (anti-modèles, modèle générique) ou la mise en place d'un système de type boucle de phonèmes pour estimer la probabilité de la séquence d'observations. D'autres méthodes ne nécessitent pas l'apprentissage de modèles supplémentaires. Celles-ci ne sont fondées que sur la connaissance des modèles existant dans le système de reconnaissance, c'est-à-dire les différents modèles en compétition pendant la phase de décodage.

Ainsi [Cox 96] propose de faire le rapport entre le modèle M et le meilleur modèle concurrent au niveau du décodage. D'autres travaux, pour un vocabulaire V de taille restreinte, prennent en compte les modèles de tous les mots du vocabulaire et définissent le rapport de vraisemblance ainsi :

$$LR = \frac{P(O|M)}{\sum\limits_{\widehat{M} \in V \setminus \{M\}} P(O|\widehat{M})} \tag{2.9}$$

La normalisation se fait donc par la somme des vraisemblances de tous les modèles concurrents. Dans [Rahim 97], les auteurs exploitent cette méthode dans un but d'acceptation/rejet de phrases constituées de chiffres connectés *via* un téléphone (environ 6000 phrases).

Cependant, avec un système grand vocabulaire, cette méthode devient difficilement réalisable à cause du nombre trop important de modèles à prendre en compte. Une solution consiste à utiliser la liste des n-meilleures phrases générées par le système de reconnaissance afin de faire le rapport entre la vraisemblance de la phrase hypothèse et celle de la deuxième meilleure phrase ou de toutes les autres meilleures phrases [Boite 93, Rueber 97, Weintraub 97]. Charlet et al. [Charlet 01] ont par exemple défini une mesure de confiance en fusionnant un tel rapport de vraisemblance fondé sur les n-meilleures phrases avec le résultat d'un réseau de neurones combinant des critères de voisement, nasalité, etc. L'objectif était encore de rejeter des phrases de faible confiance dans le cadre d'application d'interrogation de répertoire de noms. Aussi, le vocabulaire était-il de taille réduite : 1587 mots. L'analyse des mesures de confiance a été faite en terme de taux de faux rejets et de fausses acceptations.

2.3.6 Mesures fondées sur les probabilité *a posteriori*

En reconnaissance automatique de la parole, les systèmes cherchent la séquence de mots qui maximise la probabilité que cette séquence ait généré la suite d'observations du signal de parole. Cependant, une fois la séquence solution déterminée, aucun indice de qualité de cette solution n'est disponible. En effet, dans la résolution de la reconnaissance (cf. équation 1.2), la normalisation de l'équation par la probabilité de l'émission $P(O)$ a été omise car cette probabilité est indépendante de la séquence de mots considérée.

La probabilité *a posteriori* $P(W|O)$ d'une phrase ou d'une séquence de mots W pour la séquence d'observations O est donnée par l'équation suivante :

$$P(W|O) = \frac{P(O|W)P(W)}{P(O)} \tag{2.10}$$

La quantité $P(O|W)P(W)$ est calculée par le moteur de reconnaissance au cours de la phase de décodage de la suite d'observations O. Connaître $P(O)$ permettrait de pouvoir normaliser la vraisemblance de la séquence trouvée afin de calculer la probabilité *a posteriori* de la séquence W, c'est-à-dire de la phrase reconnue. De par sa définition, la probabilité *a posteriori* semble être une bonne mesure de confiance de la phrase.

De la même façon, il est intéressant de définir la mesure de confiance d'un mot comme sa probabilité *a posteriori*. Toutefois, si nous voulons calculer la probabilité *a posteriori* de chacun des mots de W, des étapes supplémentaires sont nécessaires. Pour un mot w particulier, ceci peut être réalisé en sommant la probabilité *a posteriori* de toutes les phrases contenant w à la même position dans la séquence. Soit w_n le n^{ième} mot dans une phrase W. La probabilité *a posteriori* d'un mot w s'exprime alors ainsi :

$$P(w|O) = \sum_{W/w_n=w} P(W|O) \tag{2.11}$$

Un deuxième problème est soulevé ici, car pour un mot w, il faut déterminer toutes les phrases qui contiennent w, et également, expliciter la notion de *même position dans la phrase*.

Plusieurs travaux de recherche ont donc pour objectif d'estimer cette probabilité *a posteriori*. Pour cela ils ont le plus souvent utilisé une des deux structures issues du moteur de reconnaissance : la liste des n-meilleures phrases ou les graphes de mots (voir section 1.6.3 p.18).

Toutes les méthodes qui suivent et que nous présentons diffèrent sur la façon d'estimer et d'approximer $P(O)$ ainsi que la façon dont est déterminée la position d'un mot dans la phrase.

2.3.6.1 Mesures fondées sur la liste de n-meilleures phrases

La liste des n-meilleures phrases contient une liste de séquences de mots dont la vraisemblance était parmi les n-meilleures pendant le décodage. Comme indiqué précédemment, pour calculer la probabilité *a posteriori* d'un mot, il est nécessaire d'estimer $P(O)$ et de connaître la probabilité *a posteriori* de toutes les phrases qui contiennent ce mot. Or, il est facile de connaître l'existence d'un mot et sa position dans une des séquences de la liste des n-meilleures phrases, et comme ces phrases sont parmi les n-meilleures, leur contribution est majoritaire afin d'effectuer le calcul de la probabilité *a posteriori*. Ainsi de nombreuses mesures de confiances s'appuient sur la liste des n-meilleures phrases pour leur définition [Jeanrenaud 95, Stolcke 97].

Dans [Weintraub 95, Weintraub 97], la solution adoptée pour évaluer la probabilité *a posteriori* d'un mot clé consiste à sommer les probabilités de toutes les séquences de la liste des n-meilleures phrases contenant ce mot clé à la même position dans la phrase, puis, à normaliser cette quantité par la somme des probabilités de toutes les séquences de la liste des n-meilleures phrases. Rueber [Rueber 97] propose une méthode similaire pour déterminer la confiance de mots tels que des noms propres en utilisant les probabilités re-normalisées des séquences de la liste des n-meilleures phrases. La méthode initiée par Weintraub a été étendue au calcul de la probabilité *a posteriori* de tous les mots d'une phrase par F. Wessel et al. Afin de calculer la probabilité *a posteriori* d'un mot w, il est nécessaire de déterminer l'ensemble des séquences de la liste des n-meilleures phrases contenant w à la même position. Dans cette étude, la méthode d'alignement de Levenshtein a été employée, ce qui a permis finalement d'estimer les probabilités *a posteriori* des mots [Wessel 99].

L. Chase [Chase 97] ainsi que Gillick et al. [Gillick 97] étudient la proportion de séquences contenant un mot hypothèse à la même position parmi la liste des n-meilleures phrases afin de définir la probabilité *a posteriori*.

Le calcul de la probabilité *a posteriori* en utilisant les n-meilleures phrases présente l'avantage de la simplicité d'accès aux informations telles que la position d'un mot dans la phrase. De plus, le nombre de meilleures phrases retenues étant de l'ordre de la centaine, cette méthode est peu coûteuse d'un point de vue calculatoire. Toutefois, les mesures de confiance fondées sur les n-meilleures phrases sont des approximations assez fortes de la valeur théorique de la probabilité *a posteriori* et par conséquent sont moins précises que celles estimées à partir d'un graphe de mots ayant une densité d'hypothèses plus importante [Wessel 99].

Par ailleurs, ces mesures nécessitent la génération de l'ensemble des n-meilleures phrases et donc la terminaison complète du processus de reconnaissance. Ces mesures ne sont donc pas envisageables pour des applications en flux, comme par exemple la transcription en ligne d'émissions ou de cours dans une salle de classe.

2.3.6.2 Mesures fondées sur les graphes de mots et l'algorithme de *forward-backward*

La plupart des mesures de confiance estimant la probabilité *a posteriori* à partir d'un graphe de mots utilisent l'algorithme *forward-backwark* [Kemp 97, Metze 00, Wessel 01].

Le graphe de mots est une représentation compacte et assez précise de toutes les hypothèses émises et non élaguées lors de la phase de décodage du système de reconnaissance (cf. 1.6.3). Ainsi, le calcul de la probabilité *a posteriori* à partir de ce graphe de mots permet d'obtenir une estimation fine de $P(W|O)$. C'est pourquoi ces mesures de confiance donnent en général de meilleurs résultats que les autres formes de mesures de confiance [Jiang 05].

F. Wessel décrit dans [Wessel 01] une méthode qui permet d'estimer la probabilité *a posteriori* d'un mot et ainsi de définir une mesure de confiance. Cette méthode repose sur une structure

de graphe de mots, et utilise les probabilités acoustiques ainsi que les probabilités linguistiques issues du modèle de langage des mots hypothèses contenus dans ce graphe.

Une partie de nos travaux étant fondée sur la méthode proposée par Wessel et al. pour estimer la probabilité *a posteriori* d'un mot, nous décrivons celle-ci plus en détail.

Méthode de calcul proposée par F. Wessel

Pour décrire cette mesure, nous devons introduire quelques notations, soient :
- $[w, \tau, t]$ un mot hypothèse commençant à l'instant τ et se terminant à l'instant t,
- o_τ^t la séquence d'observations du temps τ au temps t.

Une phrase sera définie comme commençant à l'instant 1 et se terminant à l'instant T. Nous définissons ainsi :
- $[w, \tau, t]$ comme le mot hypothèse dont nous voulons estimer la probabilité *a posteriori*,
- $[w, \tau, t]_1^M$ une séquence de M mots $[w_i, \tau_i, t_i]$ telle que $\tau_1 = 1$, $t_M = T$ et $t_{i-1} = \tau_i - 1$, pour $i = 2, \ldots, M$.
- $C([w, \tau, t])$ la mesure de confiance du mot hypothèse $[w, \tau, t]$.

Soit $p([w, \tau, t]_1^M | o_1^T)$ la probabilité *a posteriori* d'une séquence de M mots sachant o_1^T, les observations acoustiques correspondantes. La probabilité *a posteriori* du mot hypothèse $[w, \tau, t]$, notée $p([w, \tau, t] | o_1^T)$, est égale à la somme des probabilités *a posteriori* de toutes les phrases hypothèses contenant le mot $[w, \tau, t]$. La méthode décrite par F. Wessel dérive de l'algorithme *forward-backward* de Baum et Welch présenté section 1.4.2.1 mais appliqué au niveau du mot. Il définit donc pour un mot $[w, \tau, t]$ du graphe une probabilité *forward*, $\Phi([w, \tau, t])$, et une probabilité *backward*, $\Psi([w, \tau, t])$.

Toutefois, les scores acoustiques et linguistiques impliqués dans ces calculs ne varient pas dans les mêmes ordres de grandeur. Ce phénomène peut induire de mauvaises performances de la mesure, dans laquelle le score acoustique serait dominant. Aussi, deux facteurs d'échelle ont été introduits : α pour le score acoustique et β pour le score du modèle de langage.

Les deux probabilités *forward* et *backward* peuvent être calculées de manière récursive. Nous exprimons les équations Eq. 2.12 et Eq. 2.13 représentant respectivement les définitions de ces deux probabilités dans le cadre de modèles de langage bigramme.

$$\Phi([w, \tau, t]) = p(o_\tau^t | w)^\alpha \sum_{w_p} \sum_{\tau'} \Phi([w_p, \tau', \tau - 1])\, p(w | w_p)^\beta \tag{2.12}$$

$$\Psi([w, \tau, t]) = p(o_\tau^t | w)^\alpha \sum_{w_s} \sum_{t'} \Psi([w_s, t + 1, t'])\, p(w_s | w)^\beta \tag{2.13}$$

Dans l'équation 2.12, $[w_p, \tau', \tau - 1]$ représente tout mot du graphe qui précède $[w, \tau, t]$ et qui finit donc à l'instant $\tau - 1$. Dans l'équation 2.13, $[w_s, t + 1, t']$ représente tout mot du graphe qui suit $[w, \tau, t]$ et qui débute donc à l'instant $t + 1$. Au final, la probabilité *a posteriori* du mot $[w, \tau, t]$, avec les définitions des probabilités *forward* et *backward*, est décrite par l'équation 2.14.

$$p(w|O) = p([w, \tau, t] | o_1^T) = \frac{\Phi([w, \tau, t])\Psi([w, \tau, t])}{p(o_1^T) p(o_\tau^t | w)^\alpha} \tag{2.14}$$

Le point crucial dans le calcul de la probabilité *a posteriori* est l'estimation de la quantité $P(O) = p(o_1^T)$, qui représente la probabilité de la séquence d'observations associée à la phrase.

Cependant, à partir des équations 2.12 et 2.13, cette quantité peut être estimée par l'équation suivante :

$$P(O) = p(o_1^T) = \sum_w \sum_\tau \Phi([w, \tau, T]) \tag{2.15}$$

F. Wessel définit finalement la mesure de confiance d'un mot hypothèse $[w, \tau, t]$ comme la probabilité *a posteriori* de $[w, \tau, t]$.

$$C([w, \tau, t]) = p([w, \tau, t]|o_1^T) \tag{2.16}$$

Cependant, cette mesure est calculée pour une hypothèse de mot w avec des instants de début et de fin précisément égaux à τ et t respectivement. Or dans le graphe de mots, un même mot w peut apparaître avec des positions temporelles légèrement différentes. Par conséquent, la probabilité *a posteriori* du mot est donc répartie entre ces différentes hypothèses. La solution proposée par F. Wessel consiste à sommer les probabilités des mêmes mots hypothèses selon des critères d'intersection. Plusieurs critères ont été testés pour un mot $[w, \tau, t]$ analysé. Les mots contributeurs peuvent donc être :

- tous les mots $[w, \tau', t']$ tels que l'intersection entre les deux mots $[w, \tau, t]$ et $[w, \tau', t']$ soit non vide,
- tous les mots $[w, \tau', t']$ tels que le temps médian $(\tau + t)/2$ appartienne à l'intervalle $[\tau', t']$,
- tous les mots $[w, \tau', t']$ tels que l'instant t_{max} appartienne à l'intervalle $[\tau', t']$, t_{max} étant défini comme le temps entre τ et t maximisant la quantité suivante :

$$\max_{t_m \in [\tau, t]} \sum_{[w, \tau', t']; \tau' \leq t_m \leq t'} p([w, \tau', t']|o_1^T)$$

2.3.6.3 Mesure de confiance du système de reconnaissance *Julius*

Lee et al. [Lee 04] proposent une autre méthode afin de calculer une mesure de confiance fondée sur une approximation de la probabilité *a posteriori* pendant la deuxième passe du système de reconnaissance *Julius*.

Le système de reconnaissance *Julius* fonctionne en deux passes. Une première passe moins précise, fondée sur l'algorithme de Viterbi, permet de créer un graphe de mots contenant un ensemble restreint d'hypothèses. La deuxième, plus précise, s'appuie sur cette structure de données afin de calculer la séquence solution contenue dans le signal de parole à l'aide de l'algorithme A^* (cf. section 1.6.2 p.16). A chaque mot du graphe est associé le score de vraisemblance du meilleur chemin partiel entre le début de la phrase et ce mot, la probabilité acoustique du mot, ainsi que sa probabilité linguistique avec le mot le précédant dans le chemin.

Le principe de base du calcul de la probabilité *a posteriori* d'un mot proposé reste le même que celui présenté section 2.3.6 et par l'équation 2.11 :

- calcul de la vraisemblance des phrases contenant un mot particulier à une position particulière,
- estimation de $P(O)$,
- estimation de la probabilité *a posteriori* d'un mot.

L'idée des auteurs est d'approximer toutes ces valeurs en n'utilisant que les données disponibles au cours de la deuxième passe qui est fondée sur un algorithme A^* commençant par la fin de la phrase.

Soient un mot hypothèse $[w, \tau, t]$ et $W_{[w,\tau,t]}$ l'ensemble des phrases qui contiennent le mot hypothèse $[w, \tau, t]$. Soit $g(w)$ la vraisemblance calculée par l'algorithme A^*, de la fin de la phrase jusqu'au mot w. Soit h la fonction heuristique du système pour la recherche de solution au cours de la deuxième passe. Pour un mot w', $h(w')$ est égal à la vraisemblance calculée lors de la première passe du chemin allant du début de la phrase jusqu'au mot w'. Lee et al. définissent alors pour un mot $[w_n, \tau_n, t_n]$ du graphe la fonction $f(w_n)$, qui est en fait une approximation de la vraisemblance du chemin complet passant par $[w_n, \tau_n, t_n]$ suivant l'équation suivante :

$$f(w_n) = g(w_n) \times h(w_{n-1}) \tag{2.17}$$

Dans cette équation, w_{n-1} représente le mot pouvant temporellement précéder w_n au moment de la recherche A^* et maximisant $f(w_n)$.

La deuxième approximation faite par les auteurs consiste à considérer que les phrases passant par le mot w_n sont les phrases passant exactement par $[w_n, \tau_n, t_n]$. Or, comme le mot hypothèse $[w_n, \tau_n, t_n]$ est unique dans le graphe de mots et étant donné l'approximation du calcul de la vraisemblance f d'une phrase passant par un mot w_n, il ne peut y avoir qu'une seule phrase.

La dernière quantité nécessaire afin de pouvoir estimer la probabilité *a posteriori* d'un mot est $P(O)$. La troisième approximation que font Lee et al. consiste à estimer que $P(O)$ peut être approximer par la somme des vraisemblances des phrases passant par un mot ayant temporellement une intersection non vide avec le mot w_n dont la confiance est estimée. Soit W_c l'ensemble des mots $[w', \tau', t']$ ayant une intersection non vide avec le mot $[w_n, \tau_n, t_n]$. $P(O)$ est alors donné par l'équation suivante :

$$P(O) = \sum_{[w',\tau',t'] \in W_c} f(w') \tag{2.18}$$

La probabilité *a posteriori*, et donc la valeur de confiance, du mot hypothèse $[w_n, \tau_n, t_n]$ est alors donnée par la formule suivante :

$$p(w_n|O) = \frac{f(w_n)}{\sum_{[w',\tau',t'] \in W_c} f(w')} \tag{2.19}$$

Lee et al. ont ainsi défini une mesure de confiance calculable au cours de la phase de décodage de la deuxième passe du moteur de reconnaissance. Ce calcul demande peu d'effort car il n'utilise que des quantités déjà calculées et nécessaires au processus de décodage. En revanche, la première passe doit être complètement effectuée, ce qui est impossible pour des applications en flux pour lesquelles le signal acoustique n'a potentiellement pas de fin. Par ailleurs, du fait des nombreuses approximations faites, le mesure de confiance ainsi définie est moins précise qu'une mesure de confiance également fondée sur la probabilité *a posteriori*, mais calculée par exemple avec la méthode de Wessel et al.

2.3.6.4 Mesures fondées sur les réseaux de confusion

Un réseau de confusion est un graphe de mots simplifié dans lequel les alternatives sont exprimées en position des mots dans la phrase (cf. section 1.6.3.3). Généralement le réseau de confusion est construit à partir d'un graphe de mots préalablement existant. L'objectif étant de

simplifier le graphe de mots en regroupant des hypothèses similaires en une seule. Ainsi moins d'hypothèses sont à traiter.

Les réseaux de confusion ont également été utilisés afin de calculer des mesures de confiance fondées sur la probabilité *a posteriori*. Toutefois, afin de calculer la probabilité *a posteriori* des mots du réseau de confusion, il est nécessaire de calculer ces probabilités sur le graphe de mots. Ensuite, la probabilité *a posteriori* d'un mot du réseau est égal à la somme des probabilités *a posteriori* des mots impliqués dans la *construction* du mot du réseau.

Cette probabilité peut être utilisée directement en tant que mesure de confiance, mais celle-ci tend à surestimer la probabilité *a posteriori* réelle des hypothèses [Mangu 00, Evermann 00, Falavigna 02]. De plus, la nécessité de calculer au préalable des probabilités *a posteriori* sur le graphe de mots, bien plus dense en hypothèses que le réseau de confusion, rend cette mesure moins attractive.

2.3.6.5 Récapitulatif des mesures fondées sur une estimation de la probabilité *a posteriori*

Dans cette section, nous avons présenté des mesures de confiance estimant la probabilité *a posteriori* d'un mot avec des méthodes différentes et plus ou moins d'approximations :

- une méthode fondée sur les n-meilleures phrases. Cette méthode est légère du point de vue calculatoire, mais elle nécessite la génération de la liste des n-meilleures phrases et donc le décodage intégral de la phrase. De plus, cette méthode est une approximation assez grossière de la probabilité *a posteriori*.
- une méthode fondée sur les graphes de mots et l'algorithme *forward-backward* de Baum et Welch, dont un algorithme de calcul a été décrit par Wessel et al. Moins d'approximations sont nécessaires et cette méthode de calcul est la plus précise à notre connaissance pour estimer la probabilité *a posteriori* d'un mot. En revanche, sa complexité de calcul est supérieure à celle de la méthode des n-meilleures phrases. De plus, cette méthode nécessite la génération du graphe de mots de l'intégralité de la phrase.
- une méthode fondée sur les graphes de mots avec des approximations, comme celle implantée dans le système de reconnaissance Julius. Cette méthode estime la probabilité *a posteriori* au cours de la deuxième passe du processus de décodage, sans introduire de nouvelles variables et avec un coût calculatoire faible, mais au prix de multiples approximations. La mesure de confiance obtenue est donc moins précise que la méthode décrite par Wessel et al., tout en nécessitant comme elle la génération préalable du graphe de mots.
- une méthode fondée sur les réseaux de confusion. Afin de calculer la probabilité *a posteriori* des mots de ce réseau, cette méthode nécessite quand même le calcul de la probabilité *a posteriori* des mots du graphe de mots d'origine, non simplifié. De plus, cette méthode a tendance à surestimer les valeurs de la probabilité *a posteriori* [Evermann 00].

2.3.7 Combinaison de mesures de confiance

2.3.7.1 Combinaisons de mesures et d'heuristiques

Les premiers travaux sur les mesures de confiance ont commencé par investiguer un large panel de critères heuristiques fondés par exemple sur la stabilité acoustique, le nombre de replis dans le modèle de langage, la longueur des mots en trames, du nombre de phonèmes, etc. Généralement, ces différents indices n'ont pas été utilisés seuls mais en combinaison dans un classifieur afin de prendre une décision. Parmi les critères combinés, nous retrouvons souvent une mesure plus

statistique comme le rapport de vraisemblance ou une estimation de la probabilité *a posteriori*. La plupart des indices utilisés dans des combinaisons ont été présentés au cours de ce chapitre.

L'utilisation d'une combinaison de critères pose deux problèmes majeurs :
- quels critères choisir ?
- comment, à partir de ces indices, prendre une décision d'acceptation ou de rejet ?

Afin d'obtenir une combinaison performante, il faut que les critères qui sont associés soient le moins corrélés entre eux. En effet, si un des critères apporte une information pertinente quand les autres indices sont indécis, alors le système peut être amélioré. Par contre, si les indices sont fortement corrélés entre eux, leur combinaison n'apportera aucune information supplémentaire. Dans la plupart des travaux proposant des combinaisons de critères, ces derniers sont souvent fortement corrélés [Jiang 05]. Cependant, des études mêlant des critères acoustiques tels que la probabilité *a posteriori* et des critères purement linguistiques concluent à une faible amélioration par rapport aux performances de la probabilité *a posteriori* seule [Zhang 01, Guo 04, Mauclair 06].

Le deuxième point clé concerne la méthode de combinaison des différents critères. Plusieurs méthodes ont été proposées :
- analyse linéaire discriminante [Sukkar 94, Sukkar 96, Schaaf 97, Pao 98],
- interpolation linéaire [Guo 04],
- modèle linéaire généralisé [Schaaf 97, Gillick 97, Siu 99, Kamppari 00, Duchateau 02a, Sun 03],
- classifieur à base de mélange de Gaussiennes [Chigier 92],
- utilisation d'un réseau de neurones [Schaaf 97, Weintraub 97, San-Segundo 01, Charlet 01, Stemmer 02],
- arbre de décision [Eide 95, Neti 97],
- machines à vecteur support [Zhang 01, BenAyed 03],
- méthode de *boosting* [Moreno 01]

2.3.7.2 Combinaison de systèmes de reconnaissance

Même si les systèmes de reconnaissance optent pour des stratégies proches, les résultats qu'ils produisent ne sont pas identiques. Il est alors possible de définir un *méta-système* se basant sur les résultats de divers systèmes de reconnaissance indépendants afin de donner une solution au moins aussi bonne que celles des systèmes pris indépendamment. Ce principe a été utilisé dans le système ROVER (Recognizer Output Voting Error Reduction) [Fiscus 97].

Dans ce système, le résultat est généré à l'issue d'un vote entre les solutions des différents systèmes de reconnaissance utilisés parallèlement pour le même document sonore. La décision se situe au niveau des mots, après une phase d'alignement entre les diverses solutions suivant la formule générale suivante :

$$Score(w) = \alpha F(w) + (1 - \alpha)C(w) \tag{2.20}$$

Pour un mot w, $F(w)$ représente le nombre de fois que w est dans la solution d'un des systèmes, normalisé par le nombre total de systèmes de reconnaissance. $C(w)$ correspond à la valeur de confiance du mot w suivant la méthode de vote employée.

Trois méthodes de calcul de $C(w)$ ont été introduites et sont :

1. pas de mesure de confiance ($\alpha = 1$), seule la fréquence d'occurrence des mots dans les résultats des systèmes est prise en compte (vote majoritaire),

2. $C(w)$ est la moyenne des scores de confiance des différents systèmes,
3. $C(w)$ est le score de confiance maximal parmi les systèmes.

Suivant la méthode de vote choisie, le système ROVER choisit le mot w ayant le score maximal comme faisant partie de la phrase solution.

Un système ROVER intégrant les résultats des systèmes BBN [Kubala 98], CMU, CU, DRAGON [Wegmann 98] et SRI [Sankar 98] a été testé sur les données de la campagne d'évaluation « LVCSR '97 Hub 5E Benchark Test » et obtient de meilleures performances que le meilleur de ces systèmes en terme de taux d'erreur en mots : 39,4% pour le système ROVER avec la méthode de vote n°3 contre 44,9% pour le système BBN [Fiscus 97]. Toutefois, la mesure de confiance en elle-même apporte peu car le système ROVER fondée uniquement sur un vote majoritaire (méthode n°1) obtient un taux d'erreur de 39,7%.

Cependant, le système ROVER est suffisamment flexible pour permettre d'ajouter des connaissances externes afin d'améliorer le principe du système. Cela peut être par l'utilisation de données linguistiques ou bien en fondant la phase de sélection sur un arbre de décision [Schwenck 97, Estève 02].

2.4 Méthodes d'évaluation

Les différents travaux de recherche que nous avons décrits dans les pages précédentes ont utilisé des méthodes différentes pour évaluer leurs mesures de confiance.

Les performances d'un système de reconnaissance sont habituellement évaluées en taux d'erreur en mots. Déterminer ce taux implique un alignement de ce résultat avec les données de référence de la transcription manuelle du document sonore. De cet alignement quatre quantités sont calculées :

- le nombre de mots bien reconnus dont la position dans la phrase est correcte,
- le nombre d'omissions : les mots corrects que le système a omis,
- le nombre de substitutions : les mots mal reconnus,
- le nombre d'insertions : les mots que le système a ajoutés par rapport à la référence.

La première quantité définie les mots *corrects*, les deux dernières définissent les mots *incorrects*.

Nous allons présenter ici les méthodes d'évaluation les plus communément employées. Pour toutes ces méthodes d'évaluation, il est nécessaire de disposer de la classification d'un mot reconnu dans une des ces deux catégories : *correct* et *incorrect*.

Par ailleurs, la plupart des méthodes d'évaluation des mesures de confiance sont également fondées sur la classification des mots résultats de la reconnaissance en *acceptations* et *rejets*. Pour cela, l'utilisation d'un seuil de décision permet d'étiqueter chaque mot suivant ces deux catégories. Un mot est étiqueté *Acceptation* s'il est considéré comme juste par la mesure (valeur supérieure au seuil de décision). Il est étiqueté *Rejet* si, au contraire, il est considéré comme faux.

2.4.1 Taux d'égale erreur

Cette méthode d'évaluation consiste à comptabiliser le nombre de fois où la mesure a donné une mauvaise indication. Pour être plus précis, sont comptabilisés : les mots mal reconnus par le moteur mais considérés à tort comme bons par la mesure (fausse acceptation), et les mots bien reconnus par le moteur mais considérés comme faux par la mesure (faux rejet).

$$FA = \frac{Nb.\ de\ mots\ incorrects\ étiquetés\ Acceptation}{Nb.\ de\ mots\ incorrects} \tag{2.21}$$

$$FR = \frac{Nb.\ de\ mots\ corrects\ étiquetés\ Rejet}{Nb.\ de\ mots\ corrects} \tag{2.22}$$

Les deux quantités ainsi définies sont à mettre en relation avec la méthode des tests d'hypothèses (voir section 2.3.5.1 p.38). Nous retrouvons l'expression des erreurs de première et deuxième espèce.

Deux courbes permettent habituellement de représenter la variation de ces erreurs en fonction du seuil de décision de la mesure de confiance. Ces deux courbes sont :

- la courbe *ROC* (Receiver operating characteristic)
- la courbe *DET* (Detection Error Tradeoff)

La courbe ROC se caractèrise par la représentation du taux de fausses acceptations et du taux de vraies acceptations en fonction du seuil de décision. L'échelle utilisé est linéaire sur chaque axe.

La courbe DET représente quant-à elle le taux de fausses acceptations et le taux de faux rejets en fonction du seuil de décision. En revanche, l'échelle employée sur chaque axe est généralement log-normale (déviation par rapport à la loi normale) [Martin 97].

Toutefois, afin de pouvoir nous comparer avec la mesure de référence fondée sur la probabilité *a posteriori* calculée par la méthode décrite dans [Wessel 01], nous adoptons la même courbe pour nos évaluations. Dans leurs travaux, les auteurs définissent une courbe dite *DET* intermédiaire entre les vraies courbes ROC et DET. Afin de différencier cette courbe des deux autres, nous l'appelons *ROC-DET* car la courbe définie dans leur travaux est une courbe DET à échelle linéaire (comme une courbe ROC).

Ainsi, à partir de deux taux de fausses acceptations et de faux rejets, en faisant varier le seuil de décision, nous pouvons représenter une courbe *ROC-DET*. Cette courbe exprime les taux de fausses acceptations et de faux rejets en fonction du seuil. La figure 2.2 donne un exemple d'une courbe ROC-DET.

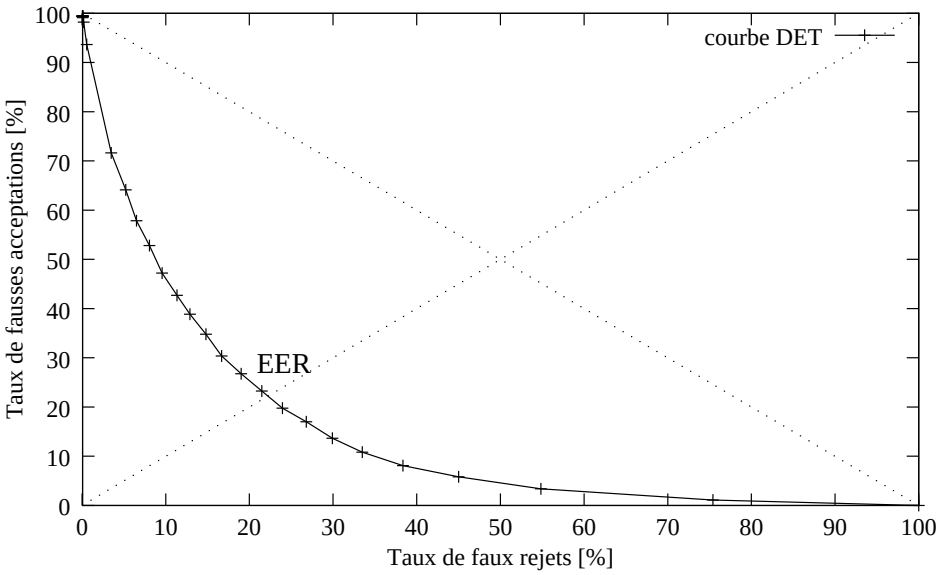


Fig. 2.2 – Exemple d'une courbe ROC-DET. L'intersection entre la première bissectrice et la courbe détermine le point EER.

La droite d'équation $y = 1 - x$ représente la courbe obtenue par une sélection aléatoire des mots à accepter ou à rejeter. Plus la courbe est proche des axes et plus la mesure de confiance est pertinente. L'intersection entre la première bissectrice et la courbe ROC-DET définit un point remarquable : la valeur du seuil pour laquelle les taux de fausses acceptations et de faux rejets sont égaux (Equal Error Rate – EER). Plus la valeur EER est faible, meilleure est la mesure. Le taux EER a été utilisé en tant que méthode d'évaluation des performances de mesures de confiance dans plusieurs travaux [Uhrik 97, Jitsuhiro 98, Siu 99].

La courbe ROC-DET présente l'avantage de contenir l'ensemble des points de fonctionnement pour une mesure de confiance. Il est en effet facile de déterminer par exemple pour un taux de fausses acceptations maximum fixé, le taux de faux rejets qui sera observé.

Une seconde courbe peut également être représentée : la courbe *ROC* (Receiver operating characteristic) [Egan 75]. Cette courbe, contrairement à la courbe ROC-DET, exprime le taux de fausses acceptations et de vraies acceptations en fonction du seuil de décision. L'analyse des acceptations est privilégiée à travers cette courbe.

La méthode d'évaluation fondée sur le taux d'égale erreur permet une analyse des performances d'une mesure de confiance de façon indépendante de toute application. Cette méthode ne cherche pas à minimiser plus spécifiquement le taux de fausses acceptations ou le taux de faux rejets. En effet, un seul point de fonctionnement est défini : le taux EER. Ceci peut représenter une limitation dans les cas où l'application visée exigerait un point de fonctionnement différent qui favoriserait le taux de fausses acceptations ou le taux de faux rejets.

2.4.2 Taux d'erreur de confiance

Le taux d'erreur de confiance (Confidence Error Rate – CER) représente de façon simple et intuitive la précision d'une mesure de confiance. Cette méthode a été utilisée dans plusieurs études [Kemp 97, Weintraub 97, Wessel 01, Mauclair 06]. Elle est définie par le rapport entre le nombre de mots incorrectement étiquetés par la mesure de confiance et le nombre total de mots reconnus :

$$CER = \frac{Nb.\ de\ Fausses\ Acceptations + nb.\ de\ Faux\ Rejets}{Nb.\ de\ mots\ reconnus} \tag{2.23}$$

Un exemple est donné en annexe A.2 page 146.

Afin de pouvoir comparer le gain d'une mesure de confiance par rapport au système de reconnaissance, le CER du système est calculé par l'équation 2.24 :

$$CER_{référence} = \frac{Nb.\ d'insertions + nb.\ de\ substitutions}{Nb.\ de\ mots\ reconnus} \tag{2.24}$$

Ceci revient à utiliser une mesure étiquetant *Acceptation* tous les mots de la phrase reconnue par le système. Une mesure de confiance qui apporte de l'information utile devrait permettre de diminuer les taux de fausses acceptations et de faux rejets. Ainsi, de l'équation 2.23, on déduit logiquement que plus une mesure de confiance sera précise et plus la valeur du CER associée sera proche de zéro. Un exemple de calcul de ce taux d'erreur sur une phrase est donné en annexe A.2 page 146.

2.4.3 Entropie croisée normalisée

La plupart des méthodes classiques d'évaluation analysent la performance des mesures de confiance par la comparaison de l'évolution du taux d'erreur en mots ou du taux d'EER. La méthode fondée sur l'entropie croisée normalisée (Normalized Cross Entropy – NCE) tente d'éva-

luer l'apport d'information que fournit une mesure de confiance par rapport au résultat du moteur de reconnaissance. Cette méthode d'évaluation a été employée dans plusieurs études [Siu 97, Kemp 97, Rueber 97, Evermann 00, Maison 01, Duchateau 02a]. Sa version non normalisée a également été utilisée dans d'autres travaux [Chase 97, Gillick 97, Weintraub 97].

Le principe est de comparer l'entropie du système à l'issue du processus de reconnaissance à l'entropie du même système mais dont les mots ont été classés en tenant compte de leur valeur de confiance. Cette méthode a été introduite lors d'une campagne d'évaluation du *NIST* de 1996 dans leur logiciel d'évaluation statistique de reconnaissance [Siu 97, Kemp 97].

Soit p_0 le taux de reconnaissance en mots du système défini comme le rapport entre le nombre de mots corrects et le nombre total de mots. L'entropie de référence $H(S)$ du système est alors définie par l'équation suivante :

$$H(S) = -p_0 \log p_0 - (1 - p_0) \log(1 - p_0) \tag{2.25}$$

Si X représente toutes les informations supplémentaires apportées au système de reconnaissance initial, l'entropie conditionnelle $H(S|X)$ peut être calculée ainsi :

$$H(S|X) = \frac{-1}{N} \Big(\sum_{w\ correct} \log pc_w \ + \sum_{w\ incorrect} \log(1 - pc_w) \Big) \tag{2.26}$$

Où N représente le nombre de mots de la phrase reconnue et pc_w la mesure de confiance associée au mot w.

La valeur NCE est alors définie par :

$$NCE = \frac{H(S) - H(S|X)}{H(S)} \tag{2.27}$$

Ainsi, si la mesure de confiance est parfaite (les mots justes ont une confiance de 1 et les mots faux ont une valeur de confiance de 0), d'après l'équation 2.26, $H(S|X) = 0$ et donc la valeur NCE associée vaut 1.

De même, si la mesure de confiance est uniformément aléatoire, $H(S|X)$ est équivalente à l'entropie du système *a priori* $H(S)$ et la valeur NCE est alors nulle.

Cependant, il est possible que la valeur NCE soit négative si l'apport d'information est trop fortement erroné. On pourra se reporter à un exemple concret de calcul de la valeur NCE en annexe A.1 page 145.

Un problème persiste dans l'utilisation de la méthode NCE. En effet, dans le cas où un mot correct a une valeur de confiance proche de 0 — réciproquement un mot faux a une valeur de confiance proche de 1 — l'équation 2.26 indique que l'entropie tend vers l'infini. Une solution consiste à seuiller les valeurs de confiance afin que celles-ci ne s'approchent pas des valeurs critiques 0 et 1 mais restent par exemple dans l'intervalle [0,1 – 0,9].

2.4.4 Coefficient de corrélation

Une autre façon de procéder pour observer la pertinence d'une mesure de confiance est d'analyser simplement l'existence d'une corrélation entre les valeurs de confiance associées aux mots reconnus et la justesse réelle de ces mots.

Pour cela nous définissons deux ensembles de données X et Y. X représente l'ensemble des valeurs de la mesure de confiance pour les mots reconnus par le système, et Y est un ensemble de

valeurs indiquant si le mot reconnu est juste ou faux relativement à la transcription de référence. L'ensemble Y est à valeur discrète dans $\{0, 1\}$.

Si les écarts-types σ_X et σ_Y sont définis et non nuls (il faut dans notre cas qu'il y ait au moins deux valeurs distinctes dans chacun des ensembles) et si les moyennes μ_X et μ_Y de ces ensembles sont également définies, alors le coefficient de corrélation $\rho_{X,Y}$ est défini par l'équation :

$$\rho_{X,Y} = \frac{E(XY) - E(X)E(Y)}{\sqrt{E(X^2) - E^2(X)}\sqrt{E(Y^2) - E^2(Y)}} \tag{2.28}$$

Où E représente l'espérance mathématique.

Une mesure de confiance idéale aura un coefficient de corrélation égal à 1. Par contre, dans le pire des cas où la mesure estime exactement l'inverse des labels réels des mots, le coefficient sera égal à -1. En effet, bien que faux, les valeurs de confiance sont corrélées avec les mots. Si une mesure accepte tous les mots, le coefficient de corrélation est nul.

En dehors des valeurs proches des extrêmes 0, 1 ou -1, il est difficile de qualifier le degré de corrélation à partir de la valeur du coefficient. Un coefficient de 0,9 peut parfois exprimer une faible corrélation à cause, notamment, de la grande sensibilité du coefficient de corrélation aux valeurs aberrantes.

2.4.5 Rappel / Précision

Dans le domaine de la recherche d'information (Information Retreival), les méthodes évaluant la pertinence habituellement utilisées sont définies par rapport aux notions de *Rappel* et de *Précision*. Ces deux notions permettent de rendre compte de l'efficacité de l'amélioration apportée au système.

Les méthodes de *rappel/précision* ont également été introduites dans le domaine des mesures de confiance en reconnaissance automatique de la parole [Cox 02]. La mesure de confiance permet d'étiqueter les mots d'une phrase en *Acceptation* et *Rejet*, comme dans le cas de l'analyse du taux d'égale erreur (EER). Les valeurs de *rappel/précision* sont définies en fonction de cet étiquetage et de la connaissance de la séquence de mots réellement prononcée. Les équations de *rappel* et de *précision* pour l'analyse des *Acceptations* s'expriment ainsi :

$$Rappel = \frac{Nb.\ de\ mots\ correctement\ étiquetés\ Acceptation}{Nb.\ total\ de\ mots\ corrects} \tag{2.29}$$

$$Précision = \frac{Nb.\ de\ mots\ correctement\ étiquetés\ Acceptation}{Nb.\ de\ mots\ étiquetés\ Acceptation} \tag{2.30}$$

La valeur de *rappel* seule n'est pas pertinente. Par exemple, une mesure de confiance qui accepte tous les mots de la phrase, et qui sera par conséquent inutile, aura une valeur de *rappel* égale à 100%. Ainsi, la valeur de *rappel* va généralement de pair avec la valeur de *précision*.

Une fonction globale définie par la moyenne harmonique de la valeur de *rappel* et de *précision* permet de regrouper ces deux indices en un seul [Van Rijsbergen 79]. Cette mesure est appelée $F-$mesure. Son équation est la suivante :

$$F_\alpha = \frac{(1+\alpha)(précision.rappel)}{\alpha.\ précision + rappel} \tag{2.31}$$

Les $F-$mesures habituellement calculées sont F_1, F_2, et $F_{0,5}$.

Les informations issues des valeurs de rappel et de précision sont ainsi représentées par un seul indice, la $F-$mesure. La comparaison des performances des différentes mesures de confiance devient alors plus directe, comme avec le taux d'égale erreur.

2.4.6 Synthèse

Nous avons présenté plusieurs méthodes d'évaluation de mesures de confiance utilisées dans la littérature. L'objectif de chacune d'elle est le même : comparer et évaluer les mesures de confiance entre elles et également par rapport au système de reconnaissance. Toutefois, chaque méthode a ses caractèristiques propres, ses avantages et ses inconvénients et tente de capturer des aspects plus ou moins liés à une application particulière. Par exemple, le taux d'erreur de confiance CER est facile à évaluer de par sa simplicité et permet de comparer les mesures de confiance par rapport à seule une valeur. De plus ce taux est également comparable avec le taux de mots corrects du système de reconnaissance. Mais un des inconvénients de cette méthode vient justement du fait qu'il n'est pas possible de distinguer les deux types d'erreurs, fausses acceptations et faux rejets, et ainsi définir différents points de fonctionnement.

La méthode proposée par le NIST fondée sur le calcul de l'entropie croisée normalisée tente d'évaluer l'apport d'information d'une mesure de confiance par rapport au résultat de la reconnaissance. Cette méthode, comme le CER, détermine une unique valeur, sans pouvoir définir de points de fonctionnement. De plus, cette méthode pose des problèmes pour des mots dont la confiance est nulle ou sûre et peut être à valeur négative.

La méthode consistant à calculer les taux de rappel et de précision est généralement utilisée en recherche d'information. L'association des deux taux permet de définir plusieurs points de fonctionnement selon le seuil de décision choisi pour la mesure. Une seconde méthode permettant également de définir divers points de fonctionnement consiste à calculer les taux de fausses acceptations et de faux rejets. Toutefois pour cette méthode, un point particulier peut être facilement mis en évidence, le point EER pour lequel la mesure de confiance fait proportionnellement autant de fausses acceptations que de faux rejets. C'est pour ces raisons que pour l'évaluation de nos mesures de confiance sur le corpus de développement nous comparons nos mesures selon leur taux d'EER : les mesures peuvent se comparer par rapport à une valeur précise et plusieurs points de fonctionnement peuvent facilement être définis.

2.5 Quelques résultats

Il est difficile de comparer les performances des mesures de confiance entre elles car les méthodes d'évaluation et les corpus utilisés sont généralement différents, souvent en lien avec l'objectif des applications.

Toutefois, nous résumons à titre indicatif dans le tableau 2.1 quelques résultats obtenus par différentes mesures de confiance. Dans ce tableau, nous indiquons pour chaque mesure l'article y faisant référence, la base de données sur laquelle elle a été évaluée, le taux d'erreur en mots *WER* du système de reconnaissance (sans mesure de confiance) ainsi que le taux d'erreur de confiance *CER*.

Le corpus SwitchBoard [Godfrey 92] est constitué d'une collection importante de conversations téléphoniques américaines. Les conversations ont été effectuées dans un environnement acoustique de type maison ou bureau. La durée de chaque conversation est de 5 minutes. Ce corpus fait partie d'une campagne d'évaluation américaine NIST pour la reconnaissance de locuteurs.

Le corpus *Broadcast news #1* est constitué d'émissions télévisées et radiophoniques en anglais américain enregistrées en 1996. Les émissions comportent six environnements d'enregistrement différents, couvrant de la parole propre à de la parole très fortement dégradée.

Une des raisons pour lesquelles il est difficile de comparer les résultats des mesures entre-elles est illustrée ici par le fait que le corpus *Broadcast news #2* n'est pas clairement défini. L'article

TAB. 2.1 – Résultats obtenus par différentes mesures de confiance sur différents corpus.

