Reconnaissance automatique de la parole guidée par des transcriptions a priori

Benjamin Lecouteux

ACADÉMIE D'AIX-MARSEILLE
UNIVERSITÉ D'AVIGNON ET DES PAYS DE VAUCLUSE

THÈSE

présentée à l'Université d'Avignon et des Pays de Vaucluse
pour obtenir le diplôme de DOCTORAT

SPÉCIALITÉ : Informatique

École Doctorale 166 I2S « Mathématiques et Informatique »
Laboratoire d'Informatique (EA 4128)

Reconnaissance automatique de la parole guidée par des transcriptions a priori

par
Benjamin LECOUTEUX

Soutenue publiquement le 5 décembre 2008 devant un jury composé de :

M.	François YVON	Professeur, LIMSI, Paris 11, France	Président, rapporteur
M.	Jan CERNOCKY	Professeur, Université de Brno, République Tchèque	Rapporteur
M.	Renato DE MORI	Professeur, LIA, Avignon, France	Examinateur
M.	Yannick ESTÈVE	Maître de Conférences, LIUM, Le Mans, France	Examinateur
M.	Jean-François BONASTRE	Professeur, LIA, Avignon, France	Directeur de thèse
M.	Georges LINARÈS	Maître de Conférences, LIA, Avignon, France	Co-Encadrant

---

Laboratoire d'Informatique d'Avignon

Remerciements

Finalement, ces trois années de thèse sont passées extrêmement vite. Quatre années auparavant, je n'aurai pas imaginé me lancer dans ce long et passionnant travail. Mon intérêt pour l'informatique s'est principalement développé au cours de mes années d'études, du deug au master, au sein de l'IUP GMI d'Avignon. Mes balbutiements en reconnaissance automatique de la parole se sont faits au cours de mon stage de maîtrise, sous la direction de Georges LINARES ; ce domaine qui me paraissait si mystique m'a immédiatement passionné. J'ai eu la chance de travailler avec l'équipe parole du LIA, qui m'a beaucoup apporté tant dans la connaissance que sur le plan humain. Je tiens particulièrement à remercier Jean-François BONASTRE et Georges LINARES qui sont les instigateurs de mon sujet de thèse et qui m'ont soutenu tout au long de ce travail. Jean-François a dirigé ma thèse et m'a aidé dans toutes les démarches relatives à celle-ci. Georges l'a encadrée au long des trois années ; il a su habilement me remotiver dans les passages à vide, et m'a fait partager énormément de connaissances et d'expériences. Je remercie également Pascal NOCERA, Driss MATROUF, Frédéric BECHET, Benoît FAVRE, Dominique MASSONIE qui, au sein du laboratoire, ont été patients vis-à-vis de mes nombreuses interrogations.

Je remercie M. François YVON et M. Yan Cernocky d'avoir accepté d'être les rapporteurs de ma thèse, mais également pour leurs corrections et remarques pertinentes vis-à-vis de mon document. Je remercie également les membres de mon jury : Yannick ESTEVE et Rénato De MORI.

Yannick ESTEVE et Guillaume GRAVIER m'ont apporté une précieuse aide lors de mes deux dernières années de thèse, dans le cadre de nos travaux sur la combinaison de systèmes en partenariat avec les laboratoires du LIUM et de l'IRISA.

Un grand merci également à l'ensemble des thésards et personnels avec qui j'ai partagé ces trois années (dans le désordre) : Nicolas, Laurianne, Nathalie, Loïc, Corinne, Frédéric D., Christoph L., Christophe S., Zac, Philou, Fabrice, Anthony, Thierry S., Pierre J., Marc, Jean-Pierre, Mickaël, Claire, Juliette, Rémy, Marie-Jean, Stanislas, Eric, Dominique S., Alexandre, William, Cathy, Gilles, Vladimir, Nimaan, Hugo, Raphaël.

Je tiens aussi à remercier mes proches Cécile, Christine et Claude, ma sœur, qui m'ont apporté leur soutien au cours de ces trois dernières années et qui m'ont supporté dans les moments de stress.

Résumé

L'utilisation des systèmes de reconnaissance automatique de la parole nécessite des conditions d'utilisation contraintes pour que ces derniers obtiennent des résultats convenables. Dans de nombreuses situations, des informations auxiliaires aux flux audio sont disponibles. Le travail de cette thèse s'articule autour des approches permettant d'exploiter ces transcriptions *a priori* disponibles. Ces informations se retrouvent dans de nombreuses situations : les pièces de théâtre avec les scripts des acteurs, les films accompagnés de sous-titres ou de leur scénario, les flashes d'information associés aux prompts des journalistes, les résumés d'émissions radio... Ces informations annexes sont de qualité variable, mais nous montrerons comment ces dernières peuvent être utilisées afin d'améliorer le décodage d'un SRAP.

Ce document est divisé en deux axes liés par l'utilisation de transcriptions *a priori* au sein d'un SRAP : la première partie présente une méthode originale permettant d'exploiter des transcriptions *a priori* manuelles, et de les intégrer directement au cœur d'un SRAP. Nous proposons une méthode permettant de guider efficacement le système de reconnaissance à l'aide d'informations auxiliaires. Nous étendons notre stratégie à de larges corpus dénués d'informations temporelles. La seconde partie de nos travaux est axée sur la combinaison de SRAP. Nous proposons une combinaison de SRAP basée sur le décodage guidé : les transcriptions *a priori* guidant un SRAP principal sont fournies par des systèmes auxiliaires.

Les travaux présentés proposent d'utiliser efficacement une information auxiliaire au sein d'un SRAP. Le décodage guidé par des transcriptions manuelles permet d'améliorer sensiblement la qualité du décodage ainsi que la qualité de la transcription *a priori*. Par ailleurs, les stratégies de combinaison proposées sont originales et obtiennent d'excellents résultats par rapport aux méthodes existantes à l'état de l'art.

Abstract

Robustness in speech recognition refers to the need to maintain high recognition accuracies even when the quality of the input speech is degraded. In the last decade, some papers proposed to use relevant meta-data in order to enhance the recognition process. Nevertheless, in many cases, an imperfect *a priori* transcript can be associated to the speech signal : movie subtitles, scenarios and theatrical plays, summaries and radio broadcast. This thesis addresses the issue of using such imperfect transcripts for improving the performance figures of automatic speech recognition (ASR) systems. Unfortunately, these *a priori* transcripts seldom correspond to the exact word utterances and suffer from a lack of temporal information. In spite of their varying quality, we will show how to use them to improve ASR systems.

In the first part of the document we propose to integrate the imperfect transcripts inside the ASR search algorithm. We propose a method that allows us to drive an automatic speech recognition system by using prompts or subtitles. This driven decoding algorithm relies on an on-demand synchronization and on the linguistic rescoring of ASR hypotheses. In order to handle transcript excerpts, we suggest a method for extracting segments in large corpora. The second part presents the Driven Decoding Algorithm (DDA) approach in combining several speech recognition (ASR) systems : it consists in guiding the search algorithm of a primary ASR system by the one-best hypotheses of auxiliary systems.

Our work suggests using auxiliary information directly inside an ASR system. The driven decoding algorithm enhances the baseline system and improves the *a priori* transcription. Moreover, the new combination schemes based on generalized-DDA significantly outperform state of the art combinations.

Table des matières

Introduction

Depuis presque trente ans, la reconnaissance automatique de la parole est un domaine qui a captivé le public ainsi que de nombreux chercheurs. À ses balbutiements, les projections sur ses applications étaient très optimistes : quoi de plus naturel que de parler à une machine, sans avoir à s'encombrer d'un clavier ? Malheureusement, malgré l'incroyable évolution des ordinateurs et des connaissances, la reconnaissance automatique de la parole n'en demeure pas moins un sujet de recherche toujours actif... et les résultats obtenus sont encore loin de l'idéal qu'on aurait pu en attendre il y a vingt ans.

Cependant, si le système de reconnaissance idéal n'existe pas encore, des applications concrètes émergent petit à petit. La reconnaissance automatique de la parole commence à équiper certains téléphones ou GPS qui, en identifiant certains mots clefs, permettent d'effectuer les tâches demandées. Les systèmes de reconnaissance sont également utilisés pour indexer de grandes bases de données audiovisuelles, pour rechercher des termes dans des flux audio ou encore comme interface de dialogue homme-machine... Dans la pratique, quand les conditions d'utilisation sont correctes, ces systèmes s'avèrent efficaces. Néanmoins, les principales limites des systèmes actuels sont relatives à leur robustesse : les conditions d'utilisation doivent être similaires à celles utilisées pour entraîner le système, l'environnement sonore peu bruité, les locuteurs ne peuvent pas parler simultanément... Souvent, l'utilisateur a dû s'adapter pour utiliser les logiciels. Un exemple en est le principe du perroquet qui s'est développé ces dernières années pour produire des sous-titres en temps réel, ou transcrire des conférences. Un opérateur répète les paroles à transcrire au système de reconnaissance en s'adaptant à celui-ci (vitesse d'élocution, vocabulaire contraint, etc.).

D'une façon générale, les performances d'un SRAP dépendent de sa capacité à modéliser les connaissances, ainsi que de la quantité et la qualité des données utilisées pour l'apprentissage des modèles. Ainsi, les systèmes nécessitent d'importants volumes de données annotées pour l'estimation des modèles acoustiques et linguistiques. En effet, malgré l'évolution continue des techniques mises en œuvre dans les SRAP, l'amélioration des modèles acoustiques reste souvent liée à l'augmentation des quantités de données d'apprentissage. De plus, cette augmentation s'est faite parallèlement à l'évolution de l'informatique : un système des années 95 était entraîné

sur une cinquantaine d'heures, alors qu'aujourd'hui les systèmes les plus aboutis ont des modèles estimés sur plusieurs milliers d'heures annotées. Il en résulte que les coûts des systèmes contemporains suivent une courbe exponentielle, en raison du travail d'annotation manuelle nécessaire. Face à ces besoins grandissants de masse de données, des stratégies visant à exploiter des données existantes, gratuites et incomplètes sont apparues.

Ce type de situation se retrouve dans de nombreux contextes où des transcriptions *a priori* sont associées aux données audio. C'est le cas du théâtre où le script des acteurs est disponible, le cinéma et ses sous-titres, les flashes d'information associés aux prompts journalistiques, etc. Ces transcriptions se présentent alors comme des corpus potentiellement exploitables pour entraîner un SRAP. Ces données approximatives doivent tout de même contenir un minimum d'information fiable, apportée manuellement. Cependant, il est nécessaire d'adopter des stratégies qui permettent d'exploiter les portions pertinentes de ces transcriptions *a priori* et/ou de les améliorer. Les techniques utilisées dans la littérature proposent généralement de compenser la faible qualité par la quantité des données. Cependant, cette stratégie conduit à une sous-exploitation des corpus et suppose que le volume de données soit suffisant. Bien que globalement efficaces, ces méthodes n'exploitent pas toute l'information disponible et ne corrigent pas les données incorrectes.

L'ensemble de nos travaux vise à l'amélioration de la qualité des transcriptions d'un SRAP par l'exploitation de transcriptions *a priori*, que l'objectif soit la transcription de ces documents ou la production de corpus "propres" à partir de transcriptions imparfaites. Cette amélioration s'effectuera en intégrant, directement dans l'algorithme de recherche, l'ensemble des informations *a priori*.

Nous formalisons le problème comme l'intégration d'un canal supplémentaire au sein du SRAP. Nos recherches explorent le potentiel de ces types d'approches dans diverses conditions : des corpus annotés temporellement ou non, plus ou moins volumineux avec des qualités de transcriptions très variables en terme de taux d'erreur mot.

En nous appuyant sur la formalisation proposée, nous généraliserons l'exploitation de transcriptions *a priori* à des transcriptions issues de systèmes de reconnaissance auxiliaires. Ainsi, le second axe de nos travaux se place dans le cadre de la combinaison de systèmes de reconnaissance. De nombreuses stratégies de combinaisons ont été proposées récemment : en amont, en aval, intégrées... Les plus couramment utilisées sont la combinaison de réseaux de confusion et la combinaison par consensus (vote entre les hypothèses des différents systèmes). Les travaux actuels tendent à montrer que les combinaisons au niveau de l'espace de recherche sont les plus efficaces, mais aussi les plus difficiles à mettre en œuvre. Nous introduisons une nouvelle stratégie dans laquelle le décodage est réalisé par un système primaire, dont l'algorithme de recherche est guidé par des transcriptions issues de systèmes auxiliaires. Cette méthode permet

d'influer dynamiquement sur l'exploration de l'espace de recherche, en fonction des informations auxiliaires disponibles. Ce document s'articule autour de trois parties :

- la première présente un état de l'art général sur les SRAP. Nous y établissons leur principe général et introduisons les éléments de base nécessaires à leur fonctionnement, notamment les modélisations acoustiques, linguistiques et les algorithmes de décodage. Les parties suivantes présentent les deux axes sur lesquels se sont développées nos recherches.

- la seconde partie du document expose nos travaux exploitant des transcriptions manuelles *a priori*. Nous présentons un état de l'art relatif à l'exploitation de ces dernières ainsi qu'aux applications qui en découlent. Les transcriptions *a priori* ont déjà été utilisées pour améliorer des modèles de langage ou pour contraindre les SRAP. Cependant, les méthodes existantes présentent certaines lacunes. Elles sont peu adaptées aux transcriptions imparfaites et exploitent très indirectement les informations (adaptation des modèles acoustiques ou linguistiques). Nous proposons une méthode originale, intégrée, permettant de guider un SRAP à l'aide de transcriptions *a priori*. Les travaux préliminaires montreront l'efficacité de l'approche sur des transcriptions parfaites. Par la suite, nous proposons d'étendre notre méthode à des corpus de texte volumineux : un algorithme original est proposé pour trouver à la volée des îlots de transcription susceptibles de guider le système. La méthode sera expérimentée en conditions réelles sur des corpus de prompts issus de la radio Belge RTBF.

- la dernière partie du document présente l'extension du décodage guidé par des transcriptions *a priori*. Nous proposons une méthode innovante permettant de combiner plusieurs systèmes de reconnaissance. Nous présentons d'abord un éventail théorique et pragmatique des stratégies de combinaison entre SRAP. Nous montrerons à quels niveaux peuvent s'effectuer les combinaisons : acoustique, linguistique, espace de recherche, *a posteriori*. Considérant que l'approche la plus efficace théoriquement consiste à intégrer l'ensemble des sources d'information disponibles dans la fonction de coût de l'algorithme de recherche, nous proposons une approche basée sur le décodage guidé dans laquelle des systèmes auxiliaires fournissent des transcriptions *a priori* et les scores de confiance associés. Nos premiers travaux sur la combinaison n'exploiteront qu'un système auxiliaire. Par la suite, nous proposons d'étendre la combinaison à n systèmes. Nous expérimentons également l'introduction d'un maximum d'informations issues des SRAP auxiliaires, en guidant le SRAP principal par des réseaux de confusion issus d'un système auxiliaire. Nous appliquerons plusieurs stratégies de combinaisons basées sur notre décodage guidé et mettrons en avant sa simplicité et son efficacité. L'ensemble de nos travaux sera présenté dans un cadre expérimental permettant de les évaluer et de les analyser.

Notre démarche s'articulera autour d'un module qui permet d'intégrer efficacement des transcriptions auxiliaires au sein d'un SRAP. Finalement, nous présenterons quelques conclusions et perspectives relatives à nos travaux.

Première partie

Principes des SRAP Markoviens

Chapitre 1

Définitions, modèles et algorithmes des systèmes de reconnaissance Markoviens

Sommaire

Dans cette partie, nous introduisons le cadre général dans lequel se sont effectués nos travaux. Nous présentons ensuite le fonctionnement général d'un SRAP, puis nous abordons plus en détail les éléments sur lesquels nous sommes intervenus.

Cet état de l'art se concentre sur les SRAP Markoviens utilisant des modèles de langages probabilistes à base de n-grammes. Nous survolons les principes des différents modèles acoustiques et linguistiques ainsi que les algorithmes liés à leur apprentissage. Nous finissons en présentant les algorithmes de décodage, ainsi que les paradigmes d'évaluation des SRAP.

1.1 Fonctionnement général d'un SRAP

Les systèmes de reconnaissance automatique de la parole (SRAP) ont pour objectif de transcrire un message oral en texte. Les principales applications utilisant des SRAP sont la transcription automatique, l'indexation de documents multimédias et le dialogue homme-machine. Les systèmes de reconnaissance automatique de la parole continue actuels se basent sur une approche statistique dont [Jelinek, 1976] a proposé une formalisation, issue de la théorie de l'information. À partir des observations acoustiques X, l'objectif d'un moteur de reconnaissance est de trouver la séquence de mots $\tilde{W}$ la plus probable parmi l'ensemble des séquences possibles. Cette séquence doit maximiser l'équation suivante :

$$\tilde{W} = \arg\max_{W} P(W|X) \tag{1.1}$$

En appliquant la théorie de Bayes, l'équation devient :

$$\tilde{W} = \arg\max_{W} \frac{P(X|W)P(W)}{P(X)} \tag{1.2}$$

$P(X) = \sum_{w} P(X|W)P(W)$ ne dépend pas d'une valeur particulière de W et peut être "sorti" du calcul de l'argmax :

$$\tilde{W} = \arg\max_{W} P(X|W)P(W) \tag{1.3}$$

Où le terme $P(W)$ est estimé via le modèle de langage et $P(X|W)$ correspond à la probabilité donnée par les modèles acoustiques. Ce type d'approche permet d'intégrer, dans le même processus de décision, les informations acoustiques et linguistiques (figure 1.1).

1.2 Modèles et paramètres acoustiques

Le signal de la parole ne peut être exploité directement. En effet, le signal contient de nombreux autres éléments que le message linguistique : des informations liées au

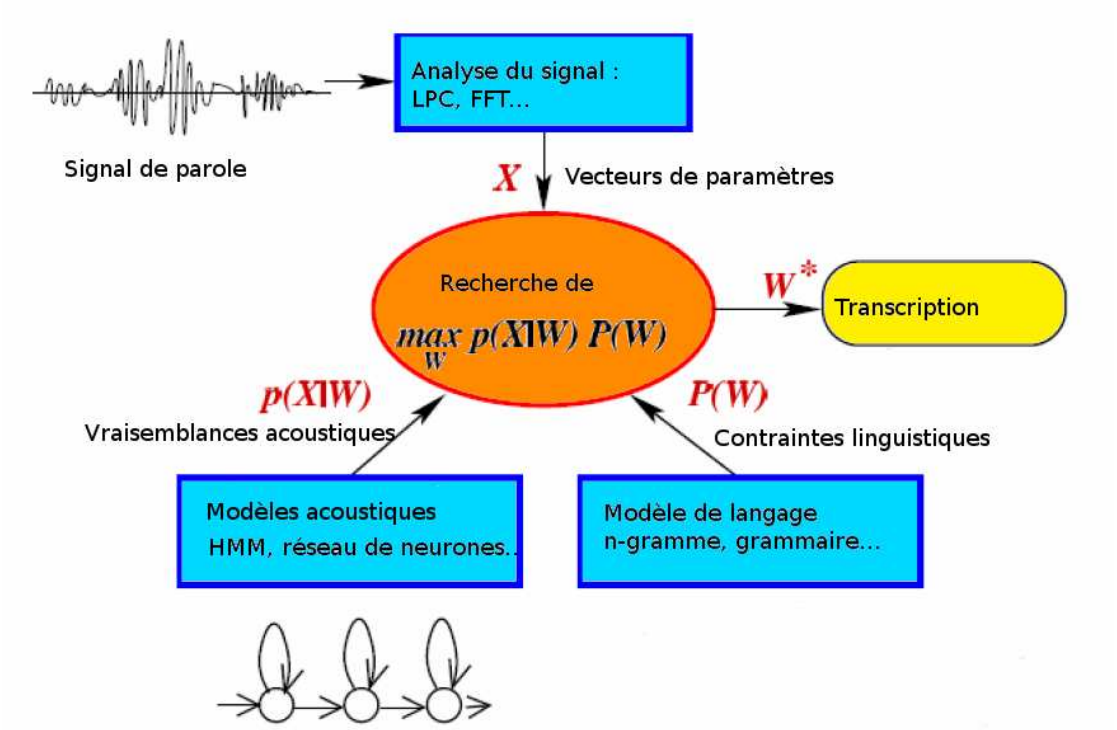

FIG. *1.1:* *Principe général des SRAP*

locuteur, aux conditions d'enregistrement, etc. Toutes ces informations ne sont pas nécessaires lors du décodage de la parole et rajoutent même du bruit. De plus, la variabilité et la redondance du signal de la parole le rendent difficilement exploitable tel quel. Il est donc nécessaire d'en extraire uniquement les paramètres qui seront dépendants du message linguistique.

Généralement, ces paramètres sont estimés via des fenêtres glissantes sur le signal. Cette analyse par fenêtre permet d'estimer le signal sur une portion du signal jugée stationnaire : généralement 10 à 30 ms en limitant les effets de bord et les discontinuités du signal via une fenêtre de Hamming.

La majorité des paramètres représentent le spectre fréquentiel et son évolution sur une fenêtre de taille donnée. Les techniques de paramétrisation les plus utilisées sont : PLP (*Perceptual Linear Prediction* : domaine spectral) [Hermansky et Cox, 1991], LPCC (*Linear Prediction Cepstral Coefficients* : domaine temporel)[Markel et Jr., 1976], MFCC (*Mel Frequency Cepstral Coefficients* : domaine cepstral).

1.2.1 Modèles de Markov Cachés (MMC)

Le signal acoustique de la parole est modélisable par un ensemble réduit d'unités acoustiques, qui peuvent être considérées comme des sons élémentaires de la langue.

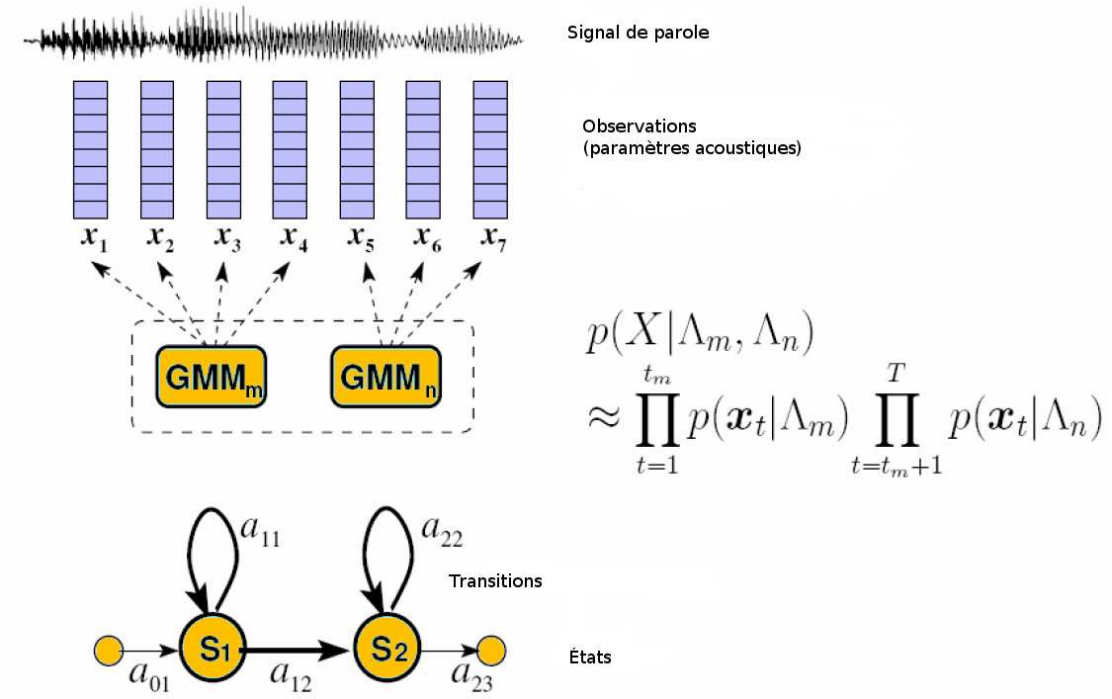

FIG. 1.2: *Paramétrisation et modèles acoustiques*

Classiquement, l'unité choisie est le phonème : un mot étant formé par leur concaténation. Des unités plus précises peuvent être employées comme les syllabes, les dissyllabes, les phonèmes en contexte, permettant ainsi de rendre la modélisation plus discriminante, mais cette amélioration théorique est limitée dans la pratique par la complexité induite et les problèmes d'estimation. Un compromis souvent employé est l'utilisation de phonèmes contextuels avec partage d'états.

Le signal de la parole peut être assimilé à une succession d'unités. Dans le cadre des SRAP Markoviens, les unités acoustiques sont modélisées par des Modèles de Markov Cachés (MMC), typiquement des MMC gauche-droite à trois états.