Mesure	Référence(s)	Base de données	WER	CER
Probabilité *a posteriori*	[Weintraub 97] [Wessel 01]	SwitchBoard Broadcast news #1	48,9% 27,7%	31,4% 20,6%
Rapport de vraisemblance	[Cox 96]	SwitchBoard	49,8%	40,0%
Score du Modèle de langage	[Weintraub 97]	SwitchBoard	48,9%	46,2%
Dépendance des mots	[Bansal 98]	Broadcast news #2	27,0%	21,8%
Probabilité *a posteriori*	[Mauclair 06]	ESTER	23,7%	18,6%

de Bansal et al. ne décrit pas plus en détail la constitution de ce corpus de plusieurs heures d'enregistrements d'émissions. Ainsi, on ne peut comparer ces résultats avec ceux de la mesure fondée sur la probabilité *a posteriori* décrite par Wessel et al. et évaluée également sur un corpus d'émissions.

Le corpus *ESTER* est un corpus de bulletins d'émissions radiophoniques francophones issu de la campagne d'évaluation du même nom. Ici encore, bien que le corpus soit des émissions radiophoniques, les résultats ne sont pas comparables avec les autres travaux du tableau 2.1 car ce corpus est en langue française alors que les autres sont en anglais.

2.6 Conclusion

Dans ce chapitre, nous avons montré qu'une mesure de confiance peut être utile quelle que soit l'application finale visée : transcription, sous-titrage, détection de mots clés, apprentissage semi-supervisé, etc.

Cependant, le choix de la mesure elle-même n'est pas aisée.

Des études présentées dans ce chapitre, nous pouvons remarquer que les critères empiriques ainsi que les critères linguistiques sont certes sources de connaissance, mais sont relativement peu efficaces pris isolément. Bien que ces critères empiriques ou linguistiques puissent être souvent calculés sans attendre la reconnaissance complète du signal de parole, leur performance sont surpassées par les mesures fondées sur des critères statistiques, en particulier la probabilité *a posteriori*. C'est pour cela que généralement, ces types de critères sont combinés avec d'autres critères empiriques ou linguistique mais essentiellement avec des critères plus statistiques tels que le rapport de vraisemblance ou la probabilité *a posteriori*.

Concernant les mesures fondées sur le rapport de vraisemblance, dans la plupart des cas, l'objectif est la détection de mots clés ou l'acceptation/rejet de mots ou phrases reconnues, le tout dans un contexte de petit ou moyen vocabulaire (5 000 ou 20 000 mots maximum). Les corpus traités sont habituellement des dialogues utilisant le téléphone comme support de communication. Quelques mesures présentées fondées sur un rapport de vraisemblance peuvent être calculées sans attendre la fin du processus de reconaissance, mais le vocabulaire limité habituellement utilisé est incompatible avec le cadre grand vocabulaire que nous nous sommes fixé.

De ces études ressort la prédominance de la probabilité *a posteriori* en tant que mesure de confiance. Celle-ci est théoriquement un très bon indice, ce qui est confirmé dans la pratique, même avec des systèmes grand vocabulaire. Toutefois, à moins de calculer une approximation

imprécise de cette probabilité, la méthode à employer pour en calculer une bonne estimation est assez complexe (méthode *forward-backward*) et exige la reconnaissance de l'intégralité de la phrase. Même pour les méthodes faisant de fortes approximations, comme par exemple la mesure du système de reconnaissance Julius, il est nécessaire d'avoir effectué la reconnaissance complète de la phrase.

Or, comme nous le verrons dans le prochain chapitre, ces mesures ne peuvent pas être utilisées dans les applications que nous avons envisagées. Aussi nous a-t-il été nécessaire de définir de nouvelles mesures de confiance en adéquation avec nos besoins.

Chapitre 3

Propositions de nouvelles mesures de confiance

Sommaire

3.1 Objectifs 58
3.1.1 Applications visées 58
3.1.1.1 Transcription d'émissions 58
3.1.1.2 Transcription de cours en salle de classe 59
3.1.1.3 Détection de mots clés 61
3.1.2 Nos mesures de confiance : dans quel but ? comment ? 61
3.1.2.1 Caractéristiques principales de nos mesures de confiance 61
3.1.2.2 Quels types de mesures de confiance ? 62
3.1.2.3 Source d'information pour calculer les mesures 62
3.1.2.4 Mesures de confiance à quel niveau ? 63
3.2 Mesures trame-synchrones 63
3.2.1 Définition des mots concurrents de l'ensemble E 64
3.2.2 Gestion des occurrences multiples 64
3.2.3 Mesure fondée sur la probabilité unigramme 65
3.2.4 Introduction de facteurs d'échelle 66
3.2.5 Mesure fondée sur la probabilité bigramme 66
3.2.6 Mesure fondée sur la probabilité trigramme 67
3.2.7 Implantation 69
3.2.7.1 Construction de l'ensemble $\widehat{E}$ 69
3.2.7.2 Calcul des mesures fondées sur les probabilités unigramme, bigramme et trigramme 69
3.3 Mesures locales 71
3.3.1 Mesures fondées sur la probabilité *a posteriori* 72
3.3.2 Définition des voisinages 73
3.3.3 Introduction d'un facteur de flexibilité η 73
3.4 Homogénéisation de la répartition des valeurs de confiance 74
3.5 Complexité de nos mesures de confiance 77
3.5.1 Mesures trame-synchrones 77
3.5.2 Mesures locales 77
3.6 Conclusion 78

3.1 Objectifs

Pour les études menées au cours de cette thèse, nous nous sommes placés dans le cadre de la reconnaissance automatique de la parole pour des systèmes grand vocabulaire.

Nous présentons plus en détail dans ce chapitre les applications que nous avons considérées ainsi que la manière dont la mesure de confiance peut être utilisée. Puis nous présentons les caractéristiques et les types de mesures de confiance que nous avons choisis et pourquoi. Ensuite, nous décrivons en premier lieu nos mesures trame-synchrones puis nos mesures locales.

3.1.1 Applications visées

3.1.1.1 Transcription d'émissions

La transcription d'émissions, qu'elles soient télévisées ou radiophoniques, permet principalement aux malentendants d'accéder au contenu des émissions. Les sous-titrages pour sourds et malentendants sont apparus à la télévision dès 1983 sur Antenne 2, puis dès 1991 pour les journaux de 13h et de 20h de la même chaîne. Mais la proportion de programmes ou d'émissions sous-titrées restait faible. Cependant, selon une loi sur le handicap datant de 2005, toutes les chaînes télévisées qui réaliseront plus de 2,5% d'audience en 2010 devront intégralement sous-titrer leurs programmes. Cet objectif reste ambitieux sachant qu'actuellement un peu plus de 50% des programmes de France Télévisions sont sous-titrés, 49% pour ceux de TF1 (valeur doublée en un an). Le principal obstacle est d'ordre économique : le prix d'un sous-titrage en direct varie de 25 à 40 euros la minute. Selon la complexité, par exemple un journal télévisé, une dizaine de personnes peuvent être employées pour la transcription. Plusieurs méthodes sont adoptées afin de réaliser les transcriptions des émissions en direct, par exemple :

- la sténographie ou une variante améliorée, la vélotypie, utilisée par exemple pour retranscrire les débats de l'assemblée nationale.
- l'utilisation d'un système de reconnaissance de la parole.

Les systèmes de reconnaissance automatique de la parole actuels permettent d'obtenir des taux de reconnaissance satisfaisants et sont donc de plus en plus employés. Une personne dite « perroquet » écoute l'émission en direct et répète dans un microphone, parfois en reformulant ou en synthétisant, ce qu'elle entend. Le microphone est relié à un système de reconnaissance automatique de la parole dont les modèles ont été adaptés au préalable aux caractéristiques vocales de la personne *perroquet*; ceci afin d'optimiser les performances du système pour ce *perroquet*. Généralement, les sous-titres s'affichent avec un délai de 2 à 4 secondes par rapport à la voix en direct. Ce décalage entre les sous-titres et la voix n'est pas un problème réel car la plupart des émissions s'imposent ou se voient imposer un décalage de l'ordre de plusieurs secondes entre le direct réel et la diffusion. Ce décalage permet par exemple de stopper la diffusion avant que tout incident majeur ne passe à l'antenne.

Dans le contexte des émissions radiophoniques en direct, nous voulons définir des mesures de confiance afin de pouvoir influer de deux façons possibles sur la réponse du système de reconnaissance :

- en intégrant la mesure de confiance dans le processus de décodage afin d'améliorer la reconnaissance en modifiant la vraisemblance des hypothèses,
- en signalant les mots ayant une faible confiance, par exemple en les mettant en couleur.

3.1.1.2 Transcription de cours en salle de classe

Une seconde application de transcription à laquelle nous allons nous intéresser consiste à transcrire pour les élèves malentendants d'une classe le contenu du cours dispensé par un professeur. Sans aide extérieure les élèves sourds ou malentendants ne peuvent suivre le cours qu'en effectuant une lecture labiale des phrases prononcées par l'enseignant. Or, dans une salle de classe, l'enseignant parle à l'ensemble de la classe, il peut être plus ou moins éloigné des élèves, il se déplace et se retourne pour écrire au tableau. De plus, la lecture labiale est une technique difficile à acquérir et possède certaines ambiguïtés. Ainsi, l'élève doit faire face à des vides dans le cours pendant lesquels il n'accède qu'à des informations très partielles voire inexistantes.

La solution habituellement pratiquée est similaire à la méthode du *perroquet* pour les émissions en direct : une personne, codeuse en langue des signes ou en Langage Parlé Complété (LPC), écoute le cours et le répète ou le synthétise en articulant les mots et en ajoutant le codage associé. Les élèves ne pouvant suivre le professeur (difficulté pour ne lire que sur les lèvres, position du professeur incompatible avec la lecture labiale) peuvent se reporter sur la personne codeuse pour suivre le cours.

Le Langage Parlé Complété a été introduit afin de lever les ambiguïtés de formes de bouche rencontrées lors de la lecture labiale. En effet, la production de phonèmes différents peut conduire à une forme de bouche identique (ouverture de la mâchoire, protusion des lèvres, partie visible de la langue). C'est le cas notamment des phonèmes ne se différenciant que par le voisement comme par exemple *b* et *p*. Un langage a alors été introduit afin de lever ces ambiguïtés de forme des lèvres : le Langage Parlé Complété. Le langage LPC[1] consiste à compléter le mouvement des lèvres à l'aide de la position d'une main par rapport au visage et de la configuration de ses doigts. Le LPC utilisé conjointement à la lecture labiale permet de *lire* les mots comme s'ils étaient écrits. Les figures 3.1 et 3.2 montrent le codage en LPC relatif aux positions de la main ainsi qu'à la configuration des doigts.

Fig. 3.1 – Les 5 positions de la main pour le codage des voyelles phonétiques en Langage Parlé Complété.

La mise en place de la solution consistant à utiliser une personne codeuse entre les élèves et le professeur fait face à plusieurs difficultés : le manque de personnes codeuses et leur coût.

Le projet LABIAO[2] (lecture LABIale Assistée par Ordinateur) a pour objectif de concevoir, développer et distribuer un ensemble de logiciels permettant aux sourds et malentendants d'être plus autonomes par exemple à l'école, dans leur travail et dans la vie de tous les jours. Notre équipe a participé à ce projet afin de proposer aux élèves sourds et malentendants deux modalités pouvant les aider : l'affichage d'une tête parlante artificielle intégrant le codage LPC et

[1] http ://www.alpc.asso.fr/
[2] Projet RIAM

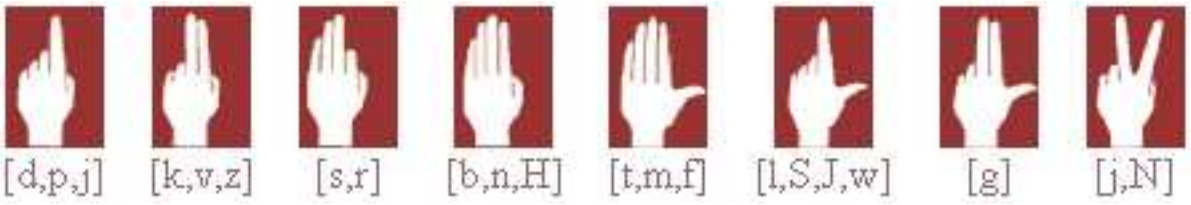


FIG. 3.2 – Les 8 configurations de doigts pour le codage des consonnes phonétiques en Langage Parlé Complété.

l'affichage d'une transcription temporisée. La tête codeuse est un visage de synthèse intégrant les mouvements des lèvres, de la langue et du menton. En plus du visage, une main artificielle vient se positionner près du visage tout en présentant une configuration des doigts selon le codage LPC. La transcription temporisée correspond à l'affichage sur un écran de la transcription du cours synchronisée avec le débit du locuteur. Les mots s'affichent séquentiellement avec une temporisation entre deux mots successifs dépendant de la durée des mots reconnus.

FIG. 3.3 – Tête codeuse de synthèse développée au Loria pour le projet LABIAO (le son « pa » en LPC).

La tête codeuse et la transcription temporisée sont pilotées par le résultat d'un système de reconnaissance. Dans un premier temps, le résultat brut de la reconnaissance a été utilisé pour des évaluations de compréhension et d'appréciation des sujets sourds pour les deux modalités, incluant les erreurs du système [Mourot 07]. Quelle que soit la modalité, la substitution d'un mot à reconnaître par un autre mot incorrect peut rendre une phrase incompréhensible, perturber le lecteur et lui faire perdre sa concentration. Nous proposons alors d'introduire des mesures de confiance afin d'indiquer, par le biais de nouvelles modalités, la confiance à avoir dans les mots fournis par le système de reconnaissance.

Plusieurs solutions sont envisagées, par exemple : dans le cas de la tête parlante et en fonction de la valeur de confiance :

- faire varier la transparence de la main codeuse,
- faire varier la couleur de la tête codeuse ;

dans le cas de la transcription temporisée :

- mettre en couleur (rouge) les mots ayant une faible confiance et laisser en couleur normale (noir) les autres,
- donner en couleur la transcription dans un langage phonétique simplifié des mots ayant une faible confiance.

Pour la dernière solution proposée, nous partons du principe qu'à partir d'un affichage phonétique simplifié des mots incorrects, il peut être plus facile de trouver le sens de la phrase que de lire un mot incorrect écrit en toutes lettres. Il est évident que cela demande un effort cognitif plus important, mais le sens de la phrase peut être plus facilement trouvé.

Un décalage temporel est introduit entre le discours de l'enseignant et le système d'assistance de l'élève, que ce soit un codeur humain, la tête codeuse artificielle ou la transcription temporisée. En effet, ce décalage est nécessaire aux élèves afin de pouvoir se synchroniser entre le professeur et le système qui fait office de *perroquet*. Ce décalage est pour le moment fixé à 5 secondes.

3.1.1.3 Détection de mots clés

Le second type d'application visée est la détection de mots clés. L'objectif est de pouvoir valider des mots clés détectés non pas dans un document sonore pré-enregistré mais dans un flux continu de parole. Cette validation doit se faire le plus tôt possible après la détection du mot clé. Le point commun avec l'application de transcription précédemment décrite est de pouvoir traiter des flux sonores continus. La contrainte intrinsèque réside dans le fait qu'il n'y a pas de fin au signal sonore. Le calcul de la mesure de confiance ne pourra donc se faire que de manière locale, c'est-à-dire avec une connaissance partielle du signal et des données générées par le système de reconnaissance. L'application peut également être de type « vérification à la demande » pour un mot dans le flux, sans nécessiter l'intégralité du signal mais juste la portion dans laquelle se trouve l'élément à vérifier.

3.1.2 Nos mesures de confiance : dans quel but ? comment ?

3.1.2.1 Caractéristiques principales de nos mesures de confiance

Si nous analysons nos besoins pour les trois applications choisies, le point important est la nécessité de pouvoir évaluer nos mesures de confiance sans avoir à attendre la fin de la phrase. Nous avons décidé de définir des mesures de confiance dites locales, c'est-à-dire qui n'ont besoin que d'une vision locale du signal pour être calculées, et non de toute la phrase. Par exemple, pour un mot w, le calcul de la mesure n'aura accès qu'aux données se trouvant dans un voisinage du mot w. Parmi ces mesures de confiance locales, certaines devront avoir une caractéristique supplémentaire forte : être trame-synchrones. Une mesure trame-synchrone est une mesure dont le calcul peut se faire en même temps que le décodage du flux sonore par le moteur de reconnaissance. Le moteur de reconnaissance procède trame par trame (les vecteurs d'observations) et génère une nouvelle meilleure hypothèse de phrase dès le traitement d'une trame terminé. Ainsi la reconnaissance se fait incrémentalement. Quand le moteur de reconnaissance a traité une trame, il génère un ensemble de nouveaux mots qui peuvent se terminer à cette trame. Ces nouveaux

mots sont intégrés dans le graphe. Dès ce moment, les mesures de confiance trame-synchrones peuvent être calculées.

3.1.2.2 Quels types de mesures de confiance ?

Hormis le fait de choisir des mesures trame-synchrones ou locales, nous avons décidé de définir nos mesures de confiance à partir :

- d'un rapport de vraisemblance obtenu à partir de données fournies par le moteur de reconnaissance et de données supplémentaires,
- d'une estimation de la probabilité *a posteriori* du mot analysé.

Comme nous l'avons vu au chapitre précédent, le calcul du rapport de vraisemblance entre deux hypothèses est souvent utilisé en tant que critère de décision dans des applications de détection de mots clés, soit en mots isolés soit en parole continue avec un vocabulaire limité, mais pas en grand vocabulaire.

Par ailleurs, les mesures fondées sur une estimation de la probabilité *a posteriori* obtiennent souvent les meilleures performances ou y contribuent grandement dans le cas de fusion de mesures [Jiang 05]. Plusieurs des méthodes que nous avons décrites dans le chapitre précédent consistaient en une estimation plus ou moins précise et donc plus ou moins coûteuse de ces probabilités *a posteriori*. Mais dans tous les cas ces méthodes nécessitaient la fin de la reconnaissance de la phrase (fin de la première passe pour la mesure de confiance de Julius [Lee 04], le graphe de mots ou la liste des n-meilleures phrases pour [Wessel 01] et [Weintraub 97]). Notre objectif a donc été de définir une estimation locale de ces probabilités *a posteriori* fondées sur les informations partielles générées par le moteur de reconnaissance à la trame t (fin du mot dont est calculée la mesure de confiance) ou un court délai après la fin du mot (voisinage local du mot).

3.1.2.3 Source d'information pour calculer les mesures

Afin de calculer nos mesures de confiance, nous devons définir les données sur lesquelles reposeront ces calculs. Nous avons présenté dans le premier chapitre (section 1.6.3) les différentes structures de données qui peuvent être extraites d'un système de reconnaissance de la parole : la liste des n-meilleures phrases, le graphe de mots et le réseau de confusion.

Le moteur de reconnaissance génère un graphe de mots interne au cours de la phase de décodage. Ce graphe est construit incrémentalement de manière trame-synchrone et n'est plus modifié par le processus de décodage au cours du traitement des trames suivantes.

La liste des n-meilleures phrases est généralement construite une fois la phase de décodage intégralement terminée. Mais il est possible de générer une liste des n-meilleures phrases partielles, uniquement valable jusqu'à l'observation traitée. Cependant, des travaux ont montré qu'il était plus judicieux d'utiliser un graphe de mots plutôt qu'une liste des n-meilleures phrases [Wessel 99].

Les réseaux de confusion ont été utilisés afin de calculer la probabilité *a posteriori* des mots. Bien que la simplicité de cette structure soit attirante, afin de calculer les probabilités *a posteriori* il est nécessaire au préalable de calculer ces probabilités sur le graphe de mots d'origine. De plus, les probabilités *a posteriori* obtenues surestiment les probabilités réelles.

Ces constatations nous ont conduit à fonder nos mesures de confiance sur les informations contenues dans le graphe de mots que génère le moteur de reconnaissance pendant la phase de décodage. Ces informations sont pour chaque mot du graphe : sa probabilité acoustique, son instant de début et de fin et son prédécesseur au sens de Viterbi sur le meilleur chemin menant à ce mot.

3.1.2.4 Mesures de confiance à quel niveau ?

Les mesures de confiance s'appliquent principalement à trois niveaux : phrase, mot et phonème. Dans le cadre des applications que nous considérons, nous allons définir des mesures de confiance pour des mots clés, mais également pour des mots hypothèses du graphe de mots interne du moteur de reconnaissance. Ainsi, nous allons naturellement proposer des mesures de confiance au niveau du mot.

Ce que nous considérerons comme un mot sera un élément du lexique. Bien que certains mots puissent avoir plusieurs prononciations (élisions de phonèmes, liaisons, noms propres), si leur graphie est identique ils seront considérés dans les calculs de nos mesures de confiance comme une seule et même entité.

3.2 Mesures trame-synchrones

Soit w le mot dont nous voulons évaluer la confiance et que nous appellerons *mot analysé*. Ce mot commence à l'instant τ et finit à l'instant t. Nous le représentons sous la forme condensée $[w, \tau, t]$.

Les mesures de confiance trame-synchrones ne sont fondées que sur des indices disponibles à l'instant de traitement du moteur de reconnaissance. Le mot analysé $[w, \tau, t]$ limite le calcul aux données présentes dans le système de décodage antérieures au temps t.

Les mesures trame-synchrones que nous avons définies sont fondées sur un rapport de vraisemblance entre l'hypothèse *nulle* et l'hypothèse *alternative*. L'hypothèse nulle représente la supposition qu'une séquence d'observations O ait été engendrée par un mot w considéré comme correct. L'hypothèse alternative représente quant à elle la supposition que la séquence O ait été engendrée par n'importe quel mot du lexique. Or généralement tout le lexique n'est pas considéré dans l'hypothèse alternative mais uniquement un sous-ensemble de celui-ci.

Les mesures de confiance que nous proposons sont définies à partir de l'équation générique suivante :

$$C([w, \tau, t]) = \frac{P(O|w)P(w)}{\sum_{w' \in E} P(O|w')P(w')} \tag{3.1}$$

$P(O|w)$ est la probabilité que la séquence O ait été engendrée par le mot w et $P(w)$ est la probabilité linguistique du mot w. Nous avons défini l'ensemble E comme un sous-ensemble de tous les mots présents dans le graphe vérifiant des contraintes que nous présentons dans le paragraphe suivant. Les mots appartenant à l'ensemble E sont dits concurrents du mot w.

L'équation 3.1 représente ainsi le rapport entre la vraisemblance du mot analysé $[w, \tau, t]$ et la vraisemblance de tous les mots qui peuvent être ses concurrents lors de la phase de reconnaissance. Cette mesure est fondée sur un principe similaire au rapport de vraisemblance avec modèles compétitifs (cf. section 2.3.5.5). Comme nous l'avons présenté paragraphe 2.3.5, ce rapport a été essentiellement utilisé pour la reconnaissance de mots isolés, en parole continue avec un vocabulaire réduit ou à partir de la liste des n-meilleures phrases. Nous le définissons dans un contexte grand vocabulaire.

3.2.1 Définition des mots concurrents de l'ensemble E

L'idée est de sélectionner parmi tous les mots possibles du graphe uniquement ceux qui peuvent se substituer à $[w, \tau, t]$ par rapport au processus de décision du moteur de reconnaissance. Ce critère repose sur la position dans le temps du mot concurrent ainsi que sur sa longueur. Une première possibilité serait de considérer les mots directement échangeables avec le mot $[w, \tau, t]$ analysé, i.e. les mots du graphe ayant exactement les mêmes temps de début et de fin que w. Toutefois, cette restriction aux mots ayant exactement la même position temporelle et la même longueur est trop forte. En effet, supposons que le mot w ne soit pas correct, il est possible que le vrai mot correct soit dans le graphe de mots à une position proche mais différente, ne serait-ce que d'une trame. La valeur de confiance de w calculée par un rapport de vraisemblance « strict » pourrait être proche de 1, alors qu'en considérant une trame de plus ou de moins, la valeur pourrait être proche de 0.

Pour tenir compte de ces variations, nous avons introduit un facteur de relâchement ε, associé à la longueur du mot $[w, \tau, t]$, qui permet d'intégrer des mots concurrents dont la position temporelle ou la longueur ne sont pas exactement identiques à celles de w. Le paramètre ε représente en fait un pourcentage de la longueur du mot w.

Soit d la longueur en trames du mot $[w, \tau, t]$ et ε le facteur de relâchement :

$$d = t - \tau + 1 \tag{3.2}$$

Un mot $[w', \tau', t']$ du graphe de mots du système de reconnaissance, de longueur d', est un mot concurrent de w s'il respecte les contraintes suivantes :

$$\tau - \varepsilon\, d \leq \tau' \leq \tau + \varepsilon\, d \tag{3.3}$$

$$t - \varepsilon\, d \leq t' \leq t \tag{3.4}$$

$$(1 - \varepsilon)\, d \leq d' \leq (1 + \varepsilon)\, d \tag{3.5}$$

L'asymétrie de la contrainte sur l'instant de fin des mots concurrents provient de notre choix d'élaborer une mesure de confiance trame-synchrone.

L'ensemble E introduit dans l'équation 3.1 est ainsi défini comme l'ensemble des mots du graphe qui valident les contraintes données par les équations 3.3 à 3.5. Ces contraintes dépendant du mot $[w, \tau, t]$ analysé, l'ensemble E est également défini relativement à $[\tau, t]$

Par ailleurs, notons que le mot $[w, \tau, t]$ lui-même respecte les contraintes des trois équations précédentes. Parmi les mesures de confiance fondées sur un rapport de vraisemblance, celles présentées paragraphe 2.3.5.5 utilisent une méthode de calcul de rapport similaire. Cependant, le rapport excluait le mot w analysé, ou la phrase analysée, de l'hypothèse alternative. Nous avons choisi d'inclure dans l'ensemble E le mot analysé lui-même comme étant un de ses propres concurrents. Ainsi le rapport obtenu est normalisé et peut être assimilé à une probabilité.

3.2.2 Gestion des occurrences multiples

Nous avons relâché les contraintes temporelles de sélection des mots concurrents de w qui appartiennent à l'ensemble E afin d'obtenir un plus grand nombre de mots représentant l'hypothèse alternative.

Toutefois ce relâchement des contraintes entraîne l'apparition d'occurrences multiples des mots concurrents w'. En effet, le moteur de reconnaissance peut avoir décodé le même mot avec des instants de début et fin légèrement décalés.

Par ailleurs, si les contraintes ont été relâchées pour les mots de l'ensemble E, elles doivent l'être également pour le mot analysé w afin que la cohérence du rapport de vraisemblance de l'équation 3.1 soit conservée. Des occurrences multiples peuvent donc apparaître pour w. Que faire de ces occurrences multiples du mot analysé et de ses concurrents ?

Nous avons choisi les deux méthodes suivantes afin de gérer les occurrences multiples dans le calcul du rapport de vraisemblance :

- faire la somme des vraisemblances des occurrences d'un même mot,
- considérer un seul représentant parmi les occurrences d'un même mot.

Concernant le choix d'un seul représentant de toutes les occurrences du mot analysé $[w, \tau, t]$, la solution que nous proposons consiste, à choisir parmi toutes les occurrences de w respectant les conditions 3.3 à 3.5, l'occurrence dont le score acoustique est maximal. Ce représentant est noté $[\widehat{w}, \hat{\tau}, \hat{t}]$. Les représentants $[\widehat{w}', \hat{\tau}', \hat{t}']$ des mots $[w', \tau', t']$ de E sont choisis de la même façon. Nous appelons alors $\widehat{E}$ l'ensemble des représentants $[\widehat{w}', \hat{\tau}', \hat{t}']$. Une stratégie identique est ainsi adoptée, que ce soit pour le mot analysé ou pour les mots concurrents de l'ensemble $\widehat{E}$.

Dans la suite de ce document, nous appellerons *méthode par sommation* la méthode consistant à sommer les vraisemblances des occurrences multiples d'un même mot, et, *méthode par maximisation* celle fondée sur la sélection d'un représentant de score acoustique maximal. Par extension, nous appellerons mots concurrents de l'ensemble E les occurrences des mots w' du graphe respectant les équations 3.3 à 3.5.

Après ces considérations introductives et les quelques notations mentionnées, nous décrivons dans les paragraphes suivants les mesures trame-synchrones que nous avons définies. Elles diffèrent les unes des autres par le niveau du modèle de langage utilisé : unigramme, bigramme ou trigramme.

3.2.3 Mesure fondée sur la probabilité unigramme

Dans cette première mesure, l'aspect local de la mesure est très marqué. Il n'est en effet tenu compte que de la probabilité acoustique des mots et de leur probabilité unigramme. Les deux méthodes de gestion des occurrences conduisent à deux équations pour cette mesure dite unigramme.

Méthode par sommation

Si une méthode de gestion par sommation des occurrences multiples a été choisie, cette mesure de confiance est définie par l'équation suivante :

$$C([w,\tau,t]) = \frac{\sum\limits_{[\widetilde{w},\tilde{\tau},\tilde{t}]\in E,\, \widetilde{w}=w} p(o_{\tilde{\tau}}^{\tilde{t}}|\widetilde{w})p(\widetilde{w})}{\sum\limits_{[w',\tau',t']\in E} p(o_{\tau'}^{t'}|w')p(w')} \tag{3.6}$$

- le mot $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$ est une occurrence du mot analysé w, contenue dans E ;
- le mot $[w', \tau', t']$ est une occurrence d'un mot w' contenue dans E ;
- $p(o_{\tilde{\tau}}^{\tilde{t}}|\widetilde{w})$ représente la probabilité acoustique du mot $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$, respectivement $p(o_{\tau'}^{t'}|w')$ représente la probabilité acoustique du mot $[w', \tau', t']$;

– $p(\widetilde{w})$ représente la probabilité linguistique unigramme du mot $\widetilde{w}$, respectivement $p(w')$ représente la probabilité linguistique unigramme du mot w'.

Comme nous l'avons précisé dans le paragraphe précédent, le mot w lui-même et toutes ses occurrences appartiennent à l'ensemble E.

Méthode par maximisation

Si la méthode de gestion des occurrences multiples est la sélection d'un représentant de score acoustique maximal, la mesure de confiance est alors définie ainsi :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})p(\widehat{w})}{\sum\limits_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')p(w')} \tag{3.7}$$

Notons bien que les éléments de l'ensemble $\widehat{E}$ dans l'équation sont des représentants de score acoustique maximal des mots concurrents de $[w,\tau,t]$. $[\widehat{w},\hat{\tau},\hat{t}]$ est le représentant de score acoustique maximal du mot analysé $[w,\tau,t]$ et se trouve à la fois au numérateur et au dénominateur.

3.2.4 Introduction de facteurs d'échelle

Les mesures de confiance que nous venons de décrire combinent probabilité acoustique et probabilité linguistique. Or, les valeurs de ces probabilités ont des ordres de grandeur différents. Afin d'équilibrer les contributions de ces deux probabilités, nous introduisons deux facteurs d'échelle : α pour la probabilité acoustique et β pour la probabilité linguistique.

Les équations précédentes 3.7 et 3.6 s'écrivent dorénavant ainsi :

Méthode par sommation

$$C([w,\tau,t]) = \frac{\sum\limits_{[\widetilde{w},\tilde{\tau},\tilde{t}]\in E,\, \widetilde{w}=w} p(o_{\tilde{\tau}}^{\tilde{t}}|\widetilde{w})^{\alpha}p(\widetilde{w})^{\beta}}{\sum\limits_{[w',\tau',t']\in E} p(o_{\tau'}^{t'}|w')^{\alpha}p(w')^{\beta}} \tag{3.8}$$

Méthode par maximisation

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha}p(\widehat{w})^{\beta}}{\sum\limits_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha}p(w')^{\beta}} \tag{3.9}$$

3.2.5 Mesure fondée sur la probabilité bigramme

Les mesures de confiance précédentes, fondées uniquement sur la probabilité unigramme des mots, n'ont qu'une vision très locale. Or, le contexte d'apparition d'un mot est important car les mots sont dépendants les uns des autres dans une phrase. C'est pourquoi nous avons modifié les précédentes mesures afin de tenir compte des mots présents dans le voisinage du mot analysé $[w,\tau,t]$. Afin de conserver le caractère trame-synchrone de nos mesures, nous ne considérons que

le voisinage passé de $[w, \tau, t]$. Comme pour la mesure dite unigramme, nous avons considéré les deux types de prise en compte des occurrences multiples.

Méthode par sommation

Dans ce cas, notre mesure dite bigramme, pour chacune des occurrences $\widetilde{w}$ du mot analysé w et pour chacune des occurrences des mots concurrents w' de E, prend en compte les probabilités bigrammes avec un ensemble de mots pouvant les précéder. Par exemple, pour une des occurrences $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$ du mot w analysé, nous considérons les probabilités bigrammes entre $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$ et tous les mots $[\widetilde{w}_p, \tilde{\tau}_p, \tilde{t}_p]$ du graphe précédant directement $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$. Nous faisons de même pour toutes les occurrences $[w', \tau', t']$ des mots de l'ensemble E avec pour chacune, les mots w'_p la précédant directement.

Nous obtenons la mesure de confiance ainsi définie :

$$C([w,\tau,t]) = \frac{\sum\limits_{[\widetilde{w},\tilde{\tau},\tilde{t}]\in E,\, \widetilde{w}=w} p(o_{\tilde{\tau}}^{\tilde{t}}|\widetilde{w})^{\alpha} \sum\limits_{\widetilde{w}_p} \left(p(\widetilde{w}|\widetilde{w}_p)p(\widetilde{w}_p)\right)^{\beta}}{\sum\limits_{[w',\tau',t']\in E} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} \left(p(w'|w'_p)p(w'_p)\right)^{\beta}} \tag{3.10}$$

Méthode par maximisation

Nous procédons de la même manière mais en ne prenant en compte que les probabilités bigrammes entre le représentant $\widehat{w}$ et tous les mots $\widehat{w}_p$ pouvant le précéder directement. Respectivement, sont prises en compte les probabilités bigrammes entre chacun des mots w' de $\widehat{E}$ et tous les mots w'_p pouvant le précéder directement. L'équation suivante définit cette mesure :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum\limits_{\widehat{w}_p} \left(p(\widehat{w}|\widehat{w}_p)p(\widehat{w}_p)\right)^{\beta}}{\sum\limits_{[w',\tau',t']\in \widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} \left(p(w'|w'_p)p(w'_p)\right)^{\beta}} \tag{3.11}$$

Définition des mots précédents

Nous avons considéré deux façons de définir l'ensemble des mots précédents :
- soit l'ensemble de tous les prédécesseurs exacts dans le graphe de mots, c'est-à-dire par exemple pour une occurrence $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$ l'ensemble des mots $[\widetilde{w}_p, \tilde{\tau}_p, \tilde{t}_p]$ du graphe tel que $\tilde{t}_p = \tilde{\tau} - 1$;
- soit l'unique mot prédécesseur au sens de Viterbi.

3.2.6 Mesure fondée sur la probabilité trigramme

Pour cette mesure, nous avons étendu davantage la portée du voisinage passé pris en compte dans la mesure en intégrant un modèle de langage trigramme. La mesure conserve son caractère trame-synchrone car le modèle de langage trigramme n'est considéré qu'avec des mots déjà décodés par le moteur de reconnaissance et antérieurs au mot analysé. Le modèle de langage

trigramme est généralement porteur d'une information plus pertinente que le modèle de langage bigramme. De même que pour les mesures de confiance unigrammes et bigrammes, nous allons exprimer cette nouvelle mesure dans le cadre d'une gestion des occurrences multiples par sommation, puis, nous la définirons pour la gestion par maximisation.

Méthode par sommation

Soit $[\widetilde{w}_p, \tilde{\tau}_p, \hat{t}_p]$ un mot du graphe précédant une occurrence $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$ du mot analysé, cette mesure de confiance prend en compte un ensemble de prédécesseurs $[\widetilde{w}_{pp}, \tilde{\tau}_{pp}, \tilde{t}_{pp}]$ de $[\widetilde{w}_p, \tilde{\tau}_p, \hat{t}_p]$ appartenant au graphe de mots. Nous faisons de même pour les occurrences $[w', \tau', t']$ des mots l'ensemble E en définissant les mots $[w'_{pp}, \tau'_{pp}, t'_{pp}]$ précédant eux-mêmes les mots $[w'_p, \tau'_p, t'_p]$, prédécesseurs de $[w', \tau', t']$

L'équation définissant cette mesure est la suivante :

$$C([w,\tau,t]) = \frac{\sum\limits_{[\widetilde{w},\tilde{\tau},\tilde{t}]\in E,\, \widetilde{w}=w} p(o_{\tilde{\tau}}^{\tilde{t}}|\widetilde{w})^{\alpha} \sum\limits_{\widetilde{w}_p} \sum\limits_{\widetilde{w}_{pp}} \left(p(\widetilde{w}|\widetilde{w}_p\widetilde{w}_{pp})p(\widetilde{w}_p|\widetilde{w}_{pp})p(\widetilde{w}_{pp})\right)^{\beta}}{\sum\limits_{[w',\tau',t']\in E} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} \sum\limits_{w'_{pp}} \left(p(w'|w'_pw'_{pp})p(w'_p|w'_{pp})p(w'_{pp})\right)^{\beta}} \tag{3.12}$$

Méthode par maximisation

Nous définissons la mesure de confiance suivant le même principe avec cette fois-ci les mots $[\widehat{w}_p, \hat{\tau}_p, \hat{t}_p]$ précédant l'occurrence $[\widehat{w}, \hat{\tau}, \hat{t}]$ de score acoustique maximal du mot analysé w. Pour chacun de ces mots $[\widehat{w}_p, \hat{\tau}_p, \hat{t}_p]$, nous considérons un ensemble de mots $[\widehat{w}_{pp}, \hat{\tau}_{pp}, \hat{t}_{pp}]$ du graphe pouvant les précéder. Nous procédons de même pour les mots $[w', \tau', t']$ de l'ensemble $\widehat{E}$ en calculant les probabilités trigrammes avec un ensemble de mots $[w'_p, \tau'_p, t'_p]$ et $[w'_{pp}, \tau'_{pp}, t'_{pp}]$.

La mesure s'exprime alors sous la forme suivante :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum\limits_{\widehat{w}_p} \sum\limits_{\widehat{w}_{pp}} \left(p(\widehat{w}|\widehat{w}_p\widehat{w}_{pp})p(\widehat{w}_p|\widehat{w}_{pp})p(\widehat{w}_{pp})\right)^{\beta}}{\sum\limits_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} \sum\limits_{w'_{pp}} \left(p(w'|w'_pw'_{pp})p(w'_p|w'_{pp})p(w'_{pp})\right)^{\beta}} \tag{3.13}$$

Définition des mots précédents

De même que pour la mesure dite bigramme, nous avons plusieurs choix possibles pour les mots précédents et les mots précédant ces mots précédents. Nous avons défini les mêmes ensembles que pour la mesure bigramme :

- soit l'ensemble de tous les prédécesseurs exacts dans le graphe de mots de tous les prédécesseurs exacts. Par exemple pour une occurrence $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$ l'ensemble des prédécesseurs est constitué des mots $[\widetilde{w}_p, \tilde{\tau}_p, \tilde{t}_p]$ du graphe tels que $\tilde{t}_p = \tilde{\tau} - 1$ et l'ensemble des prédécesseurs d'un mot $[\widetilde{w}_p, \tilde{\tau}_p, \tilde{t}_p]$ est constitué des mots $[\widetilde{w}_{pp}, \tilde{\tau}_{pp}, \tilde{t}_{pp}]$ tels que $\tilde{t}_{pp} = \tilde{\tau}_p - 1$;
- soit l'unique prédécesseur du prédécesseur d'un mot, le tout au sens de Viterbi.

3.2.7 Implantation

3.2.7.1 Construction de l'ensemble $\widehat{E}$

Nous décrivons dans cette section le principe de construction de l'ensemble $\widehat{E}$ des représentants de score acoustique maximal des occurrences des mots concurrents d'un mot analysé w, dans le cas d'une gestion des occurrences multiples par maximisation. Dans le cas de la méthode par sommation, l'ensemble E est simplement défini par l'ensemble des mots du graphe respectant les contraintes temporelles des équations 3.3, 3.4 et 3.5.

Pour un mot $[w, \tau, t]$ analysé, l'ensemble $\widehat{E}$ des représentants des mots concurrents est construit par l'algorithme 3.1 dont le principe est le suivant :

- parcours du graphe de mots,
- sélection des mots selon les contraintes temporelles données par les équations 3.3, 3.4 et 3.5,
- recherche parmi les mots sélectionnés des représentants ayant les scores acoustiques maximaux.

L'espace de recherche dépend de la longueur du mot à analyser ainsi que du paramètre de relâchement ε. L'ajout et la recherche du maximum se font en même temps.

Chacun des mots concurrents w' apparaît dans $\widehat{E}$ une seule fois, et cette unique occurrence représente le mot concurrent w' de probabilité acoustique maximale. L'ensemble $\widehat{E}$ défini, nous pouvons calculer la valeur de confiance du mot $[w, \tau, t]$.

3.2.7.2 Calcul des mesures fondées sur les probabilités unigramme, bigramme et trigramme

Pour calculer les mesures de confiance fondées sur les probabilités unigramme, bigramme ou trigramme des mots, données par les équations 3.9 à 3.13, nous devons calculer le numérateur et le dénominateur du rapport de vraisemblance. Or, comme le mot w dont nous voulons évaluer la confiance fait également partie de l'ensemble E, le calcul du numérateur sera effectué pendant une des itérations du calcul du dénominateur.

Afin de calculer le dénominateur, pour chaque mot de l'ensemble E, nous cumulons le produit de la probabilité acoustique par la probabilité soit unigramme, bigramme ou trigramme selon la mesure calculée.

Quand le mot w est rencontré dans l'algorithme de calcul, la valeur du produit de la probabilité acoustique par la probabilité linguistique est conservée pour le numérateur des équations avant d'être cumulée pour le dénominateur. La valeur de confiance pour le mot $[w, \tau, t]$ est donnée par le rapport entre le numérateur et le dénominateur.

Par ailleurs, le choix de l'ensemble des mots prédécesseurs a une incidence sur l'algorithme de calcul de la mesure de confiance. En effet, si nous ne prenons en compte que les précédents au sens de Viterbi, un seul mot est concerné et il n'est donc pas nécessaire de sommer les probabilités bigrammes ou trigrammes. Par contre, si nous choisissons de considérer tous les mots temporellement précédents, nous devons parcourir le graphe de mots. Par exemple pour la mesure bigramme, soit l'occurrence $[w', \tau', t']$, nous calculons les probabilités bigrammes entre w' et chacun des mots $[w'_p, \tau'_p, t'_p]$ du graphe dont l'instant de fin vaut $\tau' - 1$.

Comme le montre l'algorithme 3.2, pour la mesure de confiance fondée sur la probabilité trigramme, une profondeur d'itération supplémentaire est ajoutée par rapport à la mesure utilisant la probabilité bigramme. En effet, pour chaque occurrence $[w', \tau', t']$ de mots de l'ensemble E, ou $\widehat{E}$, nous devons parcourir tous les mots $[w'_p, \tau'_p, t'_p]$ précédant w' dont l'instant de fin vaut $\tau' - 1$.

Algorithme 3.1 : Algorithme décrivant la construction de l'ensemble $\widehat{E}$ associé au mot hypothèse $[w, \tau, t]$ dont nous voulons estimer la confiance.

longueur $lg = t - \tau + 1$
décalage $dec = lg * \varepsilon$
pour *(t' = (t − dec) ; t' < (t + 1) ; t'++)*
faire
  `/* parcours de tous les mots se terminant à l'instant t' */`
  pour *(i = 0 ; i < nombre de mots se terminant à l'instant t' ; i++)*
  faire
    soit $[w', \tau', t']$ le i^{e} mot
    `/* test sur les contraintes temporelles */`
    si *(τ' ≥ (τ − dec)) et (τ' ≤ (τ + dec))*
    alors
      `/* test sur la longueur du mot */`
      si *(lg' ≥ (lg - dec)) et (lg' ≤ (lg + dec))*
      alors
        `/* test si le mot est déjà dans l'ensemble E */`
        si *la graphie de w' est déjà dans E*
        alors
          si *le score acoustique de w' est meilleur que celui déjà dans E*
          alors
            on conserve w' dans E à la place de l'ancien représentant
          sinon
            on garde l'ancien représentant
          fin
        sinon
          On ajoute w' à l'ensemble E
        fin
      fin
    fin
  fin
fin

De plus, nous devons également parcourir pour chaque w'_p, tous les mots w'_{pp} dont l'instant de fin vaut $\tau'_p - 1$ puis calculer la probabilité trigramme $p(w'|w'_p w'_{pp})$.

Algorithme 3.2 : Algorithme de calcul du numérateur et du dénominateur de la mesure de confiance fondée sur la probabilité trigramme avec gestion par maximisation et précédents temporels (Eq.3.13)

numérateur = 0
dénominateur = 0
`/* parcours des toutes les occurrences des mots de l'ensemble` E `*/`
pour chaque $[w', \tau', t'] \in E$
faire
  SommeTrigramme = 0
  `/* parcours de tous les mots du graphe précédant` w' `*/`
  $t'_p = \tau' - 1$
  pour *(i = 0 ; i < nombre de mots finissant à l'instant* t'_p *; i++)*
  faire
    $[w'_p, \tau'_p, \tau' - 1]$ le i^{e} mot
    `/* parcours de tous les mots du graphe précédant` w'_p `*/`
    $t'_{pp} = \tau'_p - 1$
    pour *(j = 0 ; j < nombre de mots finissant à l'instant* t'_{pp} *; j++)*
    faire
      $[w'_{pp}, \tau'_{pp}, \tau'_p - 1]$ le j^{e} mot
      `/* calcul du score linguistique connaissant` w'_p `et` w'_{pp} `et cumul de cette valeur */`
      SommeTrigramme += $(P(w'|w'_p w'_{pp})P(w'_p|w'_{pp})P(w'_{pp}))^{\beta}$
    fin
  fin
  `/* cumul dans le dénominateur de la somme des trigrammes de` w' `*/`
  dénominateur += $P(x_{\tau'}^{t'}|w')^{\alpha}$.SommeTrigramme
  si $w' = w$ **alors**
    numérateur = $P(x_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha}$.SommeTrigramme
  fin
fin

3.3 Mesures locales

Pour un mot analysé $[w, \tau, t]$, les mesures de confiance trame-synchrones définies dans la section précédente ne sont fondées que sur des indices disponibles dans le moteur de reconnaissance à l'instant t.

Au contraire, pour les mesures de confiance que nous définissons dans cette section, nous introduisons des connaissances postérieures au temps t. C'est pourquoi ces mesures ne peuvent plus être trame-synchrones, mais uniquement locales. Un délai est introduit afin d'attendre la disponibilité des informations nécessaires au calcul de la mesure. L'intérêt principal est de prendre en compte un contexte plus large autour du mot à analyser. Des travaux ont montré que la prise en compte des mots postérieurs au mot analysé par le biais de modèles de langage inverses

représente un apport d'information important [Weintraub 97, Duchateau 02a].

Ces nouvelles mesures de confiance se distinguent entre elles par la définition du voisinage local utilisé.

3.3.1 Mesures fondées sur la probabilité *a posteriori*

Nous avons choisi de définir ces mesures de confiance locales sur l'estimation de la probabilité *a posteriori* des mots.

Le calcul effectif de nos mesures de confiance locales est fondé sur la méthode décrite par Wessel et al. dans le cadre de la définition de leur mesure de confiance estimant la probabilité *a posteriori* (cf. section 2.3.6.2) [Wessel 01]. Pour rappel, cette méthode de calcul consiste à calculer récursivement les probabilités *forward* et *backward*, $\Phi([w,\tau,t])$ et $\Psi([w,\tau,t])$, du mot $[w,\tau,t]$ analysé, selon les équations suivantes :

$$\Phi([w,\tau,t]) = p(o_\tau^t|w)^\alpha \sum_{w_p}\sum_{\tau'} \Phi([w_p,\tau',\tau-1])\, p(w|w_p)^\beta$$

$$\Psi([w,\tau,t]) = p(o_\tau^t|w)^\alpha \sum_{w_s}\sum_{t'} \Psi([w_s,t+1,t'])\, p(w_s|w)^\beta$$

La probabilité *a posteriori* est alors estimée ainsi :

$$p(w|O) = p([w,\tau,t]|o_1^T) = \frac{\Phi([w,\tau,t])\Psi([w,\tau,t])}{p(o_1^T)p(o_\tau^t|w)^\alpha}$$

Sachant que la probabilité de l'observation $p(o_1^T)$ peut être calculée de la manière suivante :

$$P(O) = p(o_1^T) = \sum_w \sum_\tau \Phi([w,\tau,T])$$

Le principe de nos mesures de confiance est de définir un voisinage autour du mot analysé en prenant en compte, de part et d'autre du mot, un nombre fixe de trames. Ainsi la taille totale en trames du voisinage V d'un mot w est la somme de la longueur du mot w et des longueurs des voisinages passé et futur. La taille totale du voisinage est donc dépendante du mot analysé.

A partir du voisinage V, nous extrayons du graphe de mots la partie correspondant à V, c'est-à-dire comprise entre le temps de début du voisinage V et le temps de fin de V. Puis dans ce sous-graphe, nous éliminons les liens passés et futurs à V. Si le voisinage V débute à l'instant d et termine à l'instant f, les liens éliminés correspondent aux mots $[w,\tau,t]$ tels que $\tau < d < t$ ou $\tau < f < t$. De plus, dans le sous-graphe, certains mots se retrouvent isolés car ils ne sont plus liés à des chemins partant du début de l'extrait à la fin de celui-ci.

Nous calculons alors sur le sous-graphe résultant la probabilité *a posteriori* du mot w, par la méthode décrite section 2.3.6.2, dont nous avons donné un bref rappel au paragraphe précédent, dans le cadre de probabilités bigrammes. Dans cette méthode, les mots isolés n'ont pas d'incidence sur le calcul de la mesure de confiance car l'algorithme de calcul *forward-backward* implique que seuls les mots appartenant à un chemin allant du début à la fin du graphe sont considérés et peuvent avoir une valeur de confiance. Toutefois il possible que le mot analysé w lui-même n'ait plus de liaison dans le graphe extrait et ainsi aucune valeur de confiance ne peut lui être attribuée par cette méthode. Ce cas se présente d'autant plus fréquemment que la taille du voisinage V est petite. Nous avons alors choisi d'affecter aux mots analysés n'ayant pas de valeur de confiance, la probabilité *a priori* qu'a un mot d'être correct. Cette probabilité *a priori* correspond au taux de mots corrects du système de reconnaissance, c'est-à-dire $(1-CER)$, CER étant le taux d'erreur de confiance présenté section 2.4.2.

3.3.2 Définition des voisinages

Nous définissons deux types de voisinage :
- un voisinage symétrique pour lequel les tailles des voisinages passé et futur sont égales,
- un voisinage asymétrique pour lequel les tailles des deux voisinages passé et futur sont définies indépendamment l'une de l'autre.

La mesure de confiance dite symétrique est dépendante d'un paramètre qui définit la taille en trame des voisinages passé et futur. La figure 3.4 représente le voisinage V associé au mot analysé w avec un paramètre de taille x.

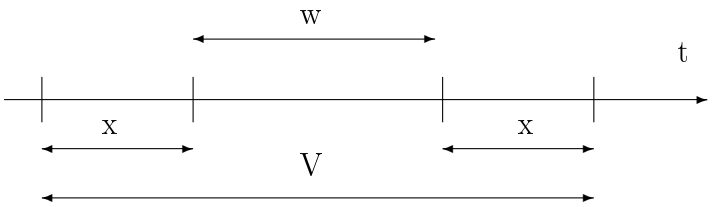


FIG. 3.4 – Illustration du voisinage pris en compte pour la mesure de confiance symétrique de paramètre de taille x.

La mesure de confiance dite asymétrique introduit deux paramètres définissant les tailles de ces deux voisinages. La figure 3.5 montre le voisinage V considéré pour le mot analysé w et une mesure de confiance asymétrique de taille de voisinage passé x et de taille de voisinage futur y.

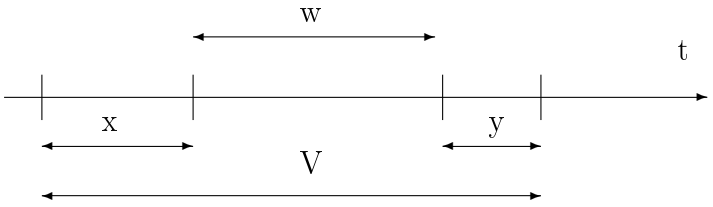


FIG. 3.5 – Illustration du voisinage pris en compte pour la mesure de confiance asymétrique de paramètre de taille x et y.

La mesure dite asymétrique permet par rapport à la mesure symétrique de prendre en compte plus d'informations issues du voisinage passé du mot analysé w, sans augmenter le délai introduit par le voisinage futur de w.

3.3.3 Introduction d'un facteur de flexibilité η

Le problème des occurrences multiples se pose aussi pour cette mesure. En effet, dans le sous-graphe extrait il est possible que plusieurs occurrences du mot analysé puissent apparaître à des positions temporelles similaires. Par la méthode de calcul *forward-backward* utilisée, une estimation de la probabilité *a posteriori* est calculée pour chaque mot du graphe, et donc pour chacune des occurrences du mot analysé. N'en retenir qu'une seule sous-estime la vraie probabilité *a posteriori* du mot.

Afin de gérer ce problème d'occurrences multiples, nous introduisons un facteur de flexibilité η et sommons les estimations des occurrences du mot analysé qui respectent un certain critère dépendant de η.

Soit le mot analysé $[w, \tau, t]$ dont nous voulons estimer la confiance, soit η le facteur de flexibilité et soit d la longueur du mot w.

$$d = t - \tau + 1 \tag{3.14}$$

Pour définir la mesure de confiance de $[w, \tau, t]$, nous sommons les valeurs de confiance des occurrences $[\widetilde{w}, \tilde{\tau}, \tilde{t}]$ de w, appartenant au sous-graphe extrait et qui réalisent les contraintes suivantes :

$$\tau - \eta\, d \leq \tilde{\tau} \leq \tau + \eta\, d \tag{3.15}$$

$$t - \eta\, d \leq \tilde{t} \leq t + \eta\, d \tag{3.16}$$

$$(1 - \eta)\, d \leq \tilde{d} \leq (1 + \eta)\, d \tag{3.17}$$

Ces contraintes s'appliquent sur le sous-graphe de mots extrait qui est associé au mot analysé. Soit F l'ensemble des occurrences d'un mot w réalisant les contraintes données par les équations précédentes, la valeur de la mesure de confiance $C([w, \tau, t])$ est ainsi donnée par l'équation suivante :

$$C(w, \tau, t]) = \sum_{[\widetilde{w}, \tilde{\tau}, \tilde{t}] \in F} p([\widetilde{w}, \tilde{\tau}, \tilde{t}] | o_d^f) \tag{3.18}$$

o_d^f est la séquence d'observations qui correspond au graphe de mots associé au voisinage défini par la mesure et par le mot analysé.

Remarque : Les équations 3.15 à 3.17 ressemblent aux équations 3.3 à 3.5 mais ce facteur de flexibilité η n'a rien à voir avec le facteur de relâchement ε introduit pour les mesures de confiance trame-synchrone :

- pour nos mesures trame-synchrones, le facteur de relâchement sert à déterminer un ensemble d'occurrences de mots concurrents du mot analysé afin de déterminer l'hypothèse alternative dans le rapport de vraisemblance,
- pour nos mesures locales fondées sur la probabilité *a posteriori*, le facteur de flexibilité permet de prendre en compte uniquement les occurrences du mot analysé dans le voisinage V, puis de sommer les valeurs de confiance de chacune d'elles.

3.4 Homogénéisation de la répartition des valeurs de confiance

Une mesure de confiance donne un indice de la fiabilité d'un mot reconnu par un système de reconnaissance. Elle doit donc refléter la qualité du système de reconnaissance et ainsi la moyenne des valeurs de confiance doit être proche du taux de mots corrects du système. Si la moyenne des valeurs de confiance est plus élevée que le taux de mots corrects, alors la mesure de confiance va surestimer la fiabilité des mots. Inversement, si la moyenne des valeurs de confiance est plus faible que le taux de mots corrects, alors la mesure de confiance va sous-estimer la fiabilité des mots.

Nous avons donc introduit une méthode permettant de rétablir une corrélation entre les valeurs de confiance calculées et le taux de mots correct du système de reconnaissance.

Le principe est le suivant :

- pour chaque mot des phrases d'un corpus de développement, nous calculons sa valeur de confiance selon la mesure choisie,
- nous trions les mots selon leur valeur de confiance,
- nous partitionnons cet ensemble de mots en N intervalles homogènes de taille identique (nombre de mots identique),
- par rapport aux transcriptions manuelles de référence du corpus nous étiquetons ces mots *correct* ou *incorrect*,
- pour chacun de ces intervalles, nous calculons à la fois la valeur de confiance moyenne et le taux de mots corrects contenus dans cet intervalle.

Le taux de mots corrects correspond au nombre de mots étiquetés *correct* par rapport au nombre total de mots reconnus. Ce taux est également obtenu à partir du taux CER du système de reconnaissance : $(1 - CER)$ (cf. section 2.4.2).

Ainsi, à chaque mot analysé, dont nous avons calculé une valeur de confiance, correspond un des N intervalles et donc un taux de mots corrects. Nous définissons alors la nouvelle mesure de confiance comme étant cette valeur du taux de mots corrects.

La figure 3.6 illustre la répartition observée entre la valeur de confiance et le taux de mots corrects, parmi tous les mots des graphes de mots correspondant à 50 phrases. Dans cet exemple, nous avons partitionné l'ensemble des mots en 20 intervalles homogènes de taille identique. La courbe en trait continu montre l'évolution de la valeur de confiance moyenne dans chacun des intervalles. Les intervalles étant triés, les valeurs moyennes sont croissantes sur l'intervalle $[0, 1]$. La seconde courbe représente le taux de mots corrects de chaque intervalle, i.e. $(1 - CER)$.

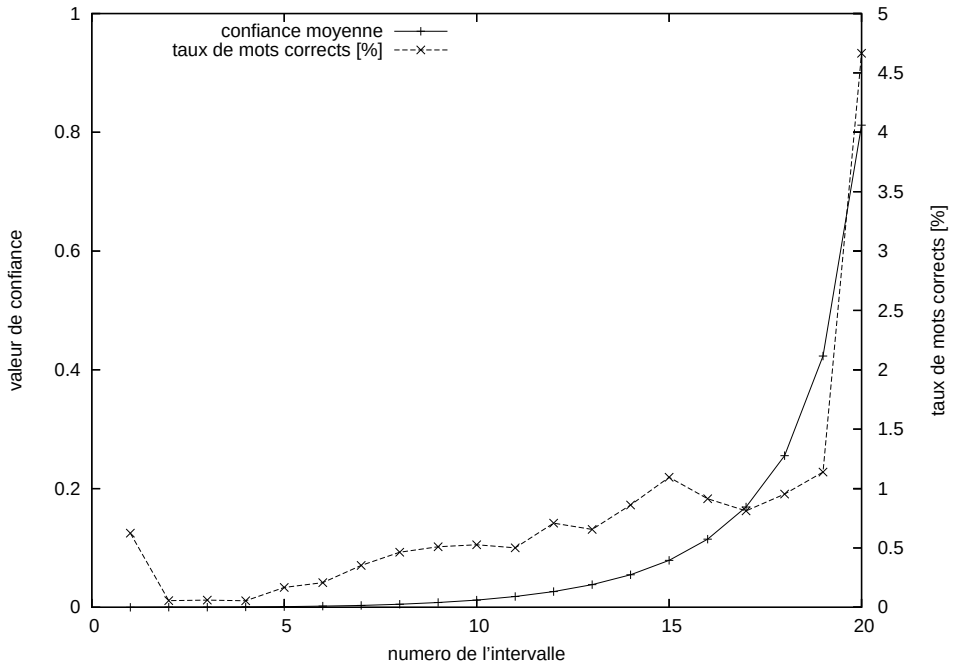


FIG. 3.6 – Distribution du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique pour tous les mots d'un ensemble de graphes de mots.

Nous pouvons remarquer le caractère non linéaire de la distribution du taux de mots corrects ainsi que des valeurs de confiance. Concernant la valeur du taux de mots corrects, nous pouvons noter qu'elle est globalement inférieure à 1% dans tous les intervalles, excepté le dernier pour lequel la valeur de confiance moyenne vaut 0,99 et le taux de mots corrects est égal à 4,7%. Bien que cela soit surprenant, ceci est en fait tout à fait normal. Dans un graphe de mots, si nous supposons que le nombre moyen d'hypothèses de mots à chaque trame est égal à 100, et que le nombre moyen de mots corrects faisant partie de la solution est égal à 2, nous obtenons un taux de mots corrects à chaque trame de 2%.

Avec une mesure de confiance idéale, la courbe représentant le taux de mots corrects de la figure 3.6 serait une droite égale à 0% avec un point à 100% pour le bloc des mots ayant une confiance de 1.

La figure 3.7 représente quant à elle la répartition observée sur un corpus de développement des valeurs de confiance d'une de nos mesures. Contrairement à la courbe précédente, l'analyse est faite sur les mots des phrases reconnues et non sur tous les mots des graphes associés. Nous avons également choisi une résolution de 20 intervalles pour cet exemple. La courbe en trait continu montre l'évolution de la valeur de confiance moyenne dans chacun des intervalles, évoluant dans l'intervalle $[0, 1]$ de manière croissante. La seconde courbe montre l'évolution du taux de mots corrects de chaque intervalle suivant la valeur de confiance de l'intervalle associé.

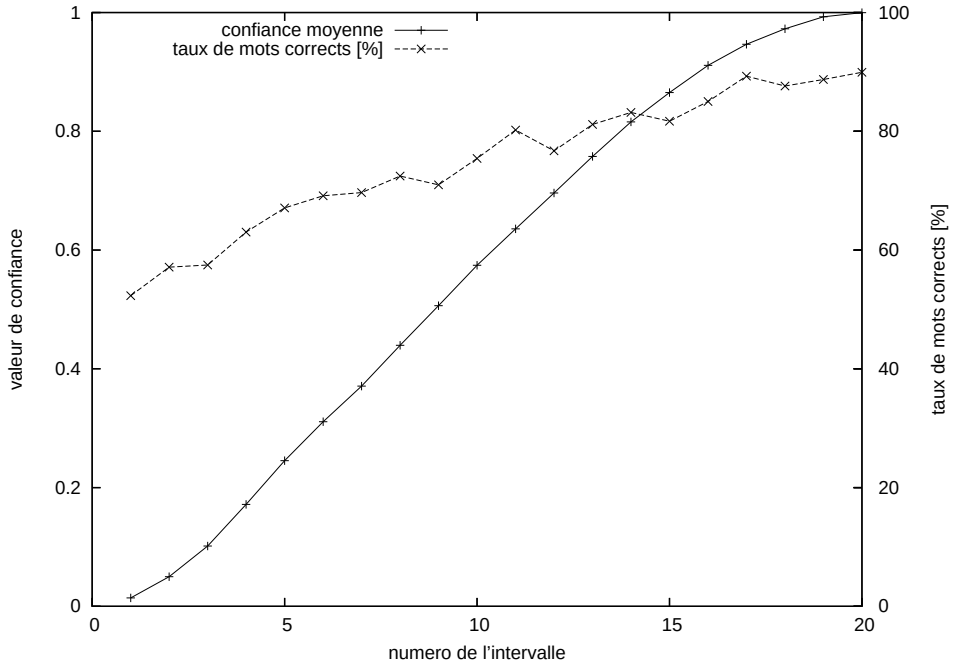


Fig. 3.7 – Distribution du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique pour les mots faisant partie d'un ensemble de phrases reconnues.

Nous pouvons remarquer que les courbes de la figure 3.7 évoluent de façon similaire. Comme cet exemple concerne les mots des phrases reconnues, le taux de mots corrects est plus élevé que pour la courbe de la figure 3.6 qui considère tous les mots des graphes de mots. En effet, le système de reconnaissance délivre plus de mots bien reconnus et donc corrects que de mots incorrects. La courbe de la figure 3.7 montre qu'un mot ayant une forte valeur de confiance est

très certainement correct, alors que ceci est moins vrai pour les mots de faible de confiance.

De ces deux exemples nous voyons donc que les valeurs de confiance ne reflètent pas le taux de mots corrects que nous pourrions espérer obtenir et que ce taux est également dépendant de données sur lesquelles il est calculé. L'intérêt d'homogénéiser les valeurs de confiance semble donc être utile dans ces cas.

Par ailleurs, dans des systèmes comme ROVER où plusieurs systèmes de reconnaissance sont combinés afin de déterminer la phrase reconnue, chaque système peut indiquer des valeurs confiance. Dans ce cas, il est préférable que les valeurs de confiance des systèmes soient normalisées entre eux pour ne pas pénaliser ou favoriser un système par rapport à un autre. Un moyen d'effectuer cette normalisation est d'homogénéiser les valeurs de confiance comme nous venons de le décrire.

3.5 Complexité de nos mesures de confiance

Nous allons analyser la complexité de nos mesures de confiance. Soient :
- N, le nombre moyen de mots se terminant à une trame t dans le graphe,
- T, le nombre de trames à traiter,
- V, la taille en trames du voisinage du mot analysé dans le cas des mesures locales,
- D, la durée moyenne en trame d'un mot.

3.5.1 Mesures trame-synchrones

Que nous considérions les gestions des occurrences multiples par maximisation ou par sommation, le même parcours du graphe de mots doit être effectué afin de construire l'ensemble E ou $\widehat{E}$. La construction de ces ensembles nécessite l'analyse d'un nombre de trames en fonction du facteur de relâchement ε utilisé. La complexité de cette construction est ainsi en $O(\varepsilon DN)$, c'est-à-dire $O(N)$.

Une fois ces ensembles déterminés, le calcul des mesures de confiance peut s'effectuer.

Pour la mesure unigramme aucune autre opération importante n'est nécessaire pour calculer la valeur de confiance d'un mot. La complexité de la mesure unigramme est donc en $O(N)$ pour un mot. La complexité obtenue pour le calcul de la valeur de confiance de tous les mots du graphe est en $O(TN^2)$ car il y a en moyenne TN mots dans le graphe.

Concernant la mesure bigramme, dans le pire cas nous devons parcourir pour chaque mot $[w', \tau', t']$ de E tous les mots $[w'_p, \tau'_p, t'_p]$ du graphe tels que $t'_p = \tau - 1$ afin de calculer les probabilités bigrammes. L'ensemble E contient au pire N éléments, mais il est plus raisonnable de considérer que E contient kN éléments avec $k < 1$. Le calcul de la mesure pour un mot est donc de complexité $O(kN^2)$ et pour le calcul sur tous les mots du graphe, la complexité est en $O(kTN^3)$, avec $k < 1$.

Pour la mesure trigramme, une profondeur de recherche supplémentaire est définie augmentant ainsi la complexité pour un mot en $O(kN^3)$ et donc pour tous les mots du graphe en $O(kTN^4)$ avec $k < 1$.

3.5.2 Mesures locales

Nos mesures de confiance locales nécessitent pour être calculées l'estimation par récurrence des probabilités *forward* et *backward* Φ et Ψ (cf. section 3.3). Pour un mot analysé $[w, \tau, t]$, ces probabilités ne dépendant que des mots $[w_p, \tau_p, t_p]$ tels que $t_p = \tau - 1$, la complexité de calcul

d'une itération est $O(N)$. Mais comme nous devons calculer Φ et Ψ pour tous les mots du sous-graphe associé au voisinage de la mesure locale, la complexité totale du calcul de $\Phi([w,\tau,t])$, et de $\Psi([w,\tau,t])$, est en $O(VN^2)$. Or ces deux quantités doivent être évaluées et donc la complexité après ces étapes est $O(2VN^2)$.

Toutefois, pour calculer la probabilité *a posteriori* d'un mot $[w,\tau,t]$ nous devons également estimer la probabilité $P(o_1^T)$ qui est la somme des probabilités *forward* des mots dont l'instant de fin correspond à la fin du voisinage. Cette opération est en $O(N)$.

Ainsi, le calcul de la valeur de confiance d'un mot est en $O(2VN^2)$. De la même façon que pour les mesures trame-synchrones, il y a en moyenne TN mots dans le graphe et donc la complexité de calcul de tous les mots est en $O(2TVN^3)$

Si nous considérons la mesure locale avec un voisinage passé qui prend en compte tous les mots depuis le début de la phrase, il est possible de diminuer la complexité de calcul de cette mesure. En effet, les probabilités *forward* peuvent être conservées d'un mot analysé à un autre. Ainsi N opérations suffisent pour calculer ces probabilités pour un mot analysé et non VN^2. La complexité du calcul de la mesure devient alors $O(VN^2)$ pour un mot et $O(TVN^3)$ pour tout le graphe.

Une estimation des quantités mises en jeu dans ces complexités sera donnée dans le chapitre suivant, une fois nos conditions expérimentales introduites.

3.6 Conclusion

Dans ce chapitre, nous avons présenté les mesures de confiance que nous avons introduites.

Nous avons défini des mesures de confiance fondées sur un rapport de vraisemblance entre l'hypothèse à tester et les mots *concurrents* de cette hypothèse. Ces mesures ont pour principale caractéristique de pouvoir être calculées de manière trame-synchrone, c'est-à-dire au fur et à mesure de la progression du décodage de la phrase par le moteur de reconnaissance. Cette caractéristique permet de les intégrer dans le processus de décodage afin d'influer sur le calcul de la vraisemblance de la solution trouvée par le système. D'autres mesures fondées sur un rapport de vraisemblance ont été définies dans l'état de l'art, mais celles-ci ne pouvaient gérer le cadre d'un système grand vocabulaire, contrairement à nos mesures.

La probabilité *a posteriori* d'un mot est présentée comme une valeur pertinente en tant que mesure de confiance dans [Jiang 05]. Plusieurs travaux ont proposé des mesures de confiance estimant la probabilité *a posteriori* d'un mot mais en nécessitant la reconnaissance complète de toute la phrase [Weintraub 97, Wessel 01]. Nous avons proposé des mesures de confiance fondées sur une estimation locale de cette probabilité en utilisant le même algorithme de calcul que celui présenté dans [Wessel 01]. Nos mesures estiment cette probabilité non pas sur la phrase entière, mais uniquement sur un voisinage du mot dont nous voulons calculer la valeur de confiance. Ce voisinage est défini par un nombre fixe de trames qui précédent et qui suivent le mot analysé. La prise en compte du contexte futur du mot induit un délai afin d'attendre la génération des données nécessaires au calcul. De ce fait, ces mesures ne sont pas trame-synchrones mais locales. Nous avons introduit deux types de voisinage, symétrique et asymétrique, afin de pouvoir prendre en compte plus d'informations passées et donc déjà décodées par le moteur de reconnaissance, sans augmenter le délai induit par le voisinage futur.

Toutefois, le court délai nécessaire à génération des informations du voisinage futur du mot analysé dans le calcul de la mesure de confiance n'est pas gênant pour les applications que nous visons. En effet, pour des applications comme la transcription d'émissions télévisées ou la transcription de cours en salle de classe pour des enfants malentendants, un décalage est

initialement introduit afin d'éviter des dérapages pour les émissions et afin de permettre la resynchronisation des enfants entre la transcription écrite et les lèvres du professeur.

Il ne suffit pas de définir de nouvelles mesures de confiance, il faut les évaluer et éventuellement les comparer à d'autres. Pour cela nous avons décidé de nous placer dans des conditions réelles : système de reconnaissance grand vocabulaire et applications concrètes de transcription et de détection de mots clés. Ces expérimentations sont décrites dans les trois chapitres qui suivent.

Chapitre 4

Conditions expérimentales

Sommaire

4.1 Introduction 82
4.2 Moteur de reconnaissance : Julius 82
4.2.1 La première passe de *Julius* 83
4.2.2 La deuxième passe de *Julius* 83
4.2.3 Options de compilation 84
4.2.4 Le graphe de mots 84
4.3 Paramétrisation 85
4.4 Modèles acoustiques 85
4.4.1 Monophones 85
4.4.2 Triphones 85
4.5 Lexique et modèle de langage 86
4.5.1 Utilisés conjointement avec les modèles monophones 86
4.5.2 Utilisés conjointement avec les modèles triphones 86
4.6 Corpus de développement et de test 87
4.7 Complexité de nos mesures de confiance 87
4.7.1 Mesures trame-synchrones 88
4.7.2 Mesures locales 88
4.8 Conclusion 88

4.1 Introduction

Dans ce chapitre, nous présentons les conditions dans lesquelles nous avons mené les expériences de validation et d'évaluation de nos travaux. Ces conditions sont liées en partie au moteur de reconnaissance Julius ainsi qu'aux modélisations acoustiques et linguistiques telles qu'elles étaient disponibles au commencement des travaux de cette thèse dans l'équipe PAROLE. Ces conditions générales d'expérimentation ont été définies et utilisées dans la phase de *test à blanc* de la campagne d'évaluation ESTER. Fondé sur des modèles acoustiques monophones, ce système a obtenu les meilleures performances parmi l'ensemble des systèmes participant à ce test à blanc. Nous avons conservé ce système pour la plupart de nos travaux afin de pouvoir comparer nos résultats entre eux tout au long de la thèse.

Toutefois, dans le cadre de la phase de test réelle de l'évaluation de la campagne ESTER, un nouveau système de reconnaissance a été défini au sein de l'équipe avec notamment l'adoption de modèles acoustiques triphones. Cette modélisation a d'ailleurs été utilisée par la majorité des participants. Nous avons choisi ce nouveau système pour nos expériences d'intégration d'une mesure de confiance au sein de la phase de décodage du moteur de reconnaissance et pour l'application de transcription pour les élèves malentendants. En effet, pour ces expériences, nous évaluons l'impact de l'intégration de la mesure de confiance sur le taux de reconnaissance du système, alors que dans les autres expériences, nous évaluons la pertinence des mesures elles-mêmes. Ainsi, changer de système pour cette tâche n'influe en rien l'interprétation des résultats obtenus.

Les points importants de l'environnement d'expérimentation sont les modélisations utilisées, le système de reconnaissance, et les corpus de développement et de test. Nous présentons dans un premier temps le moteur de reconnaissance qui est au cœur de notre étude. En effet, nos mesures sont fondées sur l'utilisation intensive du graphe de mots généré par le moteur de reconnaissance. Puis nous décrivons à la fois l'aspect acoustique de l'environnement (paramétrisation, modélisations phonétiques) et les modèles de langage. Ensuite, nous introduisons les corpus de développement et de test, issus de la campagne d'évaluation ESTER, que nous avons utilisés afin de nous placer dans des conditions expérimentales réelles.

4.2 Moteur de reconnaissance : Julius

Pour notre étude, nous avons besoin d'un système de reconnaissance de la parole. Plusieurs candidats sont possibles : Julius, ISIP, HTK, etc. Nous avons choisi le moteur de reconnaissance grand vocabulaire *Julius* car celui-ci présente plusieurs avantages :

- il intègre les dernières méthodes de reconnaissance communément utilisées et reconnues dans le domaine (algorithme de recherche de solution, modélisations acoustiques et linguistiques),
- il permet la gestion de grands vocabulaires,
- il est entièrement écrit en C, les fichiers sources sont disponibles gratuitement,
- il gère les modélisations acoustiques mono et triphones générées avec HTK,
- il est directement compatible avec les modèles linguistiques issus du CMU Toolkit,
- il utilise un graphe d'exploration interne de type graphe de mots,
- il est hautement paramétrable.

En outre, Rotovnik et al. [Rotovnik 02] ont mené une étude comparative entre plusieurs moteurs de reconnaissance grand vocabulaire en parole continue (HTK, ISIP et Julius) et ont

montré que Julius était le meilleur système que ce soit d'un point de vue vitesse d'exécution, consommation mémoire et précision de la reconnaissance.

Julius a été développé par des chercheurs de l'université de Kyoto et effectue la reconnaissance en deux passes [Lee 01]. Nous allons décrire la constitution de ces deux passes, le graphe de mots interne et l'introduction des facteurs d'échelle dans le calcul de la vraisemblance des mots d'une phrase.

4.2.1 La première passe de *Julius*

La première phase de reconnaissance de Julius consiste à construire un graphe d'exploration de façon trame-synchrone, correspondant au décodage de la phrase considérée. Cette première passe, qui s'effectue dans le sens normal de lecture, adopte plusieurs approximations afin d'accélérer le processus de décodage :

- un modèle de langage bigramme est utilisé au lieu du modèle trigramme,
- différentes techniques d'élagage des fonctions de densité gaussienne peuvent être sélectionnées,
- une limitation de la largeur du faisceau de recherche à un nombre maximal d'hypothèses,
- une approximation de la dépendance au contexte du mot suivant.

Le décodage repose sur l'algorithme de Viterbi décrit section 1.6.1 p.13. Le moteur de reconnaissance procède trame par trame. D'une trame à une autre, le système construit de nouvelles transitions entre les états actifs de la trame précédente et ceux possibles pour la trame courante, tout en respectant la topologie des modèles acoustiques. Si un état actif n'est pas un état terminal d'un mot, les transitions sont simplement celles d'un changement d'état intra-mot. Par contre, si l'état actif est terminal, dans ce cas les transitions vont vers le premier état de n'importe quel mot, en intégrant la probabilité du modèle de langage. L'introduction d'une factorisation unigramme ainsi que l'implantation du lexique sous forme d'arbre permet de réduire fortement le nombre de ces transitions vers tous les mots possibles. Les transitions ne se font plus que vers les premiers états partagés par les modèles des mots. Un élagage du faisceau de recherche est appliqué afin de ne retenir au maximum qu'une partie restreinte des hypothèses valides, de vraisemblance maximale. Un premier couple de facteurs d'échelle est utilisé au cours de la première passe afin d'équilibrer les contributions des probabilités acoustiques et linguistiques dans le calcul de la vraisemblance des hypothèses : une pondération du modèle de langage (δ) et une pénalité d'insertion (γ). L'équation définissant la vraisemblance d'un mot est la suivante :

$$\gamma.P(o_\tau^t|w_n).P(w_n|w_{n-1})^\delta \tag{4.1}$$

La première passe génère un graphe de mots contenant un ensemble restreint d'hypothèses parmi lesquelles s'effectuera la recherche de la solution du système de reconnaissance.

4.2.2 La deuxième passe de *Julius*

La deuxième passe de Julius est la dernière étape de la reconnaissance et délivre la solution du système à l'utilisateur. La seconde passe a la particularité de se dérouler dans le sens inverse de lecture : de la fin de la phrase vers le début. La reconnaissance se fait à partir d'un algorithme à pile de type A^* (voir section 1.6.2 p.16). La phase de recherche du meilleur chemin est fondée sur le graphe de mots interne généré au cours de la première passe.

Les vraisemblances calculées pendant la première phase ne servent que pour la fonction heuristique de l'algorithme A^*. L'information sur le prédécesseur au sens de Viterbi de chaque

mot est ignorée. Pendant la deuxième passe, les probabilités acoustiques et linguistiques sont recalculées avec des modèles plus fins, sans approximations (modèle trigramme, dépendance au contexte inter-mot totale). Un second jeu de facteurs d'échelle est utilisé, propre à cette seconde passe. Ne travailler que sur le graphe de mots permet un gain de temps considérable malgré une complexité supérieure due à l'augmentation de la précision.

Toutefois, une des contraintes d'application de l'algorithme A^* n'est pas continuellement vérifiée. En effet, la fonction heuristique peut fournir une estimation aussi bien supérieure qu'inférieure à la vraisemblance réelle.

La recherche n'est alors plus A^*-admissible. Ainsi la phrase candidate trouvée peut ne pas être la meilleure. La méthode employée dans le système de reconnaissance Julius consiste à calculer plusieurs solutions candidates en continuant la recherche d'hypothèses de phrase, puis de les trier afin d'obtenir la solution optimale. Le nombre de solutions explorées est limité dans le système à une valeur fixe, donnée par défaut mais paramétrable.

Une fois la séquence solution déterminée, un dernier réalignement de cette séquence est effectué.

4.2.3 Options de compilation

Le système de reconnaissance Julius dispose de trois modes de compilation différents permettant une reconnaissance plus ou moins précise par la sélection d'algorithmes d'élagage d'hypothèses et de simplification des calculs. Les trois modes de compilation possibles sont les suivants :

- standard : le calcul des triphones inter-mots est activé pour la deuxième passe, augmentant ainsi la précision des calculs et donc des résultats. Toutefois, il est possible d'utiliser des algorithmes d'élagage pour le calcul des probabilités d'émissions des GMM.
- fast : dans ce mode, les algorithmes d'élagage sont activés par défaut à tous les niveaux (graphe de mots interne, gaussiennes). La précision est moindre mais ceci permet au système de reconnaissance d'atteindre un temps d'exécution proche du temps réel.
- v2.1 : toutes les options d'accélération sont désactivées. Les calculs tiennent compte des liaisons inter-mots, font le moins d'hypothèses simplificatrices possibles et ne pratiquent aucun élagage des gaussiennes. Cette augmentation de la précision a une contrepartie : un temps d'exécution bien plus important.

Dans nos expérimentations, nous avons utilisé deux versions de compilation de Julius. La majeure partie d'entre elles ont été faites avec le mode *v2.1*, de précision, avec une largeur du faisceau de recherche importante : 8 000 hypothèses maximum à chaque trame. Le mode *fast* a, quant à lui, été utilisé dans les expériences d'intégration d'une mesure de confiance au sein même du moteur de reconnaissance et pour l'application concernant les élèves malentendants. La taille du faisceau de recherche a été fixée à une valeur de 1 500 pour permettre une exécution proche du temps réel.

4.2.4 Le graphe de mots

Le graphe de mots interne du moteur de reconnaissance est généré de manière trame-synchrone. Pour chaque trame t, le graphe contient l'ensemble des mots du lexique qui peuvent finir à cette trame après élagage. Pour chacun de ces mots, plusieurs informations sont accessibles :

- les instants de début et de fin du mot $[w, \tau, t]$,
- la probabilité acoustique du mot,
- un lien vers le mot $[w_p, \tau_p, \tau - 1]$ prédécesseur au sens de Viterbi de w,

– la probabilité bigramme $P(w|w_p)$,
– le score cumulé depuis le début de la phrase du meilleur chemin menant à w.

Avec la configuration que nous avons utilisée pour le système de reconnaissance *Julius* et le mode de compilation précis (v2.1), le graphe de mots contient en moyenne 470 mots hypothèses par trame avec un maximum de 2523 mots lors de la reconnaissance du corpus de développement.

4.3 Paramétrisation

La paramétrisation employée dans notre étude est fondée sur l'utilisation de Coefficients Cepstraux à échelle Mel (MFCC). Le signal sonore est échantillonné à une fréquence de 16 kHz. Nous avons utilisé une fenêtre d'analyse de 32 ms, de type Hamming, calculée toutes les 10 ms, impliquant ainsi un recouvrement des fenêtres. Le vecteur d'observation associé à la fenêtre de calcul est constitué de 13 coefficients cepstraux, incluant C_0, ainsi que les dérivées premières et secondes de ces coefficients. De plus, une soustraction de la moyenne cepstrale (Cepstral Mean Subtraction – CMS) est appliquée afin de réduire les effets dus aux microphones et aux canaux de transmission. Cette normalisation est importante du fait de la grande variété de microphones utilisés ainsi que des différents environnements pouvant intervenir dans des émissions radiophoniques (interventions extérieures, téléphoniques, studio, etc).

4.4 Modèles acoustiques

Comme nous l'avons expliqué dans l'introduction, dans la plupart de nos expérimentations nous avons utilisé des modèles monophones sauf pour deux applications pour lesquelles nous avons utilisé des modèles triphones : l'intégration de la mesure de confiance fondée sur la probabilité bigramme au sein même du moteur de reconnaissance et l'application de transcription pour des élèves malentendants.

4.4.1 Monophones

Nous utilisons un système de reconnaissance fondé sur une segmentation acoustique de la parole en phonèmes. Un ensemble de 40 phonèmes a été défini pour la langue française, auquel vient s'ajouter un modèle pour le silence.

Nous avons adopté une modélisation statistique des phonèmes à l'aide de modèles de Markov cachés à trois états munis d'une topologie gauche-droite. La probabilité d'émission de chaque état est, quant à elle, modélisée par une somme de 256 fonctions de densité gaussienne (GMM). Nous imposons que les matrices de covariance des GMM soient diagonales.

L'apprentissage des modèles acoustiques a été réalisé à l'aide des logiciels de la boîte à outils HTK [Young 94a] sur un corpus d'émissions radiophoniques transcrit d'environ 40 heures.

4.4.2 Triphones

Ces modèles permettent de prendre en compte un contexte acoustique plus important. Les modèles triphones exploitent les contextes intra- et inter-mots afin de définir un modèle en fonction des phonèmes le précédant et lui succédant. Ceci implique potentiellement la nécessité de modéliser un nombre très important de triphones (nombre de phonèmes au cube). Or, même avec un corpus très important, peu d'exemples de chaque triphone seront rencontrés, posant ainsi

un problème d'apprentissage. La technique que nous avons utilisée consiste à partager les états des modèles de Markov entre plusieurs modèles à l'aide d'un arbre de décision.

Pour les triphones, nous avons opté pour la même modélisation que pour les monophones : modèles de Markov cachés à trois états gauche-droit. La quantité moyenne d'exemples par modèle ayant diminué, la probabilité d'émission de chaque état est modélisée par une somme de 15 fonctions de densité gaussienne. La phase d'apprentissage a également été effectuée avec la boîte à outils HTK sur le même corpus transcrit.

4.5 Lexique et modèle de langage

4.5.1 Utilisés conjointement avec les modèles monophones

A partir d'un corpus textuel, nous avons défini un lexique et appris un modèle de langage. Le corpus est composé de 16 années du journal français « Le Monde », complété par la transcription manuelle de 90 heures de bulletins d'information radiophoniques en langue française. Ce corpus représente respectivement 366 millions de mots issus du journal et 1 million de mots provenant de la transcription manuelle.

Le lexique est constitué des 54 747 mots les plus fréquents de ce corpus ainsi que de trois mots spéciaux : début de phrase <s>, fin de phrase </s> et silence. Le modèle de langage nous permet de modéliser les probabilités unigrammes, bigrammes et trigrammes. Ce modèle a été appris sur le corpus textuel par l'intermédiaire du programme développé par le CMU [Clarckson 97].

Etant donné la séquence de mots $w_1 w_2$, la probabilité bigramme directe correspond à la probabilité $p(w_2|w_1)$. Au contraire, la probabilité bigramme inverse correspond à $p(w_1|w_2)$, apprise sur le même corpus mais en inversant l'ordre de parcours de celui-ci. A l'issue de la phase d'apprentissage, le modèle de langage utilisé permet d'estimer :

- 54 750 probabilités unigrammes,
- 2 × 2 541 882 probabilités bigrammes (directes et inverses),
- 2 × 5 827 217 probabilités trigrammes (directes et inverses).

De plus, à chaque probabilité unigramme et bigramme est associée une valeur de repli (*backoff*). La valeur de repli permet d'estimer la probabilité d'une séquence non rencontrée dans le corpus d'apprentissage.

4.5.2 Utilisés conjointement avec les modèles triphones

Dans le cadre de l'utilisation de modèles acoustiques triphones, nous avons défini un nouveau lexique et un nouveau modèle de langage. Le corpus est similaire au précédent, mais la proportion de émissions radiophoniques est plus élevée : 36 millions de mots.

La taille du lexique a été augmentée en passant à 59 551 graphies différentes. Au final, le lexique contient 119 133 prononciations possibles.

Le modèle de langage reste fondé sur une modélisation statistique des n-grammes. Cependant, nous n'avons pas estimé les probabilités trigrammes directes. Le modèle de langage regroupe :

- 59 552 probabilités unigrammes,
- 2 × 19 037 317 probabilités bigrammes (directes et inverses),
- 28 573 473 probabilités trigrammes (inverses seulement).

Les valeurs de repli ont également été calculées pour ce modèle de langage.

4.6 Corpus de développement et de test

Afin de pouvoir évaluer les mesures de confiance que nous avons précédemment introduites, nous devons définir un corpus de développement et un corpus de test.

Le corpus de développement permet de mettre au point les différents paramètres dont peuvent dépendre les mesures de confiance, notamment les valeurs des facteurs d'échelle et de relâchement mais également la valeur du seuil de décision. En effet, dans de nombreuses applications, un processus de décision d'acceptation ou de rejet d'une hypothèse est associé aux mesures de confiance. Il est donc nécessaire d'introduire un seuil représentant la frontière entre les mots considérés comme *corrects* et ceux considérés comme *incorrects*.

Le corpus de test permet, quant à lui, d'évaluer les mesures de confiance une fois les différents paramètres déterminés. Les corpus de test et de développement doivent être homogènes, c'est-à-dire représenter des conditions similaires.

La campagne d'évaluation ESTER visait à transposer en France des campagnes similaires menées aux Etats Unis par le NIST (National Institute of Standards and Technologies) [Gravier 04, Galliano 05, Galliano 06]. L'objectif d'ESTER était d'évaluer les performances des systèmes de transcription d'émissions radiophoniques de différents laboratoires francophones volontaires sur un même corpus. Le projet mettait à disposition des participants un corpus d'émissions radiophoniques francophones (France Info, France Inter, RFI ...). Ce corpus contient des émissions mêlant enregistrements en studio, interventions extérieures, jingles, etc. A partir des données de la campagne ESTER, nous avons constitué un corpus de développement et de test pour nos mesures de confiance.

Le corpus de développement est composé de 53 minutes de bulletins d'informations. Le corpus de test, quant à lui, est d'une durée de 56 minutes, également composé de bulletins d'informations. Ces corpus contiennent non seulement la partie studio du bulletin, mais également des interventions extérieures, etc. Par contre, les parties purement musicales et téléphoniques ont été supprimées de manière respectivement manuelle et automatique. Ces deux corpus contiennent respectivement 11071 mots (740 phrases) et 11296 mots (979 phrases). Le nombre moyen de mots par phrase est d'environ 11,5 sur le corpus de test.

Le taux d'erreur en mot (WER) obtenu par le système de reconnaissance *Julius* sur le corpus de test est de 33%.

Au vue de la taille des corpus de développement et de test, l'intervalle de confiance des valeurs que nous obtiendrons est d'environ 0,8% avec un niveau de confiance de 95%. Ceci concerne aussi bien les taux EER que les taux de fausses alarmes et de faux rejets que nous calculerons. Toutefois, cet intervalle de confiance est surestimé du fait que nous utilisons toujours le même corpus de test et de développement.

4.7 Complexité de nos mesures de confiance

Dans le chapitre précédent, au paragraphe 3.5, nous avons estimé la complexité des mesures de confiance que nous avons définies. Ces complexités ont été exprimées de manière théorique par rapport à trois quantités :

- N, le nombre moyen de mots se terminant à une trame t dans le graphe,
- T, le nombre de trames à traiter,
- V, la taille en trames du voisinage du mot analysé dans le cas des mesures locales.

Afin de donner une idée plus précise de ce que représentent ces complexités dans un cas concret, nous allons les exprimer vis-à-vis des conditions expérimentales que nous venons de décrire. Nous obtenons ainsi :

- $N = 470$,
- $T = 3270000$, la durée moyenne des deux corpus étant de 54,5 minutes,
- $V = 197$, en nous plaçant dans le cas de la mesure locale à voisinage symétrique de 84 trames et sachant que la longueur moyenne d'un mot est de 29 trames.