A chaque état du modèle de Markov est associée une distribution de probabilité modélisant la génération des vecteurs acoustiques via cet état. Un MMC est caractérisé par plusieurs paramètres :

- Son nombre d'états N
- L'ensemble des états du modèle $e = (e_i)_{(1 \leq i \leq N)}$
- Une matrice de transition entre les états : $A = (a_{ij})_{1 \leq i,j \leq N}$ de taille NxN
- La probabilité d'occupation d'un état à l'instant initial : $(\pi_i) \leq i \leq N : \pi_i = P(e_1 = e_i)$
- La densité de probabilité d'observation associée à l'état e_i : b_i. b_i qui est généralement modélisée par un modèle à mélange de Gaussiennes.

Un MMC est donc représenté par un ensemble de paramètres : $\theta_{MMC} = (N, A, \{\pi_i\}, \{b_i\})$. Les paramètres du MMC sont estimés empiriquement sur de grands

corpus de parole annotés.

Nous présentons succinctement dans les paragraphes suivants les techniques d'apprentissage et d'adaptation des modèles acoustiques.

1.2.2 Apprentissage et adaptation des modèles acoustiques

L'apprentissage des modèles acoustiques d'un SRAP permet de modéliser le message de la parole avec une quantité de données *a priori* annotées. Les techniques que nous présentons sont celles utilisées les plus couramment pour estimer correctement les paramètres des MMC. Cette partie décrira succinctement les principales méthodes d'apprentissage et d'adaptation pour les modèles et paramètres acoustiques.

Apprentissage par maximum de vraisemblance (ML)

L'estimation du maximum de vraisemblance (*Maximum Likelihood : ML*) est une méthode statistique utilisée pour déterminer les paramètres de la distribution de probabilité d'un échantillon X donné. Soit θ l'ensemble des paramètres associés au modèle m qui modélise le message w. Soit x une observation correspondant à l'acoustique du message w. Généralement, l'apprentissage consiste à déterminer les paramètres $\hat{\theta}$ maximisant la probabilité que l'observation x soit générée par le modèle m :

$$\hat{\theta} = \arg\max_{\theta_m} P(X = x|m) = \arg\max_{\theta} P(X = x|\theta) \tag{1.4}$$

Le paramètre θ est une variable inconnue à déterminer maximisant la vraisemblance avec l'échantillon X.

L'algorithme EM

EM est une méthode de maximisation proposée par [Dempster *et al.*, 1977], permettant de trouver le maximum de vraisemblance des paramètres de modèles probabilistes lorsque le modèle dépend de variables latentes non observables. L'algorithme EM alterne des étapes d'évaluation de l'espérance (*Expectation*), où la vraisemblance est maximisée en optimisant une fonction qui est l'espérance de la log-vraisemblance sous une distribution conditionnelle sachant les observations $E(log(P(X, Y; \theta|X, \theta)))$, et une étape de maximisation (*Maximisation*) estimant le maximum de vraisemblance des paramètres en maximisant la vraisemblance trouvée à l'étape E. Les paramètres trouvés en M sont réutilisés comme point de départ d'une nouvelle phase d'évaluation de l'espérance : la méthode est réitérée jusqu'à convergence. Cet algorithme se retrouve par exemple dans l'apprentissage des MMC avec l'algorithme de Baum-Welch.

Dans l'équation 1.4, $P(X = x|\theta)$ ne peut être maximisé directement à cause de l'incomplétude des données d'apprentissage. Ce problème est résolu par l'approche EM

qui permet en partant de conditions initiales des paramètres θ^0, d'attribuer des valeurs z^l aux données manquantes (*Expectation*) puis de trouver une nouvelle valeur θ^{l+1} des paramètres qui maximisera la vraisemblance des données complètes $P(y^l|\theta)$ avec $y^l = (x, z^l)$.

Estimation par information mutuelle maximale (MMIE)

Tandis qu'une estimation ML cherche à maximiser les vraisemblances, MMI (*Maximal Mutual Information*) est une approche discriminante : le principe général du MMI est de trouver au sein de différentes classes $G = g_1, g_2, ..., g_k$ dans un espace vectoriel défini par $F_m = V_1 \text{x} V_2 \text{x} ... \text{x} V_m$, quels sont les paramètres $< X, g >$ avec $X \in F_m$ et $g \in G$ qui discriminent le plus ces classes. Il faut donc déterminer quelles sont les composantes de X qui permettent de singulariser chaque classe. La métrique la plus appropriée pour déterminer si une composante peut s'associer à une classe est l'information mutuelle entre les valeurs de la composante et les valeurs contenues dans la classe. Afin de calculer cette quantité, on définit deux variables aléatoires : X_i la i^{me} composante de X pour un point de données et C la classe d'un point de données. L'information mutuelle entre X_i et C pour tous les points de données est :

$$Im(X_i, C) = H(C) - H(C|X_i) \tag{1.5}$$

Étant donné que $P(C)$ est identique pour toute valeur de i, et que MMI est utilisé pour ordonner, il est suffisant de calculer $P(C|X_i)$:

$$H(C|X_i) = -\sum_{c \in G} \sum_{x_i \in V_i} P(c, x_i) log P(c|x_i) \tag{1.6}$$

Où $p(c, x_i)$ est la probabilité jointe de voir une donnée de la classe c avec la composante x_i et $p(c|x_i)$ est la probabilité d'être dans la classe c de la composante x_i. Dans cette équation, les composantes les plus discriminantes obtiendront le score le plus élevé.

Cette méthode a été introduite par [Bahl *et al.*, 1986] afin d'adapter les paramètres de modèles de Markov pour les SRAP. MMIE (*Maximum Mutual Information Estimation*) a été par la suite développée pour les SRAP par [Valtchev *et al.*, 1997]. La fonction objective de MMIE est :

$$\mathcal{F}(\lambda) = \sum_{r=1}^{R} log \frac{P_\lambda(X_r|M_{w_r})P(w_r)}{\sum_{\hat{w}} P_\lambda(X_r|M_{\hat{w}})P(\hat{w})} \tag{1.7}$$

Où $\hat{w}$ représente toutes les séquences de mots possibles dans la tâche courante, $X = X_1, X_2, .., X_r$ les observations qui correspondent aux mots $w_1, w_2, .., w_R$. Les ré-estimations des moyennes et des variances des MMC sont décrites dans [Gao *et al.*, 2000].

Autres approches discriminantes : MPE, MWE, MCE

Avec l'augmentation de la puissance de calcul, ainsi que l'amélioration des méthodes d'apprentissage discriminantes telles que MMIE, plusieurs approches se sont développées. MMIE se concentre sur la maximisation des probabilités *a posteriori* des phrases d'apprentissage. MPE (*Minimum Phone Error*) [Povey et Woodland, 2002] et MWE (*Minimum Word Error*) [Heigold *et al.*, 2005, Yan *et al.*, 2008] fonctionnent sur un principe similaire à celui du MMI, mais cherchent à minimiser respectivement le taux d'erreur de phonèmes et le taux d'erreur mots. Avec un autre niveau de granularité, a été introduit le MCE (*Minimum Classification Error*) qui tend à minimiser le taux d'erreur au niveau des phrases.

Adaptation par maximum *a posteriori* (MAP)

La méthode d'estimation du Maximum *a posteriori* (MAP) peut être utilisée afin d'estimer un certain nombre de paramètres inconnus, comme par exemple les paramètres d'une densité de probabilité, reliés à un échantillon donné. Cette méthode a été introduite dans le cadre de la reconnaissance automatique de la parole par [Gauvain et Lee, 1994]. Dans le cas des modèles acoustiques, la méthode d'adaptation MAP permet de modifier les paramètres acoustiques d'un modèle générique pour rapprocher ce dernier du corpus de test. Ceci permet par exemple d'adapter un modèle acoustique générique à un locuteur spécifique. On considère un paramètre θ comme étant une variable aléatoire de distribution *a priori* $P(\theta)$. Le critère de maximum *a posteriori* cherche à maximiser la probabilité *a posteriori* $P(\theta|X)$. En appliquant la règle de Bayes, tout en considérant l'indépendance des échantillons par rapport à θ, l'adaptation de θ consiste à maximiser la valeur de $P(X|\theta)P(\theta)$, soit :

$$\begin{aligned} \theta_{map} &= \arg\max_{\theta} P(\theta|X) \\ &= \arg\max_{\theta} P(X|\theta)P(\theta) \end{aligned} \tag{1.8}$$

Par ailleurs, l'adaptation MAP est équivalente à un apprentissage par maximum de vraisemblance si la distribution *a priori* $P(\theta)$ est uniforme. L'adaptation MAP obtient de bons résultats et la quantité d'informations nécessaire à l'apprentissage est raisonnable en comparaison d'une approche par maximum de vraisemblance [Gauvain et Lee, 1994].

Adaptation par régression linéaire (MLLR)

L'adaptation de modèles par régression linéaire (*Maximum Likelihood Linear Regression* : MLLR) [Gales, 1997] est également une méthode communément utilisée pour modéliser des données *a priori*. Dans le cas de l'adaptation MLLR, l'hypothèse est faite que

les paramètres cibles peuvent être obtenus via une transformation linéaire des paramètres initiaux (figure 1.3) :

$$\hat{\mu} = A_i \mu_i + b_i \tag{1.9}$$

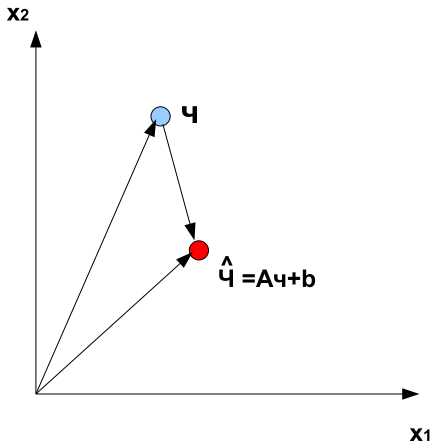

FIG. 1.3: *Apprentissage des paramètres via régression linéaire (MLLR)*

Avec $\hat{\mu}$ le vecteur cible, μ le vecteur initial, A_i la matrice de transformation et b_i le vecteur d'adaptation. Étant donné une séquence d'observation $X = x_1, ..., x_N$, l'adaptation MLLR doit trouver le nouvel ensemble de paramètres $\theta = \{A_i, b_i\}_{i=1}^{L}$ qui maximise la vraisemblance $P(X|\theta)$:

$$\hat{\theta} = \arg\max_{\theta} P(X|\theta) \tag{1.10}$$

Si des données d'apprentissage ne sont pas étiquetées, l'algorithme EM peut être appliqué pour obtenir un ensemble de paramètres optimal. L'une des contraintes de l'adaptation MLLR est la quantité très importante de paramètres à estimer. Une coupure sur les paramètres a été introduite pour résoudre ce problème : la régression par classes. Cependant l'apprentissage MLLR demande moins de données d'apprentissage que MAP ou ML lorsque le nombre de classes est réduit.

Quelques techniques d'adaptation des paramètres acoustiques

L'apprentissage fMLLR (*feature MLLR*) présentée par [Gales, 1997], contrairement aux méthodes qui modifient les modèles acoustiques (comme la méthode MLLR), s'applique sur les paramètres directement issus de l'observation. Ainsi, les modèles ne sont pas modifiés, et les paramètres sont rapprochés de ceux de l'apprentissage. Et ce tout

en modifiant leur espace de représentation. Ainsi, des caractéristiques particulières du signal seront dé-bruitées (locuteur, bruit etc.).

Une autre technique présentée par [Zhan et Waibel, 1997], *Vocal Tract Lenght Normalization* (VTLN), s'applique tout comme fMLLR sur les paramètres. Cet apprentissage s'appuie sur une normalisation du conduit vocal, qui diffère d'un locuteur à l'autre. Par cette adaptation, les variations de longueur sont éliminées par des filtres modifiant les fréquences.

1.3 Modèles de langage n-grammes

Les modèles de langage représentent un point clef du système de reconnaissance automatique de la parole. Ils introduisent les contraintes linguistiques dans le SRAP. Le modèle de langage modélise les contraintes liées à une langue, afin d'estimer la probabilité d'une suite de mots :

$$P(W_1^k) = P(w_1) \prod_{i=2}^{k} P(w_i|h_i) \tag{1.11}$$

Où h_i correspond à l'historique du mot w_i. De nombreux Systèmes de Reconnaissance Automatique de la Parole (SRAP) utilisent des modèles de langage n-grammes. Les modèles n-grammes correspondent à une modélisation stochastique du langage où l'historique d'un mot est représenté par les $n-1$ mots qui le précèdent :

$$P(W_1^k) = P(w_1) \prod_{i=2}^{n-1} P(w_i|w_1, ..., w_{i-1}) \prod_{i=n}^{k} P(w_i|w_{i-n+1}, ..., w_{i-1}) \tag{1.12}$$

Les modèles de langage n-grammes sont assez souples car ils permettent de modéliser des phrases grammaticalement incorrectes mais ils n'interdisent pas non plus de produire des phrases totalement incohérentes. Les modèles les plus couramment utilisés dans les SRAP sont les modèles d'ordre 3 ou 4 ; dans le cas d'un modèle tri-gramme, l'équation précédente s'écrit :

$$P(W_1^k) = P(w_1)P(w_2|w_1) \prod_{i=3}^{k} P(w_i|w_{i-2}w_{i-1}) \tag{1.13}$$

1.3.1 Estimation des modèles de langage

L'estimation des paramètres d'un modèle de langage n-grammes s'effectue en combinant deux composants : un modèle de décompte et un modèle de redistribution. L'ensemble des méthodes d'estimation effectuent un décompte des suites de mots observés

afin d'en extraire une probabilité d'apparition. Le principe est d'estimer toutes probabilités issues d'événements observés, puis de les redistribuer à des événements non vus. Cette seconde étape, qui correspond au lissage, permet d'associer une probabilité non nulle à des évènements jamais observés sur le corpus d'apprentissage. Les méthodes de lissage classiques calculent une probabilité non nulle en réduisant la fenêtre d'observation.

Les étapes de décompte et de redistribution sont très diverses :

- *linear Discounting*, *Absolute Discounting*, *Good-Turing* pour le décompte : chaque méthode permet d'estimer la fréquence d'un mot connaissant son historique.
- l'interpolation et le repli pour la redistribution : ces méthodes permettent de lisser les probabilités du modèle de langage afin d'estimer les évènements non observés, soit en interpolant les données inexistantes, ou en se repliant sur un historique plus restreint.

L'estimation des probabilités des suites de mots est calculée à partir de grands corpus de texte. Les méthodes d'estimation des paramètres des modèles de langage sont assez similaires à celles présentées au niveau de l'acoustique :

- Maximum de vraisemblance, pour un tri-gramme : $P_{mv} = \frac{tf(w_i w_j w)}{tf(w_i w j)}$ où $tf()$ correspond au décompte des mots (*term frequency*),
- Estimation Bayésienne, pour un tri-gramme : $P_B = \frac{tf(w_i w_j w) + 1}{tf(w_i w j) + k^2}$ où $tf()$ correspond au décompte des mots et k à la taille du vocabulaire.
- Maximum *a posteriori*
- ...

Les modèles n-grammes sont donc très dépendants du corpus d'apprentissage, et ont un champ de vision limité à la taille du n-gramme (qui est compris entre 3 et 5 généralement).

Les modèles n-grammes sont extrêmement simples, mais ont prouvé leur efficacité et leur souplesse. Ils se sont imposés dans les systèmes état de l'art bien que diverses alternatives efficaces aient été proposées dans la littérature [Schwenk et Gauvain, 2002, Schwenk, 2007], ils continuent d'être quasi systématiquement intégrés aux SRAP état de l'art.

1.3.2 Évaluation des modèles de langage

La qualité d'un modèle de langage dépend de sa capacité à orienter le SRAP afin d'en augmenter la performance. La mesure la plus couramment utilisée est la perplexité. La perplexité d'un modèle de langage correspond à sa capacité de prédiction. Plus la valeur de perplexité est petite, plus le modèle de langage possède des capacités de prédiction.

Généralement, la perplexité est estimée sur le corpus d'apprentissage pour définir

si les modèles choisis modélisent correctement le corpus. Elle est calculée sur le corpus de test, pour estimer le degré de généralisation du modèle. Cependant, bien que la perplexité permette d'estimer la capacité de représentation d'un modèle de langage, elle n'est pas systématiquement corrélée avec la qualité du décodage. Pour des modèles n-grammes, la perplexité se définit ainsi :

$$PP = 2^{-\frac{1}{n}\sum_{t=1}^{n} log_2 P(w_t|h)} \tag{1.14}$$

Où $P(w_t|h)$ est la probabilité associée au n-gramme $(w_t|h)$.

1.4 Algorithmes et stratégies de décodage

Dans cette section, nous présentons succinctement les principaux algorithmes de décodage utilisés dans les SRAP actuels. Un tour d'horizon sur l'ensemble des techniques de décodage est présenté dans [Aubert, 2002]. Nous décrirons plus en détail le principe d'un décodeur basé sur un algorithme A^*, ce décodeur étant la base du système utilisé dans nos travaux.

Lors d'un décodage, un SRAP génère, à partir des observations et de connaissances *a priori*, un ensemble d'hypothèses de mots. Cet ensemble peut être codé sous la forme d'un graphe ou treillis de mots. Le SRAP explore ce graphe afin de trouver le chemin qui maximisera la fonction de coût, qui regroupe les hypothèses linguistiques et acoustiques. Comme une exploration complète serait irréaliste, différentes heuristiques réduisent l'espace de recherche en éliminant les chemins peu probables localement. Bien que cet élagage puisse introduire des erreurs en éliminant une hypothèse globalement optimale, il permet d'obtenir de bons compromis entre temps de calcul et résultats. Nous présentons ici les principaux algorithmes d'exploration.

1.4.1 Décodage avec extension dynamique du graphe

Le principe d'un SRAP est d'explorer un graphe à la recherche d'une hypothèse maximisant à la fois les modèles acoustiques et linguistiques. Une évaluation exhaustive s'avèrerait impossible, il est indispensable d'explorer dynamiquement un graphe virtuel. Il est nécessaire d'appliquer des heuristiques qui limiteront l'espace de recherche. Deux approches se distinguent dans les approches d'exploration dynamique d'un graphe basé sur des n-grammes pour générer une séquence de mots :

– les algorithmes dits de "graphes réentrants" où une copie virtuelle du graphe est explorée pour chaque contexte linguistique. L'information relative à chaque contexte est enregistrée pour chaque chemin et combinée avec la nouvelle racine virtuelle dépendant de l'historique relatif au modèle de langage [Ney *et al.*, 1992].

– les algorithmes de "graphes synchrones" qui génèrent une copie virtuelle du graphe pour l'ensemble des hypothèses se terminant à un même temps t. Ainsi, toutes les hypothèses se finissant au même instant sont explorées via le même graphe [Ortmanns et Ney, 2000].

Les deux approches citées ont été intégrées dans le cadre des algorithmes de recherche synchrones, et la seconde a été intégrée dans les décodeurs asynchrones à pile.

1.4.2 Recherche synchrone basée sur un arbre réentrant

C'est l'algorithme le plus répandu dans les SRAP, dont la mise en œuvre la plus courante est un Viterbi en faisceau (*beam search*). Sa conception se rapproche de la programmation dynamique et s'applique de la même manière que sur des graphes statiques. Comme pour tous les algorithmes de recherche synchrones, les hypothèses sont explorées en parallèle et estimées par rapport à la même portion de signal : cet aspect permet d'appliquer facilement des heuristiques de coupure de l'espace de recherche. Cependant, l'intégration d'un modèle de langage n-grammes et des informations contextuelles aux mots n'est pas triviale.

En effet l'application d'un algorithme Viterbi est simple quand $n = 1$, mais l'intégration d'un modèle de langage n-gramme avec $n > 1$ nécessite de développer des chemins dépendants de l'historique.

1.4.3 Recherche synchrone basée sur des arbres synchrones

L'idée consiste à partager le même prochain mot pour toutes les hypothèses se terminant à un même temps t. Le mot suivant est partagé, mais associé à plusieurs historiques.

Dans cette approche l'algorithme dynamique d'alignement est appliqué pour tout nouveau mot et pour chaque trame de départ, sans avoir à conserver l'historique des mots.

Cependant, les informations liées aux contextes phonétiques et au modèle de langage sont difficiles à intégrer : les nouveaux arbres explorés ont divers historiques regroupés sur le même nœud racine. La mise en œuvre de cet algorithme s'avère rare.

1.4.4 Recherche asynchrone à pile

Le principe de ces algorithmes [Jelinek, 1969] est d'explorer en profondeur les hypothèses qui semblent *a priori* meilleures, en priorité ; en étendant mot par mot l'hypothèse sélectionnée, et sans la contrainte que les hypothèses explorées se terminent à un même temps t. Les réalisations de ce type d'algorithmes de recherche se basent sur des piles qui ordonnent les hypothèses à explorer [Paul, 1991, Nocera *et al.*, 2002b].

Les difficultés liées à cette exploration sont :

– Bien choisir les critères pour étendre un chemin donné.
– Comment évaluer correctement la qualité *a priori* des différents chemins.
– Comment évaluer les seuils de coupure d'exploration.

La mise en œuvre la plus courante est celle de l'algorithme A^*. Lors d'une première passe sur un treillis de mots ou de phonèmes, les meilleurs chemins au niveau acoustique sont retenus et ordonnés (en général via un algorithme de type Viterbi). L'algorithme A^* explore le graphe en y rajoutant ses contraintes (linguistiques ou autres) et en estimant le chemin restant (appelé sonde) grâce aux chemins pré-estimés sur l'acoustique. L'estimation sur l'acoustique garantit une sonde de coût minimal, et se rapprochant de la meilleure solution finale : la sonde peut être améliorée en intégrant des informations sur l'anticipation linguistique [Massonié *et al.*, 2005]. Sa qualité est primordiale : l'exploration plus ou moins complète du graphe dépend de cette dernière. Nous décrivons en détail l'algorithme A^* dans la section suivante.

Exemple du décodage A^*

L'algorithme A^* se déroule de la manière suivante :

1. L'algorithme A^* débute sur un nœud donné du graphe (le premier pour la condition initiale). Il applique à ce nœud un coût (composé généralement d'une partie linguistique et d'une partie acoustique), puis estime la distance séparant ce nœud du nœud terminal. La somme du coût et de l'évaluation représente le coût estimé du chemin menant à ce nœud. Le nœud est alors ajouté à une file d'attente prioritaire : l'*open list*.
2. L'algorithme retire le premier nœud de l'*open list*. Si cette dernière est vide, il n'y a aucun chemin du nœud initial au nœud d'arrivée, ce qui est une condition d'arrêt de l'algorithme. Si le nœud retenu est le nœud d'arrivée, l'algorithme reconstruit le chemin complet à partir des informations sauvegardées dans la liste *closed list* et s'arrête.
3. Si le nœud n'est pas le nœud d'arrivée, tous les nœuds adjacents sont explorés. Pour chaque nœud successif, A^* calcule et stocke son coût . Celui-ci est calculé à partir de la somme du coût de son ancêtre et du coût de l'opération pour atteindre ce nouveau nœud.
4. L'algorithme met également à jour la liste des nœuds qui ont été vérifiés, dans la liste *closed list*. Si un nouveau nœud existe déjà dans cette liste avec un coût égal ou inférieur, aucune opération n'est faite sur ce nœud ni sur son jumeau s'y trouvant.
5. Las distance évaluée entre le nouveau nœud et le nœud d'arrivée est ajoutée au coût du nœud. Ce nœud est alors ajouté à la l'*open list*, à moins qu'un nœud identique dans cette liste ne possède déjà un coût inférieur ou égal.
6. Une fois ces trois étapes réalisées pour chaque nouveau nœud adjacent, le nœud original pris dans l'*open list* est ajouté à la liste des nœuds explorés. Le nœud suivant est alors retiré de l'*open list* et le processus recommence.

1.4.5 Décodages multi-passes

Les SRAP utilisent souvent des stratégies multi-passes. Généralement la première passe génère une transcription qui sera réutilisée pour adapter les modèles acoustiques en fonction des locuteurs ou de la qualité d'enregistrement. Les premières passes permettent également de générer des graphes de mots qui peuvent être ré-explorés *a posteriori* avec des modèles de langages plus importants. Ces stratégies en plusieurs passes permettent ainsi d'introduire à chaque itération une information supplémentaire : généralement les informations rajoutées n'auraient pu l'être à l'itération précédente, car la quantité d'hypothèses en concurrence était trop importante.

1.5 Graphes de décodage

Les systèmes de reconnaissance automatique de la parole génèrent à partir du signal audio des graphes dans lesquels l'hypothèse de coût minimal est recherchée. Le treillis est une représentation d'une portion du graphe de mots qui a été effectivement développée par l'algorithme de recherche. Dans ce chapitre nous présenterons les différents travaux relatifs aux treillis, et comment ceux-ci sont exploités pour minimiser les taux d'erreur.

1.5.1 Les réseaux de confusion

[Mangu Lidia, 2000] introduisent le concept des réseaux de confusion qui peuvent être considérés comme une représentation compacte des treillis (figure 1.4). La topologie d'un réseau de confusion se définit ainsi : chaque nœud correspond à un intervalle de temps du treillis et les liens entre les nœuds correspondent chacun à un mot. Chaque lien s'associe à une probabilité, et la somme des probabilités entre deux nœuds est égale à 1. Plusieurs algorithmes comme [Xue *et al.*, 2005] permettent de réduire un treillis en réseau de confusion. Les réseaux de confusion permettent ainsi de fusionner toutes les hypothèses d'un treillis, et d'ordonner toutes les hypothèses de mots sur un seul intervalle. Ils sont surtout utilisés pour simplifier la représentation d'un treillis qui devient alors compacte et lisible. Cependant, cette simplification élargit l'espace de recherche : suite à la factorisation, les contraintes sur les chemins sont enlevées ; ils sont ainsi utilisés pour effectuer un décodage supplémentaire où certaines contraintes disparaissent. Les réseaux de confusion permettent également d'estimer des probabilités *a posteriori*. [Mangu Lidia, 2000] montrent que la bonne hypothèse se trouve dans 90% des cas parmi les dix premiers mots candidats d'un nœud du réseau de confusion.

1.5.2 Décodage par fWER

[Wessel *et al.*, 2001] introduisent une nouvelle fonction de coût s'appuyant sur les frontières de mots. Ils montrent que les erreurs sur les frontières de mots sont forte-

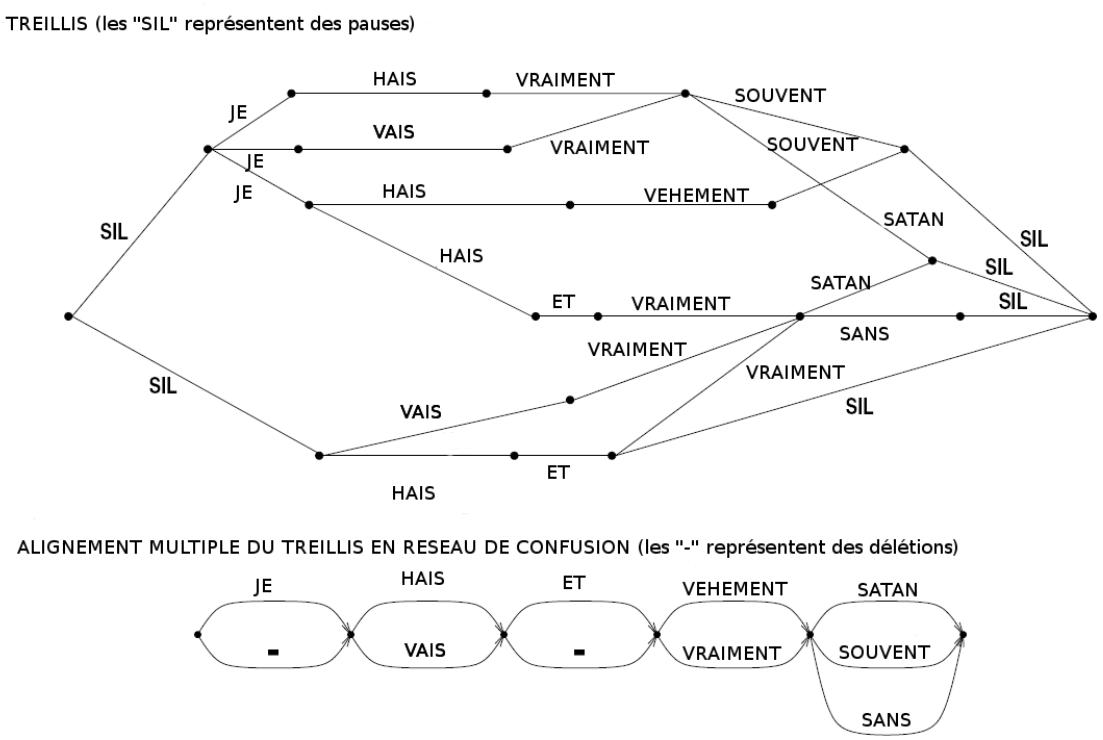

FIG. *1.4:* *Exemple de réseau de confusion à partir d'un treillis*

ment corrélées avec le taux d'erreur mots. Cette corrélation permet de se focaliser sur les zones d'un treillis où les frontières de mots diffèrent d'un chemin à l'autre. Les différences entre les frontières de mots s'expliquent par des insertions, suppressions ou substitutions de mots avant l'hypothèse considérée. Un des avantages de cette méthode est l'abstraction des hypothèses en tant que telles, étant donné que seule l'information temporelle est prise en compte pour définir les zones bien décodées (figure 1.5).

D'un point de vue formel, le *Time Frame Error* se définit ainsi : considérons deux hypothèses (suites de mots) contenues dans un graphe de mot $[w;t]_1^N$ et $[v;\tau]_1^M$. La fonction de coût sera :

$$\mathcal{C}([w;t]_1^N,[v,\tau]_1^M) = \sum_{n=1}^{N} \frac{\sum_{\hat{t}=t_{n-1}+1}^{t_n} 1-\delta(w_n,v_{\hat{t}})}{1+\alpha.(t_n-t_{n-1}-1)} \tag{1.15}$$

$v_{\hat{t}}$ est l'identifiant du mot hypothèse dans la phrase $[v;\tau]_1^M$ qui est en intersection avec la trame $\hat{t}$, α est un facteur de lissage défini empiriquement et δ une fonction de Kronecker qui renvoie une valeur binaire en fonction de l'intersection ou non de $(w_n, v_{\hat{t}})$.

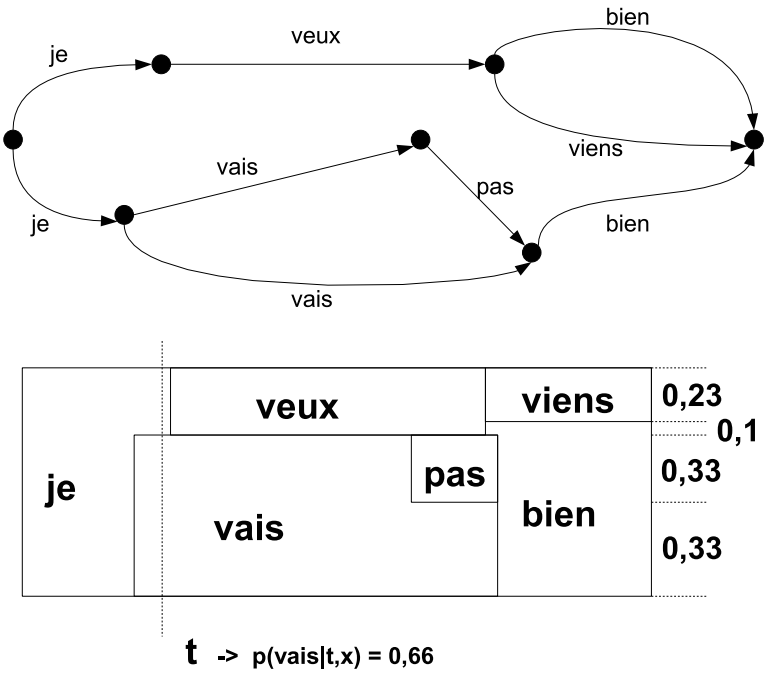

FIG. 1.5: *Exemple de calcul des probabilités a posteriori à partir d'un décodage par fWER. Les rectangles de la première figure correspondent au recouvrement temporel des hypothèses ainsi qu'à la somme des probabilités a posteriori .*

En reprenant la formulation générale de la combinaison de systèmes et en y intégrant cette fonction de coût, on obtient :

$$\{[w;t]_1^N\}_{opt} = \arg\min_{[w;t]_1^N} \left\{ \sum_{[v;\tau]_1^M} \mathcal{C}([w;t]_1^N, [v,\tau]_1^M) p([v;\tau]_1^M | x_1^T) \right\} \tag{1.16}$$

Après simplification (voir les détails dans [Wessel *et al.*, 2001]), la fonction de décision devient :

$$\{[w;t]_1^N\}_{opt} = \arg\min_{[w;t]_1^N} \left\{ \sum_{n=1}^{N} \mathcal{S}([w_n; t_{n-1}+1, t_n]) \right\} \tag{1.17}$$

$$\text{avec } \mathcal{S}([w_n; t_{n-1}+1, t_n]) = \frac{\sum_{\hat{t}=t_{n-1}+1}^{t_n} \left[1 - p(w_n|\hat{t}, x_1^T)\right]}{1 + \alpha(t_n, t_{n-1}-1)} \tag{1.18}$$

Et la probabilité d'observation d'un mot w_n à une trame $\hat{t}$ étant donné une observation acoustique x_1^T est alors donnée par :

$$p(w_n; \hat{t}, x_1^T) = \sum_{[w;\tau,t]:\tau \leq \hat{t} \leq t} \delta(w_n, v) p([v;\tau,t]|x_1^T) \tag{1.19}$$

Les principaux avantages du décodage par fWER comparé à un décodage par des réseaux de confusion sont :

- Le faible coût de calcul des alignements étant donné qu'ils ne se basent que sur les frontières de mots.
- Les zones de décodage incertaines sont facilement identifiables, car les erreurs sur les trames sont très corrélées avec le WER.
- Le décodage par fWER préserve la structure du graphe et les sorties produisent des hypothèses de mots ayant des frontières correctes. Dans les réseaux de confusion, les frontières de mots ne sont utilisées que pour aligner les mots, mais une fois le réseau de confusion construit toutes les informations temporelles sont perdues ; ce qui n'est pas le cas dans la construction d'un réseau via fWER.

1.5.3 Probabilités *a posteriori*

Les probabilités *a posteriori* [Wessel *et al.*, 1998, Evermann et Woodland, 2000a] sont estimées à partir des scores acoustiques et linguistiques d'une séquence de mots déterminée. Cette estimation est extraite du treillis de mots généré par le système de reconnaissance automatique de la parole, lors de son exploration par un algorithme Viterbi.

Du treillis au réseau de confusion

Chaque lien dans le treillis contient des informations sur le score acoustique et la vraisemblance de l'hypothèse de mots courante, et ce pour chaque mot de l'hypothèse et ses variantes dans le temps. Par ailleurs, en raison de la dépendance des modèles acoustiques et linguistiques il faut s'assurer que l'historique de chaque lien est unique : il faut étendre le treillis avec le modèle de langage.

Classiquement, à partir du treillis étendu, l'estimation des probabilités *a posteriori* est effectuée en deux étapes. La première consiste à étendre le treillis à l'aide d'un algorithme de type *forward-backward*. Au cours de la seconde étape, la probabilité *a posteriori* du lien $p(l|X)$ est définie comme étant la somme des probabilités de l'ensemble des chemins chemins Q passant par le lien l, normalisée par la probabilité du signal $p(X)$:

$$p(l|X) = \frac{\sum_{Q_l} p(q, X)}{p(X)} \tag{1.20}$$

Où $p(X)$ est la somme de tous les chemins du treillis. La probabilité d'un chemin $p(q, X)$ est composé de la vraisemblance acoustique $p_{ac}(X|q)$ ainsi que de la vraisemblance linguistique $p_l(W)$:

$$p(q, X) = p_{ac}(X|q)^{\frac{1}{\gamma}} p_l(W) \tag{1.21}$$

Où γ est un facteur d'échelle contrôlant la partie acoustique. Généralement la partie acoustique est diminuée car le score acoustique est contextuel au mot précédent. Or après factorisation en réseau de confusion, ce contexte est perdu : la partie acoustique perd de sa précision.

Les probabilités *a posteriori* peuvent être utilisées pour réévaluer un treillis afin de trouver des chemins alternatifs se basant sur des estimations plus globales. Ils sont également souvent une première étape nécessaire à l'élaboration des mesures de confiance [Wessel, 2002].

1.5.4 Les mesures de confiance

Les scores de confiance jouent un rôle important dès lors qu'on exploite les sorties d'un SRAP : ils permettent, associés à la sortie d'un système, d'estimer la probabilité qu'une hypothèse soit correcte. Les mesures de confiance sont utilisées dans de nombreuses applications : la reconnaissance de la parole, son interprétation, les systèmes de dialogue, l'adaptation des modèles acoustiques... Les propriétés des mesures de confiance sont définies ainsi : soit une suite N de mots hypothèse $w_1, ...w_N$. À chaque mot est associé une mesure de confiance $S(w)$ telle que $S(w)$ soit dans l'intervalle $[0, 1]$ et doit être perçue comme la probabilité que le mot w est correct. Dans le cadre théorique on obtient donc :

$$\mu(S) = \frac{1}{N} \sum_{i=1}^{N} S(w_i) = p_{ok} \tag{1.22}$$

Où p_{ok} est le taux de mots bien reconnus par le système. Cette métrique, comparée au taux réel d'erreur, permet d'estimer la qualité des scores de confiance. Des mesures de confiance sont parfois estimées à un autre niveau que celui des mots : les phonèmes ou les phrases [Lo *et al.*, 2004, Lo et Soong, 2005]. Les mesures de confiance se classent dans quatre catégories :

- Les mesures dérivées des probabilités *a posteriori* [Mauclair *et al.*, 2006]
- Les mesures issues de paramètres sélectionnés au cours du décodage, nommées prédicateurs [Fu et Du, 2005]
- Les mesures basées sur un score de confiance binaire accordant ou non un degré de fiabilité à l'hypothèse [Zhang et Rudnicky, 2001, Moreno *et al.*, 2001]
- Les mesures se basant sur des connaissances *a priori* : [Wiggers et Rothkrantz, 2003]

Les mesures de confiance s'estiment de multiples manières. Nous présentons non exhaustivement les plus communes.

Un score de confiance simple s'estime directement à partir du graphe de recherche [Wessel *et al.*, 1998, Wessel *et al.*, 2000, Wessel *et al.*, 2001], en calculant les probabilités *a posteriori*. Ce score de confiance pour un mot w résultera donc du ratio entre la somme des probabilités de tous les chemins passant par l'arc aboutissant à w et la somme des probabilités de tous les chemins composant le graphe. Typiquement, cette mesure se fait via un algorithme de *forward-backward*. Des variantes permettent de prendre en compte le recouvrement temporel entre les arcs, ou d'égaliser les probabilités *a posteriori* [Mauclair, 2006], méthode qui consiste à modifier l'espace de représentation des probabilités *a posteriori* par des fonctions affines afin de les projeter dans l'espace des scores de confiance.

Les scores de confiance peuvent également se calculer à partir de la liste des N meilleures hypothèses du décodeur [Souvignier et Wendemuth, 1999]. La méthode est similaire à celle utilisant les probabilités *a posteriori*, avec un espace de recherche réduit à celui des meilleures hypothèses. Ces mesures demeurent moins efficaces que les précédentes, du fait que certaines hypothèses ont été écartées et que l'espace de recherche est plus restreint.

Une méthode répandue est l'utilisation des réseaux de confusion : les scores de confiance peuvent être issus de la distribution de probabilité entre les arcs ou de la densité des nœuds.

Un autre type de mesure des scores de confiance est le rapport de vraisemblance, qui permettra de catégoriser l'hypothèse dans une classe correcte ou une classe incorrecte :

$$SV(X|W) = \frac{P(X|H_C(W))}{P(X|H_E(W)} \tag{1.23}$$

Où $H_C(W)$ représente l'hypothèse définissant W comme correcte et $H_E(W)$ comme incorrecte, X est l'observation. $H_C(W)$ peut être facilement estimé via les modèles acoustiques et linguistiques, mais $H_E(W)$ pose des problèmes : il faut modéliser ce qui est incorrect. Plusieurs méthodes ont été proposées :

- Les anti-modèles [Sukkar *et al.*, 1996, Rahim *et al.*, 1997] qui consistent à associer à chaque observation, un anti-modèle caractérisant les ambiguïtés. Cette approche s'effectue généralement au niveau phonétique.
- La catégorisation par distribution de probabilités, où l'on crée deux classes, l'une pour les incorrects et l'autre pour les corrects [Zhang et Rudnicky, 2001, Moreno *et al.*, 2001]. Un classifieur (SVM [VAPNIK, 1995, VAPNIK, 1982] ou boosting [Moreno *et al.*, 2001], [Schapire et Singer, 2000]) va apprendre sur un corpus d'apprentissage à discriminer les paramètres en fonction de multiples variables : les scores acoustiques, les durées, le nombre d'hypothèses concurrentes...
- Les modèles de rejet [Sukkar *et al.*, 1996, Rahim *et al.*, 1997] qui sont appris sur des corpus bruités et modélisés par des MMC : ils permettent de prendre en compte le bruit introduit par les mots hors vocabulaire
- Les cohortes [Rahim *et al.*, 1997] qui consistent à normaliser la probabilité du mot de l'hypothèse par le nombre de ses alternatives dans les N premières meilleures

du système :

$$SV(X|W) = \frac{P(X|W)}{\sum_{i=1,W_i \neq W}^{N} P(X|W_i)} \tag{1.24}$$

[Siu et Gish, 1999] et [Jiang, 2005] présentent respectivement des évaluations et un état de l'art sur les mesures de confiance actuelles. Ces dernières sont aussi bien utilisées pour l'auto-évaluation des systèmes que pour leur combinaison : elles constituent un estimateur de la qualité de chaque hypothèse.

Généralement, l'ensemble des mesures de confiance présentées sont efficaces. Mais, utilisées séparément, elles présentent toutes certaines faiblesses. Les meilleures mesures sont obtenues en combinant les approches et en exploitant leurs complémentarités [Mauclair, 2006].

1.6 Évaluation d'un système de reconnaissance automatique de la parole

Les SRAP sont souvent évalués en terme de taux d'erreur mot (*Word Error Rate*). Le WER est basé sur une mesure résultant de la programmation dynamique. L'hypothèse reconnue par le SRAP est alignée avec une hypothèse de référence via un algorithme d'alignement dynamique. Le WER se calcule donc :

$$WER = \frac{S + D + I}{N}.100 \tag{1.25}$$

Où S correspond aux substitutions, D aux suppressions, I aux insertions et N est le nombre de termes dans la référence.

D'autres métriques ont été introduites, notamment dans le but d'estimer la fidélité sémantique des transcriptions réalisées [Sarikaya *et al.*, 2005, San-Segundo *et al.*, 2001], pour des systèmes d'interprétation de dialogue, d'indexation...

Afin d'évaluer la fiabilité de ces mesures statistiques, il convient de calculer un intervalle de confiance relatif au nombre d'échantillons et d'erreurs. Cet intervalle de confiance est calculé en considérant que l'apparition d'une erreur de reconnaissance sur un mot ou sa non reconnaissance est associée à une variable aléatoire binomiale, dont la distribution dépend des couples (mot reconnu, mot prononcé).

Dans la situation de combinaison de systèmes, il est nécessaire d'estimer la fiabilité des résultats, qui dépend directement de la quantité d'échantillons utilisés. L'intervalle de confiance peut se calculer en supposant qu'une erreur de reconnaissance dépend d'une variable aléatoire binomiale dépendant des couples $\{motreconnu, motprononc\}$. Dans la littérature, un intervalle de confiance est proposé par [SAPORTA, 1990] :

$$wer_f - u_{\frac{\alpha}{2}} \sqrt{\frac{wer_f(1-wer_f)}{k}} < wer_p < wer_f + u_{\frac{\alpha}{2}} \sqrt{\frac{wer_f(1-wer_f)}{k}} \tag{1.26}$$

Où k est le nombre d'échantillons, wer_f est la quantité d'erreurs obtenues sur le corpus de test. α permet de définir l'intervalle de confiance : si $\alpha = 95\%$, alors wer_p sera défini avec une confiance de $+/- 0.05\%$. $u_{\frac{\alpha}{2}}$ est défini par une table de *Student* : $u_{0.425} = 1.96$.

Deuxième partie

Exploitation de transcriptions *a priori*

Chapitre 2

État de l'art : Exploiter des transcriptions *a priori*

Sommaire

Ce chapitre présente un état de l'art sur l'utilisation des transcriptions *a priori*. Les problèmes relatifs à la qualité des transcriptions sont survolés, puis nous présentons les méthodes existantes permettant d'aligner et synchroniser des transcriptions imparfaites avec les sorties d'un SRAP. Finalement cet état de l'art présente les stratégies permettant d'adapter un SRAP par l'intermédiaire de transcriptions *a priori*.

Dans de nombreuses situations, il est facile de se procurer des transcriptions correspondant à un enregistrement. Ces transcriptions sont plus ou moins fidèles en fonction du domaine. Elles sont particulièrement proches dans les pièces de théâtre avec le script des acteurs ou les films avec leurs scénarios. Des transcriptions plus approximatives peuvent être trouvées dans les prompts des journalistes, ou d'autres encore plus divergentes dans les sous-titres de films.

Ces transcriptions imparfaites introduisent deux problèmes intimement liés :
- Comment exploiter ces informations pour obtenir un corpus propre
- Comment s'appuyer sur des transcriptions pour améliorer le décodage

Cette partie décrira les travaux réalisés sur ce type d'informations en vue de les exploiter ainsi que les problèmes qu'elles soulèvent.

2.1 Qualité des prompts ou des sous-titres

Les notes des journalistes sont souvent incomplètes ou inexactes. L'audio peut contenir des parties non transcrites, c'est le cas des publicités ou des invités dans les émissions radiophoniques. Les transcriptions sont alors éclatées sur tout le signal audio. Parfois, le texte s'éloigne sur la forme tout en gardant la même sémantique. En effet, les prompts sont une aide que les journalistes ne suivent pas mot à mot. Dans d'autres situations, des parties de transcriptions n'existent pas dans l'audio : une interview annulée ou un flash oublié. De même, dans le cadre du théâtre, les acteurs s'éloignent de leur texte : une réplique est oubliée, une autre insérée tout en conservant la ligne directrice de la pièce. Dans le cas des sous-titres, [Placeway et Lafferty, 1996, Cardinal *et al.*, 2005] ont mesuré un taux d'erreur mot d'environ 10% à 20% ; dans cette situation, la contrainte pour que les sous-titres tiennent dans un espace réduit induit ce taux d'erreurs : l'idée est conservée, mais le texte ne correspond pas littéralement au contenu linguistique. Malgré ces approximations, de nombreuses informations peuvent encore être exploitées. Il est alors nécessaire de les extraire correctement afin de les utiliser. La section suivante définit les méthodes existantes pour exploiter et modéliser l'information présente dans ces corpus.

2.2 Problèmes de synchronisation et alignement

Les données associées à des flux audio sont rarement annotées précisément d'un point de vue temporel. Or, s'il est nécessaire de les utiliser avec l'audio, il est alors indispensable de synchroniser les données entre elles. Plusieurs cas se distinguent :
- Aligner des grands corpus exacts
- Aligner des grands corpus approximatifs
- Aligner des petites portions de corpus, noyées dans de grandes zones de texte

L'alignement de grands corpus soulève un problème supplémentaire : trouver des points d'ancrage à partir desquels l'alignement sera effectué. Nous décrivons, dans les

paragraphes qui suivent quelles sont les méthodes existantes pour aligner et synchroniser les corpus. Nous présentons dans le chapitre suivant une méthode originale permettant de remédier aux difficultés rencontrées.

2.3 Méthodes d'alignement

2.3.1 DTW, dérivés et améliorations

Les algorithmes d'alignement de deux séquences ont été très largement explorés [Keogh et Pazzani, 2001]. Le plus connu est le *Dynamic Time Warping* (DTW) [Wagner et Fisher, 1974], où les séquences alignées sont traditionnellement représentées comme les lignes d'une matrice. Cet algorithme consiste à effectuer une comparaison dynamique entre une matrice de référence et une matrice de test. Avec deux vecteurs définis par $H = h_1, ..., h_i, ..., h_n$ et $T = t_1, t_2, ..., t_j, ..., t_n$

Une matrice de dimension (nxm) est construite, (i, j) est la distance euclidienne entre les points H_i et T_j : $d(h_i, t_j) = (h_i - t_j)^2$

Un alignement entre T et H est le chemin W suivant les éléments contigus de cette matrice tel que $W = w_1, w_2, ..., w_k$ où $max(m, n) \leq k \leq m + n - 1$

Le chemin W doit commencer et finir aux extrémités d'une diagonale de la matrice et l'évolution est restreinte vers les éléments adjacents de la matrice. Les étapes successives dans le chemin sont réparties de manière monotone dans le temps. L'algorithme avance alors sur le chemin qui minimise la distance :

$$DTW(Q, C) = min \left\{ \sqrt{\frac{\sum_{k=1}^{K} w_k}{K}} \right\} = min \left\{ \sqrt{\frac{\sum_{k=1}^{K} (h_{i_k} - t_{j_k})^2}{K}} \right\} \tag{2.1}$$

L'objectif est d'évaluer la distance cumulée au niveau de la comparaison du couple de points (i, j), notée $\gamma(i, j)$ tel que décrit par l'équation suivante :

$$\gamma(i, j) = d(h_i, t_j) + min\ \gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1) \tag{2.2}$$

Le résultat est une mesure de la distance entre le test et la référence. Les algorithmes dynamiques se sont notamment développés dans le cadre de l'alignement de séquences moléculaires [Smith et Waterman, 1981], et des variantes ont été adoptées. C'est le cas de l'algorithme Smith-Waterman [Ahmed, 2005], qui permet tout en alignant globalement des séquences, de relever les alignements locaux. Cependant, les algorithmes d'alignement dynamiques sont peu adaptés à l'alignement de transcriptions issues de la reconnaissance automatique de la parole : leur complexité est proportionnelle à la taille des corpus alignés avec la sortie du SRAP.