4.7.1 Mesures trame-synchrones

La complexité du calcul de la confiance d'un mot pour la mesure bigramme s'exprime en $O(kN^2)$ avec $k < 1$. Or sur nos corpus, nous pouvons observer une valeur de k d'environ $1/2$. Ainsi $kN^2 = \frac{1}{2}.470^2 = 110\ 450$. La complexité du calcul pour l'ensemble des mots du graphe est en $O(kTN^3)$ avec $k < 1$. Selon nos conditions expérimentales, cette quantité est donc égale à $kTN^3 = \frac{1}{2}.3270000.470^3 = 1,7.10^{14}$.

La mesure trigramme ajoutant une profondeur de recherche supplémentaire, sa complexité pour un mot est $O(kN^3)$, c'est-à-dire dans nos conditions $kN^3 = 5,2.10^7$. Pour tous les mots, la complexité est $O(kTN^4)$ avec donc $kTN^4 = 8.10^{16}$.

4.7.2 Mesures locales

Dans le cas général, la complexité du calcul de la mesure de confiance pour un mot avec les mesures à voisinage local est en $O(2VN^2)$. Dans cette estimation, nous faisons l'hypothèse que nous considérons la mesure symétrique avec un voisinage de 84 trames. Ainsi $2VN^2 = 2.197.470^2 = 8,7.10^7$. La complexité du calcul de la valeur de confiance pour chaque mot du graphe est $O(2TVN^3)$, c'est à dire $2TVN^3 = 1,3.10^{17}$.

Nous pouvons alors remarquer que bien que la mesure locale soit d'une complexité d'ordre inférieur à la mesure trigramme, les constantes impliquées dans la complexité de la mesure locale rendent ces deux mesures de complexité équivalentes, selon nos conditions expérimentales.

4.8 Conclusion

Nous avons défini nos conditions expérimentales pour une tâche réelle : le traitement d'un flux audio continu en contexte « grand vocabulaire ». Ces conditions sont également liées au système de reconnaissance pré-existant au sein de l'équipe.

Pour l'évaluation de nos mesures de confiance par le taux d'EER traitée au chapitre 5 et deux des applications présentées au chapitre 6, nous avons conservé le même système et les mêmes conditions expérimentales afin de pouvoir comparer nos mesures de confiance entre elles de façon cohérente. Les résultats que nous obtenons sont donc de fait liés à ces choix.

Pour l'expérimentation concernant les élèves malentendants traitée au chapitre 6, nous avons introduit des corpus de test spécifiques conçus à partir de tests scolaires de lecture.

Chapitre 5

Evaluation des mesures de confiance avec le taux d'EER

Sommaire

5.1 Introduction 90
5.2 Protocole d'évaluation 90
5.3 Mesure de référence – Probabilité *a posteriori* globale 91
5.4 Mesures trame-synchrones 92
5.4.1 Mesure fondée sur la probabilité unigramme 93
5.4.1.1 Gestion des occurrences multiples par sommation 94
5.4.2 Mesure fondée sur la probabilité bigramme 95
5.4.2.1 Gestion des occurrences multiples par maximisation 95
5.4.2.2 Gestion des occurrences multiples par sommation 96
5.4.2.3 Prédécesseur au sens de Viterbi 96
5.4.2.4 Filtrage par les n-meilleures phrases 97
5.4.2.5 Probabilité bigramme seule 98
5.4.2.6 Inclusion/exclusion du mot $\widehat{w}$ dans l'ensemble $\widehat{E}$ 99
5.4.2.7 Probabilité bigramme inverse 100
5.4.2.8 Homogénéisation des valeurs 100
5.4.3 Mesure fondée sur la probabilité trigramme 102
5.4.3.1 Probabilité trigramme inverse 104
5.4.4 Synthèse 105
5.5 Mesures locales 106
5.5.1 Mesure à voisinage symétrique 106
5.5.2 Mesure à voisinage asymétrique 108
5.5.3 Homogénéisation des valeurs de confiance 110
5.5.4 Synthèse 112
5.6 Influence de la taille des mots 113
5.7 Comparaison avec la mesure de confiance intégrée dans le système de reconnaissance Julius 116
5.8 Evaluation sur le corpus de test et conclusion 117

5.1 Introduction

Nous présentons dans ce chapitre les performances des différentes mesures de confiance que nous avons définies. Afin de ne privilégier ni le taux de fausses acceptations ni le taux de faux rejets, nous adoptons une évaluation fondée sur le taux d'égale erreur *EER*. D'une manière générale, nous avons mis au point les paramètres de nos mesures $(\alpha, \beta, \varepsilon, \eta)$ sur le corpus de développement puis testé ces mesures sur le corpus de test. De plus, nous avons comparé les performances obtenues par nos mesures sur nos deux corpus par rapport à une mesure de référence que nous avons donc implantée : la mesure de confiance de F. Wessel que nous appellerons *probabilité a posteriori globale*. Nous donnerons les résultats obtenus par la mesure de référence. Puis nous décrirons les expérimentations concernant nos mesures trame-synchrones et nos mesures locales.

5.2 Protocole d'évaluation

Dans ce chapitre, nous évaluons les performances de nos mesures de confiance à l'aide du taux d'égale erreur EER (cf. section 2.4.1), qui a l'avantage de rester indépendant de toute application en ne privilégiant ni le taux de faux rejets ni le taux de fausses acceptations.

Pour cela, nous devons déterminer les mots qui ont été correctement et incorrectement reconnus par le système de reconnaissance. Dans le cadre de notre étude, nous considérons que deux mots ayant la même prononciation mais deux graphies différentes sont deux mots distincts. Par exemple, le mot « petit » de « petit bateau » et le mot « petits » de « petits bateaux » sont considérés comme deux mots différents et donc si l'un est reconnu à la place de l'autre, une erreur sera comptabilisée.

Afin de déterminer si un mot est correct ou incorrect, nous comparons le résultat de la reconnaissance à un fichier de référence contenant la transcription orthographique du corpus. Les mots de chaque phrase sont analysés à la fois selon leur graphie et selon leur position dans la phrase. Ces critères étant principalement graphiques, les phrases reconnues et les phrases de référence sont normalisées :

- écriture de tous les mots en minuscules,
- suppression de toute ponctuation,
- suppression des silences et des bruits de respirations reconnus,
- remplacement des traits d'union par un espace,
- séparation d'éventuelles séquences de mots présentes dans le lexique, par exemple la séquence `Los_Angeles` devient la suite de mots `Los` et `Angeles`,
- normalisation sous une seule graphie des noms propres à l'aide du logiciel *normalize* utilisé lors de la campagne ESTER, par exemple les écritures `schroeder` et `schröder` sont normalisées sous la forme unique `schröder`.

Ensuite, un alignement entre la phrase solution et la phrase de référence, toutes deux normalisées, est effectué par programmation dynamique *via* le logiciel *Sclite* développé par le *NIST*. Ce logiciel permet d'obtenir, pour chaque phrase et même chaque locuteur, l'alignement des phrases du corpus ainsi que plusieurs quantités permettant d'évaluer la reconnaissance :

- le nombre de mots corrects : mots dont la graphie et la position dans la phrase sont identiques,
- le nombre de substitutions : mots à une même position mais dont la graphie diverge,
- le nombre d'insertions : les mots que le système a ajoutés par rapport à la référence,
- le nombre d'omissions : les mots de la référence que le système n'a pas trouvés.

Ainsi, nous pouvons étiqueter les mots reconnus par le moteur de reconnaissance en mots *corrects* ou *incorrects*. Pour calculer le taux d'EER, nous devons également les étiqueter en *Acceptation* et *Rejet* selon leur valeur de confiance et selon un seuil de décision. Le taux d'EER correspond au plus petit seuil de décision pour lequel autant d'erreurs de fausses acceptations que d'erreurs de faux rejets sont faites. Ce seuil est mis au point sur le corpus de développement.

Même si nos mesures permettent de calculer les valeurs de confiance sur tous les mots du graphe, nous ne le faisons que sur les mots figurant dans le résultat de la reconnaissance afin de determiner le taux d'EER.

5.3 Mesure de référence – Probabilité *a posteriori* globale

Parmi les mesures de confiance proposées dans la littérature, celles fondées sur l'estimation de la probabilité *a posteriori* surpassent habituellement les autres types de mesures. Nous avons décidé de choisir une mesure appartenant à cette catégorie comme référence. Plus précisément, notre référence est la mesure de confiance estimant la probabilité *a posteriori* proposée par Wessel et al., décrite section 2.3.6.2 p.43. Cette mesure est calculée à partir du graphe de mots généré à la fin de la première passe du système de reconnaissance puisque cette méthode nécessite le décodage de la phrase entière.

En plus des deux facteurs d'échelle α et β associés respectivement au modèle acoustique et au modèle linguistique, nous introduisons un facteur de flexibilité η qui nous est propre. Ce facteur est défini et utilisé de la même manière que lorsque nous l'avons introduit pour nos mesures de confiance locales (cf. section 3.3.3). Il permet de prendre en compte les occurrences quasi simultanées du même mot dans le graphe. La mesure de confiance d'un mot est alors définie comme la somme de la probabilité *a posteriori* de chacune de ces occurrences.

Avant d'évaluer cette mesure de confiance sur les corpus de développement et de test, nous avons effectué des tests préliminaires sur un corpus indépendant de 33 phrases. Ces phrases ont été extraites de bulletins d'informations radiophoniques. Les conditions d'enregistrement sont relativement bonnes, peu ou pas de musique de fond, de doubles traductions ou de parole téléphonique.

Ces 33 phrases nous ont permis d'analyser la variation des performances en terme d'EER de la mesure de référence en fonction des facteurs d'échelle α, β parmi un large faisceau de recherche. Les plages de valeurs testées sont les suivantes : $\alpha \in [0;1]$ par pas de $0,05$, $\beta \in [0;1,3]$ également par pas de $0,05$. Le facteur de flexibilité η est fixé à 0,5.

Nous avons pu ainsi définir une plage de valeurs plus réduite pour la mise au point des paramètres optimaux (α, β, η) pour cette mesure sur le corpus de développement. Les résultats de cette mise au point figurent dans le tableau 5.1

Nous pouvons remarquer que les facteurs d'échelle ont un fort impact sur le taux EER obtenu. Le passage d'un rapport des facteurs d'échelle de 1 à 10 permet un gain absolu d'environ 13% du taux d'égale erreur EER.

L'impact du facteur de flexibilité est moins important que celui des facteurs d'échelle. Cependant pour le couple ($\alpha = 0,1$), ($\beta = 1,0$), le gain est tout de même d'environ 2%. Notons qu'à partir d'un taux de flexibilité de 0,5 le taux d'égale erreur diminue marginalement. Ceci s'explique par le fait qu'à partir de 0,5, pratiquement toutes les occurrences du mot analysé ont déjà été prises en compte.

Le taux d'égale erreur optimal pour cette mesure de référence, sur le corpus de développement, est obtenu avec le jeu de paramètres ($\alpha = 0,1$), ($\beta = 1,0$), ($\eta = 1,0$) et un seuil de décision égal

TAB. 5.1 – Taux d'EER de la mesure de référence fondée sur la probabilité *a posteriori* globale calculée sur la phrase complète avec différents facteurs d'échelle et facteur de flexibilité (corpus de développement).

	rapport β/α – $(\alpha\,;\,\beta)$			
	1	2	10	20
η	(1 ;1)	(0,5 ;1)	(0,1 ;1)	(0,05 ;1)
0,2	35,6%	31,0%	23,9%	25,1%
0,5	35,8%	30,7%	22,2%	24,3%
0,7	35,9%	30,6%	22,1%	24,2%
1,0	36,0%	30,6%	**22,0%**	24,2%

à 0,625. Ce taux d'EER vaut alors 22,0%. La figure 5.1 représente la courbe DET calculée sur le corpus de développement avec le jeu de paramètres optimal pour cette mesure de confiance.

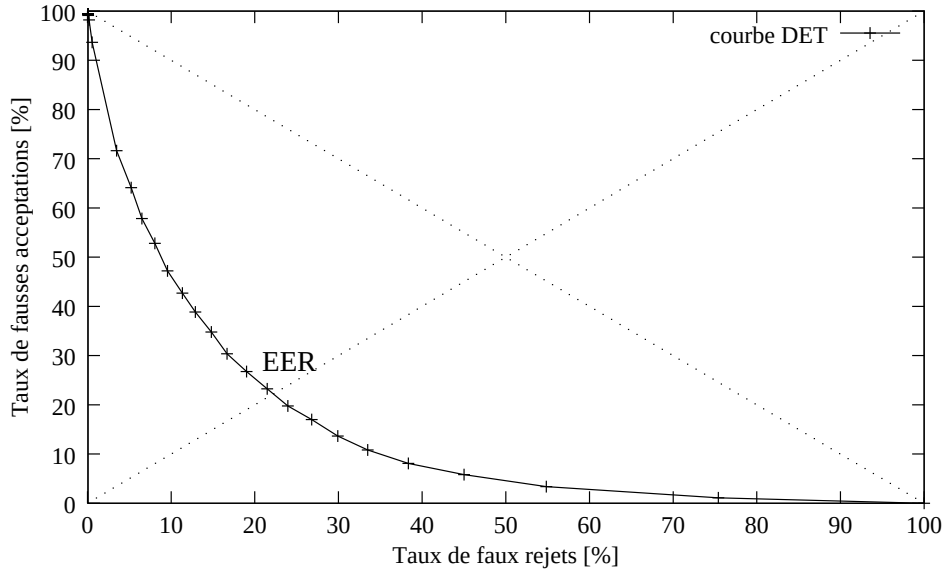


FIG. 5.1 – Courbe DET de la mesure de référence fondée sur la probabilité *a posteriori* globale $(\alpha = 0,1)$, $(\beta = 1)$ et $(\eta = 1)$. EER = 22,0% (corpus de développement).

Avec ce même ensemble de paramètres, cette mesure parvient à un taux de fausses alarmes de 24,4% et un taux de faux rejets de 21,2% sur le corpus de test.

5.4 Mesures trame-synchrones

Nous évaluons nos différentes mesures de confiance trame-synchrones d'un point de vue du taux d'égale erreur *EER*. Elles pourront ainsi être comparées entre elles ainsi qu'à la mesure de référence. Nous analysons dans l'ordre le comportement de nos mesures fondées sur les probabilités unigrammes, bigrammes et trigrammes.

5.4.1 Mesure fondée sur la probabilité unigramme

La mesure de confiance fondée sur la probabilité unigramme définie section 3.2.7.2 est une mesure assez grossière qui dispose d'informations très locales :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} p(\widehat{w})^{\beta}}{\sum\limits_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} p(w')^{\beta}}$$

Plusieurs paramètres doivent être optimisés sur le corpus de développement : les facteurs d'échelle α, β et le facteur de relâchement ε.

Nous pouvons remarquer d'après l'analyse du tableau 5.2 que l'influence des facteurs d'échelle sur les taux EER est moins importante que pour la mesure de référence. En effet, à facteur de relâchement fixé, le taux EER varie d'environ 2% sur l'ensemble des couples de facteurs d'échelle alors que la mesure de référence montre une amplitude d'environ 13%.

Nous avons introduit le facteur de relâchement afin de prendre en compte un nombre plus important de mots concurrents au mot analysé et ainsi mieux modéliser l'hypothèse alternative. Le facteur de relâchement a un impact important sur le taux d'EER (Tab. 5.2) pour cette mesure. En effet, nous pouvons constater d'une part qu'une mesure trop stricte ($\varepsilon = 0$) est moins pertinente car l'hypothèse alternative H_1 est estimée avec trop peu de modèles ; d'autre part qu'un facteur de relâchement trop important ne convient pas non plus.

Les meilleures performances sont obtenues pour un taux de relâchement de $\varepsilon = 0,1$.

Le meilleur taux EER atteint par cette mesure de confiance est de 37,6%. Il est réalisé avec les paramètres suivants : ($\alpha = 0,1$), ($\beta = 0,5$) et ($\varepsilon = 0,1$). Ce facteur de relâchement, bien qu'étant faible, permet de prendre en compte assez de diversité dans les mots concurrents du graphe sans introduire trop de mots n'ayant aucun lien de concurrence avec le mot analysé. Par exemple, pour un mot analysé de 40 trames (400 ms), un facteur de relâchement de 10% revient à considérer comme mots concurrents ceux dont la longueur est comprise entre 36 et 44 trames et dont le temps de début se situe à plus ou moins 4 trames de l'instant de début du mot analysé (idem pour le temps de fin).

Tab. 5.2 – Taux d'EER obtenus par la mesure de confiance unigramme avec différents facteurs d'échelle et de relâchement (corpus de développement).

	β/α ratio – $(\alpha;\beta)$			
	1	5	9,5	20
ε	(1 ;1)	(0,1 ;0,5)	(0,1 ;0,95)	(0,1 ;2)
0,0	41,1%	39,5%	39,6%	39,7%
0,1	39,8%	**37,6%**	38,4%	38,2%
0,2	40,6%	38,0%	38,4%	38,6%
0,3	43,0%	39,8%	39,3%	39,5%
0,5	-	43,7%	42,9%	42,1%

La figure 5.2 représente les courbes DET de la mesure de référence ainsi que celle de la mesure de confiance fondée sur la probabilité unigramme. La différence de performance entre les deux mesures se traduit par la différence de courbure entre les deux courbes. Plus la courbe se rapproche des axes et plus celle-ci est pertinente. Le point EER est en quelque sorte une façon de traduire la distance entre la courbe et les axes. Nous pouvons donc noter que la mesure unigramme a un moins bon comportement que la mesure de référence mais la mesure unigramme

est très simple, rapide à calculer et totalement trame-synchrone. En effet, la mesure unigramme ne prend en compte que les informations acoustiques et linguistiques du mot analysé et de ses concurrents.

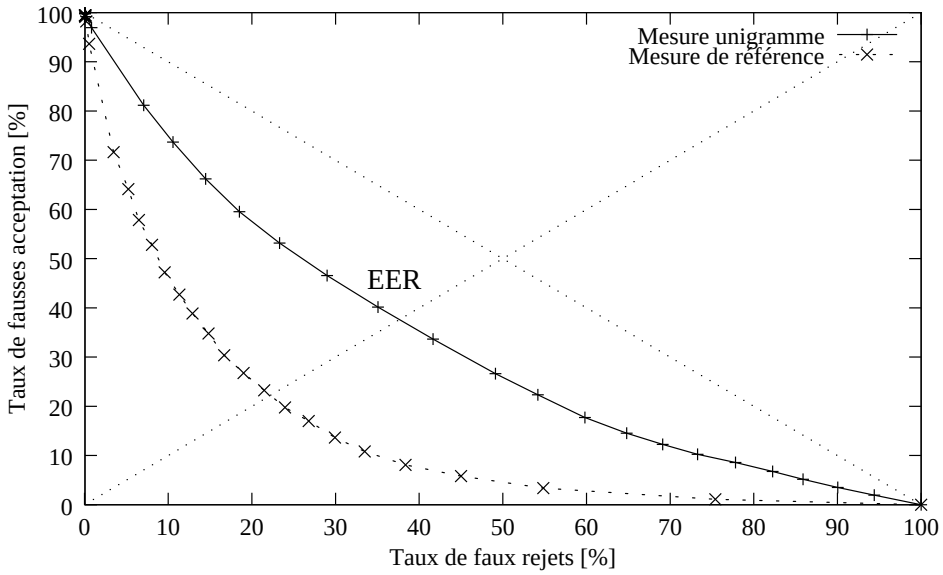


FIG. 5.2 – Courbe DET de la mesure de confiance fondée sur la probabilité unigramme ($\alpha = 0,1$), ($\beta = 0,5$) et ($\varepsilon = 0,1$). EER = 37,6% (corpus de développement).

5.4.1.1 Gestion des occurrences multiples par sommation

Le mesure de confiance fondée sur la probabilité unigramme peut être définie d'une autre manière si nous considérons la gestion des occurrences multiples par sommation, c'est-à-dire en conservant toutes les occurrences des mots de l'ensemble E. Cette mesure correspond à l'équation suivante :

$$C([w,\tau,t]) = \frac{\sum\limits_{[\widetilde{w},\tilde{\tau},\tilde{t}]\in E,\, \widetilde{w}=w} p(o_{\tilde{\tau}}^{\tilde{t}}|\widetilde{w})p(\widetilde{w})}{\sum\limits_{[w',\tau',t']\in E} p(o_{\tau'}^{t'}|w')p(w')}$$

Si nous comparons les deux méthodes de gestion des occurrences multiples, pour la mesure unigramme, nous pouvons remarquer que la méthode par maximisation présente de meilleurs résultats (cf. Tableau 5.3).

TAB. 5.3 – Taux d'EER des mesures de confiance unigramme avec gestion des occurrences multiples par maximisation et sommation avec différents facteurs d'échelle et $\varepsilon = 0,1$ (corpus de développement).

	β/α ratio – $(\alpha;\beta)$			
mode de gestion	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)
maximisation ($\widehat{E}$)	39,8%	**37,6%**	38,4%	38,2%
sommation (E)	40,1%	**38,2%**	38,5%	38,4%

5.4.2 Mesure fondée sur la probabilité bigramme

La mesure de confiance fondée sur la probabilité bigramme prend en compte, par rapport à la mesure précédente, une connaissance sur le contexte passé des mots du graphe.

Lorsque nous avons défini nos mesures bigrammes, nous avons proposé plusieurs variantes (cf. section 3.2) :

- pour la gestion des occurrences multiples :
  - par maximisation : un seul représentant est conservé parmi les occurrences d'un même mot selon le critère de probabilité acoustique maximal ;
  - par sommation : les vraisemblances de toutes occurrences sont sommées.
- pour la définition des mots précédents :
  - tous les prédécesseurs temporels directs,
  - le prédécesseur au sens de Viterbi.

De plus, nous avons introduit au cours de nos expérimentations quelques variantes supplémentaires comme par exemple l'utilisation d'une probabilité bigramme inverse. Nous allons présenter dans les paragraphes suivants les résultats concernant ces différentes variantes.

5.4.2.1 Gestion des occurrences multiples par maximisation

Pour rappel, la mesure de confiance selon la gestion des occurrences multiples par maximisation est définie par cette équation :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum\limits_{\widehat{w}_p} \left(p(\widehat{w}|\widehat{w}_p)p(\widehat{w}_p)\right)^{\beta}}{\sum\limits_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} \left(p(w'|w'_p)p(w'_p)\right)^{\beta}}$$

Afin d'étudier l'influence de la gestion par maximisation, nous devons fixer la définition des mots précédents que nous considérons. Nous avons choisi de calculer les probabilités bigrammes entre un mot et tous ses prédécesseurs temporels directs : pour le mot $\widehat{w}$, les probabilités bigrammes sont calculées entre le mot $[\widehat{w},\hat{\tau},\hat{t}]$ et tous les mots $[\widehat{w}_p,\hat{\tau}_p,\hat{t}_p]$ du graphe tels que $\hat{t}_p = \hat{\tau} - 1$.

Le tableau 5.4 récapitule les résultats obtenus par cette mesure de confiance bigramme sur le corpus de développement avec différents jeux de paramètres $(\alpha,\beta,\varepsilon)$.

Tab. 5.4 – Taux d'EER obtenus par la mesure de confiance bigramme avec gestion par maximisation pour différents facteurs d'échelle et de relâchement (corpus de développement).

	β/α ratio – $(\alpha;\beta)$			
	1	5	9,5	20
ε	(1 ;1)	(0,1 ;0,5)	(0,1 ;0,95)	(0,1 ;2)
0,0	39,7%	38,8%	38,4%	38,1%
0,1	38,9%	37,5%	**37,4%**	37,8%
0,2	40,6%	38,8%	38,6%	38,9%
0,3	42,4%	40,5%	40,3%	39,6%
0,5	45,3%	42,7%	42,0%	41,3%

Comme la mesure unigramme, cette mesure de confiance montre un impact faible des facteurs d'échelle et une influence plus importante du facteur de relâchement, de plus cette influence est

identique. Sur le corpus de développement, le meilleur résultat d'EER est de 37,4%, obtenu avec les paramètres suivants : $(\alpha = 0,1)$, $(\beta = 0,95)$ et un taux de relâchement de $(\varepsilon = 0,1)$.

Bien que les performances de cette mesure bigramme soient très légèrement meilleures que celles de la mesure unigramme, la différence entre ces deux taux n'est pas significative : 37,4% pour la mesure bigramme contre 37,6% pour la mesure unigramme.

Remarque : dans les différentes variantes présentées par la suite, lorsque nous fixons le facteur de relâchement ε à $0,1$, cela correspond à la valeur optimale de ce facteur pour la variante.

5.4.2.2 Gestion des occurrences multiples par sommation

La seconde méthode de gestion des occurrences multiples consiste à considérer toutes les occurrences des mots concurrents du mot analysé comme des mots distincts et ainsi de les prendre en compte dans le rapport. L'équation de la mesure de confiance définie selon cette méthode est la suivante :

$$C([w,\tau,t]) = \frac{\sum\limits_{[\widetilde{w},\tilde{\tau},\tilde{t}]\in E,\, \widetilde{w}=w} p(o_{\tilde{\tau}}^{\tilde{t}}|\widetilde{w})^{\alpha} \sum\limits_{\widetilde{w}_p} \left(p(\widetilde{w}|\widetilde{w}_p)p(\widetilde{w}_p)\right)^{\beta}}{\sum\limits_{[w',\tau',t']\in E} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} \left(p(w'|w'_p)p(w'_p)\right)^{\beta}}$$

Tous les éléments de l'ensemble E sont pris en compte.

Afin de pouvoir comparer cette méthode de gestion avec la méthode par maximisation, nous avons considéré, comme dans l'expérimentation précédente, que les mots précédents sont des précédents temporels directs. Le tableau 5.5 présente les taux d'EER obtenus par ces deux méthodes de gestion : par maximisation et par sommation. L'analyse est faite avec différents couples de facteurs d'échelle et à facteur de relâchement constant $\varepsilon = 0,1$.

Tab. 5.5 – Taux d'EER des mesures de confiance bigramme avec gestion des occurrences multiples par maximisation et sommation avec différents facteurs d'échelle et $\varepsilon = 0,1$ (corpus de développement).

	β/α ratio – $(\alpha;\beta)$			
mode de gestion	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)
maximisation ($\widehat{E}$)	38,9%	37,5%	**37,4%**	37,8%
sommation (E)	39,0%	38,4%	37,8%	**37,4%**

Nous pouvons remarquer que les meilleurs taux d'EER des deux mesures sont identiques donc considérer toutes les occurrences d'un mot ou prendre celle de score acoustique maximal conduit au même résultat. Dorénavant, nous ne considérerons plus que des mesures de confiance utilisant une méthode de gestion des occurrences multiples par maximisation, sur laquelle nous testerons les autres variantes.

5.4.2.3 Prédécesseur au sens de Viterbi

Dans les deux expérimentations précédentes, nous avons défini les mots précédents comme étant les précédents temporels directs. Or comme nous l'avons évoqué auparavant, nous proposons de considérer également pour les mots précédents que le précédent au sens de Viterbi.

Soit $[w, \tau, t]$ un mot pour lequel nous voulons estimer la valeur de confiance. Soit $[\widehat{w}, \hat{\tau}, \hat{t}]$ son représentant de score acoustique maximal appartenant à $\widehat{E}$. Ce représentant, comme toutes les autres occurrences de w, appartient au graphe de mots engendré par le moteur de reconnaissance. Ainsi, ce mot $[\widehat{w}, \hat{\tau}, \hat{t}]$ possède un unique prédécesseur au sens de Viterbi dans le graphe : $\widehat{w}_{pv}$. De même, chaque mot $[w', \tau', t']$ de l'ensemble $\widehat{E}$ possède un unique prédécesseur au sens de Viterbi dans le graphe : w'_{pv}. L'équation devient alors celle-ci :

$$C([w, \tau, t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha}(p(\widehat{w}|\widehat{w}_{pv})p(\widehat{w}_{pv}))^{\beta}}{\sum_{[w', \tau', t'] \in \widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha}(p(w'|w'_{pv})p(w'_{pv}))^{\beta}} \tag{5.1}$$

Tab. 5.6 – Taux d'EER des mesures de confiance bigramme avec gestion par maximisation et avec précédents temporels directs ou avec précédent au sens de Viterbi avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement).

Définition des prédécesseurs	β/α ratio – $(\alpha;\beta)$			
	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)
prédécesseurs temporels directs	38,9%	37,5%	**37,4%**	37,8%
prédécesseur Viterbi	40,7%	**40,6%**	43,0%	44,9%

Nous pouvons remarquer d'après le tableau 5.6 que prendre en compte l'ensemble des mots précédents apporte une amélioration significative par rapport à l'utilisation d'un seul bigramme sélectionné par rapport au précédent au sens de Viterbi. Ceci revient à dire que prendre en compte l'ensemble des chemins pouvant passer par le mot w considéré permet d'obtenir une mesure de confiance plus pertinente que de ne prendre que l'unique chemin déterminé au sens de Viterbi.

5.4.2.4 Filtrage par les n-meilleures phrases

En faisant abstraction de notre objectif de définition de mesures de confiance trame-synchrones, nous avons voulu évaluer notre mesure de confiance bigramme avec une méthode intermédiaire de sélection des prédécesseurs, c'est-à-dire entre celle des précédents temporels directs et celle du précédent au sens de Viterbi.

Dans l'état de l'art des mesures de confiance, les mesures fondées comme la nôtre sur un rapport de vraisemblance de modèles compétitifs (cf. section 2.3.5.5) ne considèrent souvent que la liste des n-meilleures phrases pour leur calcul [Boite 93, Rueber 97, Weintraub 97, Charlet 01].

Nous avons décidé d'utiliser la liste des n-meilleures phrases afin de filtrer les mots précédents temporels directs à un ensemble restreint de mots [Razik 05]. A partir de la liste des n-meilleures phrases générées par le système de reconnaissance, nous avons construit l'ensemble L qui contient tous les mots appartenant à cette liste. Aucune autre information que la graphie des mots n'est conservée.

Ensuite nous avons défini une mesure de confiance bigramme en ajoutant une contrainte uniquement sur les mots précédents : les mots précédents $[\widehat{w}_p, \tau_p, t_p]$ sont les précédents directs de $\widehat{w}$ ($t_p = \tau - 1$) et ces mots précédents doivent apparaître dans l'ensemble L. Respectivement, pour les mots $[w', \tau', t']$, les précédents sont les mots $[w'_p, \tau'_p, t'_p]$ du graphe tels que $t'_p = \tau' - 1$ et w'_p apparaît dans l'ensemble L. Soit F l'ensemble des mots du graphe ainsi filtrés par rapport à l'appartenance à l'ensemble L. Nous obtenons alors l'équation suivante :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum\limits_{\widehat{w}_p \in F} \left(p(\widehat{w}|\widehat{w}_p)p(\widehat{w}_p)\right)^{\beta}}{\sum\limits_{[w',\tau',t'] \in \widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p \in F} \left(p(w'|w'_p)p(w'_p)\right)^{\beta}} \tag{5.2}$$

Nous perdons le caractère trame-synchrone de notre mesure car la génération des n-meilleures phrases et la détermination de l'ensemble F nécessitent le décodage complet de la phrase par le système de reconnaissance.

Le tableau 5.7 regroupe les valeurs de taux d'EER obtenus par la mesure bigramme avec filtrage ou sans filtrage des prédécesseurs temporels directs. Les facteurs d'échelle sont fixés au couple optimal de ces mesures : $(\alpha = 0,1)$ $(\beta = 0,95)$. Nous pouvons remarquer que l'utilisation du filtrage améliore les résultats de la mesure : 37,0% avec filtrage et 37,4% sans filtrage. Prendre les mots précédents appartenant à la liste des n-meilleures phrases donne donc de meilleurs résultats, bien que non significatifs, mais au détriment de la caractéristique trame-synchrone de la mesure. La connaissance apportée par les mots des n-meilleures phrases semble affaiblie quand ces mots sont inclus parmi tous les précédents directs.

TAB. 5.7 – Taux d'EER de la mesure bigramme avec et sans filtrage des mots précédents par les n-meilleures phrases, $(\alpha = 0,1)$, $(\beta = 0,95)$ (corpus de développement).

	facteur de relâchement ε			
mesure	0,1	0,2	0,3	0,5
sans filtrage	**37,4%**	38,6%	40,3%	41,3
avec filtrage	37,6%	**37,0%**	37,5%	39,3%

Notons que dans cette expérience, nous avons autorisé le système à générer un maximum de 256 meilleures phrases et avons obtenu en moyenne 142 meilleures phrases.

5.4.2.5 Probabilité bigramme seule

Parallèlement, nous avons voulu évaluer l'impact de la probabilité bigramme dans le calcul de la vraisemblance. Nous fondons cette expérience sur la mesure bigramme calculée avec l'ensemble des prédécesseurs temporels mais également avec les uniques prédécesseurs au sens de Viterbi. Toutefois, au lieu de calculer la probabilité jointe $P(m,n) = p(m|n)p(n)$ pour deux mots quelconques m et n, nous intégrons uniquement la probabilité bigramme $p(m|n)$ sans tenir compte de la quantité relative à la probabilité unigramme.

Nous obtenons ainsi une mesure définie par l'équation suivante dans le cas des prédécesseurs temporels :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum\limits_{\widehat{w}_p} p(\widehat{w}|\widehat{w}_p)^{\beta}}{\sum\limits_{[w',\tau',t'] \in \widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} p(w'|w'_p)^{\beta}} \tag{5.3}$$

Ainsi qu'une mesure définie par cette équation dans le cas de l'unique prédécesseur au sens de Viterbi :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} p(\widehat{w}|\widehat{w}_{pv})^{\beta}}{\sum\limits_{[w',\tau',t'] \in \widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} p(w'|w'_{pv})^{\beta}} \tag{5.4}$$

Le tableau 5.8 présente les résultats du taux d'égale erreur EER pour la mesure avec tous les prédécesseurs temporels et pour la mesure bigramme avec prédécesseur au sens de Viterbi ainsi que les résultats de ces deux nouvelles mesures. Pour ces expériences, le facteur de relâchement est $\varepsilon = 0,1$, cette valeur étant la valeur optimale pour les quatres mesures. Nous pouvons remarquer que prendre en compte l'ensemble des prédécesseurs potentiels du mot analysé donne toujours de meilleurs résultats par rapport à ne prendre que le prédécesseur au sens de Viterbi, que nous tenions compte ou non de la probabilité unigramme dans le calcul de la vraisemblance. Toutefois, si pouvons également noter que quelque soit le mode gestion des prédécesseurs, ne pas tenir compte de la probabilité unigramme dans le calcul de la vraisemblance permet d'obtenir des résultats meilleurs. Le score unigramme dans le calcul de la vraisemblance semble donc pénaliser la mesure alors qu'il devrait en théorie l'aider.

Tab. 5.8 – Taux d'EER de la mesure de confiance bigramme, de la mesure bigramme seule, de la mesure bigramme avec prédécesseurs au sens de Viterbi et mesure bigramme seule avec prédécesseurs au sens de Viterbi avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement).

variante	β/α ratio – $(\alpha;\beta)$				
	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)	25 (0,1 ;2,5)
Prédécesseurs temporels	38,9%	37,5%	**37,4%**	37,8%	38,5%
Préd. temporels et bigramme seule	39,8%	38,1%	37,7%	**36,9%**	37,4%
Prédécesseur Viterbi	40,7%	**40,6%**	43,0%	44,9%	-
Viterbi et bigramme seule	40,5%	**39,1%**	40,3%	42,1%	42,6%

5.4.2.6 Inclusion/exclusion du mot $\widehat{w}$ dans l'ensemble $\widehat{E}$

Comme nous l'avons déjà dit, le mot $\widehat{w}$ que nous analysons fait également partie de l'ensemble $\widehat{E}$ contenant les mots concurrents de $\widehat{w}$. Ceci nous permet de normaliser le rapport de vraisemblance. Or dans l'état de l'art, les mesures fondées sur un rapport de vraisemblance avec des modèles compétitifs, donc similaires aux nôtres, excluent le mot analysé de l'ensemble des concurrents possibles et donc le rapport peut dépasser la valeur 1. Aussi avons-nous voulu savoir si inclure ou exclure le mot $\widehat{w}$ de l'ensemble $\widehat{E}$ avait une incidence sur les résultats de la mesure, et, dans quelle proportion.

Le tableau 5.9 regroupe les taux EER de la mesure bigramme avec prédécesseurs temporels directs incluant ou excluant $\widehat{w}$ de l'ensemble $\widehat{E}$. Nous avons évalué l'évolution du taux d'EER de ces mesures avec différents couples de facteurs d'échelle afin de déterminer les facteurs optimaux.

Tab. 5.9 – Taux d'EER des mesures de confiance bigramme, mesures incluant ou excluant $\widehat{w}$ de l'ensemble $\widehat{E}$ avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement).

modification	β/α ratio – $(\alpha;\beta)$			
	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)
$\widehat{w} \in \widehat{E}$	38,9%	37,5%	**37,4%**	37,8%
$\widehat{w} \notin \widehat{E}$	40,3%	**37,6%**	37,7%	**37,6%**

Comme pour la mesure telle que $\widehat{w} \in \widehat{E}$, le choix des facteurs d'échelle a un impact assez faible, surtout autour de la valeur optimale. Par ailleurs, la meilleure valeur atteinte par la mesure

excluant $\widehat{w}$ est légèrement inférieure à celle obtenue par la mesure incluant $\widehat{w}$. Toutefois cette différence n'est pas significative au vu de notre intervalle de confiance.

5.4.2.7 Probabilité bigramme inverse

Comme les probabilités bigrammes sont prises en compte par le moteur de reconnaissance lors de la première passe, nous avons alors évalué le comportement d'une mesure de confiance intégrant une probabilité bigramme inverse. Notre mesure devant être trame-synchrone, nous ne pouvons utiliser la probabilité bigramme inverse avec des mots dans le voisinage futur du mot analysé. Nous allons donc utiliser une probabilité bigramme inverse avec les mots dans le voisinage passé du mot analysé. L'équation de cette mesure est similaire à celle de la mesure de confiance bigramme que nous avons considérée jusque maintenant.

Cette nouvelle mesure est décrite par l'équation suivante :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum\limits_{\widehat{w}_p} \left(p(\widehat{w}_p|\widehat{w})p(\widehat{w})\right)^{\beta}}{\sum\limits_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} \left(p(w'_p|w)p(w)\right)^{\beta}} \tag{5.5}$$

Nous avons comparé la mesure de confiance fondée sur une probabilité bigramme inverse à la mesure utilisant une probabilité bigramme directe. Le tableau 5.10 regroupe les taux d'EER obtenus pas ces deux mesures de confiance par rapport à plusieurs couples de facteurs d'échelle et à un facteur de relâchement fixe. Nous pouvons remarquer que la mesure avec bigramme inverse obtient des performances légèrement meilleures, mais non significatives, que la mesure avec bigramme direct. Par ailleurs, la meilleure valeur atteinte par cette mesure indirecte l'est pour un ratio linguistique/acoustique qui fait très fortement la part belle au modèle de langage : un rapport de 20 contre un rapport de 10 pour la mesure directe. Cette amélioration est certainement due au fait que la probabilité bigramme inverse est une nouvelle information qui n'est pas utilisée par le moteur de reconnaissance lors de la première passe et donc lors de la construction du graphe de mots. Ceci laisserait penser que pour la mesure de confiance la probabilité bigramme inverse serait plus pertinente que la probabilité bigramme directe.

Tab. 5.10 – Taux d'EER des mesures de confiance fondée sur la probabilité bigramme directe et inverse avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement).

	β/α ratio – $(\alpha;\beta)$			
probabilité bigramme	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)
directe	38,9%	37,5%	**37,4%**	37,8%
inverse	38,7%	37,6%	37,3%	**37,0%**

5.4.2.8 Homogénéisation des valeurs

La caractéristique que nous voulons implicitement quand nous définissons une mesure de confiance, c'est que pour une valeur de confiance faible, le mot soit presque certainement *incorrect*, et que pour une valeur de confiance élevée, le mot soit presque certainement *correct*. Or ceci n'est pas toujours validé en pratique par les mesures que nous pouvons définir. Dans ce cas, il est possible d'effectuer une homogénéisation des valeurs afin que les valeurs finales de la mesure reflètent au mieux cette caractéristique.

La figure 5.3 présente la courbe exprimant la répartition du taux de mots corrects par intervalle de valeurs de confiance pour la mesure de confiance bigramme avec gestion par maximisation et prédécesseurs temporels. Nous avons choisi de séparer les valeurs en 20 intervalles contenant le même nombre de mots. Ce nombre d'intervalle permet d'être suffisamment précis sur l'allure de la courbe tout en effectuant un léger lissage.

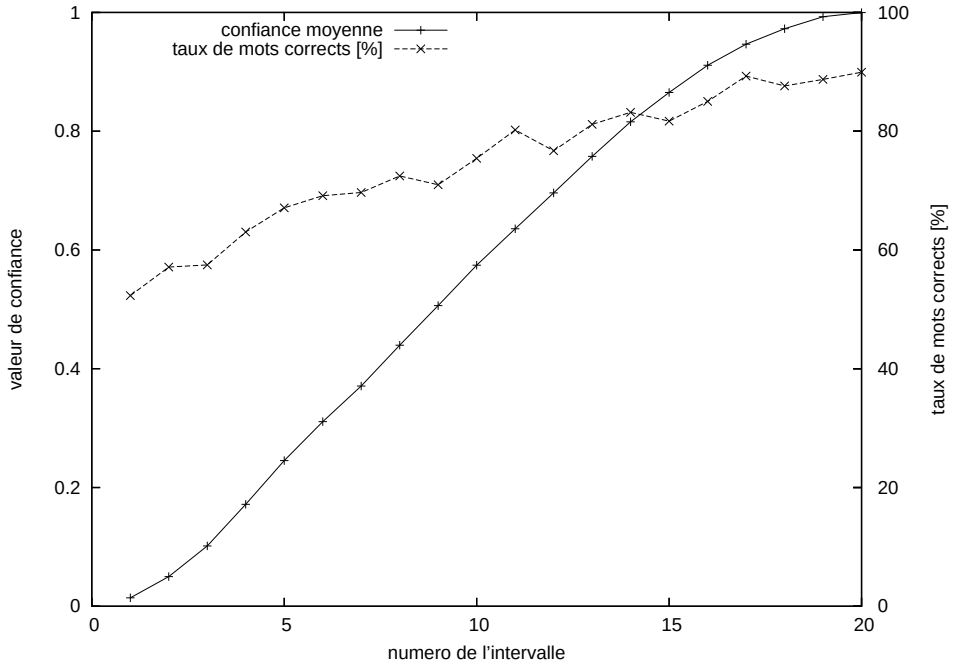


FIG. 5.3 – Distribution du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique sur le corpus de développement pour la mesure bigramme ($\alpha = 0,1$), ($\beta = 0,95$) et ($\varepsilon = 0,1$).

Nous pouvons noter que la courbe a une évolution monotone croissante, ce qui montre bien que notre mesure de confiance est porteuse d'une information utile. Toutefois, cette figure montre l'intérêt d'homogénéiser les valeurs de confiance de la mesure de départ. Pour les faibles valeurs de la mesure de confiance, nous avons en fait un peu plus de 50% de chance que le mot soit correct. En revanche, la mesure de confiance est plus pertinente pour les valeurs proches de 1. En effet, les mots ayant une valeur de confiance supérieure à 0,9 sont corrects environ 9 fois sur 10.

Homogénéiser les valeurs de la mesure de confiance et les corriger pourrait donc avoir un impact positif sur les performances de la mesure. Nous avons donc défini une nouvelle mesure de confiance par tabulation. A chaque valeur de confiance initiale correspond un intervalle de la courbe et donc également un taux de mots corrects. La mesure de confiance sera alors définie comme la valeur du taux de mots corrects associé. Nous avons évalué le taux d'EER de cette nouvelle mesure par rapport à la mesure de confiance initiale. Les résultats de cette nouvelle mesure de confiance sont légèrement meilleurs que ceux de la mesure bigramme (Tab. 5.11). Toutefois, la différence entre les deux meilleures valeurs reste faible : 37,2% pour la mesure corrigée contre 37,4% pour la mesure initiale. Au vu de l'intervalle de confiance de nos expériences, nous pouvons considérer que cette différence n'est pas significative.

Tab. 5.11 – Taux d'EER des mesures de confiance bigramme avec gestion par maximisation et tous les précédents temporels directs, avec et sans homogénéisation des valeurs de confiance avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement).

	β/α ratio – $(\alpha;\beta)$			
mesure	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)
directe	38,9%	37,5%	**37,4%**	37,8%
homogénéisée	38,8%	37,5%	**37,2%**	37,3%

5.4.3 Mesure fondée sur la probabilité trigramme

Nous évaluons à l'aide de cette mesure de confiance l'augmentation de la couverture du contexte passé des mots. Pour cela nous utilisons les probabilités trigrammes. Ces mesures de confiance ayant la contrainte d'être trame-synchrones, nous ne pouvons prendre en compte que le contexte gauche (passé) car lui seul existe au moment du traitement d'une trame t par le moteur de reconnaissance. Nous considérons dans cette expérimentation la méthode par maximisation ainsi que tous les précédents temporels directs. Pour rappel, cette mesure de confiance trigramme est définie par l'équation suivante :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum_{\widehat{w}_p} \sum_{\widehat{w}_{pp}} \left(p(\widehat{w}|\widehat{w}_p\widehat{w}_{pp})p(\widehat{w}_p|\widehat{w}_{pp})p(\widehat{w}_{pp})\right)^{\beta}}{\sum_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum_{w'_p} \sum_{w'_{pp}} \left(p(w'|w'_pw'_{pp})p(w'_p|w'_{pp})p(w'_{pp})\right)^{\beta}}$$

Les paramètres de cette mesure ont été optimisés sur le corpus de développement. La courbe représentée figure 5.4 montre l'évolution du taux d'EER de cette mesure en fonction du rapport entre le facteur d'échelle linguistique et acoustique. La valeur du facteur d'échelle acoustique α est fixée à $0,1$ et le facteur de relâchement $\varepsilon = 0,1$. Le meilleur taux d'EER est atteint pour la valeur du facteur d'échelle linguistique $\beta = 0,7$ et donc un rapport 7. Ce taux de 37,0% est le meilleur taux obtenu par cette mesure.