2.3.2 Alignement de segments audio sur transcriptions parfaites

Les travaux de [Placeway et Lafferty, 1996] ont été étendus par [Moreno *et al.*, 1998]. Les auteurs proposent un algorithme pour aligner de longs segments audio avec la sortie d'un système de RAP. Leur algorithme se base sur un alignement graduellement forcé sur sa sortie. L'alignement est essentiellement contraint par le modèle de langage ainsi que par le lexique. Afin de minimiser les erreurs, l'algorithme s'applique itérativement au fur et à mesure des passes. Le système se décompose en plusieurs modules :

1. L'analyse de la transcription pour en extraire un modèle de langage ainsi qu'un lexique. Une passe est alors effectuée sur tout l'audio en utilisant ce modèle de langage et son lexique.
2. Le résultat du SRAP est alors aligné via un algorithme dynamique avec la transcription exacte. Cette étape permet de localiser des zones de plus grande confiance, dans lesquelles nombre de mots de la transcription *a priori* correspondent à la sortie du SRAP.
3. Ces zones permettent ainsi de délimiter des segments bien alignés ainsi que des segments non-alignés. L'algorithme est réitéré sur chaque segment non-aligné : le modèle de langage et le lexique sont restreints par la zone correspondante dans la transcription *a priori*.

L'application récursive s'arrête lorsque tous les segments sont alignés et que le SRAP ne trouve plus de nouveaux mots.

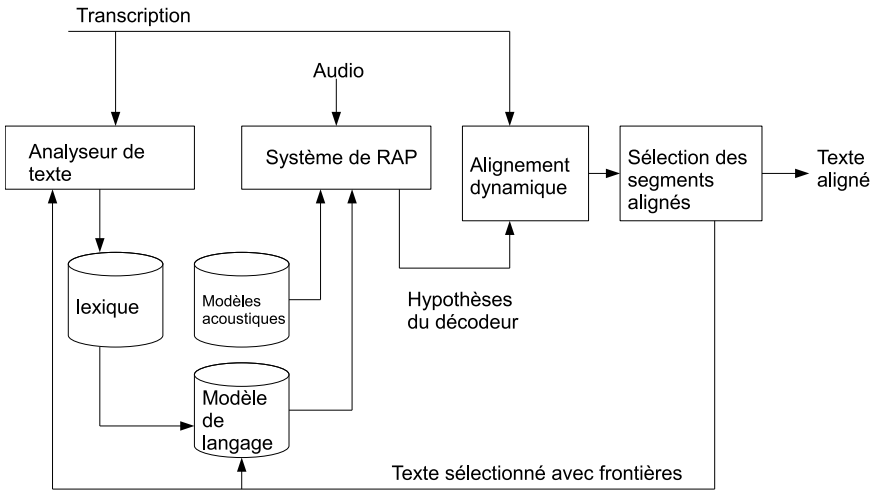

FIG. 2.1: *Diagramme du programme d'alignement présenté par [Moreno* et al.*, 1998]*

Ainsi dans l'algorithme de [Moreno *et al.*, 1998], aligne récursivement plus de 99% de la transcription : la méthode est robuste au bruit ambiant et converge, y compris sur les zones difficiles. Cependant, cette dernière se limite à des transcriptions presque parfaites. Le schéma de fonctionnement de cet algorithme est présenté sur la figure 2.1.

Un algorithme similaire est présenté dans [Robert-Ribes et Mukhtar, 1997], toutefois, il ne s'applique pas à un SRAP grand vocabulaire.

2.3.3 Alignement de segments audio sur transcriptions imparfaites

Les méthodes précédemment décrites nécessitent que les transcriptions fournies soient fidèles. Le problème de l'alignement de sorties SRAP sur des transcriptions se complique lorsqu'elles sont imparfaites. Nous allons présenter quelques techniques utilisées pour réaliser des alignements au sein de systèmes de reconnaissance automatique de la parole.

2.3.4 Exploitation de sous-titres

L'exploitation de transcriptions *a priori* a été introduite par [Placeway et Lafferty, 1996]. Les auteurs formalisent le problème comme l'ajout d'un canal d'information au sein du système de reconnaissance. Leurs travaux se basent sur l'intégration de sous-titres issus d'un journal dans un système de RAP. Le journal E est lu par un journaliste qui produit un signal acoustique X. Les sous-titres H sont créés par un scripteur qui prend en compte l'aspect visuel du sous-titre (pour qu'il rentre dans un espace relativement réduit), supprime certaines fautes, reformule des phrases, ou enlève des passages jugés sans importance. Les auteurs modélisent ces évènements sous la forme d'une distribution de probabilité conditionnelle $P(H, X|E)$.

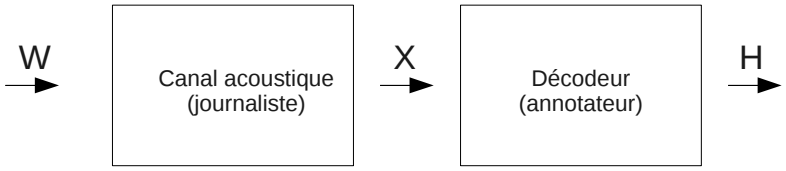

FIG. 2.2: *Formalisation de la problématique avec l'ajout d'un canal supplémentaire*

La figure 2.2 présente la problématique où il faut retrouver la séquence W la plus probable qui a permis d'émettre sur les deux canaux X et H. En intégrant ce canal à la formule générale d'un SRAP, on obtient :

$$\hat{W} = \arg\max_{w} P(W|X,H) \tag{2.3}$$

$$= \arg\max_{W} P(X|H,W)P(W)P(H|W) \tag{2.4}$$

Les auteurs émettent l'hypothèse de l'indépendance des canaux X et H. La tâche du décodeur consiste alors à maximiser la probabilité du modèle de langage, du modèle de transcription et du modèle acoustique :

$$\hat{W} = \arg\max_{W} P(X|W)P(H|W)P(W) \tag{2.5}$$

La simple interpolation d'un modèle appris sur les sous-titres avec un modèle de langage général donnant de mauvais résultats, les auteurs proposent un modèle représentant les sous-titres qui se base sur un modèle de Markov caché. Les arcs représentant alors les insertions, substitutions ou suppressions (figure 2.3). Le parcours optimal du MMC revient à minimiser la distance d'édition entre les sous-titres et l'hypothèse du SRAP. Le MMC est combiné avec le modèle de langage pour obtenir la probabilité :

$$P(w_i|w_1...w_{i-1}; h_1...h_m) \tag{2.6}$$

Où $w_1...w_{i-1}$ est estimé par le modèle de langage, et $h_1...h_m$ par la transcription approchée.

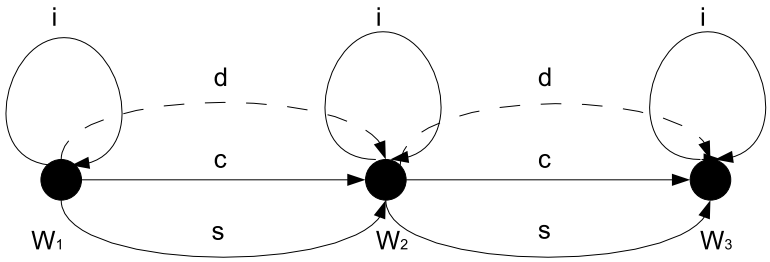

FIG. 2.3: *Représentation de la transcription sous forme de modèle de Markov caché*

Cette méthode permet d'intégrer, directement au sein du système de reconnaissance automatique de la parole, le canal supplémentaire représenté par les transcriptions *a priori* disponibles. Ils démontrent ainsi que des sous-titres peuvent être exploités pour aider efficacement les modèles acoustiques et linguistiques. Cependant, cette approche nécessite de connaître les frontières des segments ainsi que leurs correspondances avec le signal audio.

L'exploitation de sous-titres est également abordée par [Witbrock et Hauptmann, 1998] où les auteurs exploitent ceux fournis par la télévision dans le but d'estimer des modèles acoustiques. Avec cette approche, les sous-titres sont capturés et repérés par rapport au signal audio. Leur processus est récursif (figure 2.4) : le SRAP émet des transcriptions, ces dernières sont alignées via un algorithme DTW, un modèle acoustique est appris sur les données alignées et le processus se réitère. Les auteurs montrent que, en dépit d'un taux d'erreur mot dans les sous-titres supérieur à 17%, l'apprentissage des modèles acoustiques n'est pas dégradé.

Une méthode similaire est proposée dans [Jang *et al.*, 1999]. Les auteurs alignent la sortie du SRAP avec les sous-titres correspondants. Seuls les segments où un grand

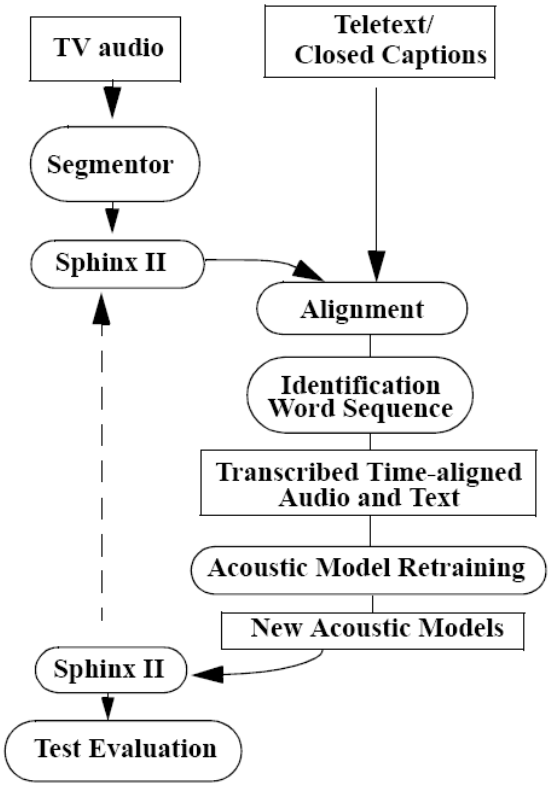

FIG. 2.4: *Exploitation de sous-titres pour apprendre des modèles acoustiques [Witbrock et Hauptmann, 1998]*

nombre de mots correspond sont sélectionnés, puis utilisés pour l'adaptation acoustique.

[Son *et al.*, 2000] proposent d'exploiter les sous-titres d'émissions TV afin de segmenter le flux audio-vidéo à partir de requêtes. L'utilisateur propose des mots clefs qui sont alors recherchés dans les sous-titres. Les segments correspondants sont extraits, puis à partir de ces derniers, un réseau de reconnaissance de mots est construit pour identifier les séquences audio dans l'ensemble du flux : ce type de réseau s'inspire de la reconnaissance de mots isolés. Ainsi, à partir des sous-titres, les zones audio sont identifiées et la vidéo peut être segmentée (figure 2.5). Dans le cadre de l'indexation automatique de documents audio, [Moissinac *et al.*, 2004] proposent une architecture permettant d'extraire des termes issus conférences. Le système de reconnaissance automatique est adapté avec toutes les données relatives à la conférence cible : textes relatifs au sujet, prompts utilisés pour la conférence. Ces données sont utilisées pour adapter les modèles acoustiques et linguistiques. Les auteurs montrent que les améliorations liées aux données ajoutées sont suffisantes dans le cadre d'un système d'indexation.

[Lamel *et al.*, 2002], proposent d'utiliser de grandes quantités de données issues de sous-titres pour adapter des données acoustiques avec une méthode légèrement supervisée : un modèle de langage est appris sur les données *a priori*. Un premier décodage

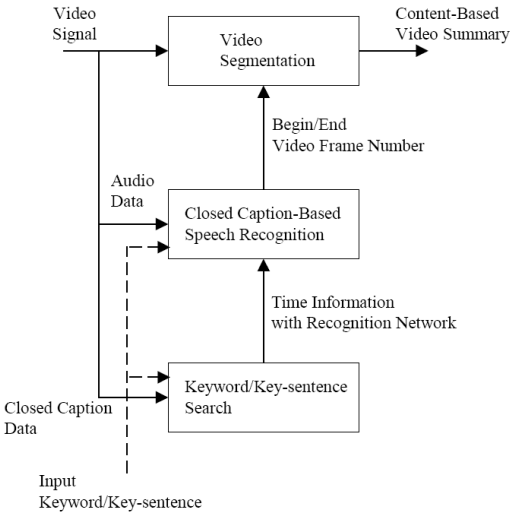

FIG. 2.5: *Méthode de [Son et al., 2000] pour segmenter des vidéos à partir des sous-titres*

est effectué, et les sorties du système sont alignées avec les données *a priori*. Les segments alignés sont utilisés pour une adaptation acoustique. Le processus est réitéré. Les auteurs démontrent que des données telles que les sous-titres peuvent être facilement exploités pour entraîner des modèles acoustiques.

[Chen *et al.*, 2004b] utilisent des sous-titres afin d'estimer plus finement des modèles acoustiques. Afin de sélectionner les parties similaires au signal audio et aux sous-titres, ils utilisent un réseau de confusion. Le SRAP décode le signal et génère un réseau pour chaque segment. Les sous-titres sont alors alignés sur le réseau de confusion pour extraire les parties du signal correspondantes : un seuil est introduit pour supprimer les hypothèses trop éloignées/improbables. Cette méthode pallie les erreurs de reconnaissance en s'appuyant sur l'ensemble des hypothèses. De plus, les auteurs obtiennent de plus grandes quantités de données d'apprentissage que dans [Lamel *et al.*, 2002].

Plus récemment, [Caillet *et al.*, 2007] proposent d'exploiter les textes issus de pièces de théâtre, afin de les aligner avec leurs enregistrements. Les transcriptions sont phonétisées et représentées sous forme de FSTs, présentant les variantes de prononciations, puis alignées via un décodage acoustico-phonétique.

2.3.5 Correction de transcriptions manuelles

Une approche intéressante de l'utilisation des SRAP est abordée par [Hazen, 2006] où les auteurs proposent de corriger des transcriptions manuelles de fichiers audio. Dans leurs expériences, les transcriptions récupérées présentent un taux d'erreur mot d'environ 10%. Leur méthode se déroule en plusieurs étapes. La première consiste à décoder l'ensemble du fichier audio. La transcription obtenue est alignée avec la transcription manuelle *a priori* exacte, via des FSTs. Le nouveau FST issu de l'alignement

concatène les transcriptions manuelle et automatique : la règle étant de conserver les parties transcrites manuellement et d'insérer la transcription automatique quand la transcription manuelle présente un vide. Le système automatique produit des transcriptions à 24% de WER. Après alignement, la transcription manuelle corrigée présente 8.8% de WER, soit un gain relatif de 12%. Leurs travaux montrent que, malgré un système automatique présentant un taux d'erreur relativement élevé, ce dernier est capable d'apporter des corrections à une transcription *a priori* presque exacte.

2.3.6 Alignement de longs segments imparfaits

[Cardinal *et al.*, 2005] proposent une méthode permettant d'aligner des transcriptions imparfaites, sans en connaître les frontières ni les correspondances avec le signal. La méthode proposée se base sur les automates à état fini (*Finite State Transducer*) FSTs [Mohri, 2002]. La stratégie employée se décompose en quatre modules (figure 2.6).

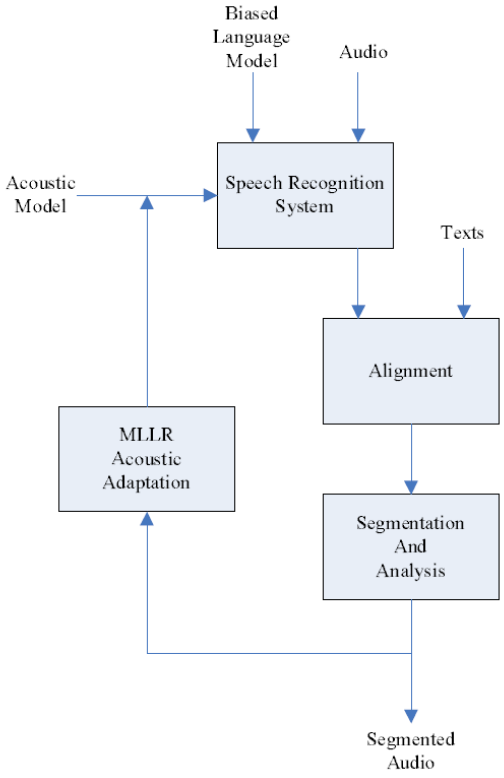

FIG. 2.6: *Schéma de l'approche pour aligner de longs segments audio avec des transcriptions imparfaites [Cardinal et al., 2005]*

1. Le système de reconnaissance automatique de la parole qui produit des transcriptions annotées avec les frontières de mots.
2. L'alignement qui reprend la transcription du SRAP et l'aligne avec les segments de texte disponibles.

3. Le module de segmentation et d'analyse prend la décision du rejet ou de l'acceptation du texte aligné.
4. Les zones correctement alignées sont injectées dans un module d'adaptation acoustique pour une passe ultérieure ainsi qu'une adaptation dynamique.

La particularité de leurs travaux réside essentiellement dans le module d'alignement qui s'appuie sur des FSTs. Les deux chaînes de caractères s_1 et s_2 appartenant respectivement au SRAP et aux segments de texte sont représentés sous forme de transducteurs T_{s1} et T_{s2} (figure 2.7).

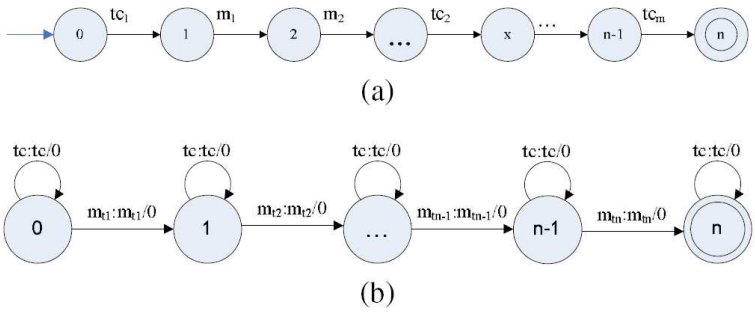

FIG. 2.7: *FSTs représentant les deux chaînes de caractères : la sortie du SRAP (a) et les segments de texte à aligner (b) [Cardinal et al., 2005]*

À partir de ces deux transducteurs, l'ensemble de tous les alignements possibles est calculé par composition des FST :

$$T_{aligns} = T_{outputASR} \oplus T_{edit} \oplus T_{ClosedCaption} \tag{2.7}$$

Le meilleur alignement est cherché dans le FST résultant de la composition, en utilisant un algorithme de recherche du meilleur chemin, dans ce cas un algorithme de Diskjstra [Cormen *et al.*, 2001] :

$$MeilleurAlignement = BPS(T_{aligns}) \tag{2.8}$$

Une fois le meilleur alignement trouvé, le module de rejet examine ce dernier et l'accepte (ou non) en fonction de seuils définis empiriquement, ainsi que de l'ordre chronologique d'apparition des segments.

Une approche plus conventionnelle est abordée par [Chih-wei, 2003], où l'auteur synchronise des sous-titres avec le signal audio en utilisant un algorithme DTW dont les fonctions d'insertion et de distance ont été modifiées afin de prendre en compte les distorsions entre le sous-titre et la transcription de l'audio :

$$ins(i,j) = a + \frac{1-a}{e^{b(t_{j+1}-t_j)}} \tag{2.9}$$

Où $0 \leq a \leq 1$ t_{j+1} et t_j correspondent aux temps de début et de fin du sous-titre, b est une constante calculée empiriquement. L'auteur met également en avant l'importance de prendre en compte la différence de la distribution des mots entre les sous-titres et la sortie du SRAP. Il propose une fonction de distance prenant en compte cette distorsion :

$$D'(i,j) = D(i,j) + c|log(P(dCS)d)| \tag{2.10}$$

Où c et d sont calculés empiriquement sur des données d'apprentissage. $P(dCS)$ est la probabilité du i^{me} mot du sous-titre d'être aligné avec le j^{me} mot du SRAP en étant éloignés d'un temps dCS. Cette approche permet d'aligner correctement l'ensemble des sous-titre à condition d'en connaître leurs temps d'apparition.

2.3.7 Alignement de segments très imparfaits

L'exploitation de transcriptions *a priori* très imparfaites est abordée par [Haubold *et al.*, 2007] qui exploitent des transcriptions automatiques devant être alignées sur des données audio provenant de locuteurs différents, d'environnements divers et de qualités variables. Les auteurs travaillent ainsi avec deux canaux fortement bruités à aligner. Cette opération est réalisée en travaillant au niveau phonétique aussi bien sur le signal audio que sur la transcription *a priori*. La transcription *a priori* est phonétisée via un phonétiseur classique. Quant à l'audio, les segments de phonèmes sont extraits par une opération de DAP. Une fois les deux chaînes phonétiques obtenues, elles sont alignées par un algorithme dynamique Smith-Waterman tel que celui utilisé pour aligner des chaînes d'ADN. Cette approche permet d'aligner des transcriptions très imparfaites.

2.4 Points d'ancrage et segmentation

Dans le cadre de l'exploitation de transcriptions imparfaites avec l'utilisation d'un SRAP, l'un des principaux problèmes est de trouver les points de synchronisation entre l'hypothèse du SRAP et la transcription. Si les transcriptions *a priori* sont volumineuses, il n'est pas envisageable d'appliquer directement des algorithmes dynamiques pour les aligner avec un SRAP, en raison de leur complexité. C'est pourquoi, de nombreuses méthodes ont été développées afin de s'adapter aux larges quantités de données. Ces méthodes sont souvent issues du domaine de la recherche d'information où un document doit être identifié/retrouvé à partir d'une requête. Nous présentons les principales approches issues de la recherche d'information permettant de segmenter et trouver des points d'ancrage rapidement.

Actuellement, l'un des modèles les plus utilisés dans la recherche d'information est basé sur la représentation dans un espace vectoriel des documents (*Vector Space Model* : VSM) [Salton, 1989]. Chaque document est considéré comme un vecteur de taille

N dont les éléments sont composés de poids assignés à chaque mot w du vocabulaire de taille N. Le poids associé à chaque mot représente son importance au sein du document/vecteur.

La qualité des poids assignés aux mots est primordiale : plusieurs méthodes existent pour pondérer les mots d'un corpus. L'une des plus simple est la fréquence du mot, correspondant au nombre d'occurrences du mot dans le corpus (*term frequency* : tf), associée à la fréquence inverse du mot (*inverse document frequency* : idf) qui permet de calculer l'importance du mot dans le document. Cette mesure calcule le logarithme de l'inverse de la proportion de documents du corpus qui contiennent le terme : les mots qui apparaissent rarement sont considérés comme plus importants. Une méthode plus sophistiquée est la mesure Okapi [Robertson et Jones, 1994] qui est dérivée de $tfxidf$.

Connaissant ces vecteurs, il est nécessaire d'estimer la distance qui les sépare. Cette estimation s'effectue soit sur un espace de dimension de la taille du plus grand vecteur, soit dans un espace plus petit dans lequel les vecteurs sont projetés. Les mesures de similarité les plus courantes sont les mesures cosinus, l'indice Dice et l'indice Jaccard, qui se formalisent ainsi :

– cosinus :

$$S(d_i, d_j) = \frac{\sum_{k=1}^{N}(w_{ki}.w_{kj})}{\sqrt{\sum_{k=1}^{N} w_{ki}^2 . \sum_{k=1}^{N} w_{kj}^2}} \tag{2.11}$$

– L'indice de Dice :

$$S(d_i, d_j) = \frac{2\sum_{k=1}^{N}(w_{ki}.w_{kj})}{\sum_{k=1}^{N} w_{ki}^2 + \sum_{k=1}^{N} w_{kj}^2} \tag{2.12}$$

– L'indice de Jaccard :

$$S(d_i, d_j) = \frac{\sum_{k=1}^{N}(w_{ki}.w_{kj})}{\sum_{k=1}^{N} w_{ki}^2 + \sum_{k=1}^{N} w_{kj}^2 - \sum_{k=1}^{N}(w_{ki}.w_{kj})} \tag{2.13}$$

Dans ces formulations, d_i et d_j désignent deux documents qui sont comparés, w_{ki} et w_{kj} sont les vecteurs qui représentent les documents d_i et d_j : ils sont composés des coefficients $tf.idf$ de chaque terme du vocabulaire de taille N.

L'approche intuitive commune à ces mesures est le calcul du degré de coocurrence entre les paramètres (mots) des vecteurs. Les résultats obtenus sont presque similaires à quelques différences près : l'indice de Jaccard pénalise, plus que celui de Dice, une faible quantité de mots communs entre deux vecteurs. La mesure cosinus est identique à l'indice de Dice si les vecteurs sont de tailles comparables, mais s'avère moins pénalisante si les nombres d'entrées nulles dans les vecteurs diffèrent beaucoup.