Ce résultat est identique à celui obtenu par la mesure de confiance bigramme intégrant les probabilités bigrammes inverses. Ces deux mesures ont un point commun, elles utilisent des informations que le moteur de reconnaissance ne prend pas en compte lors du processus de décodage de la première passe : probabilité bigramme inverse et probabilité trigramme. Bien que la différence entre les taux obtenus par la mesure trigramme et par la mesure bigramme équivalente ne soit pas significative au vu de l'intervalle de confiance, l'information supplémentaire apportée par la probabilité trigramme permet d'améliorer légèrement les performances de la mesure (0,4%).

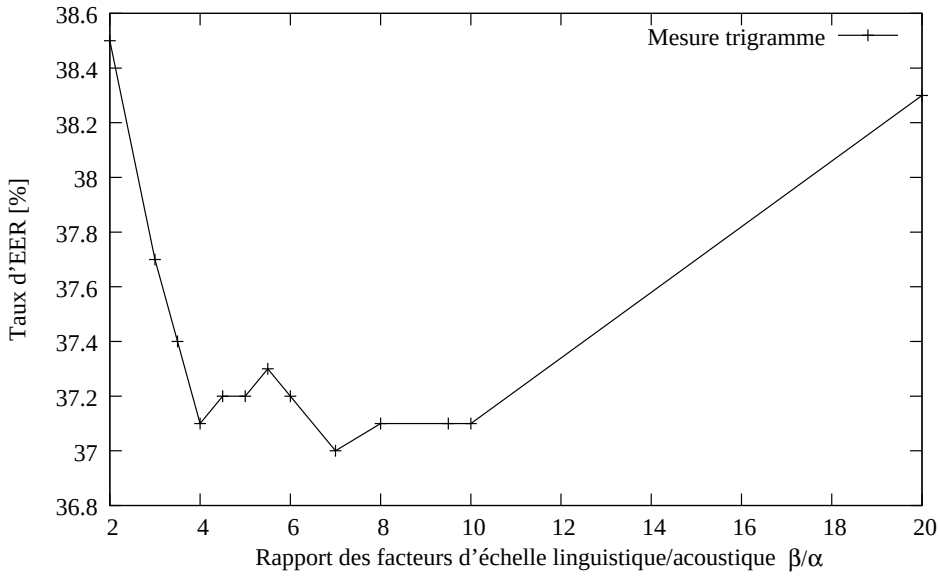


FIG. 5.4 – Variation du taux d'EER de la mesure de confiance fondée sur la probabilité trigramme, en fonction du rapport des facteurs d'échelle linguistique et acoustique β/α ($\alpha = 0,1$ et $\varepsilon = 0,1$).

Afin d'analyser l'influence de la probabilité trigramme, qui met en jeu de nombreux éléments du graphe de mots, nous avons modifié la définition de cette mesure de confiance afin de ne prendre en compte que la probabilité trigramme seule, sans probabilité bigramme et unigramme, et considérant soit les prédécesseurs temporels soit les précédents au sens de Viterbi.

L'équation définissant la mesure trigramme est alors ainsi modifiée :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum_{\widehat{w}_p} \sum_{\widehat{w}_{pp}} p(\widehat{w}|\widehat{w}_p\widehat{w}_{pp})^{\beta}}{\sum_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum_{w'_p} \sum_{w'_{pp}} p(w'|w'_p w'_{pp})^{\beta}} \tag{5.6}$$

Dans le cas des précédents au sens de Viterbi, nous ne prenons plus en compte que le triplet $\widehat{w}_{ppv}, \widehat{w}_{pv}, \widehat{w}$ appartenant au chemin du graphe au sens de Viterbi menant au mot $\widehat{w}$. $\widehat{w}_{pv}$ est l'unique prédécesseur de $\widehat{w}$ sur ce chemin et $\widehat{w}_{ppv}$ est également l'unique prédécesseur de $\widehat{w}_{pv}$ sur le même chemin, toujours selon l'algorithme de Viterbi. L'équation définissant cette modification est la suivante :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} p(\widehat{w}|\widehat{w}_{pv}\widehat{w}_{ppv})^{\beta}}{\sum_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} p(w'|w'_{pv}w'_{ppv})^{\beta}} \tag{5.7}$$

Le tableau 5.12 montre les résultats obtenus par ces deux mesures de confiance trigrammes excluant les probabilités bigrammes et unigrammes ainsi que la mesure trigramme sans ces modifications. Le choix du meilleur triplet de mots au sens de Viterbi (précédent et précédent du précédent) dégrade significativement les performances des mesures trigrammes prenant en compte les prédécesseurs temporels. Comme pour la mesure bigramme, limiter le choix des précédents à ceux du chemin calculé par l'algorithme de Viterbi réduit de façon trop importante la variabilité des hypothèses.

Une nouvelle fois dans le cas trigramme, nous pouvons remarquer qu'exclure du calcul de la mesure les probabilités d'ordre inférieur, bigramme et unigramme, améliore légèrement les performances de la mesure. Ces probabilités semblent donc également perturber la mesure alors qu'elles devraient au contraire théoriquement l'aider.

TAB. 5.12 – Taux d'EER de comparaison de la mesure de confiance fondée sur la probabilité trigramme et de sa version modifiée, $\varepsilon = 0,1$ (corpus de développement).

	β/α ratio – $(\alpha;\beta)$				
méthode	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)	25 (0,1 ;2,5)
Prédécesseurs temporels (Eq. 3.13)	38,7%	37,2%	**37,1%**	38,3%	38,4%
Préd. temporels et trigramme seule	39,5%	37,8%	37,4%	**36,7%**	36,9%
Prédecesseurs Viterbi	**43,2%**	43,9%	45,1%	-	-
Viterbi et trigramme seule (Eq. 5.7)	40,6%	**39,1%**	40,4%	41,9%	39,2%

5.4.3.1 Probabilité trigramme inverse

Comme pour les mesures bigrammes, nous avons évalué le comportement d'une mesure de confiance intégrant une probabilité trigramme inverse. Toutefois, contrairement au cas bigramme, le moteur de reconnaissance ne prend en compte aucune information de type trigramme lors de la première passe, que ce soit une probabilité directe ou inverse. De même que pour la mesure bigramme, notre mesure devant être trame-synchrone, nous ne pouvons utiliser la probabilité trigramme inverse avec des mots dans le voisinage futur. Nous allons donc utiliser une probabilité trigramme inverse avec les mots dans le voisinage passé du mot analysé. L'équation de cette mesure est similaire à celle de la mesure de confiance trigramme que nous avons considérée jusqu'à maintenant.

Cette nouvelle mesure est décrite par l'équation suivante :

$$C([w,\tau,t]) = \frac{p(o_{\hat{\tau}}^{\hat{t}}|\widehat{w})^{\alpha} \sum\limits_{\widehat{w}_p} \sum\limits_{\widehat{w}_{pp}} \left(p(\widehat{w}_{pp}|\widehat{w}_p\widehat{w})p(\widehat{w}_p|\widehat{w})p(\widehat{w})\right)^{\beta}}{\sum\limits_{[w',\tau',t']\in\widehat{E}} p(o_{\tau'}^{t'}|w')^{\alpha} \sum\limits_{w'_p} \sum\limits_{w'_{pp}} \left(p(w'_{pp}|w'_pw')p(w'_p|w')p(w')\right)^{\beta}} \tag{5.8}$$

Nous avons comparé la mesure de confiance fondée sur une probabilité trigramme inverse à la mesure utilisant une probabilité trigramme directe. Le tableau 5.13 regroupe les taux d'EER obtenus pas ces deux mesures de confiance par rapport à plusieurs couples de facteurs d'échelle et à un facteur de relâchement fixe.

Nous pouvons remarquer que la mesure avec trigramme directe obtient des performances légèrement meilleures, mais non significatives, que la mesure avec trigramme inverse. Ainsi dans ce cas, l'information issue de la probabilité inverse n'aide pas la mesure de confiance. La différence

de comportement observée entre le cas bigramme et le cas trigramme vient certainement du fait que pour le cas bigramme, l'information du modèle de langage directe est intrinsèquement présente dans la construction du graphe et se retrouve ainsi dans l'ensemble des mots compétitifs. Intégrer alors une information sur les probabilités bigrammes inverses devient un indice supplémentaire. Au contraire, dans le cas trigramme, les probabilités directes ou inverses apportent une information similaire. La mesure avec trigramme inverse obtiendrait sans doute de meilleures performances que la mesure avec trigramme directe si le graphe de mots était généré en utilisant des probabilités trigrammes directes.

TAB. 5.13 – Taux d'EER des mesures de confiance fondée sur la probabilité trigramme directe et inverse avec différents facteurs d'échelle, $\varepsilon = 0,1$ (corpus de développement).

	β/α ratio – $(\alpha;\beta)$				
probabilité trigramme	1 (1 ;1)	5 (0,1 ;0,5)	9,5 (0,1 ;0,95)	20 (0,1 ;2)	25 (0,1 ;2,5)
directe	38,7%	37,2%	**37,1%**	38,3%	38,4%
inverse	40,1%	**38,2%**	38,5%	38,4%	38,5%

5.4.4 Synthèse

Nos mesures de confiance trame-synchrones, fondées sur un rapport de vraisemblance, se distinguent principalement par le degré du modèle de langage utilisé : unigramme, bigramme et trigramme. Afin d'être trame-synchrones, ces mesures n'utilisent que des informations du voisinage passé du mot analysé. Nous avons évalué chacune de ces mesures avec pour certaines quelques variantes, principalement pour la mesure bigramme. Ces variantes concernent la gestion des mots concurrents (maximisation, sommation), la sélection des mots précédents (temporels directs, Viterbi, filtrage n-meilleures phrases) mais aussi l'homogénéisation des valeurs de confiance, l'utilisation de la probabilité bigramme inverse ou l'utilisation de la probabilité bigramme seule.

Pour chacune des mesures nous avons optimisé les paramètres suivant : les facteurs d'échelle, α et β, et le facteur de relâchement ε. La valeur optimale du facteur de relâchement ε est 10% dans la majorité de nos expérimentations. Ce facteur permet de prendre en compte plus ou moins d'occurrences de mots concurrents dans le graphe pour le rapport de vraisemblance. Le taux d'EER est croissant suivant l'augmentation du facteur de relâchement car trop de mots sans rapport avec le mot analysé sont pris en compte pour modéliser l'hypothèse alternative du rapport de vraisemblance. Au contraire, un facteur de relâchement nul ne permet pas une diversité des mots concurrents suffisante pour donner une mesure précise. Par ailleurs, le rapport entre les facteurs d'échelle linguistique et acoustique optimaux vaut souvent 10, indiquant que l'information linguistique doit être favorisée par rapport au score acoustique.

A partir de ces expérimentations, nous pouvons remarquer que plus la portée du voisinage passé pris en compte par les mesures est importante, meilleures sont ses performances. En effet, les résultats des mesures de confiance utilisant les mêmes méthodes de gestion, les mêmes définitions des précédents et les probabilités n-grammes directes, montrent que la mesure fondée sur la probabilité trigramme est légèrement meilleure que la mesure bigramme, eux-mêmes légèrement meilleures que la mesure unigramme (respectivement 37,0%, 37,4% et 37,6%). Toutefois, nous espérions un gain de performance plus significatif lors du passage d'un modèle bigramme à un modèle trigramme.

Ces différents taux d'EER sont moins bons que le meilleur taux obtenu par la mesure de référence : 22,0% Les méthodes de calcul de ces mesures trame-synchrones et de la mesure de

référence sont très différentes. Les mesures trame-synchrones sont simples et n'utilisent qu'une vision très locale du signal alors que la mesure de référence utilise les informations du décodage de l'intégralité du signal. Il est donc normal que la mesure de référence soit meilleure que nos mesures trame-synchrones.

Concernant la gestion des occurrences multiples, avec des facteurs d'échelle et de relâchement optimaux, la méthode par maximisation et la méthode par sommation sont équivalentes. Les deux mesures bigrammes ne se différenciant que par cette gestion obtiennent le même taux d'EER : 37,4%.

Les expériences concernant l'homogénéisation des valeurs de confiance ont permis de montrer qu'il y une bonne corrélation entre les valeurs de confiance calculées par nos mesures et le fait qu'un mot soit correct ou incorrect. En effet, la courbe représentant le taux de mots corrects en fonction de la valeur de confiance est monotone croissante. C'est-à-dire que lorsque la valeur de confiance augmente, la proportion de mots corrects augmente également. Par exemple, pour la mesure de confiance bigramme avec maximisation et précédents temporels directs, un mot dont la valeur de confiance vaut 1 est correct de manière presque certaine.

Nous remarquons également que parmi les variantes évaluées, les mesures sélectionnant les mots précédents uniquement au sens de Viterbi sont significativement les plus mauvaises. Les meilleurs taux d'EER de ces mesures sont en moyenne moins bons d'environ 2% en absolu par rapport aux autres mesures trame-synchrones. Nous pensons que les mesures n'utilisant que les précédents au sens de Viterbi se rapprochent trop du processus de reconnaissance et peuvent donc difficilement donner une décision différente de celle du système.

Les mesures obtenant les meilleures performances sont justement celles intégrant une connaissance qui n'est pas présente ou pas utilisée par le moteur de reconnaissance lors de la première passe. Plus précisément, la mesure trigramme et la mesure bigramme inverse. Ces deux mesures obtiennent le même taux d'EER sur le corpus de développement : 37,0%.

5.5 Mesures locales

Nous allons maintenant évaluer les mesures locales que nous avons proposées. Ces mesures ne sont pas trame-synchrones mais ont quand même une vision locale du signal. Nous avons introduit deux variantes pour ces mesures de confiance, dépendantes de la définition du voisinage locale utilisé : les mesures à voisinage symétrique et les mesures à voisinage asymétrique (voir section 3.3).

5.5.1 Mesure à voisinage symétrique

Les mesures de confiance locales que nous avons définies dans la section 3.3 sont fondées sur une estimation de la probabilité *a posteriori* sur une partie restreinte du graphe de mots. La mesure à voisinage symétrique consiste à définir un intervalle centré sur un mot w, à extraire du graphe de mots le sous-graphe correspondant à ce voisinage puis à calculer la mesure de confiance de w. La taille du voisinage est définie par le nombre de trames prises en compte de chaque côté du mot. Par exemple, pour un mot w d'une durée de 30 trames et une mesure de confiance symétrique de paramètre 84, la longueur du sous-graphe sur lequel sera calculée la mesure sera de $84 + 30 + 84 = 198$ trames.

En plus des facteurs d'échelle (α et β) nous avons introduit un facteur de flexibilité η. Nous avons tout d'abord mis au point cet ensemble de paramètres $(\alpha; \beta; \eta)$ sur le corpus de développement. Pour cela nous avons fixé la taille du voisinage à 84 trames. Cette valeur correspond à

la longueur moyenne en trames des séquences de deux mots consécutifs sur le corpus de développement. Pour information, la longueur moyenne d'un mot sur le corpus de développement est de 32 trames et une trame correspond à 10 ms.

Comme pour la mesure de référence, fondée sur la probabilité *a posteriori* globale et calculée sur le graphe de mots complet, le taux d'EER se stabilise à partir d'un facteur de relâchement de 0,5 (cf. tableau 5.14). Le meilleur couple de facteurs d'échelle est $(\alpha = 0,1)$, $(\beta = 0,95)$. Les facteurs d'échelle optimaux sont donc très proches de ceux de la mesure de référence $(\alpha = 0,1)$ et $(\beta = 1)$. Par ailleurs, les performances de notre mesure locale évoluent de façon similaire à celles de la mesure de référence quand nous faisons varier les facteurs d'échelle (cf. Tab. 5.1). Le meilleur taux d'EER de notre mesure symétrique est de 23,0% et le meilleur taux d'EER de la mesure de référence est de 22,0%.

Tab. 5.14 – Taux d'EER obtenus par la mesure de confiance locale fondée sur la probabilité *a posteriori* avec un voisinage symétrique de 84 trames, pour différents facteurs d'échelle et de relâchement (corpus de développement).

	β/α ratio – $(\alpha;\beta)$		
	1	9,5	20
η	(1 ;1)	(0,1 ;0,95)	(0,1 ;2)
0,1	36,3%	28,5%	28,8%
0,2	35,7%	24,9%	27,5%
0,3	35,8%	23,8%	27,1%
0,5	35,7%	**23,0%**	26,9%
0,7	35,7%	**23,0%**	26,7%

Nous avons évalué ensuite l'influence de la taille du voisinage sur le taux d'égale erreur obtenu par la mesure de confiance symétrique pour un jeu de paramètres (α, β, η) fixé. Le nombre de trames ajoutées de chaque côté du mot analysé varie entre 20 et 200 trames. Nous pouvons remarquer que le taux d'EER diminue fortement jusqu'à 84 trames (cf. Fig. 5.5). A partir de 84 trames, la diminution est plus lente.

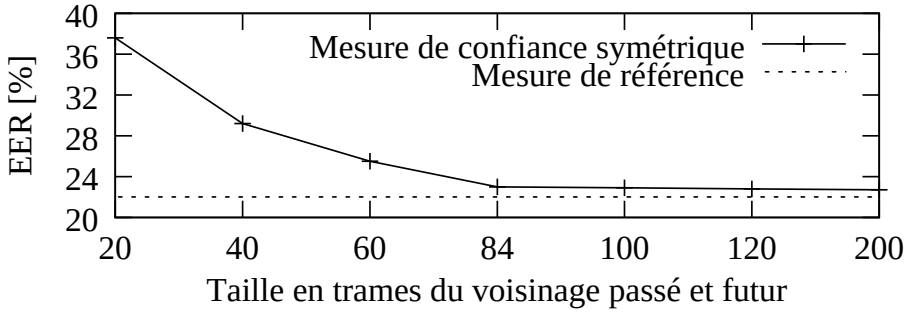


Fig. 5.5 – Courbe du taux d'EER de la mesure locale à voisinage symétrique relativement à différentes tailles de voisinage. $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\eta = 0,5)$

Pour cette mesure locale, avec un voisinage symétrique de 84 trames, le taux d'EER optimal est obtenu pour le triplet $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\eta = 0,5)$. Ce taux est très proche de celui

de la mesure de référence : 23,0% pour cette mesure contre 22,0% pour la mesure de référence. Ceci se traduit sur le corpus de test par un taux FR et FA respectivement de 23,7% et 24,4%, eux également proches des taux obtenus par la mesure de référence (21,2% de FR et 24,4% de FA).

Cette mesure locale avec un voisinage de 84 trames correspond en moyenne au calcul de la probabilité *a posteriori* sur un graphe de cinq mots. Or cette mesure obtient des performances très bonnes et très proches de la mesure de référence, qui par contre exige la phrase complète donc en général plus de cinq mots. Ainsi, la mesure de confiance locale à voisinage symétrique, bien que non trame-synchrone, peut être utile dans le cadre d'applications telles que la transcription d'émissions de télévision en direct. En effet, comme nous l'avons précédemment mentionné, un décalage est introduit pour ces émissions entre la réalisation et la diffusion. Ce décalage de l'ordre de quelques secondes est largement supérieur aux 84 trames de notre mesure de confiance locale (84 trames représentent 840 ms).

Par ailleurs, cette expérience montre qu'il est possible d'obtenir une bonne estimation de la probabilité *a posteriori* d'un mot sans avoir besoin de l'intégralité de la phrase mais seulement un voisinage du mot analysé.

5.5.2 Mesure à voisinage asymétrique

Pour la mesure précédente, la taille des voisinages passé et futur était identique de part et d'autre du mot w. Dans le cas de la mesure à voisinage asymétrique, ces deux tailles sont définies indépendamment l'une de l'autre. Le but est de pouvoir prendre en compte les informations passées, générées par le moteur de reconnaissance et donc disponibles, afin d'estimer une valeur de confiance sans augmenter la taille du voisinage futur, c'est-à-dire le délai.

Nous évaluons le comportement de cette mesure de confiance avec plusieurs valeurs de la taille du voisinage futur au mot : 0, 40, 60 et 84 trames. Pour chacun des ces voisinages, nous faisons varier le voisinage passé d'une taille minimale de 40 trames jusqu'à prendre en compte tout le graphe depuis le début de la phrase. Le triplet de paramètres $(\alpha\,;\,\beta\,;\,\eta)$ a été mis au point sur le corpus de développement et est identique à celui de la mesure de confiance à voisinage symétrique : $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\eta = 0,5)$.

Les quatre courbes de la figure 5.6 présentent les résultats de cette étude. Chacune correspond à une valeur fixe de la taille du voisinage futur. Cette évolution est fonction de la taille du voisinage passé : du début de la phrase jusqu'à 40 trames.

Nous pouvons remarquer que les quatre courbes ont le même comportement, et que, plus nous prenons en compte d'information, meilleurs sont les résultats. Avec un voisinage futur de 84 trames et un voisinage passé couvrant toutes les informations depuis le début de la phrase, nous obtenons un résultat quasi identique (22,3%) à celui de la mesure de référence qui est, elle, fondée sur l'utilisation de la phrase entière (22,0% pour $\alpha = 0,1$, $\beta = 1$ et $\eta = 1$). Avec un voisinage futur réduit à 60 trames, les taux d'EER obtenus sont proches de ceux de la mesure avec 84 trames : 23,2% et 22,2% ($\alpha = 0,1$, $\beta = 0,95$ et $\eta = 0,5$).

Les taux d'EER des mesures pour un voisinage passé commençant depuis le début de la phrase sont 30,0%, 25,8%, 23,2% et 22,3% respectivement pour un voisinage futur de 0, 40, 60 et 84 trames.

Ces mesures de confiance asymétriques permettent, sans avoir à extraire l'intégralité du graphe de mots associé à la phrase, d'obtenir des performances identiques à la mesure de référence. Les informations passées sont disponibles, même depuis le début de la phrase, et seul un court voisinage futur est nécessaire (60 ou 84 ms). Nous pouvons donc employer cette mesure, comme la précédente, pour des transcriptions d'émissions en direct.

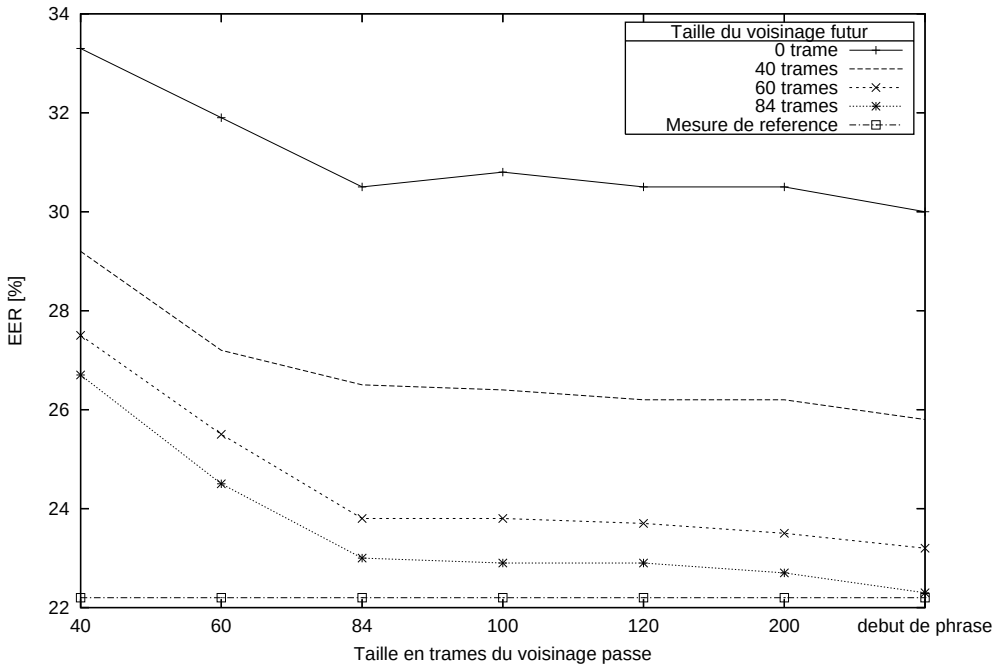


FIG. 5.6 – Taux d'EER de la mesure de confiance locale à voisinage asymétrique à taille de voisinage passé variable et taille de voisinage futur fixe (0, 40, 60, et 84 trames). Les facteurs d'échelle et de relâchement sont également fixes, $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\eta = 0,5)$ et $(\alpha = 0,1)$, $(\beta = 1)$ et $(\eta = 1)$ pour la mesure de référence.

Par ailleurs, nous pouvons remarquer que si nous fixons la taille du voisinage futur à 0, nous obtenons ainsi une mesure trame-synchrone. Si nous considérons un voisinage passé prenant en compte le graphe de mots depuis le début de la phrase, le taux d'EER de la mesure est de 30,0%. Ce qui est significativement meilleur que nos meilleures mesures trame-synchrones (37,0% pour la mesure bigramme inverse et la mesure trigramme).

Si nous comparons nos mesures trame-synchrones avec une mesure locale asymétrique calculée sur une taille de voisinage équivalente, c'est-à-dire un voisinage passé de 40 trames et un voisinage futur nul, le taux d'EER obtenu est de 33,3%.

Les résultats obtenus par cette mesure asymétrique en mode trame-synchrone explique les performances mitigées observées pour les mesures trame-synchrones fondées sur le rapport de vraisemblance (seulement 37,0%). En effet, pour les mesures trame-synchrones, le rapport de vraisemblance est calculé entre des bigrammes, c'est-à-dire des séquences d'exactement deux mots (ou trois pour nos mesures trigrammes). Or, les mesures fondées sur les probabilités *a posteriori* considèrent des chemins entre des séquences de longueur quelconque en nombre de mots mais appartenant à un voisinage de taille en trames fixée. En effet, même pour un voisinage de taille équivalente à la longueur d'une séquence bigramme, le graphe de mots associé ne contient pas uniquement des chemins contenant deux mots. De plus la probabilité *a posteriori* prend en compte plus d'informations (probabilités acoustiques des mots précédents). Ainsi, il est normal que la mesure fondée sur la probabilité *a posteriori* soit plus précise que le rapport de vraisemblance.

Notons toutefois que d'un point de vue combinatoire, les mesures locales sont plus complexes que la mesure fondée sur le rapport de vraisemblance utilisant les probabilités bigrammes.

5.5.3 Homogénéisation des valeurs de confiance

Nous allons analyser maintenant la répartition du taux de mots corrects par rapport à la valeur de confiance de mesures locales que nous avons définies.

La figure 5.7 représente cette répartition pour la mesure symétrique avec une taille de voisinage de 84 trames avec comme facteurs d'échelle et facteur de flexibilité $\alpha = 0,1$, $\beta = 0,95$ et $\eta = 0,5$. La courbe du taux de mots corrects montre un comportement très proche de l'évolution des valeurs de confiance. Ceci signifie qu'il y a bien une corrélation entre la valeur de confiance d'un mot et le fait qu'il soit réellement correct ou incorrect : un mot de faible confiance est presque sûrement incorrect et un mot ayant une forte valeur de confiance est correct de manière quasi certaine.

La figure 5.8 représente cette répartition pour la mesure locale asymétrique trame-synchrone, c'est-à-dire prenant en compte toutes les informations du graphe depuis le début de la phrase mais sans prendre en compte le voisinage futur. Les mêmes facteurs d'échelle et de flexibilité que pour la mesure locale précédente ont été utilisés : $\alpha = 0,1$, $\beta = 0,95$ et $\eta = 0,5$. Comparativement à la courbe précédente, cette mesure est moins précise pour les valeurs de confiance faible et moyenne. Toutefois, son comportement montre que cette mesure capte une information de corrélation entre la valeur de confiance et le fait qu'un mot soit correct.

Si nous comparons ces deux courbes, nous remarquons que bien que pour la deuxième courbe, la taille du voisinage passé soit bien supérieure (tout le graphe depuis le début de la phrase) la mesure est moins pertinente. Or, la deuxième différence entre ces deux mesures concerne la taille du voisinage futur. Pour la mesure locale symétrique, un voisinage futur est défini et vaut 84 trames, pour la mesure asymétrique, il n'y a aucun voisinage futur, la mesure est trame-synchrone. Nous pouvons alors conclure que la taille du voisinage passé est importante mais que ce voisinage ne peut pas rattraper l'information relative au voisinage futur. Ainsi, s'il est possible d'intégrer des informations du voisinage futur des mots considérés, le gain en pertinence pourrait

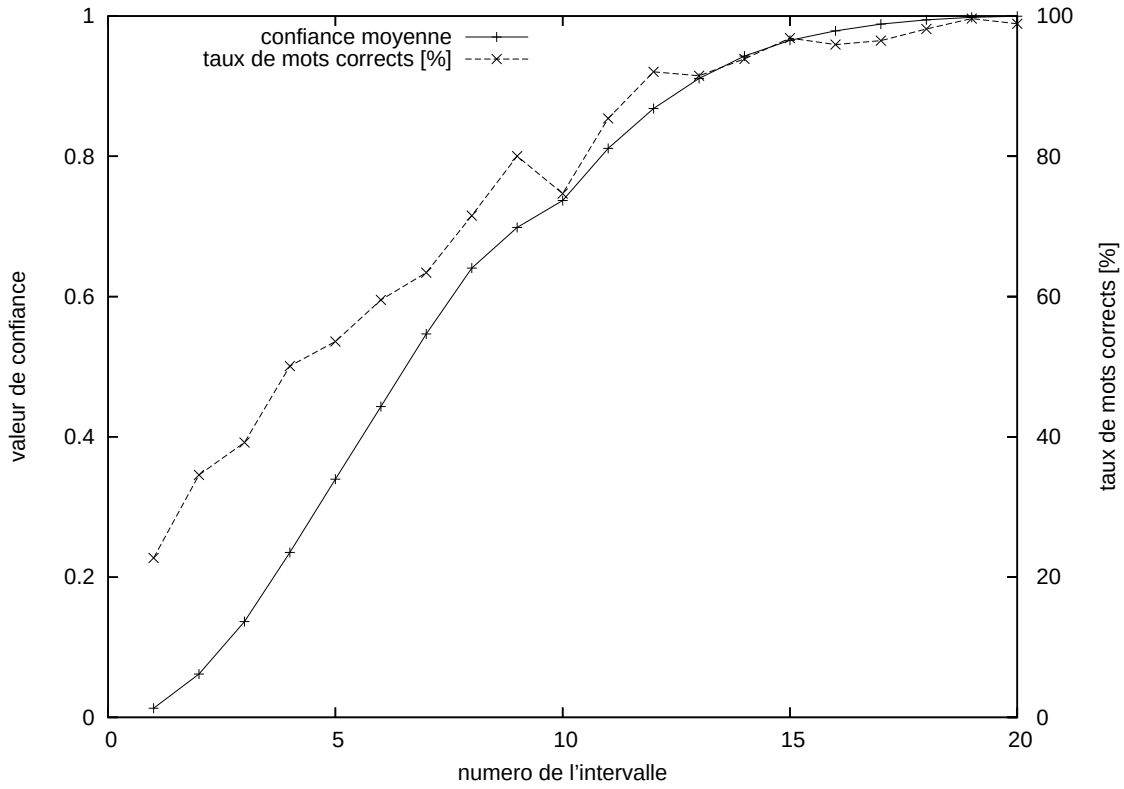


FIG. 5.7 – Répartition du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique sur le corpus de développement pour la mesure locale symétrique avec voisinage de 84 trames, $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\eta = 0,5)$

être non négligeable, même pour un voisinage court.

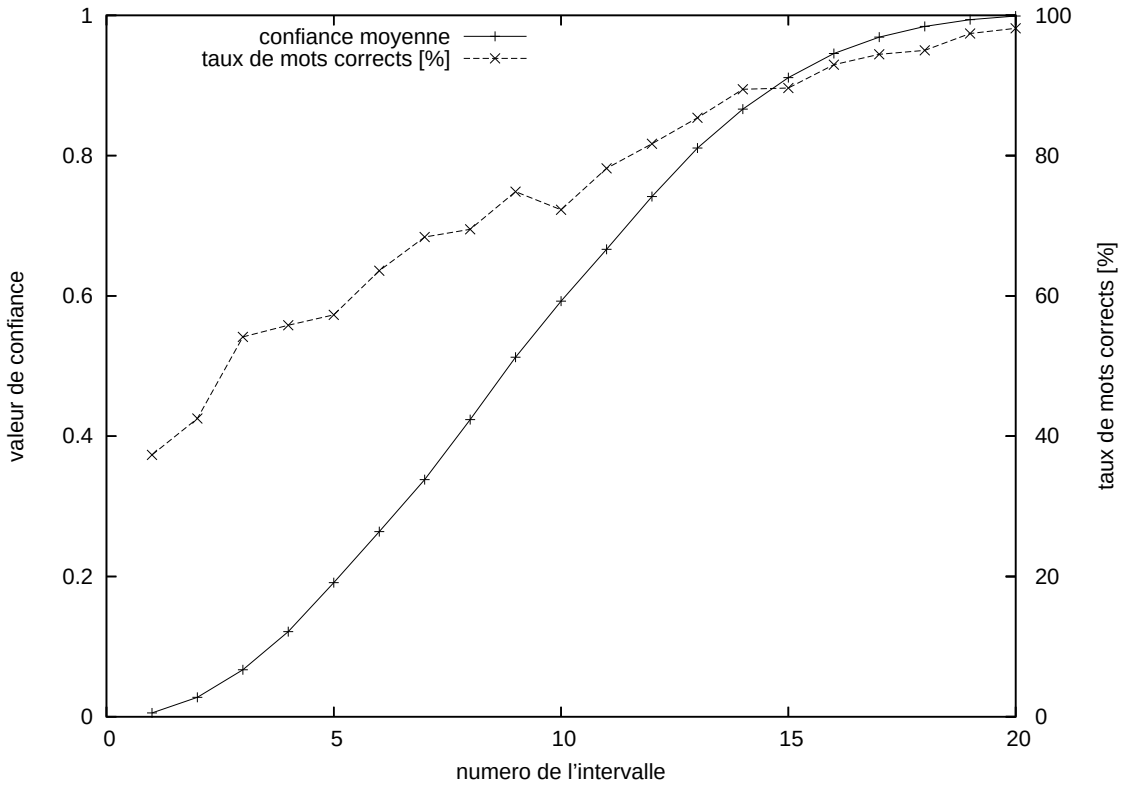


FIG. 5.8 – Répartition du taux de mots corrects et de la valeur moyenne de confiance pour 20 intervalles de taille identique sur le corpus de développement pour la mesure locale asymétrique trame-synchrone prenant en compte tout le voisinage passé depuis le début de la phrase, ($\alpha = 0,1$), ($\beta = 0,95$) et ($\eta = 0,5$)

5.5.4 Synthèse

A partir des expérimentations concernant nos mesures locales, nous pouvons conclure que ces mesures obtiennent de très bons résultats. Par exemple, la mesure locale symétrique avec un voisinage de 84 trames atteint des performances quasiment identiques à la mesure de référence (respectivement 23% et 22% d'EER). Or, le calcul de la mesure référence nécessite le traitement de l'intégralité de la phrase alors que la mesure locale se contente d'un voisinage total de 160 trames en plus de la longueur du mot.

Cette observation montre qu'il est possible de définir une très bonne mesure de confiance fondée sur la probabilité *a posteriori* uniquement à partir d'un voisinage limité du mot analysé.

Par ailleurs, les mesures locales asymétriques montrent qu'en prenant une mesure avec un voisinage passé depuis le début de la phrase et un voisinage futur de seulement 60 trames (0,6 s), nous obtenons les mêmes performances que la mesure locale symétrique avec un voisinage de 84 trames. Cette mesure locale asymétrique permet ainsi d'utiliser le maximum d'informations passées afin de compenser le manque de connaissance du futur.

De plus, il est possible de rendre nos mesures trame-synchrones en forçant un voisinage futur

nul. Par exemple, la mesure locale prenant en compte les informations depuis le début de la phrase mais avec un voisinage futur nul obtient un taux d'EER honorable de 30,1%. Ceci indique que choisir cette méthode pour définir des mesures trame-synchrones est tout à fait envisageable. De plus, d'un point de vue complexité, cette mesure locale particulière admet quasiment la même complexité que l'une de nos meilleures mesures trame-synchrones : la mesure bigramme inverse (respectivement $O(TVN^3)$ et ($O(kTN^3)$ avec $k < 1$, cf. section 3.5), tout en étant meilleure (30,1% *versus* 37,0%).

Avant de conclure sur l'évaluation sur le corpus de test de toutes nos mesures, nous allons étudier leur comportement vis-à-vis de la taille des mots analysés et comparer leurs performances à la mesure de confiance incluse dans le moteur Julius.

5.6 Influence de la taille des mots

Certaines mesures de confiance peuvent avoir un comportement différent suivant la taille des mots considérés [Duchateau 02b]. Des mesures peuvent être plus appropriées au calcul de la confiance de mots longs, tandis que d'autres mesures peuvent être plus pertinentes pour des mots courts. Nous avons mené une analyse de la dépendance de nos mesures à la longueur en phonèmes des mots.

Pour un nombre n de phonèmes, nous avons étudié le taux d'égale erreur d'une mesure de confiance en considérant uniquement les mots de n phonèmes.

Nous avons fait cette étude pour deux de nos mesures de confiance : la mesure de confiance trame-synchrone fondée sur la probabilité bigramme directe (gestion par maximisation et précédents temporels directs) et la mesure locale à voisinage symétrique de 84 trames. Pour comparaison, nous avons fait la même analyse pour la mesure de référence. Pour nos deux mesures les facteurs d'échelle utilisés sont $\alpha = 0,1$ et $\beta = 0,95$; le facteur de relâchement de la mesure trame-synchrone vaut $\varepsilon = 0,1$ et le facteur de flexibilité des mesures locales vaut $\eta = 0,5$ pour la mesure symétrique. Pour la mesure de référence, les facteurs d'échelle sont $\alpha = 0,1$ et $\beta = 1$ et $\eta = 1$.

Les figures 5.9 à 5.11 représentent l'évolution des taux d'EER respectivement de la mesure de confiance de référence, de la mesure locale symétrique et de la mesure trame-synchrone bigramme. Dans chacune des figures, la courbe correspond au taux d'EER calculé pour les mots dont le nombre de phonèmes est exactement n.

Nous pouvons remarquer que les mesures de confiance fondées sur la probabilité *a posteriori* admettent une meilleure précision pour les mots de 7 phonèmes. Pour les mots plus courts ou plus longs, le taux d'EER est plus important et même supérieur au taux global de la mesure. En revanche, pour la mesure bigramme, nous pouvons noter une baisse du taux d'EER conjointement à l'augmentation du nombre de phonèmes, avec toutefois des pics d'erreur entre deux valeurs faibles. Ces mesures de confiance semblent donc sensibles à la taille des mots sur lesquels elles sont calculées et la mesure de confiance bigramme semble avoir un meilleur comportement sur les mots longs et plus particulièrement les mots de 5 et 8 phonèmes.

Afin d'analyser l'influence de la sensibilité des mots selon leur longueur, il est utile de connaître la distribution des mots dans le corpus de développement selon leur taille en phonèmes. L'analyse pour sur les mots issus de la reconnaissance du corpus de développement, pas de la référence. Nous pouvons ainsi remarquer sur la figure 5.12 que près de 40% des mots du

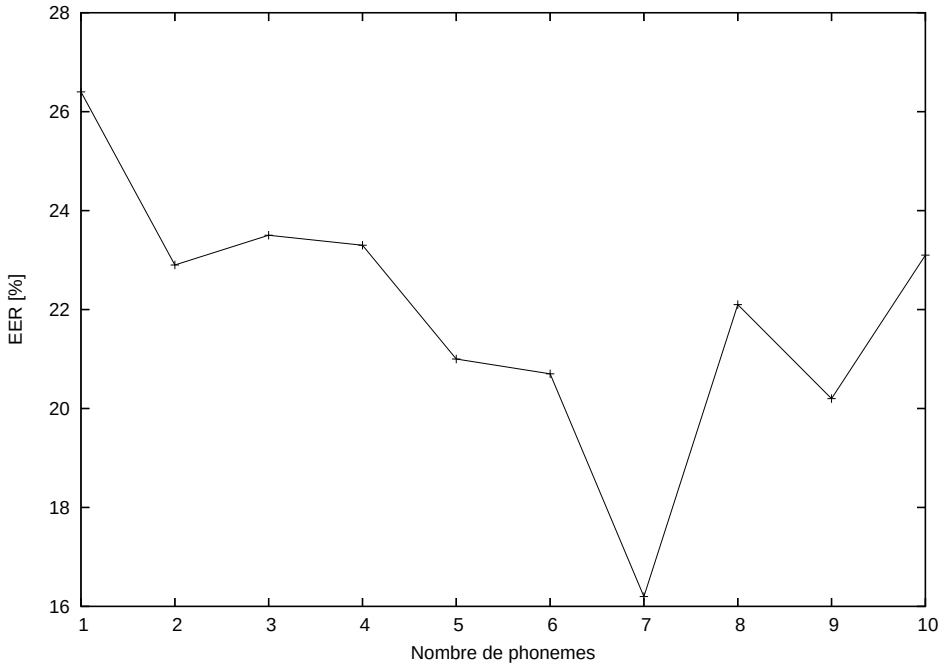


FIG. 5.9 – Evolution du taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure référence avec le jeu de paramètres ($\alpha = 0,1$), ($\beta = 1$) et ($\eta = 1$).

corpus font entre 2 et 3 phonèmes et 30% des mots du corpus font entre 5 et 8 phonèmes. Les proportions entre les zones de précision et de non-précision des mesures analysées ne sont pas égales. Toutefois ces proportions sont proches, ce qui explique sans doute le fait que le taux d'EER global des mesures est proche de la moyenne entre les zones de précision et de non-précision de ces mesures.

Dans cette étude, nous avons également décidé d'analyser l'influence de la taille des mots en séparant le corpus en deux espaces par rapport au nombre de phonèmes des mots. En effet, autant il est concevable d'utiliser deux mesures, l'une adaptée aux « grands » mots et l'autre aux « petits » mots, autant il est complexe de mettre en place une mesure différente pour chaque taille de mots. De cette expérience, nous pouvons noter que les deux types de mesures de confiance (*a posteriori* et rapport de vraisemblance) étudiées dans ce mémoire ont un comportement assez différent suivant la longueur phonétique des mots :

- les mesures de confiance fondées sur l'estimation de la probabilité *a posteriori* semblent plus enclines à l'analyse de mots de taille moyenne (4 à 6 phonèmes),
- la mesure de confiance trame-synchrone fondée sur la probabilité linguistique bigramme directe semble, quant à elle, donner de meilleurs résultats sur des mots longs.

Les courbes relatives à cette expérience sont présentées en annexe A.3 et les conclusions de cette dernière expérience corroborent les observations de l'analyse du taux d'EER des mots selon leur taille en phonèmes.

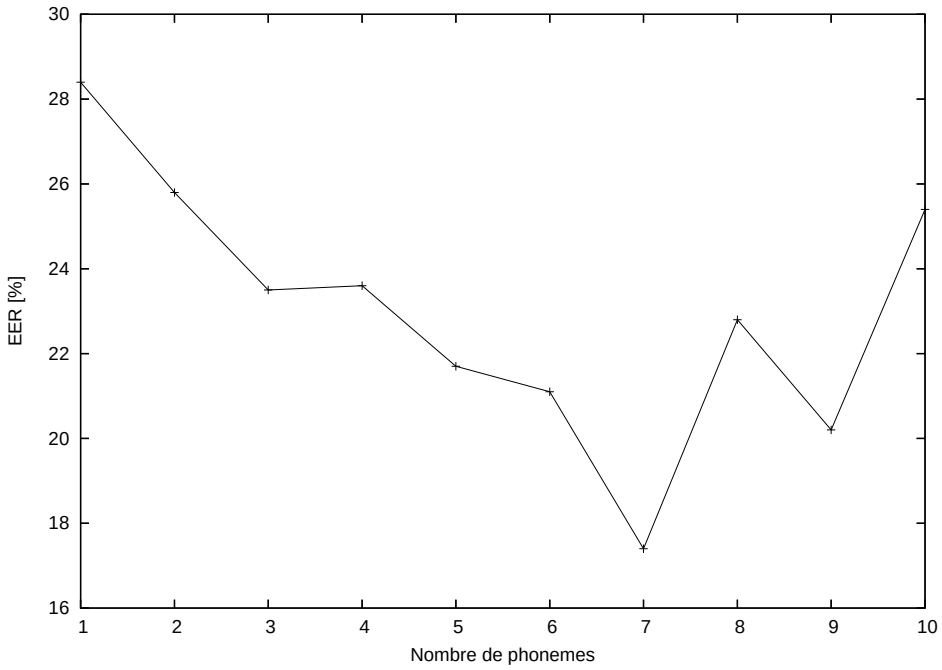


FIG. 5.10 – Evolution du taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure de confiance locale avec voisinage symétrique de 84 trames, avec le jeu de paramètres $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\eta = 0,5)$.