2.4.1 Recherche d'information basée sur des *clusters*

[Anni R. Coden, 2002] présentent une méthode de recherche d'information basée sur les méthodes de *clustering*. Leur approche consiste à découper le texte en *clusters* avec une mesure de confiance assignée à chacun. La méthode se décompose en plusieurs éléments, issus des différents travaux dans le domaine :

1. La recherche des intersections entre les morceaux de transcriptions et la requête via les occurrences de mots [Robertson et Jones, 1994] :

$$CW(w_i|D_j) = \frac{(K+1)CFW(w_i)tf(w_i,D_j)}{K((1-b)+b(NDL(D_j))+tf(w_i,Dj)} \tag{2.14}$$

   Où $CFW(w_i) = log\left(\frac{N}{n(w_i)}\right)$ est la collection de poids, N est le nombre total de documents et $n(w_i)$ le nombre de documents contenant le terme w_i. $tf(w_i,D_j)$ est la fréquence des termes w_i dans le document D_j et $NDL(D_j)$ est la taille du document D_j normalisée par la moyenne des tailles des différents documents. b et K sont des constantes définies empiriquement correspondant respectivement à l'influence de la longueur des documents et à l'influence du poids des fréquences de mots.
   À partir de ces considérations, le poids d'un document peut être estimé par :

$$DW(D_i) = \sum_{w_k \in X_i} CW(w_k) \tag{2.15}$$

2. L'estimation de l'importance d'un segment au niveau qualitatif, par sa quantité d'information, définie ainsi :

$$INTER(D_i,D_j) = \frac{DW(D_i \cap D_j)}{DW(D_i)} \tag{2.16}$$

3. Déterminer la proximité des informations basées sur le recouvrement entre les mots, s'appuyant sur une mesure de type $\mathcal{X}^2$: cette approche rejette ou accepte l'hypothèse que deux documents soient similaires :

$$\mathcal{X}^2 = \sum_{w_k \in D_i \cup D_j} \frac{\left((CW(w_k|w_k \in D_i) - CW(w_k|w_k \in D_j)\right)^2}{CW(w_k|w_k \in D_j)} \tag{2.17}$$

   À partir d'une table des valeurs de $\mathcal{X}^2$, peut être définie une valeur δ qui accepte ou rejette l'hypothèse.
4. L'attribution une mesure de similarité à chaque *cluster*, calculée à partir des éléments précédents :

$$sim(D_i,D_j) = \delta.INTER(D_i,D_j) \tag{2.18}$$

Avec ces éléments, les auteurs présentent un algorithme qui permet de segmenter le texte, en se servant d'un seuil de similarité défini empiriquement pour sélectionner les *clusters* de transcriptions qui correspondent à la requête :

1. Soit $k = 1$ l'index du *cluster* courant et i l'index du document courant.
2. Soit le document D_i qui est le centroïde C_k^* du *cluster* k
3. Déterminer tous les documents similaires à C_k^* en s'appuyant sur la mesure 2.18
4. À partir de tous les documents D_j trouvés précédemment, définir celui qui minimise le ratio :

$$D_{i^0} = \min \left\{ \frac{sim(D_j, C_k^*)}{sim(C_k^*, D_j)} \right\} \tag{2.19}$$

   Où i^0 est l'index du document à trouver.
5. Si $i \neq i^0$ aller à l'étape 2 en assignant i^0 à i.
6. Sinon le centroïde a été trouvé, et tous les documents trouvés en (3) sont assignés au *cluster* k, k est incrémenté.
7. Trouver le premier document qui n'appartient à aucun des *clusters* et mettre son index à i. Reprendre à l'étape (2)

Cette mesure définie par [Anni R. Coden, 2002] permet d'effectuer efficacement des requêtes sur des transcriptions imparfaites : elle s'adapte bien aux problèmes de recherche d'information multimédias.

[Witbrock et Hauptmann, 1997] proposent une approche très différente. Partant du constat que les systèmes de reconnaissance génèrent un certain nombre d'erreurs, ils estiment que leurs transcriptions ne peuvent être utilisées directement. Pour cette raison, ils convertissent la transcription en suite phonétique. La recherche se fait alors via un index inversé entre mots et suites phonétiques avec les algorithmes classiques de la recherche d'information [James, 1995]. Cette méthode présente plusieurs avantages, notamment la prise en considération des mots hors vocabulaire et la rapidité d'exécution, puisqu'aucune recherche n'est effectuée sur le treillis du SRAP.

2.5 Adaptation des systèmes de SRAP via des transcriptions *a priori*

En raison du coût élevé d'obtention de transcriptions manuelles de signal audio, de plus en plus de travaux se sont tournés vers l'utilisation de transcriptions existantes et imparfaites pour l'apprentissage ou l'adaptation des systèmes.

La pertinence d'exploiter des transcriptions inexactes est illustrée par [Kemp Thomas, 1998] qui montre que les modèles acoustiques d'un SRAP peuvent être entraînés et améliorés uniquement en décodant des données non-transcrites. Cette approche est d'ailleurs utilisée, à une moindre échelle dans la plupart des SRAP

effectuant plusieurs passes : leurs modèles acoustiques sont adaptés sur leurs propres transcriptions issues d'une première passe de décodage.

Ainsi, [Lamel *et al.*, 2001, Lamel *et al.*, 2002, Chen *et al.*, 2004b] proposent d'adapter des modèles acoustiques à partir de grandes quantités de données imparfaites, telles que des sous-titres ou des prompts. Leurs expérimentations montrent que l'utilisation de données parfaitement transcrites n'est pas nécessaire, si l'on adapte les stratégies d'apprentissage au fait que les transcriptions sont imparfaites. [Sundaram *et al.*, 2004] montre que dans le cas des modèles Gaussiens, beaucoup de données inexactes sont nécessaires pour les corrompre. [Lamel *et al.*, 2002] propose un modèle de langage appris sur les données, et ces dernières sont décodées pour ensuite adapter les modèles acoustiques.

La méthode présentée par [Cardinal *et al.*, 2005] et décrite précédemment permet, via un alignement, de construire une base d'apprentissage à partir de données imparfaitement transcrites. Cette méthode a pour particularité d'exploiter des données qui ne comportent pas d'informations temporelles.

Cette approche est également suivie par [Chan et Woodland, 2004], qui estiment un modèle de langage à partir de sous-titres, sur lesquels la transcription du SRAP est alignée avec un algorithme dynamique. Leur article compare quelques stratégies d'adaptation : l'une sur des segments alignés et filtrés, l'autre sur des données non filtrées. Leurs expérimentations montrent dans les deux cas une nette amélioration de la qualité des modèles. Par ailleurs, ils observent que MMIE est plus sensible à la qualité des données que ne l'est MPE. En fonction de l'apprentissage, il est donc nécessaire d'adapter la stratégie de filtrage des données.

L'approche de [Chen *et al.*, 2004b] est plus intéressante car elle considère l'ensemble de l'espace de recherche pour sélectionner les segments qui seront exploités pour l'adaptation acoustique : les sous-titres sont alignés sur les réseaux de confusion issus du SRAP. Les parties correctement alignées sont utilisées pour l'adaptation acoustique. Du fait de l'alignement sur le réseau de confusion, la quantité de données alignées et non corrompues est beaucoup plus grande que celles des méthodes précédemment citées.

[Barras *et al.*, 2004] proposent d'exploiter des transcriptions *a priori* pour l'aide à la transcription manuelle. Cette étude est faite dans le cadre de l'aide à la transcription de discussions parlementaires. Un SRAP décode le signal audio, fournit une transcription et l'aligne avec la transcription manuelle *a priori*. Les zones parfaitement alignées sont considérées comme exactes, les auteurs y trouvent environ 1% de WER. Le reste, non-aligné de la transcription (environ 25%) est transcrit manuellement, avec l'assistance du SRAP.

Une autre application de l'utilisation de transcriptions approchées est abordée dans [Paulik *et al.*, 2005] où des transcriptions aident à la traduction (figure 2.8). Dans ce cas, un locuteur parle en Espagnol et un SRAP en fournit une transcription. Par ailleurs, un traducteur traduit l'espagnol en anglais en parlant dans un autre SRAP. Les données du SRAP sur l'espagnol sont traduites par une machine de traduction automatique dont la

sortie est injectée dans le SRAP anglais. Cette injection est faite par le biais du modèle de langage qui est restreint par une interpolation et l'utilisation d'un modèle cache. Par ailleurs, les n meilleures traductions sont utilisées pour biaiser le modèle de langage. Elles sont également combinées avec les *n-best* du SRAP pour sélectionner la meilleure suite de mots. Les informations étant complémentaires, le résultat de la transcription est très largement amélioré. Une autre de leur approche consiste à exploiter des transcriptions *a priori* du locuteur espagnol qui sont traduites pour être injectées dans le SRAP. Au final, les auteurs arrivent à baisser le taux d'erreur mot d'environ 36% grâce à la complémentarité des informations apportées par les traducteurs automatiques.

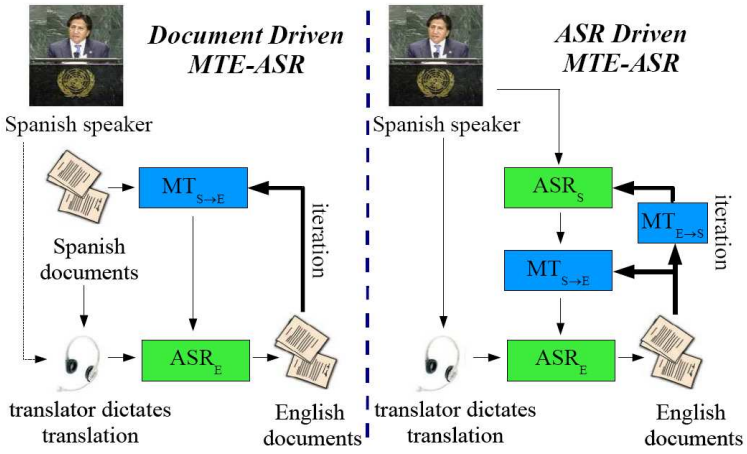

FIG. 2.8: *Système d'aide à la traduction présenté dans [Paulik et al., 2005]*

2.6 Synthèse

Nous avons présenté un panorama des techniques permettant d'exploiter des transcriptions *a priori*. Les approches les plus courantes sont une utilisation légèrement supervisée destinée à estimer/améliorer des modèles acoustiques. [Paulik *et al.*, 2005], [Placeway et Lafferty, 1996] abordent l'exploitation directe de la transcription *a priori* pour spécialiser le système de reconnaissance. Ces méthodes, bien qu'améliorant le décodage, ne permettent pas d'augmenter la qualité de la transcription *a priori* initiale. Cette analyse bibliographique montre aussi que l'exploitation de transcriptions *a priori* introduit une contrainte supplémentaire si ces dernières sont volumineuses et particulièrement inexactes. Dans ce cas, il est nécessaire de trouver des points d'ancrage pour sélectionner les bons segments.

Les autres techniques présentées dans la littérature exploitent indirectement les transcriptions disponibles, en adaptant les modèles du SRAP. De plus, elles ne s'adaptent pas à des corpus très inexacts. Dans le chapitre suivant, nous présenterons deux approches originales permettant d'intégrer une transcription *a priori* au sein du décodage d'un SRAP et de retrouver à la volée des segments noyés dans de grands corpus.

Chapitre 3

Décodage guidé par des transcriptions

Sommaire

Ce chapitre présente les principes d'un système de reconnaissance guidé par des transcriptions *a priori*. Nous abordons d'abord l'intégration de la transcription *a priori* au sein du décodeur du LIA, SPEERAL. Nous présentons ensuite les deux approches exploitant la transcription *a priori* : le mélange de modèles de langages puis le décodage guidé. Finalement, nous présentons l'ensemble des expériences menées permettant de tester nos approches.

Dans ce chapitre nous proposons une solution originale exploitant des transcriptions *a priori* au sein d'un SRAP, avec le double objectif d'améliorer la transcription imparfaite et d'en extraire du corpus de qualité correcte.

Plutôt qu'une intégration en amont ou en aval du SRAP, nous intégrons les transcriptions *a priori* comme un canal d'information supplémentaire au cœur du décodage.

Nous présentons d'abord l'intégration de la transcription dans le processus de décodage, puis nous évaluons cette méthode pour la correction de prompts d'émissions radiophoniques.

3.1 Intégration d'un canal supplémentaire au sein d'un algorithme A^*

La première partie de nos recherches s'appuie sur l'utilisation de transcriptions afin d'améliorer la qualité d'un SRAP (figure 3.1). Le comportement visé, est un alignement sur la transcription *a priori* lorsqu'elle est correcte, et de réaliser un décodage "libre" lorsqu'elle s'éloigne de l'énoncé. Nous montrons comment un SRAP peut tirer bénéfice de transcriptions parfaites ou approximatives.

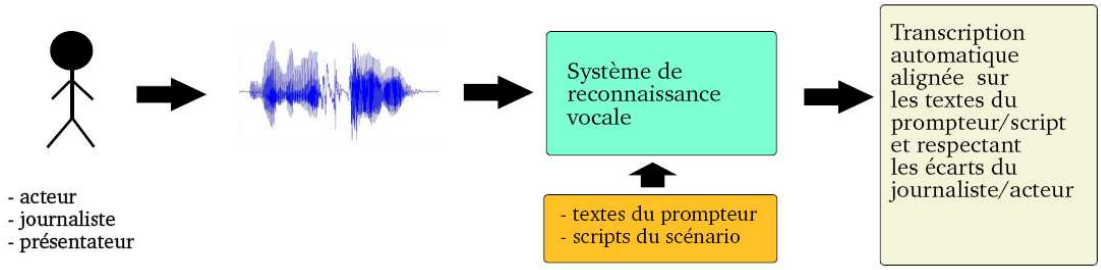

FIG. 3.1: *Principe général d'un SRAP guidé par des transcriptions approchées*

Dans ce chapitre, nous proposons une solution basée sur un décodeur à pile asynchrone, s'appuyant sur un algorithme A^* : SPEERAL [Nocera *et al.*, 2002a] et permettant d'intégrer directement au sein du décodage l'information issue des transcriptions approchées. Nous présentons les particularités de ce type de décodeur, puis les méthodes utilisées pour intégrer l'information. Enfin nous présentons le cadre expérimental permettant de tester et valider notre approche.

3.2 Le système de reconnaissance automatique de la parole SPEERAL

Nos recherches se focalisant sur l'intégration de transcriptions au sein d'un système de reconnaissance basé sur un décodeur à pile, nous présentons le système SPEERAL

[Nocera *et al.*, 2002b, Nocera *et al.*, 2002b, Nocera *et al.*, 2004], développé au LIA qui a servi de base à nos travaux.

Il s'agit d'un SRAP grand vocabulaire pour la parole continue, s'appuyant sur un algorithme de recherche A^* qui opère sur un treillis de phonèmes. Une fonction d'estimation somme pour chaque nœud n du graphe les coûts du chemin exploré et une sonde évaluant le coût du nœud n à la fin du graphe :

$$F(h_n) = g(h_n) + p(h_n) \tag{3.1}$$

Où $g(h_n)$ est la probabilité de l'hypothèse partielle arrivant au nœud n et $p(h_n)$ est la sonde estimant la probabilité du nœud n à la fin du graphe.

La sonde doit être de la meilleure qualité possible car les performances de l'algorithme de recherche dépendent d'elle. Dans SPEERAL, elle est composée de deux entités : une partie acoustique et un terme d'anticipation linguistique. Le calcul des scores acoustiques se fait grâce à un algorithme de Viterbi arrière (allant de la fin du graphe à chaque nœud intermédiaire) sur les modèles non-contextuels puis par une réévaluation du score obtenu par des modèles pseudo-contextuels [Linarès *et al.*, 2005b, Linarès *et al.*, 2005a]. L'anticipation linguistique, quant à elle, est basée sur une estimation des meilleurs tri-grammes prolongeant l'hypothèse explorée [Massonié *et al.*, 2005, Linarès *et al.*, 2007].

L'exploration du graphe est basée sur la fonction d'estimation $F(.)$. La pile des hypothèses est donc ordonnée en fonction de $F(.)$: les meilleurs chemins sont explorés en priorité. Par ailleurs, cette exploration en profondeur affine l'évaluation de l'hypothèse courante. Les chemins de faible probabilité sont coupés, provocant une recherche en arrière. Ceci induit un décodage qui est désynchronisé du flux audio.

Afin de pouvoir prendre en compte l'information résultant des transcriptions imparfaites, la fonction $F(.)$ est modifiée pour influer sur le score linguistique de l'hypothèse courante. Ce mécanisme permet de guider la recherche en réévaluant dynamiquement $g(h_n)$ en fonction des scores d'alignement. L'algorithme s'appuie sur deux étapes : la synchronisation de la transcription et son intégration au sein de la fonction d'évaluation de l'algorithme A^*.

3.3 Méthodes proposées

Cette section présente deux méthodes exploitant une transcription approchée. La première consiste à combiner un modèle de langage générique avec un modèle de langage estimé sur la transcription approchée. La seconde méthode s'appuie sur l'intégration d'un algorithme d'alignement dynamique sur la sonde de l'algorithme A^*. Nous montrons comment la fonction d'estimation de l'algorithme d'exploration est influencée par l'information issue de la transcription.

3.3.1 Méthode préliminaire : modèles de langage biaisés

Dans la situation où des transcriptions *a priori* de l'audio sont disponibles, une approche est de réduire l'espace de recherche, ou du moins le contraindre. La variabilité linguistique est directement dépendante du contexte. L'espace linguistique de la transcription *a priori* peut être modélisé, via un modèle de langage. Mais ce modèle de langage ne peut être utilisé seul à moins que la transcription soit parfaitement fidèle au signal de parole. Pour cette raison, nous proposons une interpolation de modèles entre un modèle générique apportant l'information indépendante du contexte et un modèle appris directement sur la transcription :

$$P'(w|h) = \sum_{i=1}^{k} \alpha_i P_i(w|h) \tag{3.2}$$

avec $0 < \alpha_i \leq 1$ et $\sum_i \alpha_i = 1$

où α_i représente le poids de chaque modèle.

Le décodeur SPEERAL interpole linéairement plusieurs modèles de langages. Dans nos différentes expériences, nous avons utilisé des modèles spécifiques estimés sur les transcriptions exactes ou approchées. L'information linguistique est extraite puis interpolée avec le modèle de langage générique du SRAP : les n-grammes du modèle générique sont ainsi biaisés par le contenu des transcriptions *a priori*.

3.3.2 *Driven Decoding Algorithm* (DDA) : Principe du décodage guidé

La seconde méthode que nous proposons consiste à modifier la fonction d'estimation d'un SRAP basé sur un algorithme de recherche A^*. Le principe est de synchroniser l'hypothèse courante avec la transcription *a priori*. Un score de correspondance est calculé, puis utilisé pour réévaluer les probabilités linguistiques (figure 3.2).

L'algorithme que nous proposons s'articuler en trois étapes :

1. La synchronisation entre la transcription et l'hypothèse courante.
2. Après synchronisation, l'évaluation de la correspondance entre l'hypothèse courante et la transcription
3. L'intégration de l'information issue de l'alignement synchrone à la fonction d'évaluation $F(n)$, en modifiant les probabilités linguistiques.

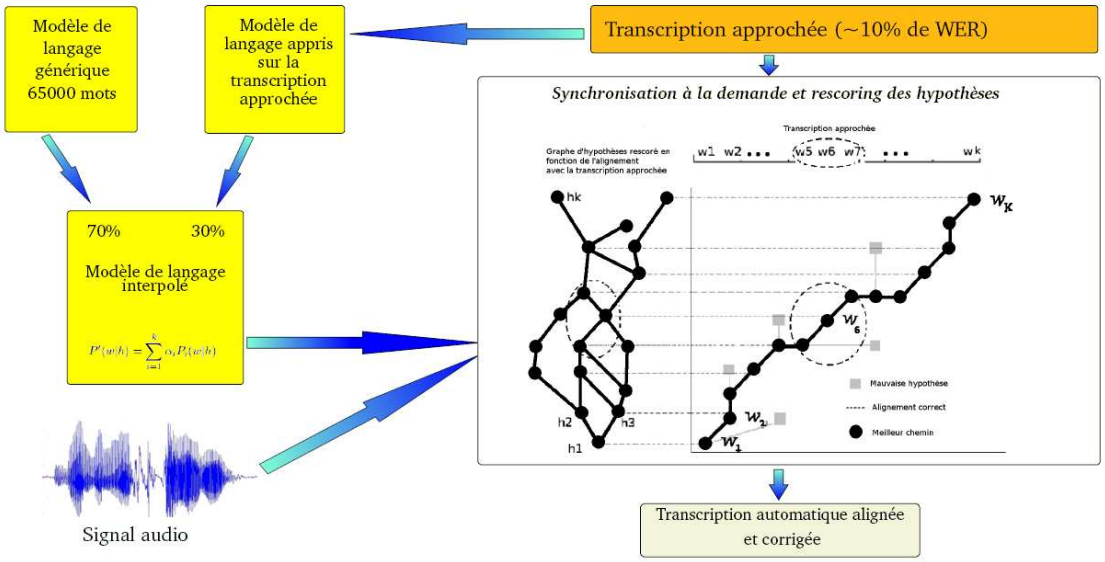

FIG. 3.2: *Principe général d'un SRAP guidé par des transcriptions approchées*

3.4 Anatomie de DDA

3.4.1 Synchronisation du flux audio et de la transcription imparfaite

Le moteur de reconnaissance construit des hypothèses au fur et à mesure qu'il avance dans le treillis de phonèmes. Les meilleures hypothèses à un instant t sont prolongées en fonction de la probabilité de l'hypothèse courante et des résultats de la sonde. Les premières modifications apportées au décodeur permettent d'aligner sur la transcription approchée chaque nouveau mot de l'hypothèse courante, ainsi que son historique. Ceci est réalisé par un algorithme d'alignement temporel (*Dynamic Time Warping*). Une hypothèse temporaire est construite à partir du mot courant et de son historique, tout au long du processus d'exploration du graphe. L'hypothèse synchronisée sur la transcription *a priori* constitue un point d'ancrage utilisé pour ré-estimer cette dernière.

Soit la matrice de dimension (nxm) dont chaque coefficient (i, j) est la distance euclidienne entre les points H_i et $T_j : d(h_i, t_j) = (h_i - t_j)^2$. L'alignement entre T et H est le chemin W suivant les éléments contigus de cette matrice tel que l'évolution soit restreinte aux éléments adjacents de la matrice. Les étapes successives telles que décrites par l'équation 3.3 sont réparties de manière monotone :

$$\gamma(i,j) = d(h_i, t_j) + min\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1) \tag{3.3}$$

Où $\gamma(i, j)$ correspond au chemin cumulé aux coordonnées (i, j) de la matrice. Le résultat est une mesure de la divergence entre le test et la référence, la distance d'édition

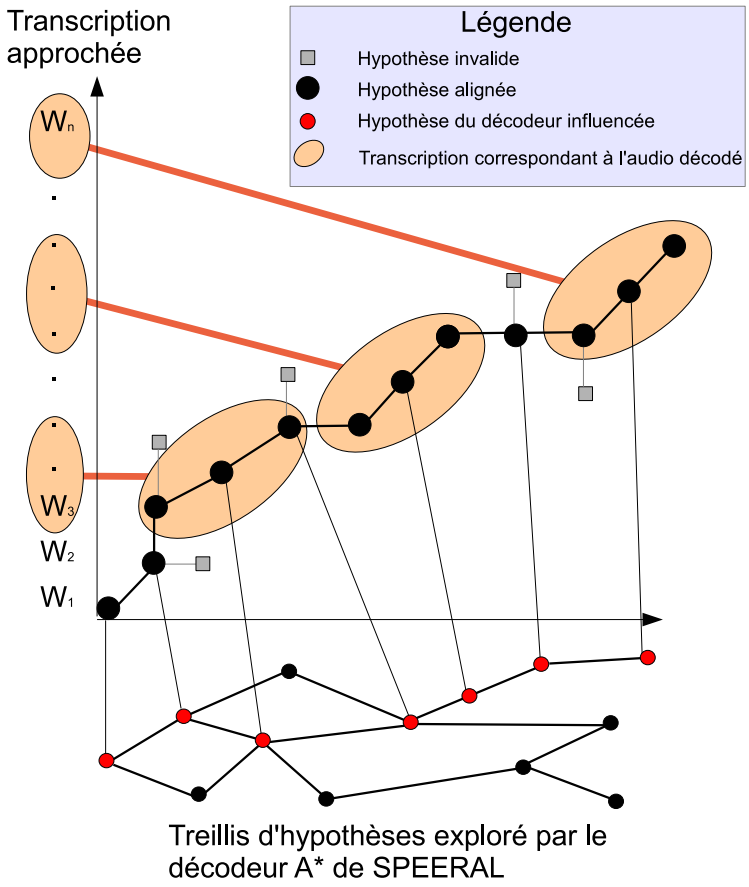

FIG. 3.3: *Synchronisation du faisceau de recherche avec la transcription imparfaite par algorithme DTW*

sert essentiellement à synchroniser les mots proposés avec la transcription.

Les hypothèses sont relativement courtes car les segments sont limités à 30 secondes. Dans les premières expériences, les segmentations sont connues *a priori* : la complexité des alignements via l'algorithme d'alignement dynamique est relativement faible. De plus, le coût supplémentaire de l'alignement est contrebalancé par l'augmentation de la vitesse de décodage sur les zones correctement transcrites.

La figure 3.3 présente l'évolution des hypothèses du décodeur à pile A^* influencées par un alignement sur une transcription approchée. L'utilisation d'une distance d'édition permet de favoriser l'hypothèse $H = h_1, h_2, .., h_i, .., h_n$ qui minimise sa distance par rapport à la transcription imparfaite $T = t_1, t_2, .., t_j, .., t_n$.

Par ailleurs, nous avons intégré un système de cache pour le calcul de la distance d'édition, qui s'adapte parfaitement à une exploration asynchrone : les parties communes aux hypothèses ne sont jamais recalculées dans la matrice, ce qui permet d'avoir un temps de calcul quasi nul. De plus, des coupures sur la matrice d'alignement limitent

la quantité de calculs nécessaires à l'alignement (figure 3.4).

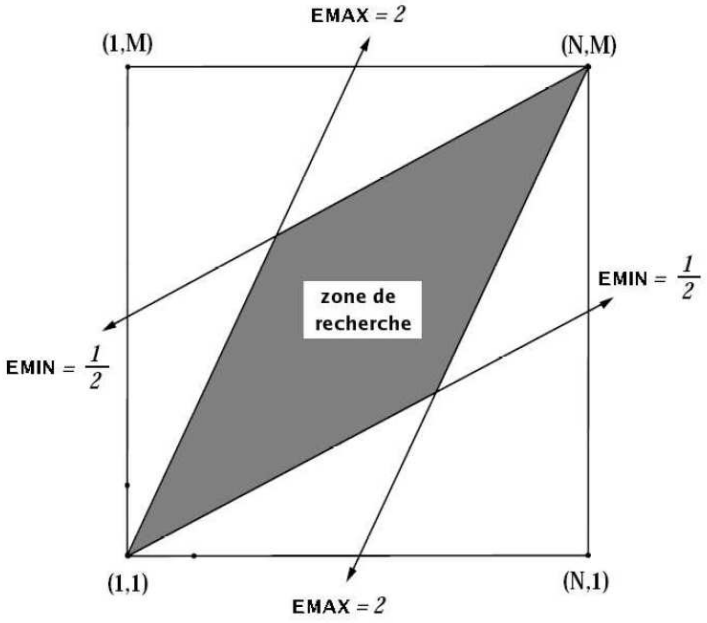

FIG. 3.4: *Limitation de l'espace de recherche au sein de l'alignement dynamique*

3.4.2 Score de correspondance et réévaluation linguistique

La fonction d'estimation calcule, pour chaque nœud du graphe, les coûts du chemin exploré ainsi qu'une sonde minimisant le coût des chemins finaux. La qualité de cette sonde influence directement les performances de l'algorithme de recherche. La solution proposée ré-hausse le score des mots présents dans le faisceau de recherche lorsqu'ils sont alignés avec la transcription approchée ; s'ils ne sont pas présents dans le faisceau, l'algorithme d'alignement n'intervient pas. Pour que notre alignement oriente le moteur de reconnaissance, il est nécessaire que le mot ait été évalué par l'algorithme A^* : le score de l'hypothèse courante sera alors modifié en conséquence. Nous ne modifions pas les scores d'anticipation linguistique.

Une fois l'hypothèse synchronisée avec la transcription, l'algorithme estime un score de synchronie locale noté α, calculé à partir du nombre de mots de l'historique correctement alignés à la transcription. Seules trois valeurs de synchronie sont utilisées, correspondant respectivement à un alignement complet du tri-gramme, l'alignement d'un bi-gramme ou du seul mot courant. Les seuils ont été définis empiriquement sur un corpus de test :

$$S = \begin{cases} 0.1 & \text{si } m_1 = ma_1 \text{ et } m_2 = ma_2 \\ 0.6 & \text{si } m_1 = ma_1 \text{ et } m_2 \neq ma_2 \\ 0.8 & \text{si } m_1 \neq ma_1 \text{ et } m_2 \neq ma_2 \\ 1 & \text{si } m \text{ non trouvé} \end{cases} \tag{3.4}$$

Lorsque le score de synchronie est connu, le score linguistique est réévalué à partir de la règle suivante :

$$L(w_i|w_{i-2}, w_{i-3}) = P(w_i|w_{i-2}, w_{i-3})^{\alpha} \tag{3.5}$$

Où $L(w_i|w_{i-2}, w_{i-3})$ est le score ré-estimé du tri-gramme courant $\{w_{i-3}, w_{i-2}, w_i\}$ et $P(w_i|w_{i-2}, w_{i-3})$ est la probabilité initiale de ce même tri-gramme.

3.5 Expérimentations

3.5.1 Cadre expérimental

L'ensemble des expériences a été effectué avec le système de "Broadcast news" développé au LIA [Nocera *et al.*, 2004] qui a été engagé dans la campagne d'évaluation ESTER [Galliano *et al.*, 2005].

Le corpus ESTER est composé d'émissions radiophoniques françaises issues du groupe de radio "Radio-France". Ce corpus est originellement destiné à évaluer les systèmes de transcription automatique de la parole. Nous avons utilisé ce corpus, car contrairement à la base de données RTBF, il contient des transcriptions d'excellente qualité. Les émissions présentes dans ESTER sont essentiellement composées de nouvelles, et plus rarement de débats ou discussions avec des journalistes par téléphone. Ceci induit la présence de locuteurs non-natifs, et de conditions acoustiques difficiles.

Le système de "broadcast news" du LIA repose sur le décodeur SPEERAL ainsi que sur le segmenteur automatique de la parole Alizé [Bonastre *et al.*, 2005]. Le modèle de langage générique est tri-gramme, estimé sur 200 millions de mots extraits du journal *Le Monde* ainsi que du corpus d'ESTER (1 million de mots). Le système utilisé lors d'ESTER fonctionne en deux passes : la première fournissant des transcriptions pour une adaptation MLLR des modèles acoustiques. La première passe s'effectue en 3x le temps réel et la seconde en 5x sur un ordinateur standard (Opteron 2Ghz). Dans les expériences qui suivent nous avons effectué une seule passe, qui consiste essentiellement à trouver des sous-séquences de mots dans le corpus.

Les modèles acoustiques sont contextuels et leur estimation se base sur des arbres de décision. Ils sont entraînés sur les données fournies lors de la campagne ESTER : environ 80 heures de parole annotée manuellement. Ils sont composés de 230000 Gaussiennes et 3600 états.

En appliquant l'équation 1.26 présentée dans le chapitre 1.6, nous estimons que, dans le cadre de nos expériences, les améliorations sont significatives dès lors qu'elles sont supérieures à 0.2%.

Expériences de référence et développement

Dans nos expériences de référence, nous avons cherché à identifier, aussi clairement que possible, les différents types d'erreurs incompressibles commises par le système sur lesquelles notre approche n'aura aucune influence. Notre objectif est d'être capable de mesurer l'effet réel des techniques proposées sur les performances du décodeur. Il y a quatre principales sources d'erreurs :

- **Les mots hors vocabulaire (MHV)** : ces mots n'existent pas dans le lexique du moteur de reconnaissance. Lors du décodage, ils ne peuvent pas être reconnus. Ceci entraîne des effets de bord : avec les modèles probabilistes tri-grammes, la non-reconnaissance d'un mot peut provoquer des erreurs sur les mots adjacents. De proche en proche, ces erreurs peuvent théoriquement se propager. Pratiquement, il est assez difficile des mesurer précisément l'impact des MHV sur les performances, sauf en les intégrant au lexique et en comparant les résultats obtenus. Nous avons analysé la transcription exacte pour en extraire les mots absents du lexique initial. Ils représentent 2% des mots. Ces mots ont ensuite été phonétisés et ajoutés au lexique. Le modèle de langage a été ré-estimé avec ce lexique enrichi.

- **Une segmentation inexacte** : le système "broadcast news" réalise automatiquement l'ensemble des traitements qui permettent de passer du flux audio brut à la transcription synchronisée. Une émission radiophonique comporte, en plus des segments de parole, des parties musicales, des événements acoustiques divers qui ne sont pas de la parole (bip, jingle, etc.). La première étape du processus global de décodage consiste à extraire les zones de parole du flux audio. Cette segmentation automatique n'est pas exacte et les segments de parole sont parfois mal isolés. Le moteur cherche alors à décoder une zone qui ne contient pas de parole ou inversement, ne décode pas des zones de parole qui ont été supprimées par erreur par le segmenteur. Quelque soit la stratégie de décodage utilisée en aval, les erreurs liées à la segmentation restent irrécupérables.

- **La linguistique** : la qualité du modèle de langage dépend de l'adéquation du corpus d'apprentissage et des conditions d'utilisation du système. Un moteur de reconnaissance susceptible de décoder des messages linguistiquement variés devra utiliser un modèle de langage codant cette variabilité, ce qui nécessite des corpus représentatifs généralement très volumineux. Bien entendu, en augmentant le champ des hypothèses linguistiquement acceptables, on augmente aussi les risques de confusion. Ici, le domaine linguistique peut être réduit puisqu'on dispose d'une transcription exacte ou approchée du discours. Le gain maximal qu'on peut obtenir en réduisant globalement l'espace linguistique peut être es-

timé en apprenant un modèle de langage sur la transcription exacte elle même. Nous avons donc cherché à évaluer les performances d'un système qui disposerait d'un modèle de langage tri-gramme parfaitement adapté aux données à traiter. Bien entendu, cette expérience ne permet pas d'imputer toutes les erreurs au seul modèle de langage, l'exploration du graphe d'hypothèses combinant simultanément les scores acoustiques et linguistiques (la bonne qualité linguistique d'une hypothèse peut compenser sa mauvaise qualité acoustique). De plus, ce taux très bas est obtenu en utilisant la transcription sans erreurs, ce qui ne correspond pas à un contexte d'utilisation réaliste.

– **L'acoustique et les heuristiques de décodage** : l'algorithme de recherche ne fait pas une exploration exhaustive du graphe d'hypothèses. La complexité d'un tel parcours serait bien trop importante pour des systèmes à grand vocabulaire, et un certain nombre d'heuristiques accélèrent l'exploration en écartant des hypothèses jugées très improbables. Dans SPEERAL, les critères permettant de réduire l'espace de recherche sont à la fois d'ordre acoustique et linguistique. Théoriquement, ces coupures doivent introduire très peu d'erreurs de décodage ; cependant, lorsque le contexte acoustique est très mauvais, les meilleures hypothèses (en terme de taux d'erreur) peuvent se trouver exclues du faisceau de recherche. Dans ce cas, une stratégie basée sur la "promotion" des hypothèses du faisceau coïncidant avec la transcription ne permet pas de récupérer ces erreurs. On peut quantifier de façon approximative la perte correspondant à cette situation en utilisant le moteur de reconnaissance (avec des seuils de coupure standards) pour faire un alignement forcé de la transcription exacte sur le signal (tableau 3.3).

Corpus utilisé et transcription approchée

Nos expérimentations se sont focalisées sur trois heures extraites du corpus de développement d'ESTER : France Inter (1)-1 heure, France Inter (2)-1 heure et France Info-1 heure. Les transcriptions imparfaites ont été simulées en ajoutant manuellement des erreurs dans les transcriptions exactes : nous avons pris soin de garder une forme journalistique correcte pour respecter le style classique d'une émission radiophonique. Nous simulons ainsi une transcription imparfaite proche de ce que serait le script d'une émission de ce type. Dans les deux premières heures, 10% de WER ont été introduits, et 20% de WER on été introduits dans la dernière. Ces erreurs comprennent aussi bien des changements au niveau de l'organisation des phrases, que des substitutions par des synonymes et parfois quelques suppressions ou insertions.

3.5.2 Interpolation avec modèle de langage 'exact'

Dans les sections qui suivent, le corpus utilisé est composé d'une émission d'une heure de France Inter (1), utilisée ici comme corpus de développement. Cette heure

nous permet de valider l'ensemble des méthodes proposées. Nous utilisons désormais une segmentation automatique, afin de projeter la stratégie dans des conditions d'utilisation réelles.

Les expériences proposées dans cette section permettent d'évaluer les erreurs incompressibles commises par le SRAP. Un modèle de langage a été appris sur la transcription exacte, puis combiné avec un modèle de langage générique (65000 mots appris sur Le Monde). Les mots hors vocabulaire ont été extraits de la transcription pour être phonétisés et ajoutés au modèle de langage. Les expériences préliminaires ont été réalisées sur la première heure : France Inter (1). Le tableau 3.1 présente les résultats d'interpolation d'un modèle de langage appris sur la transcription exacte avec le modèle de langage générique.

Protocole	Taux d'erreur
ML-G 100%	22.7%
ML-TrEx 100%	5.2%
ML-G 70% + ML-TrEx 30%	13.0%
ML-G 50% + ML-TrEx 50%	11.5%
ML-G 30% + ML-TrEx 70%	10.8%

TAB. *3.1:* *Résultats des expériences de référence avec interpolation du modèle de langage générique (ML-G) et du modèle appris sur la transcription exacte (ML-TrEx). Les pourcentages correspondent aux poids d'interpolation des modèles de langage*

Ces premières expériences montrent qu'en limitant le faisceau d'exploration de l'algorithme de recherche à la seule transcription exacte, le taux d'erreur mots atteint 5.2% contre 22.7% pour un modèle générique. Ces 5.2% d'erreur mots résiduels peuvent être liés à la perplexité du modèle de langage ou des problèmes de modélisation acoustique : l'hypothèse n'apparaît jamais dans le faisceau. Les expériences interpolant les deux modèles de langage montrent que le modèle générique génère beaucoup de bruit : +5.6% d'erreurs dans le meilleur des cas. Ces expériences nous indiquent qu'un modèle de langage de type n-gramme présente des limites intrinsèques à sa modélisation.

3.5.3 Interpolation avec modèle de langage 'approché'

Afin de mesurer l'impact des erreurs au sein des transcriptions, nous avons estimé un autre modèle de langage à partir des transcriptions approchées. Les expériences utilisant ce modèle de langage combiné au modèle de langage générique sont présentées dans le tableau 3.2.

L'ensemble de ces expériences montre qu'un modèle de langage estimé sur des transcriptions imparfaites a la capacité d'améliorer considérablement la qualité de la reconnaissance automatique. Cependant, les taux d'erreurs restent supérieurs à ceux de la transcription imparfaite *a priori*. Ceci signifie que sans autre source d'information, le système de reconnaissance converge vers ses limites imposées par ses erreurs initiales. Cette technique semble donc limitée pour exploiter pleinement des transcriptions im-

Protocole	Taux d'erreur
ML-TrErr seul	16.3%
ML-G 70% + ML-TrErr 30%	16.2%
ML-G 50% + ML-TrErr 50%	15.4%
ML-G 30% + ML-TrErr 70%	15.2%

TAB. 3.2: *Résultats des expériences d'interpolation de modèles de langage Générique (ML-G) et appris sur la transcription approchée (ML-TrErr). Les pourcentages correspondent aux poids d'interpolation des modèles de langage*

parfaites. D'ailleurs, les expériences réalisées en utilisant une transcription exacte pour modéliser le modèle de langage montrent que le modèle de langage générique génère trop de bruit par rapport au modèle exact : seule l'utilisation d'un modèle appris sur la transcription exacte, sans interpolation avec modèle générique, permet d'obtenir un taux d'erreur mot acceptable (5.2%). Mais n'utiliser que la transcription comme source du modèle de langage s'avèrera vite dangereux si cette dernière s'éloigne de ce qui est prononcé : le SRAP convergera alors vers les erreurs de la transcription approchée et n'aura à sa disposition aucune autre alternative. Il est donc nécessaire d'exploiter autrement cette information disponible pour qu'elle influence plus significativement le SRAP.

3.5.4 Expériences avec modèle de langage 'exact' et DDA

Après avoir expérimenté des modèles interpolés, nous avons évalué la méthode de décodage guidé par la transcription *a priori* exacte. Bien que cette approche puisse permettre de dépasser certaines des limites observées dans la combinaison de modèles, des sources potentielles d'erreur subsistent. En particulier, des heuristiques sont utilisées dans le décodeur pour réduire l'espace de recherche et accélérer le décodage. Dans des conditions d'utilisation normales, les coupures n'introduisent que peu d'erreurs. Mais lorsque le contexte acoustique est de mauvaise qualité, les meilleures hypothèses peuvent être écartées de la pile du décodeur : ceci est d'autant plus vrai en utilisation temps réel du décodeur qui opère des coupures bien plus sévères.

Dans ces cas liés aux coupures, notre stratégie basée sur la "promotion" des hypothèses du faisceau coïncidant avec la transcription ne permet pas de récupérer ces erreurs. On peut quantifier de façon approximative la perte correspondant à cette situation en utilisant le moteur de reconnaissance pour faire un alignement forcé de la transcription exacte sur le signal.

Les expériences combinant l'interpolation des modèles de langage avec un alignement sur la transcription exacte sont présentées dans le tableau 3.3.

Nous obtenons dans ce cas un taux d'erreur mots de 3.5%. Ce niveau d'erreur peut être considéré comme minimal pour une méthode réévaluant les hypothèses concurrentes dans le faisceau d'hypothèses sans remettre en cause le contenu même de ce

Protocole	Taux d'erreur
ML-G seul + alignement TrEx	6.1%
ML-TrEx seul + alTrEx	4.1%
ML-G70%+ML-TrEx30%+alTrEx	3.7%
ML-G50%+ML-TrEx50%+alTrEx	**3.5%**
ML-G30%+ML-TrEx70%+alTrEx	3.7%

TAB. 3.3: *Résultats des expériences interpolant modèle de langage générique (ML-G) avec le modèle de langage appris sur la transcription exacte (ML-TrEx) et s'alignant sur la transcription exacte (alTrEx). Les pourcentages correspondent aux poids d'interpolation des modèles de langage*

faisceau. Cette méthode valide l'hypothèse que le modèle de langage n'est pas suffisant dans cette configuration : l'apport du décodage guidé améliore sensiblement la qualité du décodage. Nous avons retenu cette stratégie pour exploiter des transcriptions imparfaites.

3.5.5 Expériences avec modèle de langage 'approximatif' et alignement

Les expériences précédentes nous ont permis de mesurer le potentiel de notre stratégie. Les expériences suivantes présentent l'interpolation du modèle de langage générique avec un modèle de langage estimé sur les transcriptions imparfaites. Le tableau 3.4 reprend les expériences précédentes mais en remplaçant la transcription exacte par la transcription approchée. Le modèle de langage est également remplacé par le modèle appris sur les transcriptions imparfaites.

Protocole	Taux d'erreur
ML-TrErr + alignement TrEr	9.9%
ML-G + alignement TrEr	7.7%
ML-G70%+ML-TrEr30%+alTrEr	7.2%
ML-G50%+ML-TrEr50%+alTrEr	7.4%
ML-G30%+ML-TrEr70%+alTrEr	8.6%

TAB. 3.4: *Résultats des expériences avec interpolant le modèle de langage générique (ML-G) avec le modèle appris sur la transcription approchée (ML-TrEr), et s'alignant sur la transcription approchée (alTrEr). Les pourcentages correspondent aux poids d'interpolation des modèles de langage*

Les meilleurs résultats sont obtenus en combinant le modèle de langage générique avec un poids de 70% avec le modèle appris sur la transcription approchée et en réalisant un alignement sur cette dernière. L'alignement fait descendre le taux d'erreur mots jusqu'à 7.2%. L'alignement dynamique permet d'apporter une information temporelle qui est mal prise en compte par le modèle de langage. L'utilisation d'une distance d'édition associée à l'interpolation des modèles montre à nouveau un gain.

Cette expérience montre qu'un équilibre peut être atteint pour exploiter l'information approchée sans pour autant reproduire la majorité des erreurs qu'elle comporte.

Les meilleurs résultats sont obtenus en utilisant à la fois le modèle de langage générique biaisé et un alignement sur la transcription *a priori*. L'information erronée ne se trouve que dans la transcription sur laquelle le moteur essaie de s'aligner. Quand il ne trouve aucun alignement, il se replie exclusivement sur l'utilisation du modèle de langage générique. Par ailleurs, la légère interpolation avec le modèle de langage appris sur la transcription approchée permet de corriger certaines erreurs inhérentes au modèle de langage générique. Dans ces conditions, le système tire avantageusement parti de la transcription approchée lorsqu'elle est correcte et bascule en mode de reconnaissance automatique lorsque les observations acoustiques ne correspondent pas à la transcription proposée (figure 3.5).

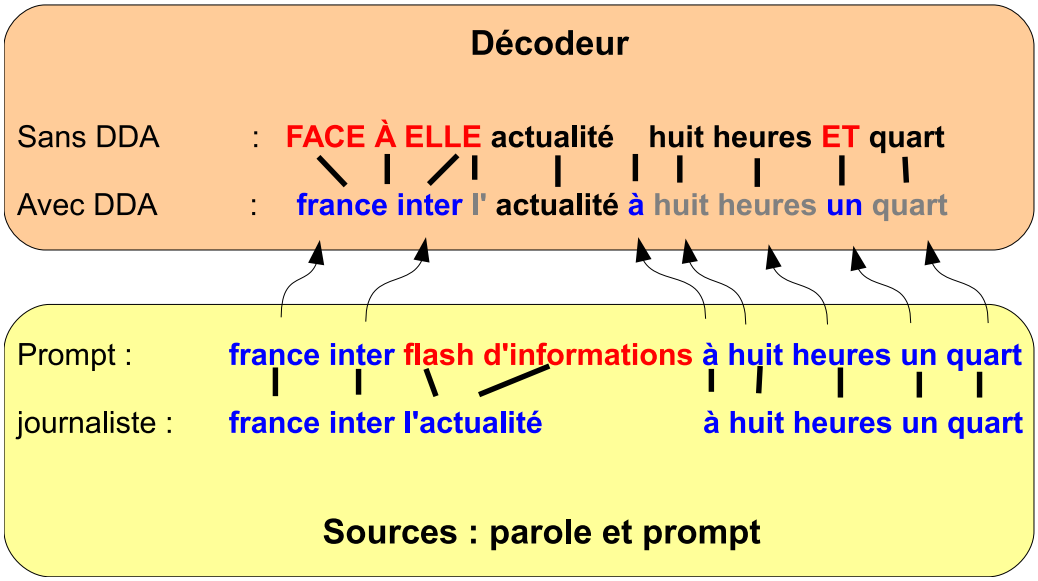

FIG. 3.5: *Exemple d'un résultat obtenu via le décodage guidé : de nouvelles hypothèses apparaissent, en modifiant l'exploration du graphe*

Les expériences montrent que le décodage guidé est complémentaire d'une simple interpolation de modèles de langages : il intègre parfaitement l'information de la transcription dans la fonction de recherche. Cependant des causes potentielles d'erreurs subsistent : les heuristiques de coupure et les portions acoustiques mal modélisées.

3.5.6 Expériences sur le corpus d'évaluation

Afin de valider l'ensemble de nos résultats, nous testons la meilleure configuration de notre système sur un corpus plus large. Nous procédons à l'évaluation sur deux heures supplémentaires, en suivant le même protocole. Les résultats sont reportés dans le tableau 3.5.

Nous observons à partir de ces résultats que les gains de performance sont relativement indépendants de la qualité initiale des transcriptions *a priori*. Il est intéressant de noter que les améliorations relatives les plus importantes sont observées sur l'heure de

Émissions	SRAP de référence	Taux d'erreur Transcriptions	Décodage guidé
dév : France Inter (1) - 1 heure	22.7%	10.1%	7.2%
test : RTM - 15 minutes	19.2%	9.1%	7.9%
éval : France Inter (2) - 1 heure	21.1%	10.2%	7.7%
éval : France Info - 1 heure	24.3%	20.3%	12.1%

TAB. 3.5: *WER obtenus sur le système de référence, WER contenu dans les transcriptions a priori et WER obtenu après un décodage guidé par les transcriptions a priori*

France Info pour laquelle la transcription fournie est la plus mauvaise (20% de WER). Ce résultat confirme que l'approche est robuste face aux imperfections présentes dans les transcriptions.

Ces expériences montrent aussi que des transcriptions imparfaites peuvent être exploitées très favorablement au cours d'un processus de décodage. Une partie de ces travaux a été présentée par [Lecouteux *et al.*, 2006], [Lecouteux *et al.*, 2006b].

3.6 Conclusion

Nous avons évalué notre stratégie, étape par étape, puis estimé les performances maximales pouvant être apportées par chacune des deux méthodes proposées. Elles permettent d'exploiter l'information contenue dans une transcription imparfaite pour améliorer les performances d'un SRAP.