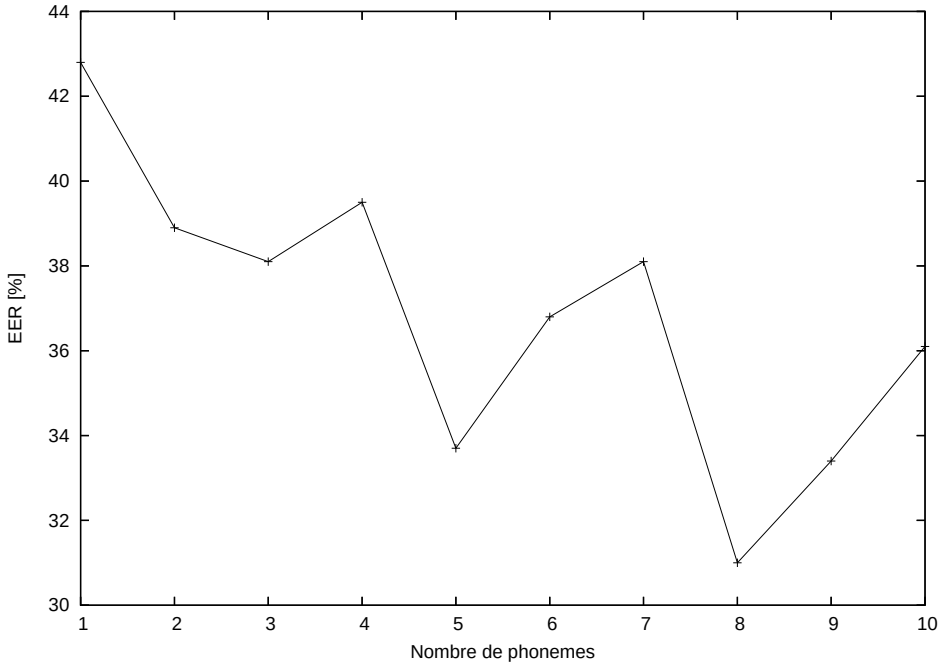


FIG. 5.11 – Evolution du taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure de confiance trame-synchrone bigramme direct avec le jeu de paramètres $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\varepsilon = 0,1)$.

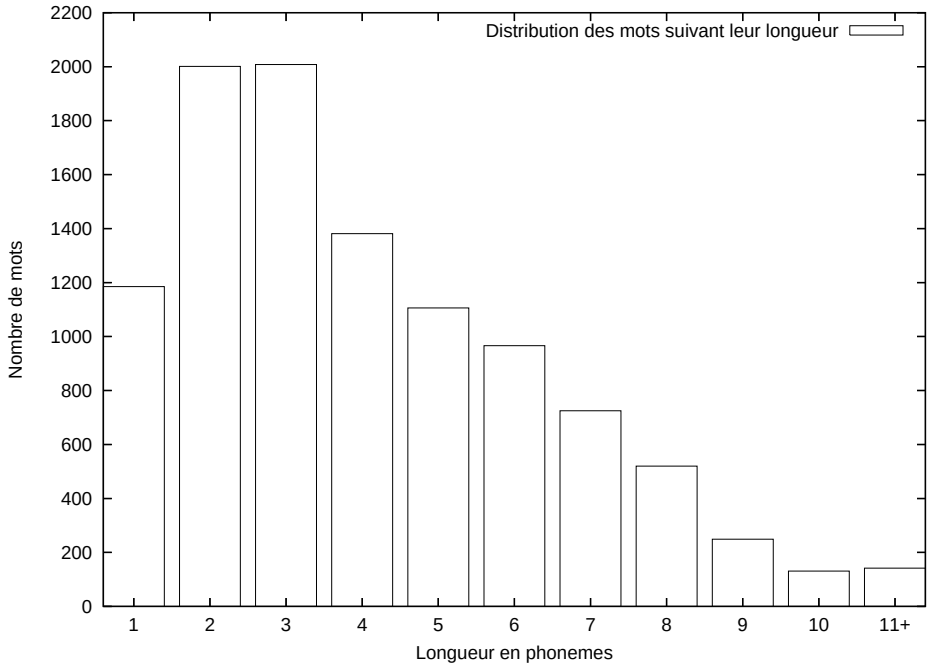


FIG. 5.12 – Répartition des mots de la reconnaissance pour le corpus de développement selon leur taille en phonèmes .

5.7 Comparaison avec la mesure de confiance intégrée dans le système de reconnaissance Julius

Le système de reconnaissance que nous utilisons, Julius, intègre le calcul d'une mesure de confiance que nous avons présentée section 2.3.6.3. Cette mesure de confiance est calculée au cours de la seconde passe du moteur et nécessite la génération intégrale du graphe de mots associé à la phrase. Cette mesure de confiance repose sur une approximation de la probabilité *a posteriori*.

Bien que la mesure soit calculée au cours de la deuxième passe du processus de reconnaissance, cette valeur de confiance n'est pas impliquée dans ce processus. Le calcul de la mesure n'a donc aucune incidence sur le résultat de la reconnaissance.

Sur le corpus de développement, la mesure de confiance du système de reconnaissance Julius obtient un taux d'EER de 31,2%. Pour cette mesure, nous avons utilisé Julius dans les mêmes conditions que pour effectuer la reconnaissance de nos corpus. C'est-à-dire que les paramètres dont dépend le système de reconnaissance ont été optimisés pour le taux de reconnaissance et non pas pour la mesure de confiance intégrée dans ce système.

Si nous comparons le résultat obtenu par la mesure de Julius par rapport à celui obtenu par notre mesure trame-synchrone bigramme inverse, le résultat de la mesure de Julius est meilleure (37,0% pour la mesure bigramme inverse). Cette différence est à relativiser car la mesure de Julius utilise beaucoup plus d'informations que nos mesures trame-synchrones. De plus, la mesure de Julius nécessitant l'exécution de l'intégralité de la première passe, elle ne permet pas une utilisation trame-synchrone.

La figure 5.13 représente les courbes DET de ces trois mesures de confiance : Julius, locale

symétrique et trame-synchrone bigramme inverse. Si nous comparons la mesure de Julius avec notre mesure locale symétrique avec un voisinage de 84 trames, le résultat de notre mesure est très significativement meilleur. Nous obtenons en effet un taux d'EER de 23,0% pour la mesure avec les facteurs d'échelle $\alpha = 0,1$, $\beta = 0,95$ et $\eta = 0,5$. Bien que notre mesure n'utilise que les données d'un voisinage local du mot analysé, les approximations que nous faisons sont moins fortes que celles faites par la mesure de Julius. Un des objectifs de la conception de la mesure de Julius était d'obtenir une mesure qui puisse être calculée rapidement et donc plusieurs approximations ont été réalisées.

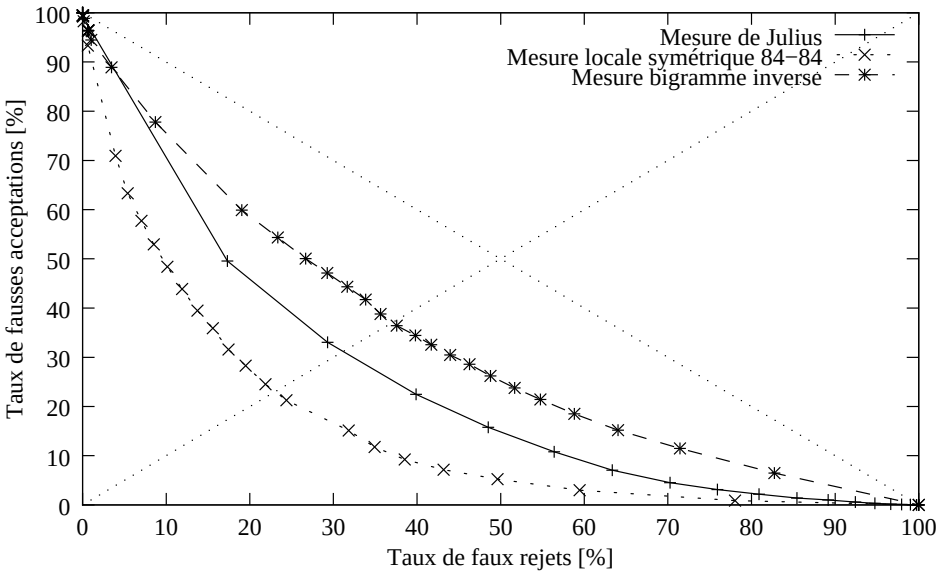


FIG. 5.13 – Courbe DET de la mesure de confiance intégrée dans Julius ainsi que celles de la mesure locale symétrique avec voisinage de 84 trames et de la mesure trame-synchrone bigramme inverse.

A partir du taux d'EER obtenu sur le corpus de développement, nous déterminons un seuil de décision afin de calculer les taux de fausses acceptations et de faux rejets sur le corpus de test. La mesure de Julius obtient un taux de faux rejets de 25,1% et un taux de fausses acceptations de 36,6%. La mesure bigramme inverse obtient des taux de faux rejets et de fausses acceptations de 35,1% et 37,2%. Notre mesure locale symétrique avec un voisinage de 84 trames obtient quant à elle des taux de 23,7% et 24,5%. Nos mesures semblent moins sensibles que la mesure de Julius à un changement de corpus. Si nous comparons le taux CER sur le corpus de test, nous obtenons : 23,9% pour notre mesure locale, 28,3% pour la mesure de Julius et 35,7% pour la mesure trame-synchrone.

5.8 Evaluation sur le corpus de test et conclusion

Une fois les mesures de confiance optimisées sur le corpus de développement, le taux d'EER nous permet de déterminer un seuil de décision. Ce seuil et les paramètres optimaux sont utilisés pour chaque mesure sur le corpus de test afin de calculer le taux de fausses alarmes et de faux rejets et ainsi évaluer les mesures de confiance. Le tableau 5.15 regroupe les taux d'EER sur le corpus de développement ainsi que les taux de fausses acceptations (FA) et de faux rejets (FR)

pour plusieurs de nos mesures de confiance.

Cependant, il est difficile de comparer les résultats obtenus sur le corpus de test au travers des deux taux FA et FR sans se placer dans le cadre d'une application spécifique. C'est pourquoi nous avons également calculé le taux d'erreur de confiance (CER) qui permet à partir de ces taux d'obtenir un critère de comparaison. Pour rappel, le taux CER est détermine ainsi :

$$CER = \frac{Nb.\ de\ Fausses\ Acceptations + nb.\ de\ Faux\ Rejets}{Nb.\ de\ mots\ reconnus}$$

Le tableau 5.15 est trié selon la valeur CER croissante sachant que plus le taux CER est faible, meilleure est la mesure.

Nous pouvons remarquer que les résultats des différentes mesures de confiance du tableau 5.15 sont homogènes entre le passage du corpus de développement au corpus de test. En effet, il n'y a pas d'écart significatif entre le taux d'EER sur le corpus de développement et le taux d'EER que nous pouvons estimer par rapport aux FA et FR observés.

A partir de ce tableau, nous pouvons également noter que la hiérarchie observée sur le corpus de développement entre les différentes mesures est respectée sur le corpus de test : les mesures locales utilisant un voisinage futur sont en tête, suivies de la mesure locale trame-synchrone, puis les mesures trame-synchrones fondées sur la vraisemblance et enfin en bas de tableau, la mesure n'utilisant que les prédécesseurs au sens de Viterbi.

La mesure locale asymétrique prenant en compte le passé depuis le début de la phrase et un voisinage futur de 60 trames (0,6 s) obtient un résultat proche de celui de la mesure de référence en terme de CER (respectivement 23,1% et 22,0%). La mesure locale symétrique avec un voisinage de 84 trames de part et d'autre du mot analysé obtient une valeur proche des deux précédentes : 23,9%. Ainsi, cette mesure n'utilisant qu'un voisinage court à la fois dans le passé et dans le futur peut être employée par exemple dans des applications dont seul un extrait du signal est disponible (vérification à la demande).

La mesure locale trame-synchrone obtient quant à elle des performances intermédiaires entre les mesures locales précédentes et les mesures trame-synchrones fondées sur un rapport de vraisemblance (29,6% contre 34,7% de CER pour la mesure trigramme).

L'apport d'informations extérieures au système de reconnaissance, comme la probabilité trigramme ou la probabilité bigramme inverse, permet aux mesures trame-synchrones intégrant ces données d'être significativement meilleures que les autres mesures trame-synchrones.

Nous pouvons également observer que les mesures prenant en compte un voisinage passé plus important (trigramme, bigramme et unigramme) se distingue significativement les unes par rapport aux autres.

En revanche, comme pour le corpus de développement, le fait de ne considérer que le prédécesseur au sens de Viterbi dégrade fortement les performances des mesures.

Le taux CER peut être mis en relation avec le taux de reconnaissance du système. En effet, le CER du système est défini comme le taux d'erreur en ne considérant que les insertions et les substitutions alors que le taux d'erreur en mots habituel inclus également les omissions. Sur le corpus de test, le taux CER du système de reconnaissance est de 27,4% (le taux d'erreur en mots est de 33%).

TAB. 5.15 – Synthèse des résultats obtenus par nos mesures de confiance ainsi que par la mesure de référence sur corpus de développement en taux d'EER et sur le corpus de test en taux de fausses alarmes (FA), taux de faux rejets (FR) et de CER.

mesure	corpus dév.	corpus test		
	EER	FR	FA	CER
référence	22,0%	21,2%	24,4%	22,1%
locale début-60	23,2%	23,1%	23,2%	23,1%
locale 84-84	23,0%	23,7%	24,5%	23,9%
locale 60-60	25,5%	27,3%	24,1%	26,4%
locale début-0	30,1%	30,3%	27,9%	29,6%
trigramme avec maximisation	37,1%	34,5%	35,4%	34,7%
bigramme inverse avec maximisation	37,0%	35,1%	37,2%	35,7%
bigramme avec sommation	37,4%	36,2%	35,6%	36,0%
bigramme avec maximisation	37,4%	36,6%	35,8%	36,4%
bigramme avec maximisation tabulée	37,2%	38,8%	31,7%	36,9%
unigramme	37,6%	38,8%	33,9%	37,4%
bigramme avec prédécesseur Viterbi	40,6%	39,4%	40,0%	39,6%

Chapitre 6

Evaluation dans le cadre d'applications spécifiques

Sommaire

6.1 Introduction 122
6.2 Application à la détection de mots clés 122
6.3 Intégration d'une mesure de confiance dans le moteur de reconnaissance 125
6.3.1 Méthodologie 125
6.3.2 Expérimentation 126
6.4 Transcription de cours en salle de classe 129
6.4.1 Présentation du système initial 129
6.4.2 Utilisation de la mesure de confiance 130
6.4.3 Protocole de test 131
6.5 Conclusion 135

6.1 Introduction

Dans le chapitre précédent, les mesures de confiance ont toutes été comparées suivant le critère du taux d'égale erreur. Ce taux permet de donner un aperçu de leur performance de manière égale, que ce soit pour les fausses acceptations ou pour les faux rejets. Nous allons désormais étudié le comportement des mesures de confiance que nous avons proposées par rapport à des objectifs précis : la détection de mots clés, l'intégration de la mesure de confiance dans le moteur de reconnaissance et la transcription de cours pour des élèves sourds ou malentendants.

6.2 Application à la détection de mots clés

Cette détection pourra se faire par exemple dans un flux continu de parole, avec une vérification à l'aide d'une mesure de confiance des mots clés repérés par le système de reconnaissance. Par rapport à un seuil de décision fixé, un mot supposé clé sera alors accepté ou rejeté suivant sa valeur de confiance. Dans notre approche, nous nous focalisons sur la diminution du nombre de fausses acceptations de façon à ne conserver que des vraies alarmes. Toutefois, une diminution du nombre de fausses acceptations implique une augmentation du nombre de faux rejets. Nous devons alors trouver un point de fonctionnement qui correspond à un seuil de décision qui nous permette de concilier à la fois une diminution significative du nombre de fausses acceptations et une faible augmentation du nombre de faux rejets.

Pour cette étude de faisabilité, nous avons défini une liste de 33 mots clés (cf. Tab. 6.1), sélectionnés sur le corpus de développement selon un critère de fréquence d'apparition (variant entre 6 et 32 occurrences par heure) et un critère de longueur minimale (3 phonèmes).

Tab. 6.1 – Liste des 33 mots clés.

europe	guerres	pragmatismes	rapport	figaro
pays	gouvernement	pragmatique	rapports	
président	gouvernements	pragmatiques	libération	
présidents	politique	millions	libérations	
aujourd'hui	politiques	million	ambassadeur	
irak	république	démocratie	ambassadeurs	
chirac	républiques	démocraties	quotidien	
guerre	pragmatisme	bagdad	quotidiens	

Dans cette application, nous avons pris un critère de comparaison orthographique strict, c'est-à-dire par exemple que nous comptons comme une fausse acceptation un pluriel reconnu à la place d'un singulier. Un critère plus souple aurait pu être défini en considérant comme équivalents des mots de même racine, par exemple pragmatique et pragmatisme. Quelques mots clés n'apparaissent pas dans le corpus de développement et/ou de test, principalement à cause de la distinction des pluriels et des singuliers dans la liste des mots clés.

Dans le corpus de développement, 152 mots clés sont présents. Le système de reconnaissance en a reconnu 130 : 122 sont réellement des mots clés (noté VA), 8 sont des erreurs dues au système de reconnaissance. Ces 8 erreurs représentent les fausses acceptations (noté FA). A l'aide des mesures de confiance, nous voulons donc réduire le nombre de fausses acceptations (8) tout en conservant le maximum de mots clés trouvés (122).

La figure 6.1 représente pour plusieurs mesures de confiance l'évolution du nombre de fausses acceptations et du nombre de bons mots clés restant par rapport à la variation du seuil de

décision. Une mesure de confiance idéale aura un seuil pour lequel le nombre de fausses alarmes est nul sans perdre de mots justes. Les courbes de cinq mesures de confiance sont représentées :

- la mesure trame-synchrone fondée sur la probabilité trigramme (bigramme),
- la mesure trame-synchrone fondée sur la probabilité bigramme (trigramme),
- la mesure locale fondée sur la probabilité *a posteriori* à voisinage symétrique de 84 trames (Locale 84-84),
- la mesure locale asymétrique fondée sur la probabilité *a posteriori* avec un voisinage passé débutant dès le début de la phrase et avec voisinage futur de 84 trames (Locale début-84),
- la mesure locale trame-synchrone fondée sur la probabilité *a posteriori* avec un voisinage passé débutant dès le début de la phrase et avec voisinage futur nul (Locale début-0).

Les courbes commencent par le point de coordonnées (122 ;8) correspondant au nombre de vraies acceptations et au nombre de fausses acceptations trouvées par le système de reconnaissance (seuil de décision nul). Selon l'augmentation du seuil de décision, le nombre d'acceptations à la fois vraies et fausses diminue. Nous devons déterminer un point de fonctionnement pour chaque mesure correspondant à un seuil pour lequel le compromis entre diminution du nombre de fausses acceptations et du nombre de vraies acceptations nous semble acceptable.

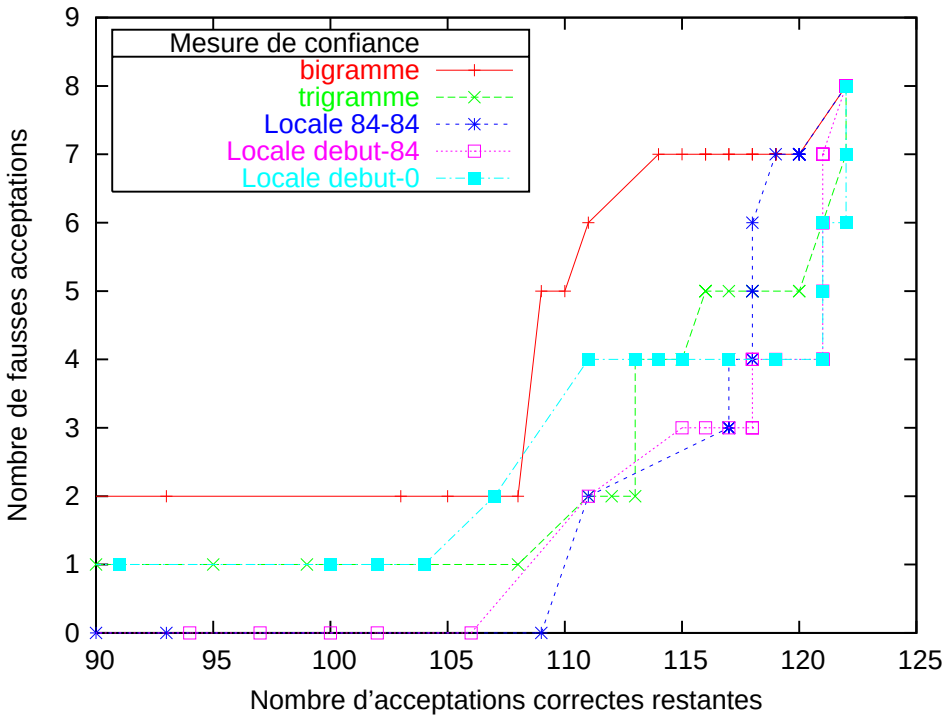


FIG. 6.1 – Evolution du nombre de fausses acceptations et du nombre de bons mots clés restant en fonction du seuil de décision (corpus de développement).

Si nous considérons un point de fonctionnement exprimant la diminution de 50% du nombre de FA, nous pouvons remarquer que la hiérarchie observée selon le critère du taux d'EER pour nos mesures de confiance est conservée. Les deux mesures locales prenant en compte le voisinage passé depuis le début de la phrase obtiennent la même diminution du nombre de VA (121 au lieu de 122). Pour la mesure locale symétrique, cette même diminution est très proche de ces deux mesures (118). La mesure trigramme montre un résultat meilleur que la mesure bigramme avec

une diminution à 115 des VA restantes contre 109 pour la mesure bigramme.

En revanche, si nous considérons comme point de fonctionnement le seuil pour lequel le nombre de fausses acceptations est nul ou stagnant, les observations sont légèrement différentes. En effet, la mesure locale symétrique est celle conservant le plus de vraies acceptations pour une élimination totale des fausses acceptations. En terme de pourcentage de réduction, cette mesure admet un taux de réduction du nombre de VA de 10,7% alors que la mesure locale début-84 admet une réduction de 13,1%. Les mesures bigramme et trigramme obtiennent respectivement une diminution de 14,7% et 11,5% du nombre de VA pour une réduction de 57,5% du nombre de FA. Le mesure locale trame-synchrone atteint une diminution du nombre de VA de 11,5% pour une réduction de seulement 75%.

Dans le corpus de test, le nombre de mots clés est plus important. Sur les 333 mots clés présent le système en a détecté 284 : 273 vraies acceptations et 11 fausse acceptations. Les courbes montrant l'évolution du nombre de FA et du nombre VA des cinq mesures de confiance sont représentées figure 6.2.

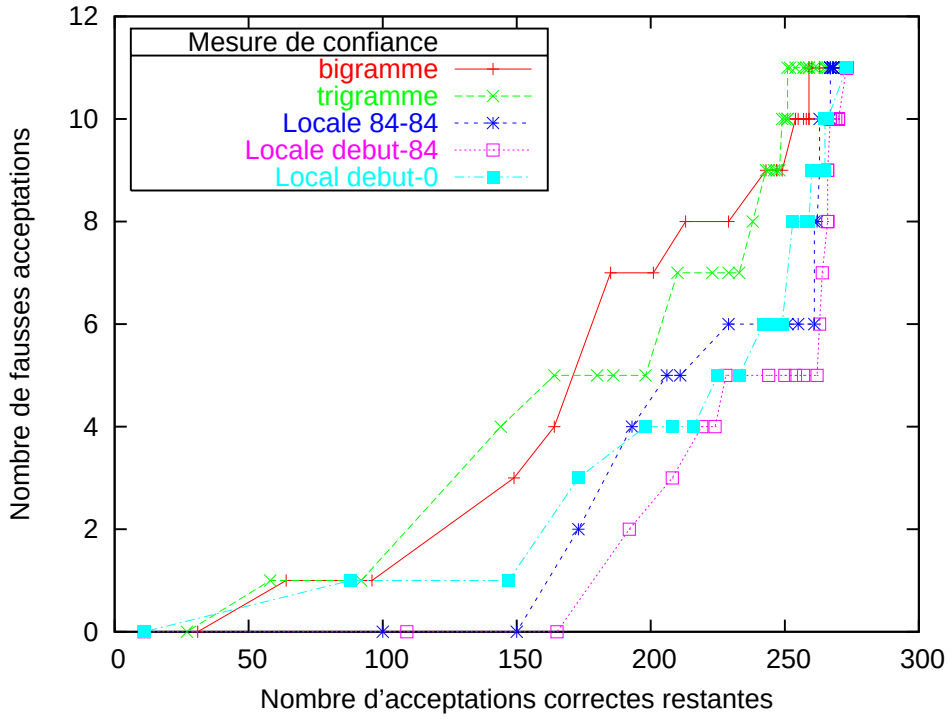


FIG. 6.2 – Evolution du nombre de fausses acceptations et du nombre de bons mots clés restant en fonction du seuil de décision (corpus de test).

A partir des seuils correspondant aux points de fonctionnement définis précédemment (diminution total ou presque totale du nombre de FA) sur le corpus de développement, nous allons alors évaluer les diminutions de FA et de VA sur le corpus de test. L'ordre de grandeur de la proportion de diminution de ces mesures de confiance sur le corpus de test n'est pas le même que celui observé sur le corpus de développement. Le nombre de fausses acceptations et de vraies acceptations ont diminué de :

- 63,6% de FA et 20,9% de VA pour la mesure bigramme,
- 36,4% de FA et 18,3% de VA pour la mesure trigramme,

– 45,5% de FA et 16,1% de VA pour la mesure locale 84-84,
– 54,5% de FA et 16,5% de VA pour la mesure locale début-84,
– 27,3% de FA et 16,1% de VA pour la mesure locale trame-synchrone début-0.

Sur le corpus de test, le compromis entre la diminution du nombre de fausses acceptations au détriment de la diminution du nombre de vraies acceptations n'est plus du même ordre que sur le corpus de développement. Les mesures sont moins fiables d'un corpus à l'autre. Toutefois, ces résultats étant obtenus avec un nombre réduit de mots clés, ces observations sont à confirmer avec une liste de mots clés plus importante.

6.3 Intégration d'une mesure de confiance dans le moteur de reconnaissance

Une des applications visées par la définition de nos mesures de confiance est la transcription automatique d'émissions radiophoniques ou télévisuelles en direct. Nous avons vu au chapitre 3 que les mesures de confiance pouvaient être utilisées de deux manières :

– la mise en évidence des mots de faible confiance dans la transcription,
– l'intégration de la mesure de confiance dans le moteur de reconnaissance.

Nous présentons ici cette seconde utilisation, la première étant développée dans la section suivante sur la transcription de cours.

6.3.1 Méthodologie

Un des intérêts pour lesquels nous avons défini des mesures trame-synchrones est de pouvoir calculer celles-ci pendant le processus de décodage du système de reconnaissance. La vraisemblance d'une phrase, donnée par l'équation 1.7, est modifiée afin de prendre en compte la valeur de confiance $C(w)$ du mot w. Nous obtenons alors l'équation générique suivante :

$$P(O|M) = \max_{W \in \Xi} \pi_{w_0} \prod_{w_i \in W} P(O|w_i) P(w_i|w_{i-1}) C(w_i) \tag{6.1}$$

Ξ représente l'ensemble des séquences de mots appartenant au lexique qu'il est possible de construire.

Plus précisément, nous intégrons la mesure de confiance dans la première passe du système de reconnaissance Julius. A chaque trame t de signal, le moteur de reconnaissance détermine un ensemble de mots pour lesquels le modèle HMM associé se trouve dans un état final, c'est-à-dire l'ensemble des mots du graphe dont l'instant de fin vaut t. Chacun de ces mots est ajouté dans le graphe de mots avec sa vraisemblance. C'est précisémment au moment où l'ensemble des mots terminant à cette trame est déterminé que nous pouvons calculer la valeur de confiance de chacun de ces mots. Une fois ces valeurs disponibles, nous modifions le calcul de la vraisemblance de ces mots en intégrant leur valeur de confiance selon l'équation suivante :

$$\gamma . P(o_\tau^t | w_n) . P(w|w_p)^\delta . C(w)^\nu \tag{6.2}$$

Le facteur d'échelle ν que nous avons ajouté permet de pondérer la contribution de la valeur de confiance dans le calcul du score associé à un mot. Un mot du graphe dont la confiance est faible sera ainsi défavorisé dans la phase de décodage alors qu'un mot ayant une forte confiance le sera beaucoup moins.

Notons que lors de la deuxième passe, la vraisemblance est de nouveau calculée avec toutefois des modèles plus précis, que ce soit pour le modèle de langage ou pour les modèles acoustiques

(pas d'approximation du calcul des gaussiennes). Ce calcul se fait de la fin de la phrase vers le début de la phrase suivant l'équation suivante :

$$\gamma_2.P(o_\tau^t|w_n).P(w|w_s w_{ss})^{\delta_2} \tag{6.3}$$

w_s étant le mot suivant w_n et w_{ss} le suivant de ce suivant. Les paramètres γ_2 et δ_2 sont définis indépendamment des paramètres γ et δ de la première passe. Il n'y a pas de coefficient de confiance car nous intégrons notre mesure uniquement dans le calcul de la vraisemblance de la première passe.

Cette manière simple d'intervenir sur le score de vraisemblance de tous les mots du graphe influe sur le système de reconnaissance avec plusieurs conséquences :

- modification du graphe de mots à cause de l'élagage. La mesure de confiance pénalisant plus ou moins des mots avant élagage, certains mots peuvent être conservés alors que d'autres peuvent être éliminés ;
- modification de la solution à la fin de la première passe. Le graphe de mots et les vraisemblances étant modifiés, la phrase hypothèse de score maximal peut être différente ;
- modification de la solution à la fin de la deuxième passe. En effet, bien que la mesure de confiance ne modifie pas les calculs effectués au cours de la seconde passe, le graphe de mots n'est plus le même. De plus, comme la fonction heuristique utilisée par l'algorithme A^* en seconde passe est fondé sur les vraisemblances calculées lors de la première passe, cette heuristique est également différente.

En revanche, si nous avions décidé d'intégrer une mesure de confiance au niveau de la deuxième passe, le graphe de mots et la solution en fin de première passe seraient identiques entre un système avec la mesure et un système sans la mesure.

Nous pouvons ainsi noter que la portée des modification dues à l'intégration d'une mesure de confiance dans un système de reconnaissance sont dépendantes des algorithmes utilisés dans ce système.

Par ailleurs, le nombre de mots dans le graphe pouvant être important le taux d'erreur en mot sur l'ensemble du graphe sera proche de 100%. En effet, seuls les mots de la phrase sont justes et ces mots ne représentent qu'une faible partie des mots présents dans le graphe. Or, comme nous estimons une valeur de confiance pour tous les mots du graphe, il est intéressant d'effectuer une homogénéisation des valeurs de confiance selon le taux de mots corrects dans le graphe.

6.3.2 Expérimentation

Pour cette étude, nous avons défini un sous-ensemble du corpus de développement, contenant 51 phrases uniformément réparties suivant leur taux d'erreur en mots. De plus, nous avons utilisé des modèles acoustiques triphones ainsi qu'un lexique et un modèle de langage différent (cf. section 4.4.2 et 4.5.2) et une compilation du système de reconnaissance Julius en mode *fast* (cf. 4.2.3).

La mesure de confiance que nous avons intégrée dans le système de reconnaissance est la mesure trame-synchrone fondée sur le rapport de vraisemblance avec les probabilités bigrammes. La gestion des occurrences multiples est réalisée par maximisation et nous considérons comme mots précédents tous les prédécesseurs temporels directs. Les facteurs d'échelle et le facteur de relâchement ont été optimisés sur le corpus de développement selon le critère du taux d'EER ; $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\varepsilon = 0,1)$.

Nous définissons alors 4 systèmes de reconnaissance Julius :

- sans mesure de confiance,

– avec l'intégration d'une mesure trame-synchrone,
– avec l'intégration d'une mesure trame-synchrone tabulée (homogénéisation des valeurs de confiance),
– avec l'intégration d'une mesure trame-synchrone bornée de façon à éviter à la mesure de prendre les valeurs aux extrémités de l'intervalle $[0, 1]$.

La figure 6.3 représente le taux de mots corrects selon la valeur de confiance attribuée pour tous les mots des graphe des 51 phrases. Le taux de mots corrects dans le graphe étant proche de zéro, la valeur de confiance calculée par la mesure surestime en moyenne la probabilité qu'un mot du graphe soit *a priori* correct. C'est pourquoi nous avons décider de définir un système de reconnaissance intégrant la mesure bigramme mais avec homogénéisation des valeurs de confiance. Nous avons défini une homogénéisation par tabulation des valeurs sur 20 intervalles. Cette résolution permet d'être suffisamment précis tout en ayant un lissage des valeurs. Nous avons également défini un autre système toujours à partir de la mesure bigramme mais en bornant les valeurs que peut prendre la mesure. Cette mesure n'est pas homogénéisée.

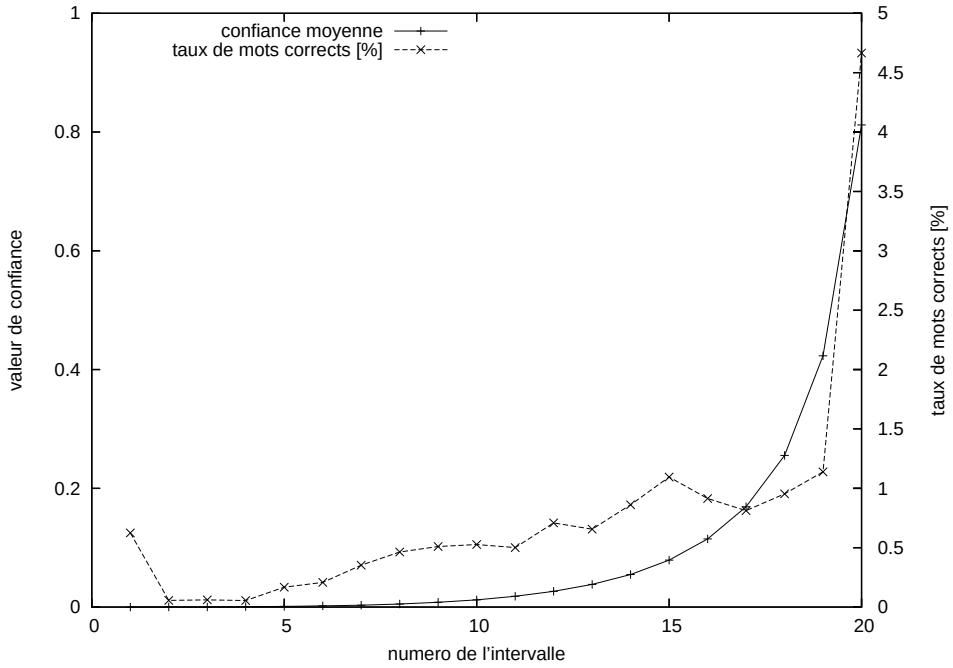


FIG. 6.3 – Distribution du taux de mots corrects en fonction de la valeur moyenne de confiance pour 20 intervalles de taille identique sur le corpus utilisé dans le cadre de la mesure bigramme intégrée dans le moteur de reconnaissance.

Nous avons dans un premier temps évalué nos 4 systèmes de reconnaissance sur les 51 phrases. Les paramètres γ, δ, γ_2 et δ_2 du système Julius simple ont été optimisés dans leur ensemble selon le taux d'erreur en mots du système complet (le taux d'erreur en mot prend en compte les insertions, les substitutions et les omissions). Pour les 3 systèmes incluant les mesures de confiance, nous avons optimisé les paramètres γ, δ et ν de l'équation 6.2 pour tenir compte de l'introduction de la valeur de confiance dans le calcul. Nous n'avons pas modifié les facteurs utilisés par le moteur de reconnaissance pour le calcul de la vraisemblance au cours de la deuxième passe.

Le système de reconnaissance Julius permettant de stopper le processus de reconnaissance à

la fin de la première passe. Le système délivre alors la meilleure phrase solution selon l'algorithme de Viterbi. Le tableau 6.2 présente les taux d'erreur en mots des trois systèmes intégrant une mesure de confiance en considérant les solutions obtenues à l'issue de la première passe. Dans ce tableau, les taux d'erreurs sont évalués selon le facteur de pénalité γ. Les facteurs δ et ν sont fixé à leur valeur optimale ($\delta = 10$) et ($\nu = 1$).

TAB. 6.2 – Taux d'erreur en mots à la fin de la première passe suivant différentes intégrations de la valeur de confiance.

mesure de confiance	pénalité γ		
	-7	-5	0
directe	39,8%	38,6%	38.4%
seuillée	38,6%	37,8%	37,8%
avec homogénéisation	39,7%	38,7%	**37,1%**

Le système de reconnaissance Julius sans mesure de confiance obtient un taux d'erreur en mots de 38,3% avec les paramètres ($\gamma = -7$) et ($\delta = 10$). Nous pouvons remarquer qu'utiliser la mesure de confiance directement n'améliore pas les résultats en fin de première passe. Les deux systèmes intégrant soit des valeurs de confiance bornées soit l'homogénéisation des valeurs montrent une diminution du taux d'erreur par rapport au système Julius de base. La mesure avec homogénéisation obtient le taux le plus bas avec 37,1% d'erreur contre 38,3% pour le système Julius simple.

Comme nous l'avons mentionné, l'intégration de la mesure de confiance telle que nous l'avons faite peut avoir un impact non seulement pour le résultat de la première passe mais aussi pour celui de la deuxième passe. Le tableau 6.3 contient les taux d'erreur en mots de nos trois systèmes intégrant une mesure de confiance mais en poursuivant le processus de reconnaissance jusqu'à la fin. Les systèmes sont utilisés avec les mêmes paramètres que précédemment : ($\delta = 10$) et ($\nu = 1$).

TAB. 6.3 – Taux d'erreur en mots à la fin de la deuxième passe suivant différentes intégrations de la valeur de confiance.

mesure de confiance	pénalité γ		
	-7	-5	0
directe	32,1%	31,5%	31,9%
seuillée	30,9%	31,0%	31,0%
avec homogénéisation	32,5%	**29,6%**	31,2%

Le taux d'erreur en mots du système Julius sans mesure de confiance est de 30,8%. Nous pouvons noter qu'en considérant le processus de reconnaissance complet, l'intégration de la mesure de confiance avec homogénéisation permet de diminuer le taux d'erreur en mots à 29,6%. Ceci montre bien que l'intégration de la mesure de confiance en première passe a une influence sur le résultat global du système de reconnaissance.

Ces résultats indiquent que pour l'intégration d'une mesure de confiance dans un système de reconnaissance comme Julius, l'homogénéisation des valeurs de confiance permet d'améliorer la pertinence de la mesure de confiance.

Nous avons donc montrer que l'intégration d'une mesure de confiance trame-synchrone permet d'améliorer un système de reconnaissance que le processus soit arreté à la première passe ou qu'il

soit exécuté intégralement. Cette preuve de faisabilité ayant été réalisée sur un corpus de 51 phrases, les améliorations observées sont à confirmer sur le corpus de test.

6.4 Transcription de cours en salle de classe

6.4.1 Présentation du système initial

Notre équipe est impliquée dans le projet RIAM LABIAO, *lecture LABIale Assistée par Ordinateur* dont l'objectif de développer un ensemble de logiciels permettant aux sourds ou malentendants d'être plus autonomes (cf. section 3.1.1.2).

Au cours de ce projet, la participation de l'équipe s'est concrétisée par la création de logiciels proposant deux modalités pouvant aider les personnes sourdes ou malentendantes :

- l'affichage d'une tête parlante artificielle intégrant le codage Langage Parlé Complété (LPC),
- l'affichage d'une transcription synchronisée avec la voix.

La figure 6.4 montre le principe de la tête parlante LABIAO. Un système de reconnaissance reconnaît une phrase et pilote la tête parlante qui va à la fois articuler les phonèmes et ajouter le codage en LPC des sons à l'aide d'une main artificielle. La modalité de transcription correspond à l'affichage sur un écran de la transcription du cours de façon synchronisée avec la voix, même si un délai existe entre l'affichage du texte et la production du son.

Fig. 6.4 – Tête codeuse de synthèse développée au Loria pour le projet LABIAO (le son « pa » en LPC).

6.4.2 Utilisation de la mesure de confiance

Dans le système LABIAO initial, le résultat brut de la reconnaissance a été utilisé pour ces deux modalités. Ce résultat brut pouvant contenir des erreurs, nous avons alors proposé d'introduire une mesure de confiance afin d'indiquer les erreurs possibles. Nous pensons que ces indications pourront aider à la compréhension du texte. Plusieurs solutions sont envisagées, par exemple : dans le cas de la tête parlante et en fonction de la valeur de confiance :

- faire varier la transparence de la main codeuse,
- faire varier la couleur de la tête codeuse ;

dans le cas de la transcription synchronisée :

- mettre en couleur (rouge) les mots ayant une faible confiance et laisser en couleur normale (noir) les autres,
- donner en couleur la transcription dans un langage phonétique simplifié des mots ayant une faible confiance.

Dans notre étude, nous avons considéré la modalité de transcription synchronisée ainsi que les deux nouvelles modalités de coloration des erreurs. Soit la phrase prononcée suivante :

... le circuit troublé seulement par un hélicoptère qui bourdonne là-haut ...

Le système de reconnaissance a reconnu ceci :

... le circuit troublée seulement par un hélicoptère qui bourreau donnent là-haut ...

En utilisant la mesure de confiance locale symétrique avec un voisinage de 84 trames (840 ms), le tableau 6.4 montre les valeurs de confiance calculées pour les mots de la phrase reconnue. Cette phrase présente deux erreurs : une substitution (bourdonne et bourreau) et une insertion (donnent). Ainsi, à l'aide de la mesure de confiance, ces deux erreurs seront mises en valeur selon les modalités que nous avons proposées :

- coloration des mots :
  le circuit troublée seulement par un hélicoptère qui **bourreau donnent** *là-haut*

- coloration et phonétisation simplifiée des mots :
  le circuit troublée seulement par un hélicoptère qui **b_ou_r_o d_o_n_e** *là-haut*

Pour la phonétisation, les mots de faible confiance sont ré-écrits en utilisant un alphabet phonétique simplifié permettant une lecture aisée du son sans avoir le sens. En effet, beaucoup de gens ne connaissent pas les conventions d'écriture des phonèmes et leur affichage n'est pas toujours possible. Ainsi, certains phonèmes ont été remplacés par un groupe de lettres de l'alphabet français indiquant la prononciation à observer. Par exemple, le mot *bourreau* s'écrira phonétiquement *b ou r o*.

Tab. 6.4 – Exemple des valeurs de confiance des mots d'une phrase.

le	circuit	troublée	seulement	par	un	hélicoptère	qui	bourreau	donnent	là-haut
0,98	0,97	0,70	0,70	0,70	0,97	0,99	0,70	0,01	0,29	0,96

6.4.3 Protocole de test

Ces nouvelles modalités définies, nous devons les évaluer par rapport aux élèves sourds selon deux critères :

– l'amélioration de la compréhensibilité,

– l'appréciation de ces modalités.

Le projet LABIAO a permis une évaluation de la compréhensibilité d'une tête parlante ou d'une transcription synchronisée pour des étudiants sourds ou malentendants. Pour cette évaluation, les tests se passaient ainsi : le résultat de la reconnaissance d'un texte d'une leçon était diffusé soit avec la tête parlante LPC soit avec la transcription synchronisée. Puis, une série de questions sur le contenu de la leçon était posée afin d'évaluer le niveau de compréhension des participants. Des questions d'appréciation étaient également posées afin d'avoir un retour sur le ressenti des participants vis-à-vis de l'utilisation d'une tête parlante versus celle d'une transcription synchronisée. Ce travail d'évaluation a été réalisé notamment avec des élèves de l'école d'orthophonie de Nancy [Mourot 07].

Le protocole de test que nous avons défini s'inspire de celui utilisé pour le projet LABIAO tout en se concentrant sur notre objectif : est-ce-que les modalités issues de la mesure de confiance peuvent aider ?

Notre test se compose de 4 textes , dits calibrés, associés à une série de questions permettant d'évaluer le niveau de compréhension du lecteur. A l'origine, ces textes ont été définis afin de connaître le niveau de lecture d'élèves à l'entrée en sixième. Les sujets abordés sont assez vastes[3] : conte suédois [Lobrot 70], histoire portant sur les 24 heures du Mans [Lobrot 70], un récit d'une expédition en avion [Chevrier-Muller 97] ainsi qu'une enquête policière autour du vol d'un ordinateur [Boutard 06].

Chaque texte a été lu et enregistré par une même personne à l'aide d'un micro-casque, en intégrant une prosodie et des pauses semblables à celles rencontrées pour des cours. Puis, ces enregistrement ont été.transcrits par le système de reconnaissance de notre équipe : Julius en mode *fast*, modèles acoustiques triphones, lexique et modèle linguistique associés aux modèles triphones. Aucune adaptation au locuteur ou à l'environnement acoustique n'a été réalisée. Le taux de reconnaissance sur ces textes est de 71,4%.

Pour chaque texte transcrit, nous avons calculé la confiance de chaque mot reconnu. Nous avons choisi d'utiliser la mesure de confiance locale symétrique avec un voisinage 84 trames. Cette mesure obtenait en effet de bons résultats sur nos corpus de développement et de test et nécessite seulement un délai de 84 trames.

Selon les valeurs de confiance calculées, nous devons décider si un mot est correct ou incorrect. Nous utilisons comme seuil de décision celui du taux d'EER déterminé sur le corpus de développement pour cette mesure afin de ne favoriser ni les fausses acceptations ni les faux rejets. Il n'est cependant pas établi que ce point de fonctionnement soit optimal dans ce cadre applicatif. En effet, l'exploration d'autres points de fonctionnement basés sur des critères perceptifs pourrait être menée afin d'évaluer l'influence des fausses acceptations et des faux rejets sur l'utilisateur.