- La première consiste à extraire du script l'information linguistique sous forme d'un modèle de langage tri-gramme appris sur la transcription approchée. Nos expérimentations montrent que l'interpolation de ce modèle avec le modèle de langage générique permet d'améliorer significativement le décodage. Il ne permet cependant pas de dépasser la qualité de la transcription approchée fournie, ce qui limite son intérêt.
- La seconde approche présentée consiste à orienter l'algorithme de recherche vers la transcription en synchronisant à la volée les hypothèses en cours d'évaluation et la transcription dont on dispose. Cette méthode permet de combiner efficacement les scores linguistiques avec les scores d'alignement. Cette stratégie permet d'obtenir des gains significatifs, même si la qualité de la transcription *a priori* est mauvaise. Le gain relatif de WER obtenu sur l'ensemble des trois heures est compris entre 28% et 40% par rapport aux transcriptions imparfaites fournies.

L'un des intérêts du décodage basé sur A^* est la facilité avec laquelle des sources d'informations supplémentaires sont intégrées au cœur même de l'algorithme de recherche. La difficulté majeure est la synchronisation entre la sonde et la transcription imparfaite : nous proposons une synchronisation à la demande basée sur un algorithme d'alignement dynamique, permettant de combiner un SRAP asynchrone avec une transcription linéaire. Le système est ainsi capable d'exploiter la transcriptions ap-

prochée lorsqu'elle correspond au signal et bascule en reconnaissance automatique autonome lorsque les observations acoustiques ne correspondent pas à la transcription *a priori*.

Ici, l'évaluation des hypothèses guidée par la transcription atteint les objectifs fixés tout en accélérant le décodage. En effet, nous observons un gain sur le temps d'exécution dû tant à la réduction de l'espace de recherche qu'à une meilleure anticipation des chemins optimaux, qui correspondent souvent aux hypothèses alignées. Ce gain en terme de vitesse de décodage peut probablement être augmenté en introduisant plus tôt des heuristiques basées sur la transcription approchée, notamment au niveau de la sonde elle-même. L'ensemble de ces travaux a été présenté dans [Lecouteux *et al.*, 2006].

Bien que ces premiers résultats montrent l'intérêt d'un alignement sur une transcription approchée, ces expériences ont été effectuées dans des conditions contrôlées : niveau de bruit relativement réduit, transcription relativement proche de la transcription exacte, etc. Nous proposons d'adapter notre système à des conditions plus difficiles et dans des conditions réelles. La prochaine section présente les travaux menés dans des conditions réelles, sur des corpus volumineux où l'information est dispersée.

Chapitre 4

Détection d'îlots de transcription

Sommaire

Ce chapitre présente nos travaux permettant d'exploiter/retrouver de l'information noyée dans une grande masse de données. Le décodage guidé présenté dans le chapitre précédent est limité à des segments de petites tailles. Or, dans la pratique, les transcriptions associées (prompts, sous-titres, scénarios) sont dépourvues d'informations temporelles ; de plus elles sont volumineuses et imparfaites. Il est nécessaire d'établir une stratégie qui trouve des îlots de transcription dans l'ensemble du corpus.

Les deux difficultés majeures pour exploiter ces transcriptions *a priori* sont leur imperfection, que nous avons traitée précédemment, et leur manque d'informations temporelles : les alignements linéaires sont inadaptés (les segments peuvent être mélangés, manquants, insérés...). De plus, la quantité de données devient en elle-même un problème : la complexité des algorithmes d'alignement étant trop grande (la complexité est proportionnelle à la taille des corpus alignés).

Dans le chapitre précédent, nous avons vu qu'un certain nombre de travaux proposent des solutions en effectuant des synchronisations en plusieurs passes [Cardinal *et al.*, 2005, Lamel *et al.*, 2002, Witbrock et Hauptmann, 1998] , mais ces approches sont lourdes, non triviales. Elles reposent sur un processus dans lequel synchronisation et décodage sont dissociés, induisant une perte potentielle de précision.

Nous présentons dans ce chapitre, une méthode identifiant à la volée des îlots de transcription correspondant au signal décodé, au sein de corpus de taille importante. Nous proposons également de coupler cette méthode avec le décodage guidé, afin d'avoir un système capable d'exploiter tout type de transcription.

L'algorithme de décodage guidé présenté précédemment aligne et corrige des transcriptions imparfaites via le SRAP. Nous avons montré que cet algorithme permet d'améliorer la transcription finale aussi bien par rapport à un décodage normal que par rapport à la transcription utilisée comme support. Cependant, des suppressions de morceaux de transcription ou des insertions importantes, peuvent corrompre la recherche des points d'ancrage. De plus, la transcription *a priori* doit correspondre à des zones de taille limitée, isolées dans une masse importante de signal. Nos expériences ont montré que lorsque le système de reconnaissance est guidé par des transcriptions qui ne contiennent pas d'information corrélée avec le signal audio, le décodage devient deux à trois fois plus lent, tandis que le taux d'erreur mots reste inchangé.

Dans l'objectif d'adapter le décodage guidé à de grands corpus, nous développons un algorithme de détection de segments de texte. Cet algorithme permet d'isoler des îlots de transcription dans l'ensemble de la masse de signal. Il doit être rapide, et rechercher à la demande, pendant le décodage, des points d'ancrage dans les transcriptions.

4.1 Stratégie proposée

La méthode que nous présentons exploite les transcriptions imparfaites lorsqu'elles sont disponibles alors qu'aucune information temporelle ne permet de les localiser dans la masse globale de corpus. Par ailleurs, le décodage guidé est capable d'intégrer au sein de l'algorithme de recherche des informations relatives à un prompt ou une transcription. Nous proposons de combiner l'algorithme de décodage guidé avec un détecteur d'îlots de transcription. Ainsi, à chaque nœud exploré dans le graphe du SRAP, l'algorithme explore l'ensemble des corpus à la recherche d'accroches. Étant donné que le graphe de recherche se développe dynamiquement, cette accroche doit s'effectuer à la volée.

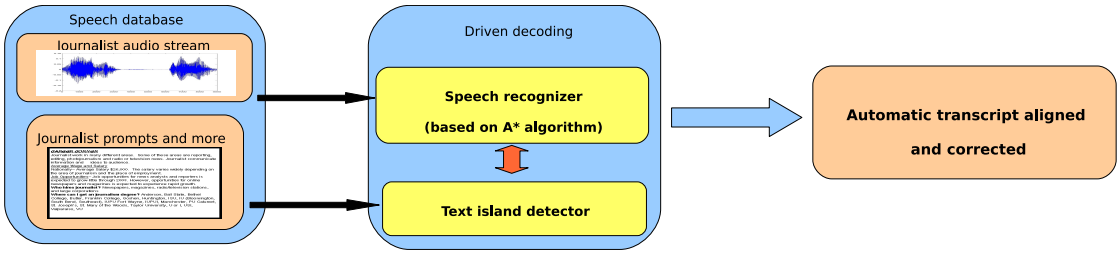

FIG. 4.1: *Principe général du système exploitant de grands corpus*

Le principe de la méthode se rapproche des techniques issues de la recherche d'information. Mais dans ce cas, l'hypothèse joue le rôle d'une requête du système dont la réponse est à chercher dans un corpus de texte volumineux où un îlot de transcription répondra à la requête. Dans une approche vectorielle, les moteurs de recherche tentent de trouver les documents qui concordent le mieux avec la requête. Pour cela, l'hypothèse et le corpus sont indexés, avec leurs mots et leurs positions. La plupart des algorithmes construisent un ensemble de réponses qui sont ordonnées par leur confiance. Nous présentons un concept similaire qui est à la fois efficace et adapté à cette tâche.

4.2 Définition de notre algorithme de recherche

Chaque dimension de l'espace lexical L_s est associée à un mot. L'ensemble des documents (les transcriptions), y compris l'hypothèse elle-même sont représentés dans cet espace lexical par des vecteurs de fréquence. Les coefficients des vecteurs représentent la fréquence des mots dans le document, associée à la position du mot.

Au fur et à mesure que les hypothèses sont développées, un ensemble de mots C_i est construit et mis à jour (figure 4.2). Ces partitions sont le résultat de l'intersection de l'hypothèse h_c avec un îlot de confiance I_i. Ainsi, pour chaque nouveau mot ajouté à l'hypothèse h_c, des îlots de transcription sont considérés comme des candidats afin de guider la recherche du système de reconnaissance. Cette compétition est arbitrée par un score de confiance S_i qui est calculé de la manière suivante :

$$S_i(I_i) = \frac{|I_i(t)|}{|h_c(t)|} \sum_{w_k \in C_i}^{k} idf(w_k) \tag{4.1}$$

où $|I_i(t)|$ et $|h_c(t)|$ sont respectivement les cardinalités de l'îlot I_i et de l'hypothèse courante h_c. C_i est une partition résultant de l'intersection entre l'hypothèse courante et la transcription, telle que $C_i = h_c \cap I_i$. $idf(w)$ correspond à la mesure classique de la fréquence de mot inverse :

$$idf(w) = \frac{NbDocuments}{tf(w)} \tag{4.2}$$

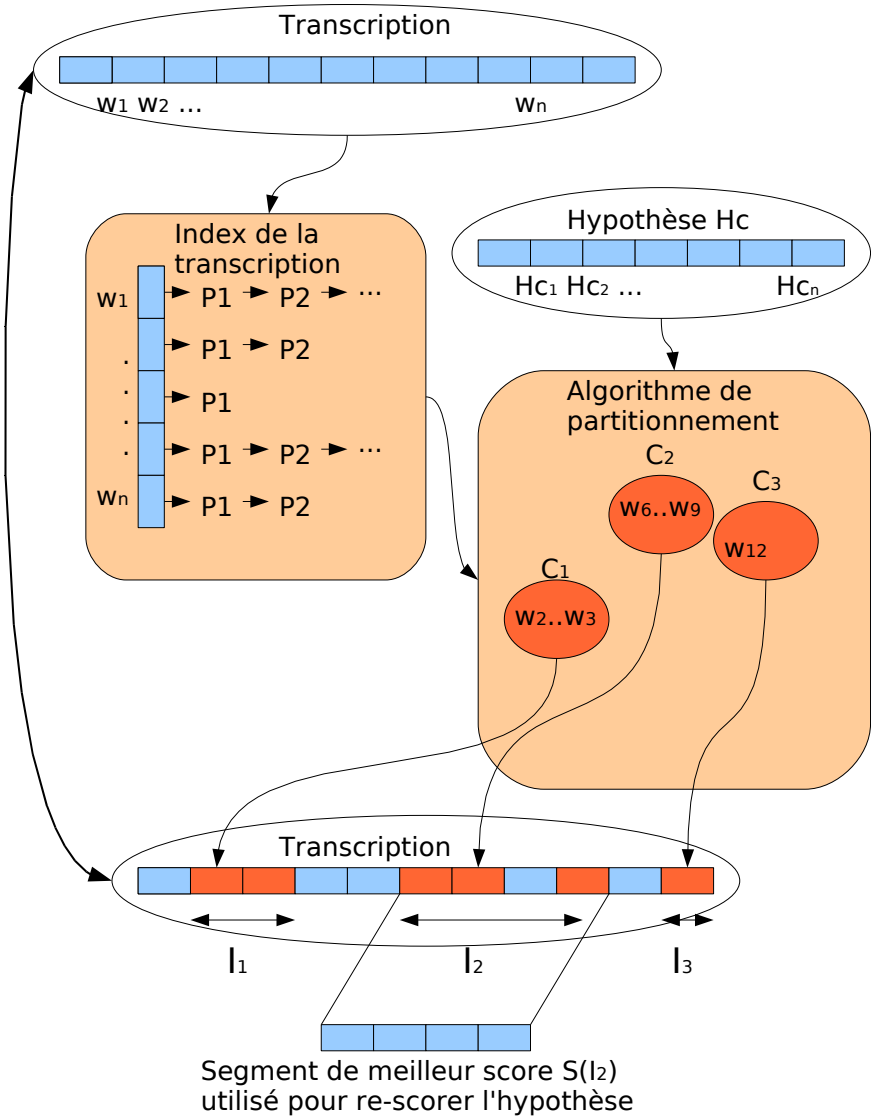

FIG. 4.2: *Identification des zones d'alignement, P_i représente la position d'un mot dans la transcription*

Dans notre cas, $NbDocuments = 1$. Ce score de confiance s'interprète comme le degré de similarité entre l'hypothèse courante et chacun des îlots de transcription considérés. Cette mesure s'appuie sur le poids sémantique de chaque mot qui dépend de sa fréquence relative dans le document.

Le score de confiance doit être supérieur à un seuil τ fixé *a priori*, pour que l'algorithme considère que cet îlot est suffisamment fiable pour guider le système de reconnaissance.

Un autre aspect, développé dans cette technique afin de contrôler la complexité de

l'algorithme, est l'adaptation dynamique du nombre de partitions. Ce nombre $N(C)$ est compris entre deux seuils :

$$N_{min} \leq N(C) \leq N_{max} \tag{4.3}$$

Où N_{min} et N_{max} correspondent aux limites fixées par l'utilisateur : N_{min} permettra d'explorer un minimum d'îlots et N_{max} réduira le bruit apporté par les mots à forte fréquence.

Ainsi, le nombre maximal de partitions estimées sera constant. Pour agir dynamiquement sur le nombre de partitions, les mots considérés comme porteur de sens sont sélectionnés en priorité : cette sélection dépend directement de leur fréquence, en considérant qu'un mot de faible fréquence apporte plus de sens. Une partition est définie par :

$$C = Tr \cap h_c \tag{4.4}$$

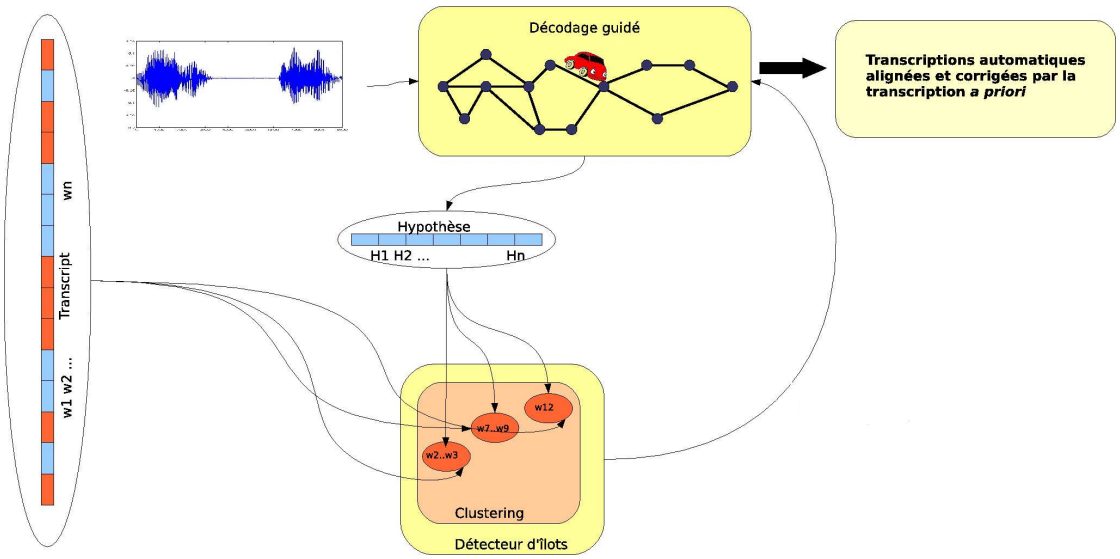

FIG. 4.3: *Principe général du système exploitant de grands corpus : le détecteur d'îlots de transcriptions construit des partitions qui correspondent à l'hypothèse courante, la meilleure guide alors le système*

Le nombre de partitions $N(C)$ dépend donc des mots contenus dans h_c. En sélectionnant les mots plus ou moins pertinents de h_c, le nombre d'intersections diminue ou augmente. La sélection se fait sur la fréquence des mots dans la transcription *a priori*. Ainsi, quelles que soient les hypothèses du SRAP, l'algorithme de recherche s'adapte dynamiquement pour ne pas être trop exhaustif ou sans réponse. Cette méthode permet, sur le début des hypothèse de SRAP de limiter le nombre de partitions, un faible nombre de

mots ayant plus de chances de se retrouver en divers endroits. Quand l'hypothèse est développée, le nombre d'intersections diminue car les partitions commencent à cibler précisément des zones de transcription. Cette diminution induit alors un relâchement sur la coupure des mots sélectionnés, améliorant la précision de la recherche. Cette approche adapte la reconnaissance guidée à de grands corpus, sans ralentir l'ensemble du système (figure 4.3).

4.3 Déroulement de l'algorithme lors du décodage

L'algorithme se décompose ainsi :

1. Indexation de l'ensemble des transcriptions : À chaque mot sont associés un identifiant w_i et une position p_i, le nombre de partitions est prédéfini à N_{max}, le seuil sur la fréquence des mots S est initialisé à 0, les fréquences de chaque mot w_i $tf(w_i)$ sont calculées. L'ensemble des partitions C est vide. Les partitions sont composés de positions de mots et une position de mot p_i est considérée comme appartenant à la partition $C_n = \{p_0, .., p_m\}$ si elle vérifie les conditions suivantes :

$$\arg\min\{p_0, ..., p_m\} - 2 \leq p_i \leq \arg\max\{p_0, ..., p_m\} + 5 \tag{4.5}$$

   Les bornes sont définies de manière à favoriser l'avancée des îlots sur le contexte droit des transcriptions.
2. Le SRAP propose une hypothèse $h_c = \{w_1, ..., w_n\}$
3. Pour chaque mot w_j de h_c vérifiant la condition $tf(w_j) > S_{freq}$, pour chaque position p_g du mot w_j :
   – Si $p_g \in \{C\}$, p_g est rajouté à la première partition pouvant le contenir
   – Si $p_g \notin \{C\}$, une nouvelle partition contenant p_g est créée
   Si deux partitions se chevauchent, elles sont fusionnées.
4. Si $|\{C\}| > N_{max}$ alors S_{freq} est incrémenté.
5. Si $|\{C\}| < N_{max}$ alors S_{freq} est décrémenté.
6. Un score de confiance est calculé pour chaque *îlot* I_i :

$$S_i(I_i) = \frac{|I_i(t)|}{|h_c(t)|} \sum_{w_k \in (h_c \cap I_i)}^{k} idf(w_k) \tag{4.6}$$

7. L' îlot ayant un score de confiance maximum est sélectionné. La portion de texte correspondant à cet îlot est alors utilisée pour le décodage guidé si elle est supérieure à un seuil fixé *a priori*.

Par la suite, nous avons proposé une autre mesure de confiance :

(6b) On calcule l'alignement optimum entre h_c et I_i : $DTW(h_c, I_i)$ pour en déduire les deux composantes suivantes :

$$Best = \frac{NbAlignes}{|h_c|} \text{ et } \quad Distorsion = \frac{NbAlignes}{DernierAligne - PremierAligne} \tag{4.7}$$

Où *DernierAligne* et *PremierAligne* correspondent aux index respectifs des premier et dernier mots alignés de l'hypothèse h_c par l'algorithme d'alignment dynamique. *NbAlignes* correspond au nombre de mots alignés entre l'hypothèse h_c et l'îlot courant I_i. La distorsion permet de mesurer la quantité de bruit introduite dans le segment qui correspond *a priori*.

Le score de confiance est alors défini par :

$$S_i(I_i) = Distorsion.Best.\frac{\sum_{i=0}^{n} idf(w_i)}{|I_i|} \tag{4.8}$$

Cette mesure est utilisée dans d'autres travaux d'alignement de grands corpus et présente de meilleurs taux d'alignement. Elle s'avère extrêmement efficace, car l'alignement s'effectue sur une fenêtre réduite à $\min\{|h_c|, |I_i|\}$: le temps de calcul de l'algorithme d'alignement est donc négligeable, mais apporte des informations précieuses sur les distorsions entre l'hypothèse et l' îlot. Ainsi, des îlots présentant des mots identiques dans le désordre sont directement éliminés de la recherche (figure 4.4).

4.4 Expériences

L'ensemble des expériences a été réalisé avec le SRAP du LIA utilisé dans le cadre de la campagne d'évaluation ESTER tel que présenté précédemment.

4.4.1 Les corpus d'évaluation

Afin d'évaluer notre méthode nous avons utilisé deux corpus. Le premier se base sur le corpus fourni lors de la campagne ESTER. Nous avons modifié les transcriptions exactes afin de simuler la fragmentation des transcriptions que l'on pourrait avoir dans le cadre d'expériences en conditions réelles. Le second corpus est en condition réelle, et se base sur des enregistrements de la radio RTBF, pour laquelle certaines parties de l'audio sont associées à des prompts.

4.4.2 Le corpus ESTER

Les premières expériences effectuées sont basées sur le corpus de la campagne d'évaluation d'ESTER.

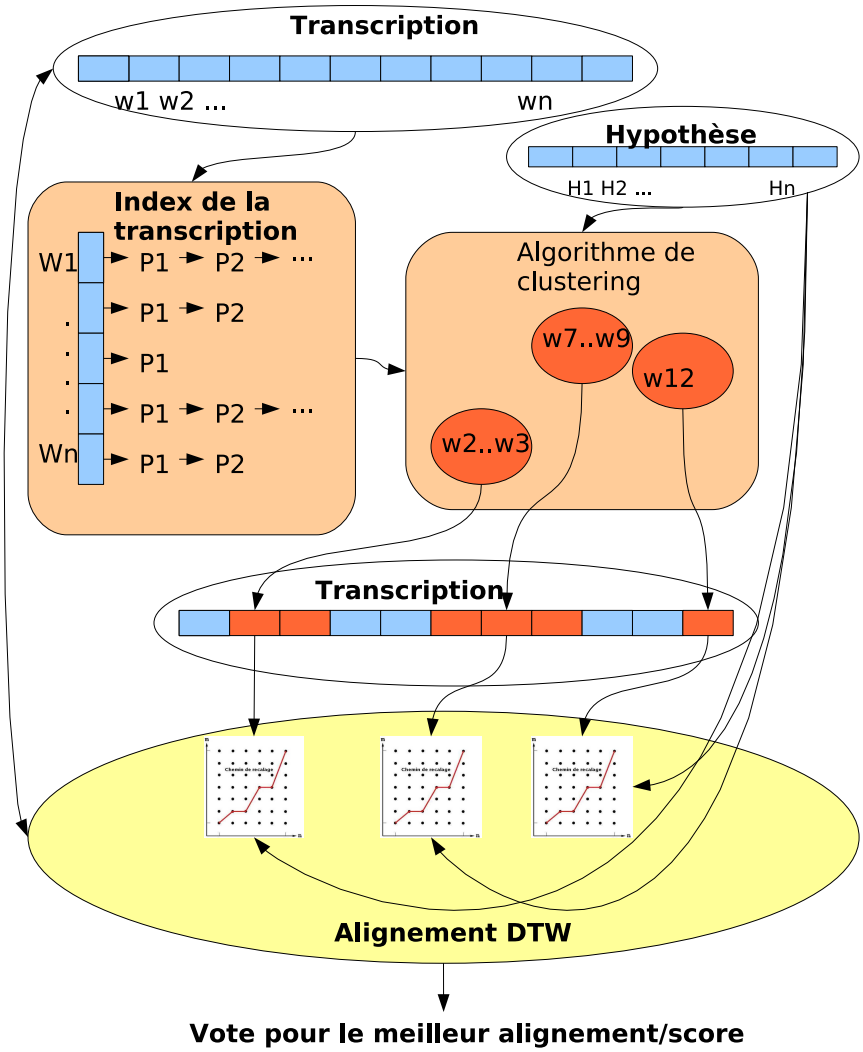

FIG. 4.4: *Identification des zones d'alignement : le détecteur d'îlots de transcriptions construit des partitions à partir du corpus indexé, qui correspondent à l'hypothèse courante, le meilleur îlot guide alors le système. Dans cette version, un alignement dynamique a été intégré pour prendre en compte l'ordre des mots de la requête*

Nous avons réutilisé les transcriptions imparfaites simulées avec erreurs rajoutées manuellement, 10% de WER dans les deux premières heures et 20% dans la seconde (cf. chapitre 3.5.1). De plus, afin de simuler des transcriptions incomplètes, nous avons supprimé 50% des segments.

4.4.3 Le corpus RTBF

Le second corpus utilisé a été extrait du cadre de travail du corpus fourni dans le projet AIDAR [Tshibasu-Kabeya *et al.*, 2006]. Ce corpus comprend environ 1000 heures

d'émissions issues de la Radio Télévision Belge Française (RTBF). Les émissions sont en langue Française et enregistrées dans de bonnes conditions audio. Elles contiennent essentiellement des nouvelles dont le style se rapproche de celui de la campagne d'ESTER. Le corpus RTBF est composé d'environ 60 heures de prompts dispersées dans environ 300 heures de signal audio. Ces prompts sont utilisés par les journalistes, mais ne correspondent pas toujours au signal audio. Par ailleurs, les prompts ne sont associés à aucune information temporelle, ce qui empêche leur localisation précise par rapport au signal audio. Afin d'évaluer notre algorithme sur cette base de données nous avons manuellement annoté 22 heures de cette base : cette annotation concerne exclusivement le temps d'apparition de chacun des segments.