Pour chaque texte, nous avons sélectionné une partie ou l'intégralité de la transcription (240 mots en moyenne) et préparé quatre à cinq questions. La première question consiste à ré-écrire une partie indiquée de la transcription telle qu'elle aurait du être si la reconnaissance était parfaite (60 mots en moyenne). Les autres questions portent sur des points précis du texte, des mots qui ont été bien ou mal reconnus. Parallèlement, quatre questions subjectives d'appréciation, de difficulté sont posées pour chaque texte. Enfin, un questionnaire d'appréciation global, sur l'ensemble des textes est proposé.

[3] Nous ne pouvons fournir les textes en intégralité pour des raisons de droits

L'expérience consiste à présenter successivement les transcriptions des quatre textes avec leur questionnaire associé à chaque sujet, puis d'évaluer le niveau de compréhension du lecteur.

- une première transcription issue du système de reconnaissance, avec ses erreurs mais sans aucune indication de confiance ou de mots faux ;
- une deuxième transcription présentée en incluant la nouvelle modalité de coloration des mots de faible confiance ;
- une troisième transcription utilisant la coloration et la phonétisation simplifiée des mots de faible confiance ;
- une quatrième transcription utilisée comme *oracle*, c'est-à-dire que les mots colorés sont exactement les mots incorrects, sans utiliser de mesure de confiance.

Les différentes transcriptions et modalités sont distribuées aléatoirement de sorte que tous les sujets n'aient pas le même texte présenté avec la même modalité.

Dans notre expérience, le test est effectué par 20 sujets entendants car le concours de sujets sourds ou malentendants nécessite les disponibilités de ces sujets et des orthophonistes qui les accompagnent afin de calibrer l'expérience. Les sujets ayant uniquement accès à la transcription écrite sur une feuille, le fait d'être entendant ou non ne joue aucun rôle à ce niveau. Chaque sujet doit traiter quatre textes, chacun avec une modalité différente. Une durée limite de 15 minutes par texte a été fixée pour laisser le temps au sujet de lire la transcription et de répondre aux questions. A la fin des 15 minutes, la transcription et les réponses sont reprises en échange de la transcription suivante.

Voici un exemple de la transcription présentée lorsque celle-ci est directement issue du système de reconnaissance :

> *lars abandonner sa vieille mère qui lui demander de travailler à la ferme ils ont voulu beaucoup parce qu' elle ne voulait pas que maître qui deviendrait quelqu'un d' important en allant faire danser les villageois par les balles un jour dans une forêt qui lança au génie de lui so le défilé joue du violon aussi bien que lui à cet instant d' une jeune fille est apparu et qui est lui le mandat de la faire danser lars accepta accorda son violon commença à jouer la jeune fifi quelque part plus arrêta presque aussitôt casque tu jour fit elle ça manque d' encre que le musicien choisi un nerf plus vif le jeune fils était toujours pas satisfaite est ce que je peux danser sur un air aussi l' ambition dit garce attaque un air le plus amer qui connaît situe n' est pas content de celui là dit il qu' il faudra faire venir un musicien plus habile que moi*

Le même texte mais en mettant en valeur les mots incorrects se présente ainsi (mode oracle) :

> *lars* **abandonner** *sa vieille mère qui lui* **demander** *de travailler à la ferme* **ils ont voulu** *beaucoup parce qu' elle ne voulait pas* **que maître qui** *deviendrait quelqu'un d' important en allant faire danser les villageois* **par** *les* **balles** *un jour dans une forêt* **qui** *lança au génie* **de lui so** *le* **défilé joue** *du violon aussi bien que lui à cet instant* **d'** *une jeune fille* **est apparu** *et* **qui est** *lui* **le mandat** *de la faire danser lars accepta accorda son violon commença à jouer la jeune* **fifi quelque part plus** *arrêta presque aussitôt* **casque** *tu* **jour** *fit elle ça manque d'* **encre que** *le musicien* **choisi** *un* **nerf** *plus vif* **le** *jeune* **fils** *était toujours pas satisfaite est ce que je peux danser sur un air aussi* **l' ambition** *dit* **garce**

attaque un *air le plus* **amer qui connaît** **situe** *n'* **est** *pas* **content** *de celui là dit il* **qu'** *il faudra faire venir un musicien plus habile que moi*

La modalité intégrant la mesure de confiance pour ce passage se traduit ainsi :

lars abandonner sa **vieille mère qui** *lui* **demander** *de travailler à* **la ferme** *ils ont voulu beaucoup parce qu' elle ne* **voulait** *pas* **que maître** *qui deviendrait quelqu'un d'* **important** *en allant faire danser les villageois par les balles un jour dans une forêt* **qui** *lança au génie de* **lui** *so le* **défilé joue** *du violon aussi bien que lui* **à** *cet instant* **d'** *une jeune fille* **est** *apparu et* **qui est lui le** *mandat de la* **faire danser** *lars accepta* **accorda** *son violon commença à jouer* **la jeune fifi quelque** *part plus arrêta presque aussitôt casque tu jour fit* **elle ça** *manque d'* **encre que** *le musicien choisi* **un nerf** *plus vif* **le** *jeune* **fils** *était toujours pas* **satisfaite** *est ce que je peux danser sur un air aussi* **l' ambition dit garce** *attaque* **un** *air le plus amer qui connaît* **situe** *n' est pas content de* **celui là** *dit* **il** *qu' il faudra faire venir un musicien plus* **habile** *que* **moi**

Enfin, la même transcription intégrant la phonétisation des mots de faible confiance :

lars abandonner sa **v_y_ai_y_m_ai_r_k_i** *lui* **d_e_m_an_d_é** *de travailler à* **l_a_f_ai_r_m** *ils ont voulu beaucoup parce qu' elle ne* **v_ou_l_é** *pas* **k_m_ai_t_r** *qui deviendrait quelqu'un d'* **in_p_o_r_t_an** *en allant faire danser les villageois par les balles un jour dans une forêt* **k_i** *lança au génie de* **l_u_i** *so le* **d_é_f_i_l_é_j_ou** *du violon aussi bien que lui* **a** *cet instant* **d** *une jeune fille* **é** *apparu et* **k_i_é_l_u_i_l** *mandat de la* **f_ai_r_d_an_s_é** *lars accepta* **a_k_o_r_d_a** *son violon commença à jouer* **l_a_j_eu_n_f_i_f_i_k_ai_l_k_e** *part plus arrêta presque aussitôt casque tu jour fit* **ai_l_e_s_a** *manque d'* **an_k_r_e_k_e** *le musicien choisi* **in_n_ai_r** *plus vif* **l_e** *jeune* **f_i_l** *était toujours pas* **s_a_t_i_s_f_ai_t** *est ce que je peux danser sur un air aussi* **l_an_b_i_s_y_on_d_i_t_g_a_r_s_e** *attaque* **in** *air le plus amer qui connaît* **s_i_t_u** *n' est pas content de* **s_e_l_u_i_l_a** *dit* **i_l** *qu' il faudra faire venir un musicien plus* **a_b_i_l** *que* **m_ou_a**

Voici la version d'origine qui a été prononcé :

Lars a abandonné sa vieille mère qui lui demandait de travailler à la ferme. Il lui en voulait beaucoup parce qu'elle ne voulait pas admettre qu'il deviendrait quelqu'un d'important en allant faire danser les villageois dans les bals. Un jour, dans une forêt, il lança au génie des ruisseaux le défi de jouer du violon aussi bien que lui. A cet instant, une jeune fille apparut et lui demanda de la faire danser. Lars accepta, accorda son violon, commença à jouer ; la jeune fille fit quelques pas, mais s'arrêta presque aussitôt. « Qu'est-ce que tu joues ? fit-elle. Ça manque d'entrain. » Le musicien choisit un air plus vif, mais la jeune fille n'était toujours pas satisfaite. « Est-ce que je peux danser, sur un air aussi languissant ? dit-elle. » Lars attaqua l'air le plus alerte qu'il connût . « Si tu n'es pas contente de celui-là, dit-il, il faudra faire venir un musicien plus habile que moi. »

Pour chaque test et chaque modalité, nous avons évalué la question concernant la ré-écriture d'une partie de la transcription en calculant le taux d'erreur en mot (taux d'insertion, omission, substitution) en ne tenant pas compte des fautes d'orthographe ou de grammaire afin que chaque sujet ne soit pas pénalisé par son niveau de français. Cette approximation n'est pas un problème car nous voulons tester la compréhension et ce type de faute gène généralement peu la compréhension.

Le tableau 6.5 présente les résultats en taux d'erreur en mots obtenus par texte et par modalité. Nous pouvons remarquer que sans aucune indications, les sujets sont à même de corriger des erreurs dans le texte et donc diminuer le taux d'erreur de départ des transcriptions. C'est par rapport à ce taux obtenu pas les sujets sur la transcription issue de la reconnaissance que nous basons nos comparaisons. Notons également que les passages choisis pour la ré-écriture ne sont pas égaux en difficultés avec pour certains des taux d'erreur bien supérieur à celui du texte dans son intégralité.

Si nous analysons plus précisément les résultats du tableau, nous pouvons remarquer les indications, quelque soit leur origine et leur forme (oracle, confiance, phonétique), permettent de diminuer le taux d'erreur en mots de chaque partie de texte sélectionnée. La méthode oracle, indiquant exactement les mots faux de la transcription, obtient dans 3 cas sur 4 des taux d'erreur plus faible. Par contre, la méthode proposant une phonétisation simplifiée des mots de faible confiance obtient quant-à elle un plus faible taux d'erreur dans 3 cas sur 4.

Bien que le faible nombre de sujets par texte et modalité (5) ne permettent pas de dire que les taux observés sont significatifs, nous ne pouvons que remarquer la tendance qu'ont les méthodes apportant une information de confiance à aider à diminuer le taux d'erreur en mots de la transcription d'origine par rapport la présence d'aucune aide.

Tab. 6.5 – Taux d'erreur en mots sur les parties retranscrites des textes suivant les différentes modalités.

texte	départ	aucune	confiance	oracle	phonétique
Le Mans	18,8%	11,0%	10,4%	9,0%	7,5%
Conte suédois	31,3%	11,6%	11,1%	16,4%	10,6%
Première expédition	43,9%	40,4%	27,4%	36,1%	29,1%
Vol du PC	20,5%	17,4%	16,4%	12,3%	14,9%

De plus, selon les réponses données par les sujets aux questions subjectives d'appréciation, la transcription phonétique semble plus aider à corriger les textes et donc les comprendre que les transcriptions avec coloration des mots de faible confiance. En effet, beaucoup de sujet indiquent que les mots entier écrits en couleurs ont tendance à les laisser croire que les autres mots sont justes. De plus, les mots en couleur ont également tendance à orienter l'esprit sur une recherche de substitution plus proche en sens ou de même racine, alors que la transcription phonétique laisse plus de liberté à rechercher des sons phonétiquement proches. Toutefois, le fait que les sujets ont préféré la modalité utilisant la phonétique doit être validé avec de vraies personnes sourdes ou malentendantes car les sujets entendants n'ont sans doute pas le même rapport à une mémoire phonétique que des gens n'ayant jamais entendus.

Les questions d'appréciations sont données en annexe A.4.

Concernant les questions, nous avons attribué des points selon le nombre de bonnes réponses trouvées puis, nous avons normalisé les scores afin d'obtenir une valeur entre 0 et 1. Plus le score

est proche de 1 et mieux les sujets ont répondu aux questions. Le tableau 6.6 montre les résultats obtenus aux questionnaires de compréhension. Malheureusement ces résultats ne permettent pas de conclure sur un effet ou non de l'introduction d'une notion de confiance pour les réponses aux questions. Une analyse plus approfondie montre que pour la plupart des questions, soit tout le monde a trouvé, soit personne n'a trouvé. Dans quelques cas, certains sujet trouve une réponse mais cela représente un ou deux sujet sur les 20 du test. Ainsi, les questions portant précisément sur des mots bien ou mal reconnus ne sont pas un moyen de distinguer une influence des mesures de confiance ou tout autre aide à la compréhension. Il est nécessaire d'explorer une autre méthode ou se concentrer sur la ré-écriture d'une partie du texte pour pouvoir évaluer des travaux similaires.

Tab. 6.6 – Taux de réponse aux questions des textes selon les différentes modalités.

texte	aucune	confiance	oracle	phonétique
Le Mans	0,9	0,9	0,9	1,0
Conte suédois	0,9	0,7	0,9	0,8
Première expédition	0,7	0,6	0,6	0,7
Vol du PC	0,3	0,5	0,5	0,3

6.5 Conclusion

Dans ce chapitre, nous avons évalué la faisabilité et l'intérêt des mesures de confiance pour trois applications :

- la diminution du taux de fausses alarmes dans le cas de la détection de mots clés,
- la diminution du taux d'erreur en mots d'un système de reconnaissance par l'intégration d'une mesure de confiance dans le processus de décodage,
- l'amélioration de la compréhensibilité par la mise en valeur de mots de faible confiance, notamment par leur phonétisation.

Bien que ces expériences soient des études de faisabilité, les observations faites sont prometteuses à tous points de vue.

Pour la tâche de détection de mots clés, la mesure de confiance permet de rejeter la totalité des fausses alarmes avec une diminution de seulement 10% du nombre de vraies alarmes restantes sur le corpus de développement. Sur le corpus de test, les résultats obtenus sont différents, ce qui laisse penser que la liste des mots clés n'est sans doute pas assez importante.

L'intégration d'une mesure de confiance trame-synchrone dans le processus de décodage du moteur de reconnaissance a permis de diminuer le taux d'erreur en mots du système, que ce soit en fin de première passe (diminution de 4% en relatif) ou à la fin du processus complet de reconnaissance (diminution de 6% en relatif).

Enfin, dans un cas concret d'aide aux personnes sourdes et malentendantes, les mesures de confiance ont montré qu'elles pouvaient améliorer la compréhensibilité d'une transcription issue d'un système de reconnaissance en mettant en valeur les mots de faible confiance.

Conclusion et perspectives

Au cours de cette étude, nous avons étudié la problématique des mesures de confiance en reconnaissance automatique grand vocabulaire de la parole en flux continu.

Nous nous sommes principalement intéressés à trois applications dans ce contexte : la détection de mots clés, l'intégration d'une mesure de confiance dans le moteur de reconnaissance et la transcription de cours en salle de classe.

La mesure de confiance peut être utilisée de multiples façons en post-traitement : mise en valeur des mots de faible confiance, intégration dans la phase de décodage du système de reconnaissance. Par exemple, dans une application de transcription, les erreurs ainsi mises en valeur pourront aider à leur correction manuelle sans nécessiter de traiter l'intégralité du texte.

Les applications que nous visons impliquent des contraintes sur le calcul des mesures de confiance : elles ne peuvent utiliser que des données disponibles à l'instant de traitement du moteur de reconnaissance.

Nous avons ainsi défini des mesures de confiance trame-synchrones et locales :

- les mesures trame-synchrones peuvent être calculées exactement à la même trame que le processus de décodage et n'utilisent que le voisinage passé du mot analysé.
- les mesures locales n'utilisent que des informations limitées à un voisinage, passé et futur, du mot dont nous voulons estimer la confiance.

Nous avons choisi de fonder ces mesures soit sur un rapport de vraisemblance, soit sur une estimation de la probabilité *a posteriori*.

Les données nécessaires au calcul de ces deux types de mesure de confiance sont extraites du graphe de mots que le moteur de reconnaissance génère pendant la phase de décodage de la phrase.

Les mesures de confiance trame-synchrones, fondées sur un rapport de vraisemblance, se distinguent principalement par le degré du modèle de langage utilisé pour prendre en compte le contexte passé du mot analysé : unigramme, bigramme et trigramme. Nous avons défini, principalement pour la mesure bigramme, quelques variantes concernant la gestion des mots concurrents (maximisation, sommation), la sélection des mots précédents (temporels directs, Viterbi, filtrage n-meilleures phrases) mais aussi l'homogénéisation des valeurs de confiance, l'utilisation de la probabilité bigramme inverse ou l'utilisation de la probabilité bigramme seule.

Les mesures de confiance locales estiment la probabilité *a posteriori* d'un mot uniquement à partir du sous-graphe de mots associé à un voisinage du mot analysé. Ces mesures n'ont à leur disposition que des informations sur un voisinage qui couvre de façon limitée à la fois le passé et le futur du mot analysé. Nous avons défini des mesures à voisinage symétrique : le voisinage de part et d'autre du mot analysé est de même taille (en trames) ; et des mesures à voisinage asymétrique : les deux voisinages passé et futur sont définis indépendamment. Les mesures asymétriques permettent d'augmenter la portée des informations passées afin d'améliorer la pertinence des mesures, sans augmenter le voisinage futur.

En plus de nos mesures de confiance, nous avons choisi une mesure de référence fondée sur

l'estimation de la probabilité *a posteriori* par la méthode proposée par Wessel et al. [Wessel 01] mais dont le calcul nécessite le décodage de l'intégralité de la phrase.

Pour pouvoir évaluer et comparer nos mesures de confiance, nous avons défini deux corpus d'émissions radiophoniques :

- un corpus de développement d'une heure permettant de déterminer pour chaque mesure ses paramètres optimaux,
- un corpus de test permettant d'évaluer nos mesures et de les comparer entre elles mais également avec la mesure de référence.

Toutes nos évaluations sont tributaires du choix du système de reconnaissance utilisé. Nous avons choisi *Julius*, un système de reconnaissance grand vocabulaire combinant une passe avant fondée sur l'utilisation de l'algorithme de Viterbi et une passe arrière fondée sur l'algorithme A^*.

Un seuil de décision est habituellement associé à la définition de mesure de confiance. La comparaison entre la valeur de confiance d'un mot et ce seuil détermine si le mot accepté ou rejeté par la mesure de confiance comme étant correct. Sur le corpus de développement, nous avons évalué nos différentes mesures suivant le critère du taux d'égale erreur (EER) qui permet de comparer les mesures indépendamment de toute application, en ne favorisant ni le taux de faux rejets (FR), ni le taux de fausses acceptations (FA).

A partir du taux d'EER, et du seuil de décision associé, calculé pour chaque mesure sur le corpus de développement, nous déterminons les taux FA et FR de celle-ci. Pour comparer plus facilement sur le corpus de test les mesures entre elles ainsi qu'avec la mesure de référence, nous avons utilisé le taux d'erreur de confiance CER.

En plus d'une évaluation sur le critère du taux d'EER et de taux CER, nous avons mis en œuvre certaines mesures de confiance dans trois applications : la détection de mot clés, l'intégration d'une mesure de confiance dans le moteur de reconnaissance et la transcription de cours pour des élèves sourds ou malentendants.

Conclusion au niveau des mesures de confiance

Mesures trame-synchrones

Nos mesures de confiance trame-synchrones sont fondées sur l'utilisation de connaissances issues du contexte passé du mot analysé. Ces mesures se distinguent principalement par le degré du modèle de langage utilisé : unigramme, bigramme et trigramme. Nous avons évalué chacune de ces mesures avec pour certaines quelques variantes, principalement pour la mesure bigramme. Ces variantes concernent la gestion des mots concurrents (maximisation, sommation), la sélection des mots précédents (temporels directs, Viterbi, filtrage n-meilleures phrases) mais aussi l'homogénéisation des valeurs de confiance, l'utilisation de la probabilité bigramme inverse ou l'utilisation de la probabilité bigramme seule.

A partir de ces expérimentations, nous pouvons remarquer que plus la mesure de confiance prend en compte un contexte passé important, meilleures sont ses performances. En effet, les résultats des mesures de confiance utilisant les mêmes méthodes de gestion, les mêmes définitions des précédents et les probabilités n-grammes directes, montrent que la mesure fondée sur la probabilité trigramme est significativement meilleure que la mesure bigramme, eux-mêmes significativement meilleures que la mesure unigramme ; respectivement 34,7%, 36,4% et 37,5% de taux CER sur le corpus de test pour les mesures avec gestion par maximisation et prédécesseurs temporels. L'écart entre les performances de ces mesures est moins marqué sur le corpus de développement avec respectivement 37,0%, 37,4% et 37,6% de taux EER.

De plus, les mesures obtenant les meilleures performances sont celles intégrant une connaissance qui n'est pas présente ou pas utilisée par le moteur de reconnaissance. Plus précisément, la mesure trigramme et la mesure bigramme inverse. Ces deux mesures obtiennent le même taux d'EER sur le corpus de développement (37,0%) et sont les deux meilleures mesures trame-synchrones sur le corpus de test : 34,7% de taux CER pour la mesure trigramme et 35,7% pour la mesure bigramme inverse. Ces mesures étant très proches des calculs effectués par le moteur de reconnaissance pendant la phase de décodage.

Si nous comparons la mesure bigramme prenant en compte pour un mot tous les prédécesseurs temporels se trouvant dans le graphe et la mesure bigramme ne prenant en compte que le prédécesseur au sens de Viterbi, nous remarquons que cette dernière est significativement plus mauvaise d'environs 3% en absolu sur le corpus de développement (en terme d'EER) et également sur le corpus de test (en terme de CER). Nous pensons que les mesures n'utilisant que les précédents au sens de Viterbi se rapprochent trop du processus de reconnaissance et peuvent donc difficilement donner une décision différente de celle du système. De plus ne prendre en compte que les précédents au sens de Viterbi est beaucoup trop restrictif pour obtenir une modélisation de l'hypothèse alternative du rapport de vraisemblance plus précise.

Par ailleurs, les expériences sur le corpus de développement concernant l'homogénéisation des valeurs de confiance a permis de montrer qu'il y a une bonne corrélation entre les valeurs de confiance calculées par nos mesures et le fait qu'un mot soit correct ou incorrect. En effet, nous avons observé que lorsque la valeur de confiance augmente, la proportion de mots corrects augmente également. Ceci permet de déduire que nos mesures extraient bien une information de corrélation entre les valeurs de confiance calculées et le taux de mots corrects.

Nous avons également analysé la pertinence de nos mesures de confiance en fonction de la taille des mots pour lesquels elles sont calculées. Nous avons alors observé que les mesures de confiance fondées sur la probabilité *a posteriori* semblent plus enclines à l'estimation de la confiance des mots de taille moyenne (4 à 6 phonèmes) alors que les mesures trame-synchrones sont plus précises pour les mots longs (plus de 6 phonèmes).

Sachant que notre meilleure mesure locale obtient un taux d'EER de 22,3% et que notre meilleure mesure trame-synchrone obtient un taux d'EER de 37%, nous pourrions nous étonner de ces résultats mitigés. Mais si nous considérons la mesure locale asymétrique calculée sur une taille de voisinage passé de 40 trames et un voisinage futur nul, le taux d'EER obtenu est de 33,3% contre 37,0% pour notre mesure bigramme inverse alors que la taille du voisinage passé de ces deux mesures est équivalente. En revanche, les mesures bigrammes utilisent moins d'informations que les mesures locales (probabilités acoustiques des mots précédents). De plus, pour les mesures trame-synchrones, le rapport de vraisemblance est calculé entre des bigrammes, c'est-à-dire des séquences d'exactement deux mots. Or, les mesures fondées sur les probabilités *a posteriori* considèrent des chemins entre des séquences de longueur quelconque en nombre de mots mais appartenant à un voisinage de taille en trames fixée. En effet, même pour un voisinage de taille équivalente à la longueur d'une séquence bigramme, le graphe de mots associé ne contient pas que des chemins contenant deux mots. Ainsi, il est normal que la mesure fondée sur la probabilité *a posteriori* soit plus précise que le rapport de vraisemblance.

Mesures Locales

Nos mesures locales obtiennent de très bons résultats, que ce soit sur le corpus de développement ou sur le corpus test. Par exemple, notre mesure locale symétrique avec un voisinage de 84 trames atteint des performances quasiment identiques à la mesure de référence sur le corpus de développement (respectivement 23% et 22% d'EER). De même, sur le corpus de test, les taux de

CER de ces deux mesures sont respectivement de 23,9% pour la mesure locale et 22,1% pour la mesure de référence. Or, le calcul de la mesure référence nécessite le traitement de l'intégralité de la phrase alors que la mesure locale se contente d'un voisinage total de 168 trames en plus de la longueur du mot (1 trame vaut 10 ms). Ceci montre qu'il est possible de définir une bonne mesure de confiance fondée sur la probabilité *a posteriori* mais calculée que sur un court voisinage du mot analysé.

Par ailleurs, les mesures locales asymétriques montrent qu'en prenant une mesure avec un voisinage passé depuis le début de la phrase et un voisinage futur de seulement 60 trames (0,6 s), nous obtenons les mêmes performances que la mesure locale symétrique avec un voisinage de 84 trames sur le corpus de développement. Cette mesure locale asymétrique permet ainsi d'utiliser le maximum d'informations passées afin de compenser le manque de connaissance du futur. Sur le corpus de test, cette mesure asymétrique est même meilleure que la mesure locale symétrique (respectivement 23,1% et 23,9%).

De plus, il est possible de rendre nos mesures locales trame-synchrones en forçant un voisinage futur nul. Par exemple, la mesure asymétrique locale prenant en compte les informations depuis le début de la phrase mais avec un voisinage futur nul obtient un taux d'EER honorable de 30,1%, confirmé sur le corpus de test avec un taux CER de 29,6%.

Ceci indique que choisir cette méthode pour définir des mesures trame-synchrones est tout à fait envisageable. De plus, d'un point de vue complexité, cette mesure locale particulière admet quasiment la même complexité que notre meilleure mesure trame-synchrone, la mesure bigramme inverse (respectivement $O(TVN^3)$ et $(O(kTN^3)$ avec $k < 1$, cf. section 3.5).

Ainsi, nous avons réussi à définir des mesures de confiance qui peuvent être utilisées dans des applications de reconnaissance automatique de la parole grand vocabulaire et en flux.

Ces mesures peuvent être trame-synchrones et fondées soit sur un rapport de vraisemblance soit sur la probabilité *a posteriori*, et obtenir des résultats de l'ordre de 30% d'EER ; ou bien ces mesures peuvent nécessiter un court délai avant de pouvoir être calculées mais obtenir des performances très proches de la mesure de référence qui nécessite le décodage de l'intégralité du signal. Par exemple, 60 trames (0,6 s) de délai suffisent pour notre mesure locale asymétrique prenant en compte toutes les informations depuis le début de la phrase pour obtenir un taux d'EER de 23,2% contre 22,0% pour les mesure de référence.

Nous pouvons également utiliser une mesure locale calculée uniquement sur une portion d'une phrase comme notre mesure symétrique avec un voisinage de 84 trames, par exemple pour une vérification à la demande par exemple. Cette mesure obtient en effet des performances également proches de la mesure de référence (1% à 2% que ce soit en EER ou en CER).

Conclusion au niveau des applications visées

Nous nous sommes intéressés à trois applications : la détection de mots clés, l'intégration d'une mesure de confiance dans le moteur de reconnaissance et la transcription de cours en salle de classe pour des étudiants sourds ou malentendant. Nous allons présenter la façon dont nous avons utilisé les mesures de confiance pour ses applications ainsi que les conclusions que nous avons pu observer.

Détection de mots clés

Dans le cadre de l'application de détection de mots clés, nous avons utilisé et comparé différentes mesures trame-synchrones et locales que nous avons définies afin d'étudier la faisabilité de l'utilisation d'une de nos mesure de confiance afin de trouver le meilleur point de fonctionnement qui permet de concilier au mieux la diminution du nombre de fausses acceptations et la diminution du nombre de vraies acceptations.

A notre connaissance, les mesures de confiance fondées sur le rapport de vraisemblance ou sur une estimation de la probabilité *a posteriori* n'ont pas été utilisées dans le cadre de détection de mots clés en grand vocabulaire.

Pour cette application, nous avons défini une liste courte de mots clés (33). A partir de l'analyse sur le corpus de développement de l'évolution du nombre de fausses et de vraies acceptations en fonction du seuil de décision de la mesure de confiance, nous avons décidé de choisir comme point de fonctionnement le seuil à partir duquel le nombre de fausses acceptations est quasi nul et ne décroît que lentement. Sur le corpus de développement (1 heure d'émission radiophonique), la mesure trigramme permet une diminution de 87,5% du nombre de fausses acceptations avec une perte des vraies acceptations de 11,5%. La mesure locale symétrique avec voisinage de 84 trames rejette toutes les fausses acceptations avec une perte de seulement 10,7% de vraies acceptations.

Une fois les seuils associés aux points de fonctionnement choisis, nous évaluons les mêmes mesures de confiance sur le corpus de test. La meilleure mesure sur ce corpus est la mesure bigramme avec maximisation, celle-ci obtient une diminution de 63,6% du nombre de fausses acceptations et une perte de 20,9% de vraies acceptations. La meilleure mesure locale considère un voisinage passé depuis le début de la phrase et un voisinage futur de 84 trames (840 ms). Cette mesure diminue le nombre de fausses acceptations de 54,5% avec une perte de 16,5% de vraies acceptations.

Nous pouvons remarquer que l'utilisation de nos mesures de confiance, trame-synchrones ou locales, peut aider à la diminution du nombre de fausses acceptations avec une perte assez faible du nombre de vraies acceptations.

Les résultats obtenus par nos mesures dans cette application montrent un comportement et des points de fonctionnement différent des mesures entre le corpus de développement et de test. L'origine de ces divergences vient sans doute de la trop petite liste de mots clés utilisée dans cette étude de faisabilité. Afin de confirmer ou affiner ces résultats et observation, ces expériences devraient être menées avec une liste plus importante, contenant plusieurs centaines de mots clés, définis par exemple conjointement avec les besoins d'une entreprise.

Intégration d'une mesure de confiance dans le système de reconnaissance Julius

Nous avons réalisé une étude de faisabilité de l'intégration d'une valeur de confiance dans le processus de décodage du système de reconnaissance Julius. Pour cela, nous avons défini deux systèmes : Julius sans notre mesure de confiance et Julius avec notre mesure de confiance.

La façon d'intégrer une mesure de confiance dans un système de reconnaissance est très dépendant des algorithmes employés dans celui-ci. Nous avons décidé d'utiliser une mesure trame-synchrone afin de pouvoir l'intégrer au cours de la phase de décodage du moteur. La mesure de confiance est ainsi calculée pour chaque mot apparaissant dans le graphe généré par le moteur. Puis nous définissons pour ces mots un nouveau score de vraisemblance. Pour chaque mot, nous combinons la vraisemblance calculée par le moteur à la valeur de confiance de ce mot. Le processus de décodage se poursuit alors à partir de ces nouveaux scores. Par conséquent, à la fois le graphe

de mots et le score heuristique utilisé par l'algorithme A^* sont modifiés.

Pour cette expérimentation, nous avons choisi d'intégrer dans le moteur Julius la mesure de confiance bigramme avec homogénéisation des valeurs. Nous avons extrait 51 phrases du corpus de développement pour lesquelles nous avons calculé le taux d'erreur en mots du système intégrant ou non la mesure de confiance. Cette expérience montre que le système Julius intégrant la mesure de confiance améliore la reconnaissance si nous ne considérons que le résultat à l'issue de la première passe : 37,1% avec la mesure et 38,3% sans la mesure. Par ailleurs, si nous poursuivons le processus de reconnaissance avec l'exécution de la deuxième passe, le taux d'erreur en mots du système est également amélioré pour Julius intégrant la mesure de confiance : 29,6% avec la mesure et 30,8% sans la mesure.

Toutefois, le corpus utilisé ne contenant que 51 phrases, les conclusions de cette étude de faisabilité devront être confirmées sur le corpus de test complet. De même, il serait intéressant de comparer l'influence de l'intégration d'une mesure trame-synchrone à la première passe du moteur de reconnaissance par rapport à l'utilisation d'une mesure en deuxième passe.

Transcription de cours en salle de classe

La dernière application pour laquelle nous avons introduit nos mesures de confiance concerne la transcription de cours en salle de classe pour des élèves sourds ou malentendants.

Cette application a été mise en place dans le cadre du projet RIAM LABIAO dont l'objectif est de développer un ensemble de logiciels permettant aux sourds ou malentendants d'être plus autonomes. La participation de notre équipe de recherche à ce projet s'est concrétisé par la création d'un logiciel proposant deux modalités visuelles distinctes :

- une tête parlante qui va à la fois articuler les phonèmes et ajouter à l'aide d'une main artificielle le codage en Langage Parlé Complété des sons,
- une transcription dont le rythme d'affichage suit le rythme d'élocution du locuteur.

Ces deux modalités sont pilotées par le résultat d'un système de reconnaissance.

Or pour ces deux modalités, le résultat brut de la reconnaissance a été directement utilisé, mêlant mots corrects et mots incorrects sans distinctions. Nous avons alors proposé d'utiliser nos mesures de confiance afin d'indiquer les mots potentiellement incorrects à l'aide d'une modalité complémentaire : changement de couleur de la transcription ou la tête parlante selon la valeur de confiance du mot. Nous espérons que grâce à ces indications les étudiants sourds ou malentendants pourront corriger plus facilement la transcription et retrouver le sens de la phrase d'origine.

Le but de l'expérience que nous avons menée est d'évaluer l'influence de l'utilisation d'une indication de confiance sur la compréhensibilité des étudiants sourds ou malentendants. Pour cela, nous avons considéré la modalité de transcription rythmée. Nous avons proposé les deux modalités suivantes pour les mots dont la confiance est inférieure à un seuil de décision :

- la coloration de ces mots,
- la coloration et la phonétisation de ces mots dans un alphabet phonétique simplifié.

Nous avons utilisé la mesure de confiance locale symétrique avec un voisinage de 84 trames car celle-ci obtient de bons résultats tout en ne nécessitant que la connaissance d'un court voisinage du mot analysé. Le seuil de décision a été choisi comme le seuil associé au taux d'EER de cette mesure sur le corpus de développement d'une heure.

Afin d'évaluer l'influence de ces nouvelles modalités, nous avons défini un corpus de textes calibrés utilisés dans des tests de lecture dans les écoles. Les textes et les questions associées permettent de déterminer le niveau de compréhension de l'élève. Ces textes ont été enregistrés par une même personne dans des conditions similaires puis ces enregistrements ont été transcrits par le système de reconnaissance. Nous calculons pour chaque mot de ces transcriptions brutes

une valeur de confiance. Ensuite pour une des modalités intégrant les valeurs de confiance, nous évaluons la compréhensibilité des testeurs ainsi que leur appréciation vis-à-vis de la modalité.

Nous avons réalisé la reconnaissance des enregistrements et avons également calculé les valeurs de confiance de chaque mot des transcriptions brutes obtenues. Les tests des différentes modalités sont prêts, toutefois l'évaluation de cette expérimentation n'étant pas encore terminée, nous ne savons pas encore comment sont perçues ces nouvelles modalités.

Perspectives

Des perspectives à court terme peuvent être explorées en relation avec nos mesures de confiance afin de compléter certaines observations :

- nous avons remarqué que parmi les mesures trame-synchrones bigrammes, celle utilisant une connaissance qui n'est pas prise en compte dans le processus de décodage obtiennent les meilleurs résultats (mesure bigramme inverse). Nous pourrons évaluer si la prise en compte la probabilité trigramme inverse mène à des observations similaires ;
- d'un point de vue applicatif, nous avons obtenu des résultats prometteurs concernant le fait d'intégrer une mesure de confiance trame-synchrone dans le moteur de reconnaissance. Cette étude de faisabilité, réalisée sur 51 phrases du corpus de développement, nécessitera d'être validée sur le corpus de test. En outre, l'utilisation de la mesure bigramme inverse au lieu de la mesure bigramme directe devra être expérimentée. En effet la mesure bigramme inverse est la meilleure de nos mesures fondées sur un rapport de vraisemblance, tout en restant de complexité raisonnable. Nous avons également défini des mesures locales trame-synchrone fondées sur la probabilité *a posteriori*. Il pourra ainsi être intéressant d'intégrer une de ces mesures dans le moteur de reconnaissance (par exemple la mesure prenant en compte tout le voisinage passé et un voisinage futur nul) ;
- nous avons montré qu'à partir d'une liste de mots clés restreinte, l'utilisation de mesures de confiance pour la détection de mots clés permet de diminuer le nombre de fausses acceptation tout en conservant un maximum de vraies acceptations. Une validation sur une liste de mots clés plus importante reste cependant nécessaire ;
- l'application des mesures de confiance pour la transcription de cours en salle de classe nécessite la prise en compte de critères perceptifs. Dans les travaux que nous avons réalisés, nous avons choisi le seuil de décision associé au taux d'EER. Il n'est cependant pas établi que ce point de fonctionnement soit optimal dans ce cadre applicatif. En effet, l'exploration d'autres points de fonctionnement basés sur des critères perceptifs devra être menée ; en d'autres termes nous pourrons évaluer l'influence des fausses acceptations et des faux rejets sur l'utilisateur. Notre étude est fondée sur la modalité de transcription rythmée mais il faudra également faire les mêmes tests perceptifs pour la modalité utilisant la tête parlante.

A plus long terme, nous pourrons étudier la fusion de nos mesures de confiance avec des critères ou indices non utilisés par le moteur de reconnaissance ou extérieurs à celui-ci. En effet, parmi nos mesures nous avons observé que les mesures utilisant des connaissances différentes de celles impliquées dans le processus de décodage (bigramme inverse, trigramme) sont significativement meilleures. Ainsi la fusion avec des critères externes également trame-synchrones pourront améliorer la pertinence de la mesure. Des critères simples, présents dans le système de reconnaissance mais non utilisés pourront être considérés, comme par exemple la distribution des trames sur les états du modèle HMM d'un mot.

De plus, nous pourrons explorer l'utilisation d'autres critères potentiellement porteurs d'informations utiles tels que la prosodie, la vitesse d'élocution et d'autres critères phonétiques. Une direction très intéressante sera la prise en compte d'indices sémantiques et contextuels. En effet, une personne est capable d'identifier avec certitude un mot incorrect à partir du sens ou du contexte de la phrase. Nous pourrons utiliser des connaissances de différents domaines connexes comme l'ontologie et la fouille de données afin de déterminer le sens d'une phrase ou un ensemble de mots sémantiquement liés. L'objectif sera de définir ces indices ou critères de manière trame-synchrone.

Annexe A

A.1 Entropie croisée normalisée

Exemple du calcul de l'entropie croisée normalisée (NCE)

Soit la phrase issue du moteur de reconnaissance constituée des deux mots w_1 et w_2. Supposons que w_1 soit correctement reconnu et que par contre w_2 soit faux. Supposons également qu'une mesure de confiance calcule une valeur pc_1 et pc_2 pour respectivement les mots w_1 et w_2.

mot	reconnu	valeur de confiance
w_1	correct	pc_1
w_2	incorrect	pc_2

L'entropie initiale du système $H(S)$ est définie par l'équation 2.25 avec dans cet exemple $p_0 = 1/2$. Ainsi :

$$H(S) = -p_0 \log p_0 - (1-p_0)\log(1-p_0) = -\log 1/2 \simeq 0,3$$

L'entropie du système en prenant compte des indices issus de la mesure de confiance est définie par l'équation 2.26. Nous considérons un cas pour lequel la valeur de confiance attribuée est non informative ($pc_i = 0,5$) et 4 cas suivant pour lesquels $pc_i = \{0,9; 0,1\}$:

- le cas favorable où le mot correctement reconnu a une valeur de confiance forte ($pc_1 = 0,9$) et le mot incorrectement reconnu a une valeur de confiance faible ($pc_2 = 0,1$)
- le cas défavorable dans lequel la mesure de confiance a indiqué exactement l'inverse du résultat espéré
- le cas où les deux mots ont une valeur de confiance faible
- le cas où les deux mots ont une valeur de confiance forte

c_1	c_2	$H(S\|X)$	$H(S\|X) \simeq$	$NCE \simeq$
0,5	0,5	$-\frac{1}{2}(\log(0,5) + \log(0,5))$	0,3	0
0,9	0,1	$-\frac{1}{2}(\log(0,9) + \log(0,9))$	0,05	0,8
0,1	0,9	$-\frac{1}{2}(\log(0,1) + \log(0,1))$	1	-2,3
0,1	0,1	$-\frac{1}{2}(\log(0,1) + \log(0,9))$	0,5	-0.7
0,9	0,9	$-\frac{1}{2}(\log(0,9) + \log(0,1))$	0,5	-0.7

A.2 Taux d'erreur de confiance

Exemple de calcul du taux d'erreur de confiance (CER)

Soit un échantillon audio dont le contenu correspond à cette phrase :

```
REF:  POUR  L'  INSTANT  sur  france  inter  il  est  sept  heures
```

Supposons que le système de reconnaissance ait reconnu le résultat suivant :

```
HYP:  ****  EN  LAISSANT  sur  france  inter  il  est  sept  heures
```

Nous avons dans ce résultat : une omission (`POUR`) et deux substitutions (`L'` → `EN` et `INSTANT` → `LAISSANT`).

Le taux CER de référence correspond au taux d'une mesure de confiance qui accepte tous les mots comme justes (Eq. 2.24). Ceci correspond au taux de mots mal reconnus : on ne tient pas compte des omissions qui n'apparaissent pas dans le résultat. Ainsi pour la phrase ci-dessus que le système a déterminée, il y a 9 mots dont 2 faux et donc le CER de référence équivaut à $2/9 \simeq 22\%$.

Supposons maintenant qu'une mesure de confiance détermine pour chaque mot de la phrase son appartenance aux classes *Acceptation* (Acc) et *Rejet* (Rej) et que la phrase résultat soit étiquetée ainsi :

```
HYP:  ****  EN  LAISSANT  sur  france  inter  il   est   sept  heures
CONF:       Rej   Acc     Rej    Acc    Acc   Acc  Acc   Acc     Acc
```

Dans cet exemple, le mesure de confiance a déterminé une fausse acceptation (deuxième mot) et un faux rejet (troisième mot).

Le taux CER pour une mesure de confiance est calculée par le rapport entre le nombre de faux rejets et de fausses acceptations sur le nombre de mots reconnus (Equ. 2.23). Dans notre cas, le taux CER équivaut à $2/9 \simeq 22\%$, ce qui est exactement le taux de référence.

Supposons maintenant que la mesure de confiance ait étiquetée la phrase de cette manière :

```
HYP:  ****  EN  LAISSANT  sur  france  inter  il   est   sept  heures
CONF:       Rej   Acc     Acc    Acc    Acc   Acc  Acc   Acc     Acc
```

Dans ce cas, il n'y a qu'une seule fausse acceptation (deuxième mot) et aucun faux rejet. Le taux CER équivaut maintenant à $1/9 \simeq 11\%$. La mesure de confiance apporte un gain, ce que confirme la décroissance du taux CER vers 0.

A.3 Influence de la taille des mots

Pour un nombre n de phonèmes, nous avons étudié le taux d'égale erreur de la mesure de confiance en considérant uniquement les mots dont la décomposition en phonèmes est :

- supérieure ou égal à n phonèmes,
- strictement inférieure à n phonèmes.

Nous avons fait cette étude pour deux de nos mesures de confiance : la mesure de confiance trame-synchrone fondée sur la probabilité bigramme directe (gestion par maximisation et précédents temporels directs) et la mesure locale à voisinage symétrique de 84 trames. Pour comparaison, nous avons fait la même analyse pour la mesure de référence. Pour deux mesures les facteurs d'échelle utilisés sont $\alpha = 0,1$ et $\beta = 0,95$; le facteur de relâchement de la mesure trame-synchrone vaut $\varepsilon = 0,1$ et le facteur de flexibilité des mesures locales vaut $\eta = 0,5$ pour la mesure symétrique. Pour la mesure de référence, les facteurs d'échelle sont $\alpha = 0,1$ et $\beta = 1$ et $\eta = 1$.

Les figures A.1 à A.3 représentent l'évolution des taux d'EER respectivement de la mesure de confiance de référence, de la mesure locale symétrique et de la mesure trame-synchrone bigramme. Dans chacune des figures, les courbes correspondent au taux d'EER calculé pour les mots dont le nombre de phonèmes est soit supérieur soit inférieur à n. C'est-à-dire que pour un nombre de phonèmes valant 5, la courbe en trait plein indique le taux d'EER sur les mots de plus de 5 phonèmes (5, 6, etc.), la courbe en pointillés indique le taux d'EER pour les mots de strictement moins de 5 phonèmes (1, 2, 3 et 4 phonèmes).

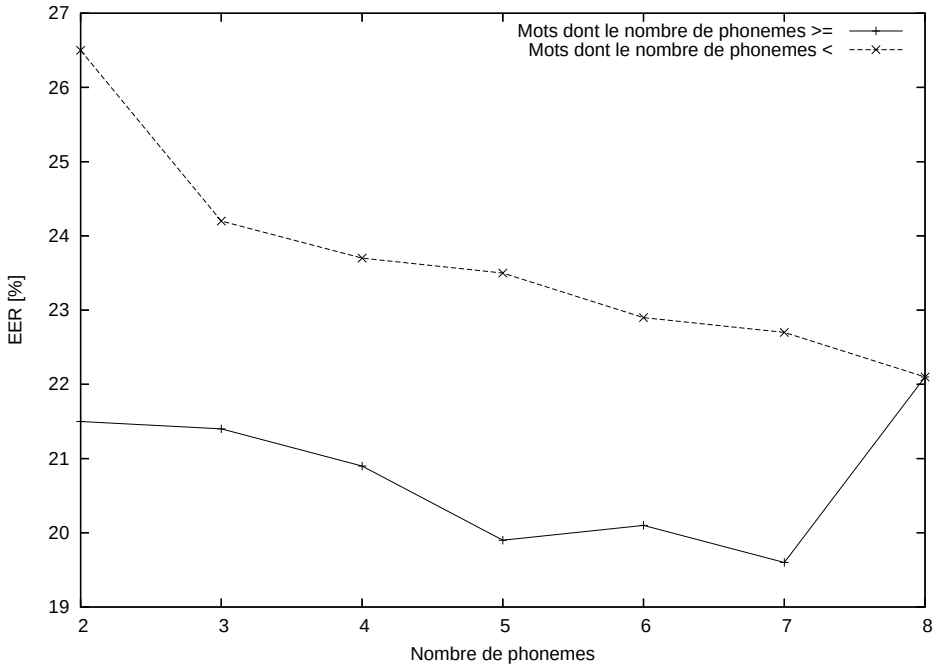


FIG. A.1 – Evolution des taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure référence avec le jeu de paramètres $(\alpha = 0,1)$, $(\beta = 1)$ et $(\eta = 1)$.

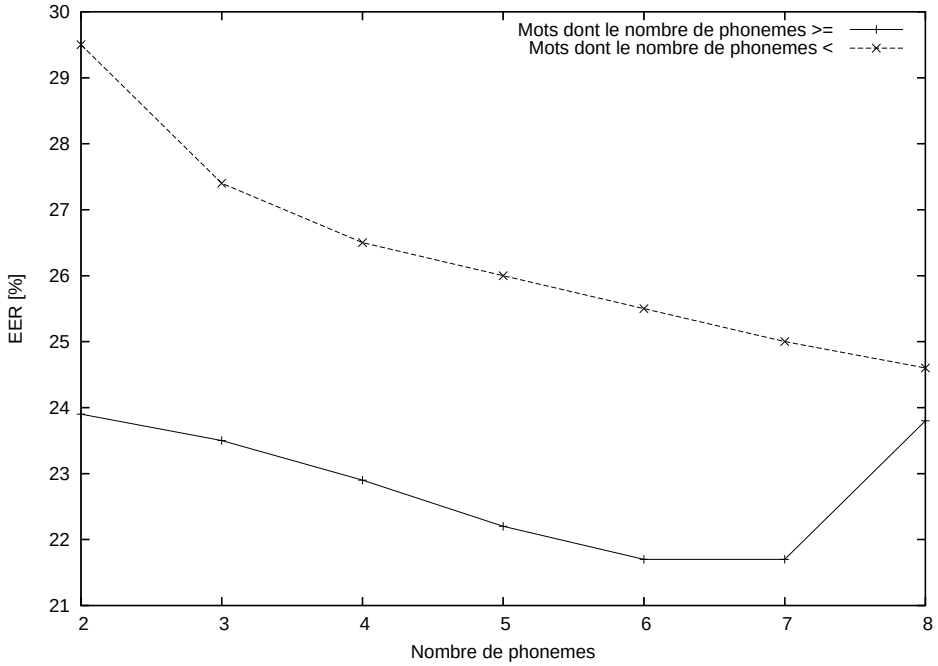


FIG. A.2 – Evolution des taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure de confiance locale avec voisinage symétrique de 84 trames, avec le jeu de paramètres $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\eta = 0,5)$.

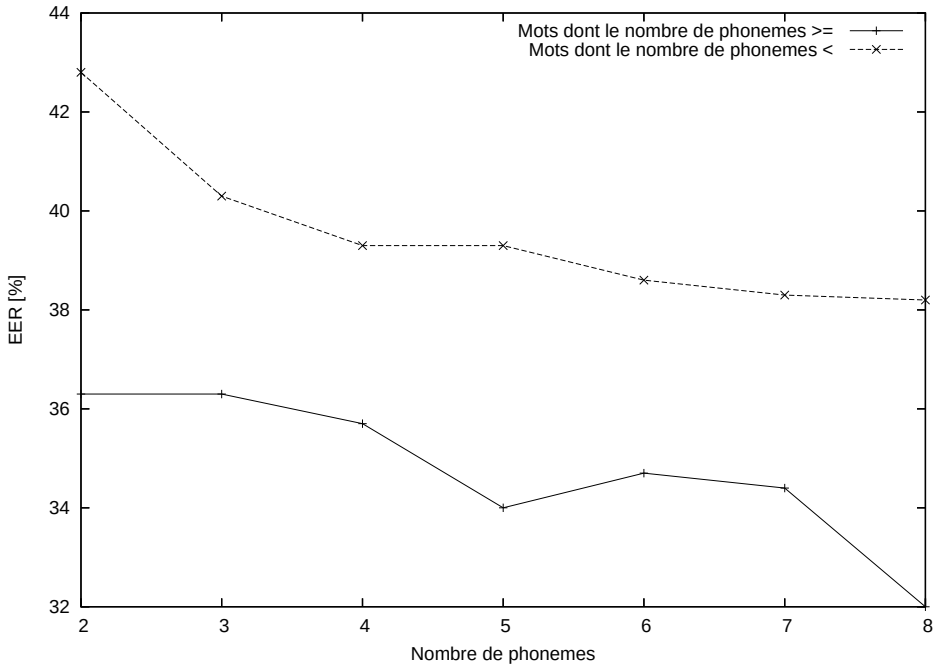


FIG. A.3 – Evolution des taux d'EER suivant la taille en phonèmes des mots analysés pour la mesure de confiance trame-synchrone bigramme direct avec le jeu de paramètres $(\alpha = 0,1)$, $(\beta = 0,95)$ et $(\varepsilon = 0,1)$.

A.4 Questionnaire pour l'évaluation des transcriptions pour malentendants

Voici les questions posées à la fin de chaque texte :

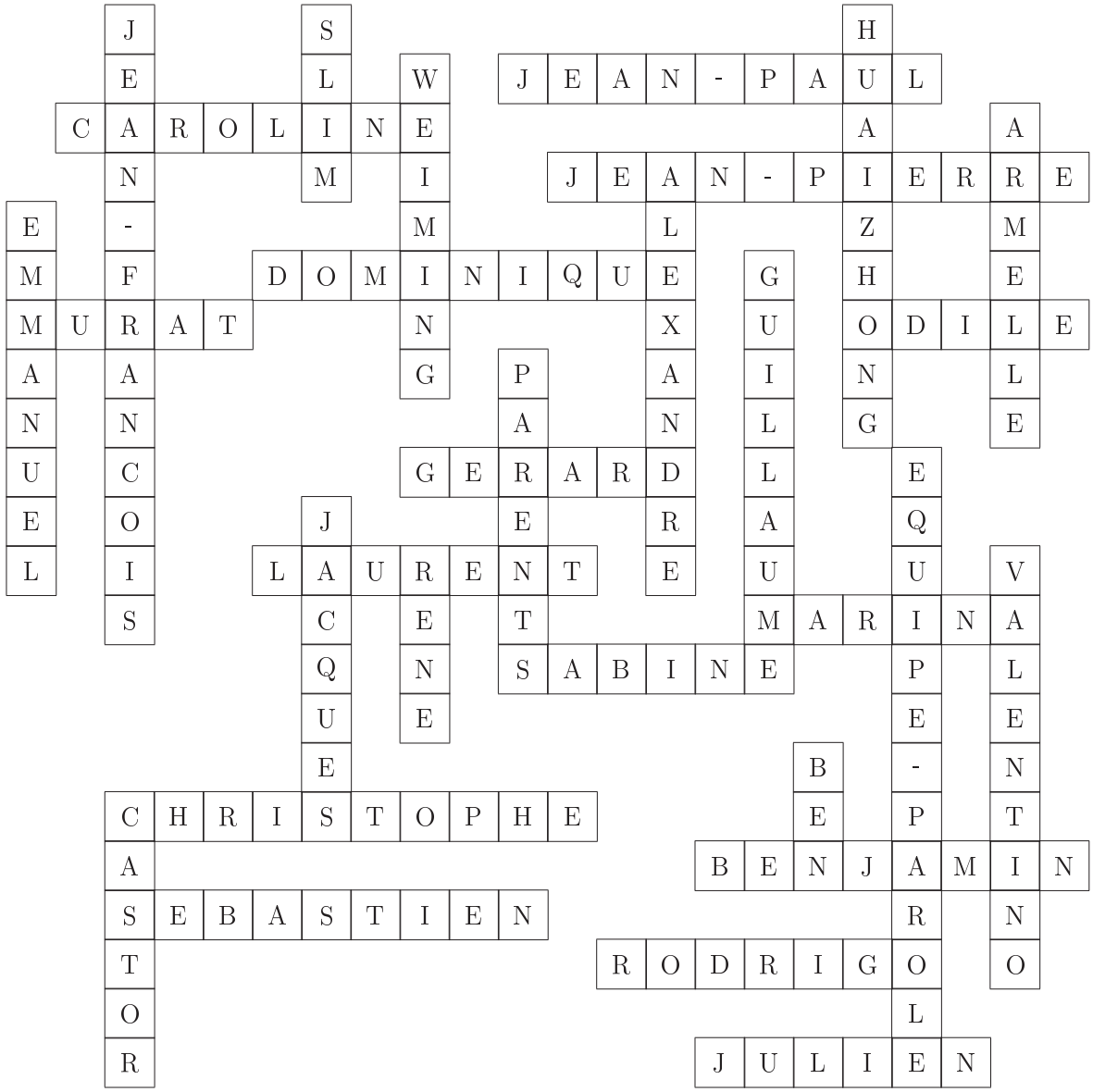
1. Est-ce qu'il a été difficile de répondre aux questions (1 très facile, 5 très difficile)
2. Est-ce qu'il a été difficile de corriger les erreurs de reconnaissance (1 très facile, 5 très difficile)
3. Que pensez-vous du texte ?
4. Que pensez vous de la manière de présenter le texte ?

A la fin du test complet, après les quatres textes, des questions d'ordre plus générales sont posées :

1. Que pensez vous des différentes méthodes de transcription en terme de :
   - compréhensibilité
   - effort pour suppléer

2. Que pensez-vous de l'ensemble des textes et du protocole ?
3. Avez vous des idées pour d'autres méthodes de transcription ?
4. Autres remarques :

Solution des remerciements



Glossaire

ASRU : Automatic Speech Recognition and Understanding

BBT : Bark Bilinear Transform

CER : Confidence Error Rate
CMN : Cepstral Mean Normalisation
CMS : Cepstral Mean Subtraction
CSLP : Chinese Spoken Language Processing

DET : Detection Error Tradeoff

EER : Equal Error Rate
EM : Expectation Maximisation
EUROSPEECH : European Conference on Speech Communication and Technology

GMM : Gaussian Mixture Model

HMM : Hidden Markov Model

ICASSP : International Conference on Acoustics, Speech, and Signal Processing
ICSLP : International Conference on Spoken Language Processing
IEEE Trans. ASSP : IEEE Transactions on Acoustics, Speech, and Signal Processing
IEEE Trans. PAMI : IEEE Transactions on Pattern Analysis and Machine Intelligence
IEEE Trans. SAP : IEEE Transactions on Speech and Audio Processing

JASA : Journal of the Acoustical Society of America
JEP : Journées d'Etude sur la Parole

LABIAO : lecture LABIale Assistée par Ordinateur
Latent Semantic Analysis : LSA
LLR : Log Likelihood Ratio
LPC : Langage Parlé Complété
LPC : Linear Predictive Coding
LR : Likelihood Ratio
LREC : International conference on Language Resources and Evaluation

MAP : Maximum *a posteriori*
MFCC : Mel Frequency Cepstral Coefficient
ML : Maximum Likelihood
MLLR : Maximum Likelihood Linear Regression
MMI : Maximum Mutual Information

NCE : Normalized Cross Entropy
NIST : National Institute of Standards and Technologies

PLP : Perceptual Linear Prediction

ROC : Receiver operating characteristic
ROVER : Recognizer Output Voting Error Reduction

SVD : Singular Values Decomposition

TLDP : Two-Level Dynamic Programming

WER : Word Error Rate

Bibliographie

[Agbago 04] A. Agbago et C. Barriere. *Fast two-level-dynamic-programming algorithm for speech recognition.* In ICASSP, pages 129–132, 2004.

[Anastasakos 98] T. Anastasakos et S.V. Balakrishman. *The Use of Confidence Measures in Unsupervised Adaptation of Speech Recognizers.* In ICSLP, pages 2303–2306, 1998.

[Baker 75] J.K. Baker. *The DRAGON system - An overview.* IEEE Trans. ASSP, vol. 23, no. 1, pages 24–29, 1975.

[Bansal 98] D. Bansal et M.K. Ravishankar. *New Features for Confidence Annotation.* In ICSLP, pages 2391–2394, 1998.

[Baum 70] L.E. Baum, T. Petrie, G. Soules et N. Weiss. *A maximization technique occuring in the stat. analysis of probabilistics functions of Markov chains.* Annals of Mathematical Statistics, vol. 41, pages 164–171, 1970.

[Bellagarda 98] J.R. Bellagarda. *A multispan language modeling framework for large vocabulary speech recognition.* IEEE Trans. SAP, vol. 6, no. 5, page 456, 467 1998.

[BenAyed 03] Y. BenAyed. *Détection de mots clés dans un flux de parole.* Ecole, Ecole Nationale Supérieure des Télecommunications (Paris), Dec 2003.

[Boite 93] J.M Boite, H. Boulard, B. D'hoore et M. Haesen. *A new approach towards keyword spotting.* In EUROSPEECH, volume 2, pages 1273–1276, 1993.

[Boutard 06] C. Boutard, I. Claire et L. Gret-Chanousay. Isbergues, 2006.

[Brill 98] E. Brill, R. Florian, J.C. Henderson et L. Mangu. *Beyond n-grams : can linguistic sophistication improve language modeling ?* In ACL, pages 186–190, 1998.

[Brown 92] P.F. Brown, V.J. DellaPietra, P.V. deSouza, J.C. Lai et R.L. Mercer. *Class based n-gram models of natural language.* Computational Linguistics, vol. 18, no. 4, pages 467–478, 1992.

[Charlet 01] D. Charlet, G. Mercier et D. Jouvet. *On combining confidence measures for improved rejection of incorrect data.* In EUROSPEECH, pages 2113–2116, 2001.

[Chase 97] L. Chase. *Word and acoustic confidence annotation for large vocabulary speech recognition.* In EUROSPEECH, Rhodes, pages 815–818, 1997.

[Chen 99] S.F. Chen et J.T. Goodman. *An Empirical Study of Smoothing Techniques for Language Modeling.* Computer Speech and Language, vol. 13, no. 4, pages 359–393, 1999.

[Chevrier-Muller 97] C. Chevrier-Muller, A.M. Simon et F. Fournier. L2ma. 1997.

[Chigier 92] B. Chigier. *Rejectin and keyword spotting algorithms for a directory assistance city name recognition application.* In ICASSP, volume 2, pages 92–96, 1992.

[Clarckson 97] P.R. Clarckson et R. Resenfeld. *Statistical Language Modelling Using the CMU-Cambridge Toolkit.* In EUROSPEECH, Rhodes, pages 2707–2710, 1997.

[Cohen 04] I. Cohen, F.G. Cozman, N. Sebe, M.C. Cirelo et T.S. Huang. *Semisupervised learning of classifiers : theory, algorithms, and their application to human-computer interaction.* IEEE Trans. PAMI, vol. 26, no. 12, pages 1553–1567, 2004.

[Cox 96] S. Cox et R. Rose. *Confidence measures for the switchboard database.* In ICASSP, pages 511–514, 1996.

[Cox 00] S. Cox et S. Dasmahapatra. *A Semantically-based confidence measure for speech recognition.* In ICSLP, pages 206–209, 2000.

[Cox 02] S. Cox et S. Dasmahapatra. *High-Level Approaches to Confidence Estimation in Speech Recognition.* IEEE Trans., pages 460–471, 2002.

[Davis 80] S. Davis et P. Mermelstein. *Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences.* IEEE Trans. ASSP, vol. 28, pages 357–366, 1980.

[Decadt 01] B. Decadt, J. Duchateau, W. Daelemans et P. Wambacq. *Phoneme-to-grapheme conversion for out-of-vocabulary words in large vocabulary speech recognition.* In ASRU Workshop, 2001.

[Decadt 02] B. Decadt, J. Duchateau, W. Daelemans et P. Wambacq. *Transcription of Out-Of-Vocabulary Words In Large Vocabulary Speech Recognition Based On Phoneme-To-Grapheme Conversion.* In ICASSP, pages 861–864, 2002.

[Deshmukh 99] N. Deshmukh, A. Ganapathiraju, J. Hamaker, J. Picone et M. Ordowski. *A public domain speech-to-text system.* In EUROSPEECH, pages 2127–2130, 1999.

[Deviren 02] M. Deviren. *Dynamic Bayesian networks for speech recognition.* In SIGART/AAAI Doctoral Consortium, 2002.

[Deviren 03] M. Deviren et K. Daoudi. *Frequency filtering or wavelet filtering ?* In ICANN/ICONIP, 2003.

[Deviren 04] M. Deviren. *Systèmes de reconnaissance de la parole revisités : Réseaux Bayesiens dynamiques et nouveaux paradigmes.* PhD thesis, UHP Nancy 1, 2004.

[Duchateau 02a] J. Duchateau, K. Demuynck et P. Wambacq. *Confidence Scoring Based on Backward Language Models.* In ICASSP, pages 221–224, 2002.

[Duchateau 02b] J. Duchateau et P. Wambacq. *Unconstrained versus constrained acoustic normalisation in confidence scoring.* In ICSLP, pages 1617–1620, 2002.

[Egan 75] J.P. Egan. Signal detection theory and roc analysis. Academic Press, 1975.

[Eide 95] E. Eide, H. Gish, P. Jeanrenaud et A. Mielke. *Understanding and improving speech recognition performance through the use of diagnostic tools.* In ICASSP, pages 221–224, 1995.

[Estève 02] Y. Estève. *Intégration de sources de connaissances pour la modélisation stochastique du langage appliquée à la parole continue dans un contexte de dialogue oral homme-machine.* PhD thesis, Université d'Avignon et des Pays de Vaucluse, 2002.

[Evermann 00] G. Evermann et P.C. Woodland. *Posterior Probability Decoding, Confidence Estimation and System Combination.* In NIST Speech Transcription Workshop, 2000.

[Fabian 05] T. Fabian, R. Lieb, G. Ruske et M. Thomae. *A Confidence-Guided Dynamic Pruning Approach - Utilization of Confidence Measurement in Speech Recognition -.* In EUROSPEECH, pages 585–588, 2005.

[Falavigna 02] D. Falavigna, R. Gretter et G. Riccardi. *Acoustic and Word Lattice Based Algorithms for Confidence Scores.* In ICSLP, Colorado, pages 1621–1624, 2002.

[Federico 98] M. Federico et R. De Mori. Language modelling. Academic Press, 1998.

[Ferrer 01] L. Ferrer et C. Estienne. *Improving Performance of a Keyword Spotting System by Using a New Confidence Measure.* In EUROSPEECH, Aalborg, pages 2561–2564, 2001.

[Finke 96] M. Finke, T. Zeppenfeld, M. Maier, L. Mayfield, K. Ries, P. Zhan, J. Lafferty et A. Waibel. *Switchboard April 1996 evaluation report.* Rapport technique, Interactive Systems Laboratories, 1996.

[Fiscus 97] J.G. Fiscus. *A Post-Processing System to Yield Reduced Word Error Rates : ROVER.* In ASRU Workshop, 1997.

[Fohr 00] D. Fohr, O. Mella et C. Antoine. *The automatic speech recognition engine ESPERE : experiments on telephone speech.* In ICSLP, pages 246–249, 2000.

[Foote 95] J.T. Foote, G.J.F Jones, K. Spärck Jones et S.J. Young. *Talker-independent keyword spotting for information retreival.* In EUROSPEECH, pages 2145–2148, 1995.

[Forney 73] G.D. Forney. *The Viterbi algorithm.* IEEE, vol. 61, no. 3, pages 268–278, 1973.

[Galliano 05] S. Galliano, E. Geoffrois, D. Mostefa, K. Choukri, J.F. Bonastre et G. Gravier. *The ESTER phase II evaluation compaign for the rich transcription of french broadcast news.* In EUROSPEECH, 2005.

[Galliano 06] S. Galliano, E. Geoffrois, G. Gravier, J.F. Bonastre, D. Mostefa et K. Choukri. *Corpus description of the ESTER Evaluation Campaign for the Rich Transcription of French Broadcast News.* In LREC, pages 315–320, 2006.

[Garcia-Mateo 99] C. Garcia-Mateo, W. Reichl et S. Ortmanns. *On combining confidence measures in HMM-based speech recognizers.* In ASRU, pages 201–204, 1999.

[Gauvain 94] J.L Gauvain et C.H. Lee. *Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains.* IEEE Trans. SAP, vol. 2, pages 291–298, 1994.

[Gauvain 05] J.L. Gauvain, G. Adda, M. Adda-Decker, A. Allauzen, V. Gendner, L. Lamel et H. Schwenk. *Where are we in transcribing French broadcast news.* In EUROSPEECH, pages 1665–1668, 2005.

[Gillick 97] L. Gillick, Y. Ito et J. Young. *A probabilistic approach to confidence measure estimation and evaluation.* In ICASSP, Munich, pages 879–882, 1997.

[Godfrey 92] J. Godfrey, E. Holliman et J. McDaniel. *Switchboard : Telephone speech corpus for research development.* In ICASSP, volume 1, pages 517–520, 1992.

[Gopalakrishnan 95] P.S. Gopalakrishnan, L.R. Bahl et R.L. Mercer. *A tree search strategy for large-vocabulary continuous speech recognition.* In ICASSP, pages 572–575, 1995.

[Gorin 97] A.L. Gorin, G. Riccardi et J.H. Wright. *How May I Help You ?* Speech Communication, vol. 23, pages 113–127, 1997.

[Gravier 02] G. Gravier, F. Yvon, B. Jacob et F. Bimbot. *Sirocco, un système ouvert de reconnaissance de la parole.* In JEP, pages 273–276, 2002.

[Gravier 04] G. Gravier, J.F. Bonastre, S. Galliano et E. Geoffrois. *The ESTER evaluation compaign of rich transcription of french broadcast news.* In LREC, 2004.

[Gunawardana 03] A. Gunawardana et A. Acero. *Adapting acoustic models to new domains and conditions using untranscribed data.* In EUROSPEECH, pages 1633–1636, 2003.

[Guo 04] G. Guo, C. Huang, H. Jiang et R.H. Wang. *A comparative study on various confidence measures in large vocanulary speech recognition.* In International Symposium on CSLP, pages 9–12, 2004.

[Hai 03] Jiang Hai et Er Meng Joo. *Improved linear predictive coding method for speech recognition.* In ICSP, volume 3, pages 1614–1618, 2003.

[Hazen 02] T.J. Hazen, S. Seneff et J. Polifroni. *Recognition confidence scoring and its use in speech understanding systems.* Computer Speech and Language, vol. 16, pages 49–67, 2002.

[Hermansky 90] H. Hermansky. *Perceptual linear predictive (PLP) analysis of speech.* JASA, vol. 87, no. 4, pages 1738–1752, 1990.

[Hernández-Abrego 00] G. Hernández-Abrego et J.B. Marino. *Contextual confidence measures for continuous speech recognition.* In ICASSP, pages 1803–1806, 2000.

[Jeanrenaud 93] P. Jeanrenaud, K. Ng, M. Siu, J.R. Rohlicek et H. Gish. *Phonetic-based word spotter : Various configrations and application to event spotting.* In EUROSPEECH, pages 1057–1060, 1993.

[Jeanrenaud 95] P. Jeanrenaud, M. Siu et H. Gish. *Large vocabulary word scoring as a basis for transcription generation.* In EUROSPEECH, pages 2149–2152, 1995.

[Jelinek 76] F. Jelinek. *Continuous Speech Recognition by Statistical Methods.* In IEEE, volume 64, pages 532–556, 1976.

[Jiang 05] H. Jiang. *Confidence measures for speech recognition : A survey.* Speech Communication, vol. 45, pages 455–470, 2005.

[Jitsuhiro 98] T. Jitsuhiro, S. Takahashi et K. Aikawa. *Rejection of Out-Of-Vocabulary Words Using Phoneme Confidence Likelihood.* In ICASSP, Seattle, pages 217–220, 1998.

[Jouvet 99] D. Jouvet, K. Bartkova et G. Mercier. *Hypothesis dependent threshold setting for improved out-of-vocabulary data rejection.* In ICASSP, volume 2, pages 709–712, 1999.

[Kamppari 00] S.O. Kamppari et T.J. Hazen. *Word and Phone Level Acoustic Confidence Scoring.* In ICASSP, Istanbul, 2000.

[Kemp 97] T. Kemp et T. Schaaf. *Estimating Confidence using Word Lattices.* In EUROSPEECH, Rhodes, pages 827–830, 1997.

[Kemp 98] T. Kemp et A. Waibel. *Unsupervised training of a speech recognizer using TV broadcast.* In ICSLP, pages 2207–2210, 1998.

[Kemp 99] T. Kemp et A Waibel. *Unsupervised training of a speech recognizer : recent experiments.* In EUROSPEECH, pages 2725–2728, 1999.

[Kubala 98] F. Kubala, J. Davenport, H. Jin, D. Liu, T. Leek, S. Matsoukas, D. Miller, L. Nguyen, F. Richardson, R. Schwartz et J. Makhoul. *The 1997 BBN Byblos system applied to broadcast news transcription.* In DARPA Broadcast News Transcription and Understanding Workshop, 1998.

[Kuhn 90] R. Kuhn et R. De Mori. *A cache-based natural language model for speech recognition.* IEEE Trans. PAMI, vol. 12, no. 6, pages 570–582, 1990.

[Lamel 02] L. Lamel, J.L. Gauvain et G. Adda. *Lightly supervised and unsupervised acoustic model training.* Computer Speech and Language, vol. 16, pages 115–129, 2002.

[Lamere 03] P. Lamere, P. Kwok, W. Walter, E. Gouvea, R. Singh, B. Raj et P.P. Wolf. *Design of the CMU Sphinx-4 Decoder.* In EUROSPEECH, volume 2, 2003.

[Landauer 97] T.K. Landauer et S.T. Dumais. *A solution to Plato's problem : representation of knowledge.* Psychological Review, vol. 104, pages 211–240, 1997.

[Lavecchia 06] C. Lavecchia, K. Smaïli et JP Haton. *How to handle gender and number agreement in statistical language models ?* In EUROSPEECH, 2006.

[Lee 91] C.H. Lee, C.H. Lin et B.H. Juang. *A study on speaker adaptation of the parameters of continuous density hidden Markov models.* IEEE Trans. Signal Processing, vol. 39, pages 806–814, 1991.

[Lee 00] A. Lee, T. Kawahara, K. Takeda et K. Shikano. *A new phonetic tied-mixture model for efficient decoding.* In ICASSP, pages 1269–1272, 2000.

[Lee 01] A. Lee, T. Kawahara et K. Shikano. *Julius - an open source real-time large vocabulary recognition engine.* In EUROSPEECH, Aalborg, pages 1691–1694, 2001.

[Lee 04] A. Lee, K. Shikano et T. Kawahara. *Real-time Word Confidence Scoring Using Local Postrior Probabilities on Tree Trellis Search.* In ICASSP, Montreal, pages 793–796, 2004.

[Leggetter 95] C.J. Leggetter et P.C. Woodland. *Maximum likelihood linear regression for speaker adaptation of continuous density hidden Markov models.* Computer Speech and Language, vol. 9, pages 171–186, 1995.

[Linares 05] G. Linares, P. Nocera, D. Matrouf, F. Béchet, D. Massonié et C. Fredouille. *Le système de transcription du LIA pour ESTER-2005.* In Workshop ESTER, 2005.

[Lleida 96] E. Lleida et R. Rose. *Efficient decoding and training procedures for utterance verification in continuous speech recognition.* In ICASSP, pages 507–510, 1996.

[Lobrot 70] M. Lobrot. Orlec. 1970.

[Ma 07] J. Ma et S. Matsoukas. *Unsupervised Training on a Large Amount of Arabic Broadcast News Data.* In ICASSP, volume 2, pages 349–352, 2007.

[Maison 01] B. Maison et R. Gopinath. *Robust confidence annotation and rejection for continuous speech recognition.* In ICASSP, volume 1, pages 389–392, 2001.

[Mangu 00] L. Mangu, E. Brill et A. Stolcke. *Finding consensus in speech recognition : word error minimization and other applications of confusion networks.* Computer Speech and Language, vol. 14, pages 373–400, 2000.

[Mari 97] J.F. Mari, J.P. Haton et A. Kriouile. *Automatic word recognition based on second-order hidden Markov models.* IEEE Trans. SAP, no. 5, pages 22–25, 1997.

[Markel 76] J.D. Markel et A.H. Gray. Linear prediction of speech. Springer Verlag, 1976.

[Martin 97] A. Martin, G. Doddington, T. Kamm, M. Ordowski et M. Przybocki. *The Det Curve In Assessment Of Detection Task Performance.* In EUROSPEECH, pages 1895–1898, 1997.

[Mauclair 06] J. Mauclair, Y. Estève et P. Deléglise. *Probabilité* a posteriori *: amélioration d'une mesure de confiance en reconnaissance de la parole.* In JEP, Dinard, pages 421–424, 2006.

[Mengusoglu 03] E. Mengusoglu. *Confindence measure based model adaptation for speaker verification.* In CIIT, 2003.

[Metze 00] F. Metze, T. Kemp, T. Schaaf, T. Schultz et H. Soltau. *Confidence measure based language identification.* In ICASSP, pages 1827–1830, 2000.

[Moreau 00] N. Moreau, D. Charlet et D. Jouvet. *Confidence Measure and Incremetal Adaptation for the Rejection of Incorrect Data.* In ICASSP, pages 1807–1810, 2000.

[Moreno 01] P.J. Moreno, B. Logan et B. Raj. *A boosting approach for confidence scoring.* In EUROSPEECH, pages 2109–2112, 2001.

[Mourot 07] Lorène Mourot et Marie Rovel. *Evaluation of a talking head for helping HOH people in the classroom.* PhD thesis, School of Speech therapist, 2007.

[Neti 97] C.V. Neti, S. Rouskos et E. Eide. *Word-based confidence measures as a guide for stack seach in speech recognition.* In ICASSP, pages 883–886, 1997.

[Ney 94] H. Ney et X.Aubert. *A word graph algorithm for large vocabulary continuous speech recognition.* In ICSLP, Yokohama, pages 1355–1358, 1994.

[Nguyen 05] L. Nguyen, B. Xiang, M. Afify, S. Abdou, S. Matsoukas, R. Schwartz et J. Makhoul. *The BBN RT04 English broadcast news transcription system.* In EUROSPEECH, pages 1673–1676, 2005.

[Ngyen 99] P. Ngyen, P. Gelin, J.C Junqua et J.T. Chien. *N-Best Based Supervised and Unsupervised Adaptation for Native and Non-Native Speakers in Cars.* In ICASSP, pages 173–176, 1999.

[Nilsson 71] N.J. Nilsson. *Problem-solving methods in artificial intelligence.* McGraw-Hill, 1971.

[Ortmanns 97] S. Ortmanns, H. Ney et X. Aubert. *A word graph algorithm for large vocabulary continuous speech recognition.* Computer Speech and Language, vol. 11, no. 1, pages 43–72, 1997.

[Pao 98] C. Pao, P. Schmid et J. Glass. *Confidence scoring for speech understanding systems.* In ICSLP, pages 815–818, 1998.

[Pitz 00] M. Pitz, F. Wessel et H. Ney. *Improved MLLR speaker adaptation using confidence measures for conversational speech recognition.* In ICSLP, pages 548–551, 2000.

[Qiu 96] H. Qiu. Confidence measures for speech recognition. Master's thesis, Carnegie Mellon University, 1996.

[Rabiner 78] L.R. Rabiner et R.W. Schafer. Digital processing of speech signals. Prentice Hall, 1978.

[Rabiner 89] L.R. Rabiner. *A tutorial on Hidden Markov Models and selected applications in speech recognition.* Proceedings of the IEEE, vol. 77, no. 2, pages 257–286, 1989.

[Rabiner 93] L. Rabiner et B.H. Juang. Fundamentals of speech recognition. Prentice Hall PTR, 1993.

[Rahim 95] M. Rahim, C.H. Lee et B.H Juang. *Robust utterance verification for connected digits recognition.* In ICASSP, volume 1, pages 285–288, 1995.

[Rahim 96] M. Rahim et B.H. Juang. *Signal bias removal by maximum likelihood estimation for robust telephone speech recognition.* IEEE Trans. SAP, vol. 4, pages 19–30, 1996.

[Rahim 97] M.G. Rahim, C.H Lee et B.H. Juang. *Discriminative utterance verification for connected digits recognition.* IEEE Trans. SAP, vol. 5, no. 3, pages 266–277, 1997.

[Ramesh 98] P. Ramesh, C.H. Lee et B.H. Juang. *Context dependent anti subword modeling for utterance verification.* In ICSLP, pages 3233–3236, 1998.

[Ravishankar 96] M.K. Ravishankar. *Efficient algorithms for Speech Recognition.* PhD thesis, School of Computer Science, CMU, 1996.

[Razik 05] J. Razik, O. Mella, D. Fohr et J.P. Haton. *Local Word Confidence Measure Using Word Graph and N-Best List.* In INTERSPEECH, Lisbon, pages 3369–3372, 2005.

[Robinson 88] A. J. Robinson et F. Fallside. *A dynamic connectionist model for phoneme recognition.* In nEuro, 1988.

[Robinson 94] A. J. Robinson. *An application of recurrent nets to phone probability estimation.* IEEE Transactions on Neural Networks, vol. 5, no. 2, pages 298–305, 1994.

[Rose 95a] R.C. Rose. *Keyword detection in conversational speech utterances using hidden markov model based continuous speech recognition.* Computer, Speech and Language, vol. 9, pages 309–333, 1995.

[Rose 95b] R.C. Rose, B.H. Juang et C.H. Lee. *A training procedure for verifying string hypothesis in continuous speech recognition.* In ICASSP, pages 281–284, 1995.

[Rosenfel 96] R. Rosenfel. *A ME Approach to Adaptative Statistical Language Modeling.* PhD thesis, School of Computer Science, CMU, 1996.

[Rotovnik 02] T. Rotovnik, M.S. Maučec, B. Horvat et Z. Kačič. *A Comparison of HTK, ISIP and Julius in Slovenian Large Vocabulary Continuous Speech Recognition.* In ICASSP, pages 681–684, 2002.

[Rueber 97] B. Rueber. *Obtaining Confidence Measures from Sentence Probabilities.* In EUROSPEECH, Rhodes, pages 739–742, 1997.

[San-Segundo 01] R. San-Segundo, B. Pellom et K. Hacioglu. *Confidence Measures For Spoken Dialogue Systems.* In ICASSP, pages 393–396, 2001.

[Sankar 98] A. Sankar, F. Weng, Z. Rivlin, A. Stolcke et R.R. Gaddeal. *Development of SRI's 1997 broadcast news transcription system.* In DARPA Broadcast News Transcription and Understanding Workshop, 1998.

[Schaaf 97] T. Schaaf et T. Kemp. *Confidence measures for spontaneous speech recognition.* In ICASSP, pages 875–878, 1997.

[Schaaf 01] T. Schaaf. *Detection of OOV Words Using Generalized Word Models and Semantic Class Language Model.* In EUROSPEECH, pages 2581–2584, 2001.

[Schwartz 90] R. Schwartz et Y.L. Chow. *The N-best Algorithm : An efficient and exact procedure for finding the N mots likely sentence hypotheses.* In ICASSP, Albuquerque, pages 81–84, 1990.

[Schwenck 97] H. Schwenck et J.L. Gauvain. *Combining multiple speech recognizers using voting and language model information.* In ICSLP, pages 915–918, 1997.

[Setlur 96] A.R. Setlur, R.A. Sukkar et J. Jacob. *Correcting recognition errors via discriminative utterance verification.* In ICSLP, pages 602–605, 1996.

[Siu 97] M.H. Siu, H. Gish et F. Richardson. *Improved Estimation, Evaluation and Applications of Confidence Measures for Speech Recognition.* In EUROSPEECH, Rhodes, pages 831–834, 1997.

[Siu 99] M. Siu et H. Gish. *Evaluation of word confidence for speech recognition systems.* Computer Speech and Language, vol. 13, pages 299–319, 1999.

[Smith 95] J. Smith et J. Abel. *The Bark bilinear transform.* In IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, 1995.

[Stemmer 02] G. Stemmer, S. Steidl, E. Nöth, H. Niemann et A. Batliner. *Comparison and combination of confidence measures.* In International Conference on Text, Speech and Dialogue, pages 181–188, 2002.

[Stolcke 97] A. Stolcke, Y. König et M. Weintraub. *Explicit xord error rate minimization in N-best list rescoring.* In EUROSPEECH, pages 163–166, 1997.

[Sukkar 94] R.A. Sukkar. *Rejection for connected digit recognition based on GDP segmental discrimination.* In ICASSP, volume 1, pages 393–396, 1994.

[Sukkar 96] R.A. Sukkar et C.H. Lee. *Vocabulary independent discriminative utterance verification for nonkeyword rejection in subword based speech recognition.* IEEE Trans. SAP, vol. 4, pages 420–429, 1996.

[Sun 03] H. Sun, G. Zhang, F. Zheng et M. Xu. *Using Word Confidence Measure for OOV Words Detection in a Spontaneous Spoken Dialog System.* In EUROSPEECH, pages 2713–2716, 2003.

[Tebelskis 95] J. Tebelskis. *Speech Recognition using Neural Networks.* PhD thesis, School of Computer Science, Pittsburgh, 1995.

[Uhrik 97] C. Uhrik. *Confidence metrics based on n-gram language model backoff behavior.* In EUROSPEECH, pages 2771–2774, 1997.

[Van Rijsbergen 79] K. Van Rijsbergen. Information retrieval. Butterworths, 1979.

[Vergyri 00] D. Vergyri. *Use of Word Level Side Information to Improve Speech Recognition.* In ICASSP, pages 1823–1826, 2000.

[Viterbi 67] A. Viterbi. *Error bounds for convolutional codes and an asymptotically optimum decoding algorithm.* IEEE Transactions on Information Theory, vol. 13, no. 3, pages 260–269, 1967.

[Wallhoff 00] F. Wallhoff, D. Willett et G. Rigoll. *Frame-Discriminative and Confidence-Driven Adaptation for LVCSR.* In ICASSP, Istanbul, pages 1835–1838, 2000.

[Wegmann 98] S. Wegmann, P. Zhan, I. Carp, M. Newman, J. Yameon et L. Gillick. *Dragon systems' 1997 broadcast news transcription system.* In DARPA Broadcast News Transcription and Understanding Workshop, 1998.

[Weintraub 95] M. Weintraub. *LVCSR Log-Likelihood ratio scoring for keyword spotting.* In ICASSP, Detroit, pages 297–300, 1995.

[Weintraub 97] M. Weintraub, F. Beaufays, Z. Rivlin, Y. Konig et A. Stolcke. *Neural-network based measures of confidence for word recognition.* In ICASSP, Munich, pages 887–890, 1997.

[Wessel 99] F. Wessel, K. Macherey et H. Ney. *A comparison of word graph and N-best list based confidence measures.* In EUROSPEECH, pages 315–318, 1999.

[Wessel 01] F. Wessel, R. Schlüter, K. Macherey et H. Ney. *Confidence Measures for Large Vocabulary Continuous Speech Recognition.* IEEE Trans. SAP, vol. 9, pages 288–298, 2001.

[Wessel 05] F. Wessel et H. Ney. *Unsupervised training of acoustic models for large vocabulary continuous speech recognition.* IEEE Trans. SAP, vol. 13, no. 1, pages 23–31, 2005.

[Wilcox 91] L.D. Wilcox et M.A. Bush. *HMM based wordspotting for voice editing and indexing.* In EUROSPEECH, pages 25–28, 1991.

[Williams 04] J.D. Williams et S.M. Witt. *A comparison of dialog strategies for call routing.* Int. Journal of Speech Technology, vol. 7, pages 9–24, 2004.

[Wilpon 90] J.G. Wilpon, L.R. Rabiner, C. Lee et E.R. Goldman. *Automatic recognition of keywords in unconstrained speech using hidden Markov models.* IEEE Trans. ASSP, vol. 38, pages 1870–1878, 1990.

[Young 94a] S. Young. *The HTK hidden Markov model toolkit : Design and philosophy.* Rapport technique, Cambridge University Engineerind Department, UK, 1994.

[Young 94b] S.J. Young. *Detecting misrecognitions and out-of-vocabulary words.* In ICASSP, volume 2, pages 21–24, 1994.

[Zavaliagkos 98] G. Zavaliagkos et T. Colthurst. *Utilizing untranscribed data to improve performance.* In Broadcast News Transcription and Understanding Workshop, pages 301–305, 1998.

[Zhang 01] R. Zhang et A.I. Rudnicky. *Word level confidence annotation using combinations of features.* In EUROSPEECH, 2001.

Publications personnelles

[Razik 03] J. Razik, C. Sénac, D. Fohr, O. Mella et N. Parlangeau-Vallès. *Comparison of Two Speech/Music Segmentation Systems For Audio Indexing on the Web.* In 7th World Multiconference on Systemics, Cybernetics and Informatics - SCI'2003, 2003. 6 pages.

[Razik 04] J. Razik, D. Fohr, O. Mella et N. Parlangeau-Vallès. *Segmentation Parole/Musique pour la transcription automatique.* In Journées d'Etude sur la Parole - JEP 2004, Fès, Maroc, pages 417–420, Avril 2004.

[Razik 05] J. Razik, O. Mella, D. Fohr et J.P. Haton. *Local Word Confidence Measure Using Word Graph and N-Best List.* In INTERSPEECH, Lisbon, pages 3369–3372, 2005.

[Razik 06] J. Razik, O. Mella, D. Fohr et J.P. Haton. *Mesures de confiance trame-synchrone.* In Journées d'Etude sur la Parole - JEP 2006, Dinard, France, 2006. 4 pages.

[Razik 07] J. Razik, O. Mella, D. Fohr et J.P. Haton. *Frame-Synchronous And Local Confidence Measures For On-The-Fly Keyword Spotting.* In International Symposium on Signal Processing and its Applications - ISSPA 2007, 2007. 4 pages.

Nancy-Université

Université Henri Poincaré

Monsieur **RAZIK Joseph**

DOCTORAT DE L'UNIVERSITE HENRI POINCARE, NANCY 1

en INFORMATIQUE

VU, APPROUVÉ ET PERMIS D'IMPRIMER n° 1395

Nancy, le 23/10/2007

Le Président de l'Université

J.P. FINANCE

UNIVERSITÉ HENRI POINCARÉ, NANCY 1 ★ LE PRESIDENT

Université Henri Poincaré, Nancy I
24-30 rue Lionnois - B.P. 3069 - 54013 NANCY Cédex
Tél. : 03 83 68 20 00 - Fax : 03 83 68 21 00

Up the Irons!!!

Résumé

En reconnaissance automatique de la parole, les mesures de confiance tentent d'estimer la confiance qu'on peut accorder au résultat fourni par le moteur de reconnaissance : l'apport de la mesure de confiance permettant par exemple de mettre en évidence les mots mal reconnus ou hors vocabulaire.

Dans cette thèse nous proposons des mesures de confiance capables de faire cette estimation dans le cas d'applications nécessitant une reconnaissance « grand vocabulaire » en flux continu comme l'indexation en mots clés ou la transcription en ligne d'émissions radiophoniques, ou bien encore la transcription du cours d'un enseignant dans une salle de classe pour des élèves malentendants.

Dans ce cadre, nous avons défini deux types de mesures de confiance. Les premières, fondées sur des rapports de vraisemblance, sont des mesures trame-synchrones qui peuvent être calculées au fur et à mesure de la progression du moteur de reconnaissance au sein de la phrase à reconnaître. Les secondes, fondées sur une estimation de la probabilité *a posteriori* limitée à un voisinage du mot considéré, nécessitent seulement un court délai avant de pouvoir être calculées.

Ces mesures ont été évaluées et comparées à une mesure de l'état de l'art fondée sur la probabilité *a posteriori* mais nécessitant la reconnaissance de toute la phrase. Cette évaluation a été faite d'une part dans une tâche de transcription automatique d'un corpus réel d'émissions radiophoniques (ESTER) en utilisant le taux d'EER ; d'autre part dans une tâche de détection de mots clés sur le même corpus. Des performances très proches de celles de la mesure de l'état de l'art ont été obtenues par nos mesures locales avec un délai de moins d'une seconde.

Nous avons également intégré l'une de nos mesures trame-synchrones dans le processus de décodage du moteur de reconnaissance et ainsi diminué le taux d'erreur en mots du système initial d'environ 6% en relatif. Enfin, une de nos mesures de confiance a permis par la mise en valeur des mots de faible confiance d'améliorer la compréhension de malentendants.

Mots-clés: mesure de confiance, mesure trame-synchrone, mesure locale, détection de mots-clés, reconnaissance automatique de la parole, sourds, malentendants

Abstract

In automatic speech recognition, confidence measures aim at estimating the confidence we can give to a result (phone, word, sentence) provided by the speech recognition engine ; for example, the contribution of the confidence measure allows to highlight the misrecognized or out-of-vocabulary words.

In this thesis, we propose several confidence measures which are able to provide this estimation for applications using large vocabulary and on-the-fly recognition, as keyword indexation, broadcast news transcription, and live teaching class transcription for hard of hearing children.

In this framework, we have defined two types of confidence measures. The first, based on likelihood ratio, are frame-synchronous measures which can be computed simultaneously with the recognition process of the sentence. The second ones are based on an estimation of the posterior probability limited to a neighborhood of the considered word, and need only a short delay before being computed.

These measures were assessed and compared to a state-of-the-art one, based on posterior probability but which requires the recognition of the whole sentence. Two evaluations were performed on a real broadcast news corpus (ESTER). The first one used the EER criterion in an automatic transcription task. The second evaluation was performed in a keyword spotting task. We achieved performance close to our reference measure with our local measures and a delay of less than one second.

We also integrated one of our frame-synchronous measures in the decoding process of the recognition engine and achieved to decrease the word error rate of the original system of around 6% in relative. One of our confidence measure acheived to increase the comprehension of hard of hearing children by highlighting words of low confidence.

Keywords: confidence measure, frame-synchronous measure, local measure, keyword detection, automatic speech recognition, hard of hearing people