4.4.4 Résultats expérimentaux

L'ensemble de nos expériences évalue les performances de l'algorithme proposé. L'objectif est de construire un large corpus de parole annoté, d'une qualité convenable. La méthode est totalement automatisée et s'appuie sur des prompts associés au signal audio. Les performances du système de recherche sont évaluées en terme de taux de précision/rappel sur les deux corpus présentés précédemment. Nous décrivons également le calcul de la F-mesure qui donne une information sur la qualité globale du système de recherche.

Dans un premier temps, nous avons testé l'alignement et les corrections effectuées sur le corpus ESTER, en utilisant des transcriptions exactes et imparfaites. Puis nous comparons les résultats au niveau du WER afin de quantifier la qualité des transcriptions produites grâce à notre procédure. Finalement nous évaluons notre méthode sur le corpus RTBF qui se base sur des prompts réels.

Recherche d'îlots sur des transcriptions exactes issues du corpus ESTER

Dans cette partie, nous évaluons les performances de l'algorithme de recherche en utilisant des transcriptions exactes. Les morceaux de transcriptions supprimés ont été choisis à partir de la référence en sélectionnant aléatoirement 50% des segments. Ces segments sont de tailles variables, d'une dizaine à une quarantaine de mots.

Les résultats expérimentaux sont présentés dans la table 4.1. Ces résultats montrent que dans des conditions simulées, les performances en terme de détection sont bonnes : plus de 95.3% des segments sont retrouvés avec une précision de 96.7%. De plus, les résultats semblent relativement indépendants des performances initiales du SRAP, qui varient selon les heures : de 27.2% pour RFI à 22.6% pour les émissions de France-Inter.

Recherche d'îlots sur des transcriptions imparfaites issues du corpus ESTER

La seconde suite d'expériences est effectuée sur des transcriptions imparfaites afin d'évaluer l'impact des erreurs sur les performances globales de notre méthode. Les

Radio	Précision	Rappel	F-mesure	Nombre de segments
INTER	90.9%	98.89%	94.8%	478
INFO	93.7%	92.9%	91.5%	468
RFI	98.9%	97.8%	98.4%	812
Moyenne	95.3%	97.3%	95.5%	1758

TAB. 4.1: *Détection d'îlots sur le corpus d'ESTER. Les expériences sont effectuées sur 3 heures de sources différentes : France Inter (INTER), France Info (INFO) et Radio France International (RFI). Ici, les transcriptions sont exactes*

résultats sont présentés dans la table 4.2. Les expériences montrent que les précisions et rappels sur des transcriptions imparfaites sont très similaires à ceux obtenus sur des transcriptions exactes. Ces expériences ont été effectuées avec des transcriptions ayant un taux d'erreurs d'environ 10% à 20% pour la dernière, ce qui correspond au taux d'erreur dans les prompts et les sous-titres. On peut penser qu'une large augmentation du taux d'erreur impacterait très négativement les performances de l'algorithme de détection de segments qui s'appuie essentiellement sur les similarités entre mots. Une situation extrême serait de fournir à l'algorithme une transcription qui ne correspond en rien au signal : aucun point d'accroche ne serait trouvé.

Station radio	Précision	Rappel	F-mesure	Nombre de segments
FrInter	90.7%	96.9%	93.7%	478
FrInfo	93.4%	89.7%	91.5%	468
RFI	98.8%	97.8%	98.4%	812
Moyenne	95.1%	95.4%	95.2%	1758

TAB. 4.2: *Détection d'îlots sur le corpus d'ESTER. Les expériences sont effectuées sur 3 heures de sources différentes : France Inter (INTER), France Info (INFO) et Radio France International (RFI). Ici, les transcriptions sont imparfaites : environ 10% de WER. 50% des segments ont été supprimés afin de tester la qualité du détecteur de termes*

Îlots de transcription et décodage guidé

Comme nous l'avons présenté dans le chapitre précédent, le décodage guidé permet d'améliorer les taux de reconnaissance d'un SRAP, en s'appuyant sur des transcriptions, même imparfaites. Dans ce paragraphe, nous évaluons la qualité des transcriptions fournies par le SRAP SPEERAL, via un décodage guidé. Nous avons effectué deux expériences que nous comparons au système de référence (décodage normal et sans transcription).

Dans un premier temps, nous évaluons le taux d'erreur mot en utilisant des segments issus de la transcription exacte. Ensuite, nous expérimentons l'utilisation de transcriptions imparfaites. Les résultats sont présentés dans la table 4.3. Le système de référence est SPEERAL, sans décodage guidé, en une seule passe. DDA+IT désigne un décodage guidé via des transcriptions imparfaites et DDA-PT correspond à un déco-

dage guidé via des transcriptions exactes. Les expériences sont effectuées sur 3 heures du corpus de développement d'ESTER, en utilisant des transcriptions *a priori* présentant de 10% (France Inter et France Info) à 20% de WER (RFI). Comme dans les expériences précédentes, 50% des segments ont été supprimés afin de tester la qualité de l'algorithme de détection de segments.

Système	Système de référence	DDA+IT	DDA+PT
INTER	22.6 %	17.9%	17.1%
INFO	23.4 %	21.7%	18.3%
RFI	27.2 %	23.0%	20.3 %
Moyenne	24.4 %	20.9 %	18.6 %

TAB. 4.3: *Efficacité de l'algorithme de détection de segments et influence sur le WER, DDA+IT correspond à un décodage guidé par une transcription imparfaite et DDA+PT une transcription parfaite*

L'ensemble de ces résultats montre que le décodage est nettement amélioré par l'algorithme de détection de segments par rapport au système de référence. Ce dernier fournit les portions de texte adéquates au système de décodage guidé. Sans algorithme de détection d'îlots, le DDA n'aurait pu aligner ces portions de texte étant donné qu'aucune segmentation préalable n'a été fournie.

L'utilisation de transcriptions exactes s'avère bien entendu plus efficace, mais les transcriptions imparfaites apportent cependant un gain de WER relatif moyen d'environ 14%, tandis que les transcriptions exactes permettent d'apporter un gain relatif d'environ 24%.

Détection d'îlots de transcription en condition réelle

Dans ce paragraphe, nous présentons les travaux effectués en conditions réelles sur le corpus RTBF. Les expériences ont été réalisées sur 22 heures pour lesquelles les temps d'apparition des segments ont été annotés manuellement. Le WER contenu dans les transcriptions RTBF est évalué aux alentours de 5% lorsque celles-ci sont présentes : en effet, seule une petite partie des 300 heures d'audio a été annotée. Cependant, ce WER reste très faible et correspond parfaitement à la tâche recherchée.

	Précision	Rappel	F-mesure	Nombre de segments
RTBF shows	99.28 %	97.13 %	98.41 %	501

TAB. 4.4: *Précision, rappel et F-mesure de l'algorithme de recherche lors des expériences réalisées sur les émissions radio de la RTBF couplées aux prompts des journalistes.*

Les résultats des expériences effectuées sur le corpus RTBF sont excellents et même meilleurs que ceux obtenus sur le corpus d'ESTER. Près de la moitié des 22 heures (10 sur 22) sont totalement alignées sans erreur, et la F-mesure pour ce corpus est supérieure à 98% (tableau 4.4). Ces résultats prometteurs sont en parti dus au fait que les

prompts ont une structure cohérente par rapport à l'audio : les sujets sont bien découpés, les locuteurs identifiés. L'algorithme de détection est donc parfaitement adapté à ce type de segmentation très structurée et trouve efficacement ses points d'accroche : les thèmes correspondants aux mots sont regroupés dans des zones précises du corpus. L'algorithme peut alors facilement créer ses *îlots* avec des confiances élevées. Un autre paramètre important à souligner est la taille des segments présents dans RTBF. Ils sont sensiblement plus grands que ceux d'ESTER : la moyenne est d'environ 20 segments par heure pour plus de 400 heures, ce qui représente des segments de longueur variant d'une centaine à cinq cent mots. Du fait de leur grande taille et de leur nombre réduit, le risque *a priori* d'en oublier est beaucoup plus faible qu'avec les corpus d'ESTER. Ces travaux sont présentés dans [Lecouteux et Linarès, 2008].

4.5 Améliorer et augmenter la quantité de données

Nous avons montré que l'algorithme de recherche est robuste dans des conditions de WER raisonnables et, qu'associé à un décodage guidé, il permet d'améliorer qualitativement les transcriptions automatiques. Nous présentons ici une méthode directement dérivée des travaux précédents, qui permet d'augmenter la masse de données nécessaire à l'apprentissage automatique de modèles acoustiques. Nous exploitons 200 heures de données audio issues de la radio RTBF associées à leurs prompts. Les prompts sont regroupés par mois et sont imparfaits/incomplets.

Notre objectif est d'exploiter au mieux toute l'information disponible afin d'extraire le maximum de corpus. Par ailleurs, les données étant destinées à des apprentissages discriminants (MMIE), il est impératif qu'elles comportent un minimum d'erreurs. Nous proposons une approche non-supervisée utilisant des prompts pour guider le décodage.

4.5.1 Stratégie basée sur notre algorithme de détection de segments

La stratégie que nous avons élaborée, s'appuie sur quatre étapes :

1. Ajout de tous les mots hors-vocabulaire contenus dans les prompts dans le modèle de langage du SRAP. Les mots sont phonétisés automatiquement via les outils du LIA : LIA_PHON [Bechet, 2001]. Ensuite, un modèle de langage est appris sur toutes les données issues du corpus RTBF (environ 2400000 mots). Ce dernier est fusionné avec le modèle de langage générique (un très faible poids est accordé au modèle RTBF).
2. Décodage utilisant à la fois l'algorithme de détection de segments et le décodage guidé. Ce décodage fournit des transcriptions où les mots alignés avec le prompt sont indiqués.
3. Extraction de la transcription de tous les mots alignés : en effet, s'ils correspondent au prompt, la probabilité de leur exactitude est extrêmement élevée.

4. Utilisation des données considérées comme exactes pour estimer les modèles acoustiques.

Afin d'estimer la quantité de données supplémentaires récoltées par notre méthode, nous avons mené des expériences comparatives :

– Une expérience de référence où le module de décodage guidé a été désactivé : le système fournit donc uniquement les alignements avec les prompts, sans qu'il n'y ait eu d'incidence sur les résultats.
– Dans la seconde expérience, le décodage guidé est activé, et influence le décodage.

Le tableau 4.5 présente les résultats de ces expériences.

Système	Référence	Décodage guidé
# Heures	200	200
# Segments	50370	50370
# Mots décodés	2 497 125	2 515 503
# Segments alignés	11158 (22%)	11487 (23%)
# Mots alignés	380042 (15% :̃30 heures)	615481 (25% :̃50 heures)

TAB. 4.5: *Mots correspondants entre le prompt et la sortie du SRAP. La première colonne montre le système de référence et la seconde montre le résultat lorsque le décodage guidé est activé*

Ces résultats montrent que, lorsque le décodage guidé est couplé à l'algorithme de détection de segments, 38% de données supplémentaires sont correctement alignées : ce qui représente environ 50 heures de parole annotée contre seulement 30 lors d'un décodage normal. De plus, le nombre de segments identifiés reste similaire, ce qui démontre la robustesse de l'algorithme de détection de segments.

Ces résultats montrent une amélioration significative de la quantité de données utilisable. Le décodage guidé permet de corriger à la volée le SRAP, ce qui correspond à l'amélioration de WER observée dans le chapitre précédent. Nous obtenons les résultats attendus : un plus grand corpus de meilleure qualité. Par ailleurs, les résultats présentés dans la littérature présentent des taux de données extraites inférieurs. L'ensemble de ces travaux a été présenté dans [Lecouteux *et al.*, 2007a, Lecouteux et Linarès, 2008].

4.5.2 Conclusions sur l'algorithme de recherche d'îlots de transcription

Dans ce chapitre, nous avons proposé une méthode permettant de généraliser le décodage guidé à des corpus de grande taille où l'information est dispersée. Notre méthode permet de retrouver des îlots de transcription dans de grandes bases de données audio. L'algorithme permet de construire pour un coût réduit des corpus d'apprentissage destinés aux SRAP. La recherche d'îlot est directement intégrée au sein du décodage guidé. Son rôle consiste essentiellement à détecter dynamiquement des îlots de transcription dans la transcription, au fur et à mesure que le SRAP explore des hypothèses. Par ailleurs, une fois ces îlots de transcription identifiés, ils sont injectés dans le

SRAP pour le guider. Nos résultats et expériences montrent d'excellents résultats sur le corpus RTBF. De plus l'algorithme est robuste avec des transcriptions imparfaites.

Nous avons évalué cette technique dans le cadre de la construction de corpus à bas coût, en s'appuyant sur des ressources limitées telles que des prompts. Les résultats montrent, sur notre corpus, un gain d'environ 38% de données supplémentaires extraites, lorsque nous exploitons la recherche d'îlots combinée au décodage guidé. Cette méthode permet de produire des corpus propres en quantité dès lors que des transcriptions *a priori* sont disponibles, ce qui est particulièrement adapté pour les méthodes d'apprentissage nécessitant des données peu corrompues.

4.6 Conclusion

Nous avons proposé une méthode exploitant pleinement des transcriptions imparfaites, via un décodage guidé. Nous avons montré qu'une transcription imparfaite peut être améliorée par un décodage guidé, impliquant de fait l'amélioration du décodage lui-même. La stratégie consiste à intégrer directement l'information au sein de la fonction d'estimation du décodeur : le SRAP s'aligne sur la transcription imparfaite lorsque cette dernière correspond, et redevient autonome dès que la transcription s'éloigne du signal.

Nous avons également proposé un algorithme étendant l'utilisation de transcriptions imparfaites à des segments dispersés dans de grands corpus : le SRAP détecte à la volée des îlots de texte, afin qu'ils puissent le guider.

L'ensemble des techniques proposées s'intègrent dans de nombreuses applications telles que la correction automatique de transcriptions [Lecouteux *et al.*, 2007b], la génération automatique de corpus de qualité [Lecouteux et Linarès, 2008], la transcription automatique assistée par des scripts *a priori*, ou la recherche de sous-séquences [Rouvier *et al.*, 2008, Lecouteux *et al.*, 2007a].

Troisième partie

Combinaison de systèmes automatiques de la parole

Chapitre 5

État de l'art : stratégies globales de combinaisons entre SRAP

Sommaire

Ce chapitre propose d'abord un survol théorique des stratégies de combinaison de classifieurs statistiques. Nous présentons ensuite les méthodes existantes permettant de combiner des SRAP à différents niveaux : acoustiques, linguistiques, espace de recherche et combinaison sur les sorties des décodeurs. Un tour d'horizon présente les stratégies globales de combinaison de SRAP. Nous finissons par présenter les préalables nécessaires à une combinaison efficace : la complémentarité et l'équivalence entre SRAP.

Dans la partie précédente, nous proposions des méthodes exploitant des transcriptions *a priori* produites manuellement. L'approche proposée permet l'intégration de ces dernières directement au cœur de l'algorithme d'exploration. Nous considérerons maintenant, le cas particulier où la transcription *a priori* est produite par un système de reconnaissance automatique de la parole. Nous proposons le principe de DDA, formulé dans le cadre de la combinaison de systèmes. Plusieurs schémas de combinaisons seront évalués. Nous dressons dans cette partie un état de l'art des techniques de combinaisons.

Dans cet état de l'art sur la combinaison, nous présentons d'abord le cadre théorique général de la combinaison de probabilités. Nous abordons ensuite les combinaisons au travers des divers niveaux d'un SRAP : acoustique, linguistique, espace de recherche et sorties. Finalement nous expliquons à quel niveau se place notre contribution.

5.1 Modèles théoriques de combinaison

Cette section présente dans un cadre général les techniques de combinaison de classifieurs statistiques. Nous verrons ensuite comment les combinaisons s'appliquent dans le contexte des SRAP.

Diverses approches théoriques (figure 5.1) pour combiner les probabilités provenant de différents classifieurs sont présentées par [Kittler *et al.*, 1998]. Leur approche est formalisée ainsi : On considère une observation X, un ensemble de classes $M = \{w_1, w_2, ..., w_m\}$, et R classifieurs. Dans le cas de la reconnaissance automatique de la parole, il est nécessaire d'affecter l'observation R à l'une des classes de M. Dans ce cas x_i représente l'observation de X vue par le classifieur. En appliquant la théorie de Bayes, la classe $w_{selection}$ qui représentera le mieux l'observation X, sera celle qui maximisera la probabilité *a posteriori* de l'hypothèse w_k sachant l'observation x_i :

$$w_{selection} = \arg\max_{w_k} P(w_k|x_1...x_R) \tag{5.1}$$

Il est nécessaire de calculer cette probabilité à partir des données disponibles. En appliquant le théorème de Bayes, la probabilité *a posteriori* se décompose ainsi :

$$P(w_k|x_1...x_R) = \frac{P(x_1...x_R|w_k)P(w_k)}{P(x_1...x_R)} \tag{5.2}$$

avec $P(w_k)$ la probabilité *a priori* de la classe w_k et $P(x_1...x_R|w_k)$ la probabilité jointe des mesures extraites par les différents classifieurs sachant la classe et $P(x_1...x_R)$ la probabilité jointe des différentes observations. [Kittler *et al.*, 1998] propose alors différentes règles pour combiner les classifieurs.

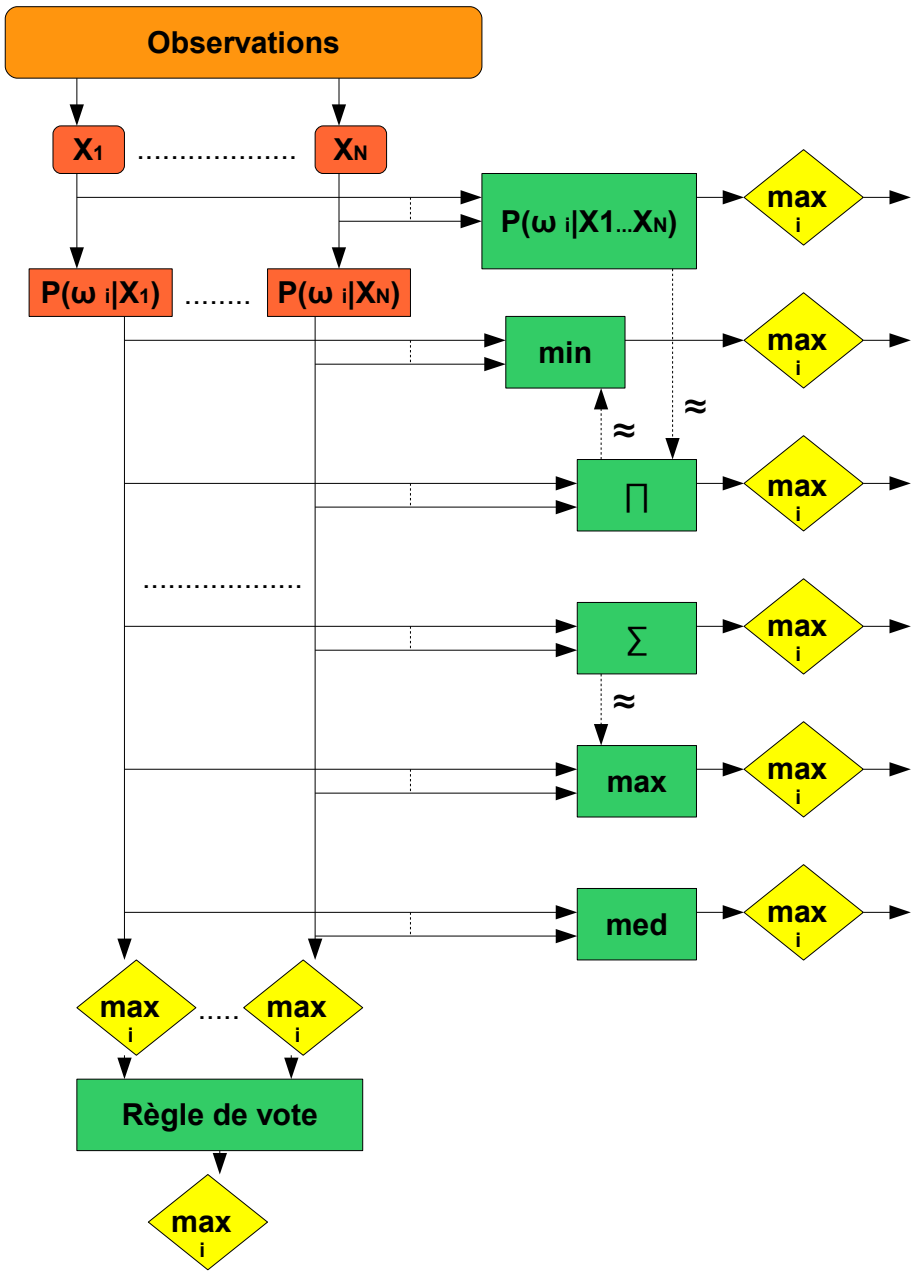

FIG. 5.1: *Schéma général des règles de combinaisons présentées par [Kittler et al., 1998]*

5.1.1 Combinaison via un produit

Il s'agit d'une combinaison simple dans laquelle la probabilité jointe est vue comme le produit des probabilités marginales. Ceci est théoriquement exact lorsque les x_i sont "différentes" versions de la même observation.

$$P(x_1...x_R|w_k) = \prod_{i=1}^{R} P(x_i|w_k) \tag{5.3}$$

Où $p(x_i|w_k)$ représente la mesure par le modèle i. En conservant l'hypothèse d'indépendance des observations, on obtient :

$$w_{selection} = \arg\max_{w_k} \prod_{i=1}^{R} P(x_i|w_k)P(w_k) \tag{5.4}$$

Comme discuté précédemment, la probabilité jointe peut être vue comme la probabilité du vecteur $(x_1, ..., x_n)$ dans l'espace produit. Cette approche correspond à un "et" logique qui n'est pas forcément le comportement souhaité intuitivement, dans lequel la règle de décision devrait intégrer la probabilité qu'un classifieur se trompe ou qu'un des estimateurs de $P(x_i|w_k)$ soit localement de mauvaise qualité. Un seul modèle peut donc "inhiber" les autres, ce qui en général n'est pas souhaitable : il est plus intéressant de prendre en compte les décisions de tous les modèles.

5.1.2 Combinaison via une somme

En considérant que la règle du produit ne peut s'appliquer que dans les cas où l'ensemble des probabilités sont relativement proches, ce qui n'est pas le cas dans des conditions où les probabilités sont très éparses, [Kittler *et al.*, 1998] posent alors l'approximation suivante :

$$P(w_k|x_i) = P(w_k)(1 + \delta_{ki}) \text{ avec } \delta_{ki} << 1 \tag{5.5}$$

La règle de combinaison via une somme est alors définie ainsi par les auteurs :

$$w_{selection} = \arg\max_{w_k} \sum_{i=1}^{R} P(w_k|x_i) \tag{5.6}$$

La combinaison via une somme permet de prendre en compte l'ensemble des classifieurs sans que l'un d'eux ait la possibilité, à lui seul, de compromettre la décision finale. Ce modèle aura d'ailleurs tendance à favoriser les classifieurs présentant de fortes probabilités. Ce type de combinaison, quant à elle, correspond à un "ou" logique.

5.1.3 Combinaisons linéaire et log-linéaire

Les combinaisons basées sur des sommes ou produits peuvent être généralisées en pondérant les scores de chacun des classifieurs. Ceci conduit aux combinaisons linéaires et log-linéaires. La combinaison linéaire d'hypothèses se définit ainsi :

$$P(w_k|x_i) = \sum_{i=1}^{R} \alpha_i P(w_k|x_i) \tag{5.7}$$

Où α_s détermine la confiance dans le classifieur s, et se définit par $\sum_{s=1}^{S} \alpha_s = 1$. La combinaison linéaire est celle utilisée dans la combinaison des réseaux de confusion. Elle correspond à une généralisation de la combinaison par somme.

La combinaison log-linéaire, quant à elle est définie ainsi :

$$P(w_k|x_i) = \frac{1}{Z} exp\left(\sum_{i=1}^{R} \alpha_i log\left(P(w_k|x_i)\right)\right) \tag{5.8}$$

Où α_i détermine la confiance dans le classifieur i, et se définit par $\sum_{s=1}^{S} \alpha_s = 1$. La combinaison log-linéaire correspond à une généralisation de la combinaison par produit.

5.1.4 Combinaisons basées sur un maximum ou minimum

[Kittler *et al.*, 1998] montrent que les deux règles énoncées précédemment peuvent être approximées. En effet, la règle de combinaison via un produit aura tendance à favoriser l'ensemble ayant une mauvaise probabilité, tandis que la règle de combinaison via une somme aura la tendance inverse. À partir de ce constat, ils définissent :

$$P(w_k|x) = \frac{\max_{i=1}^{R} P(w_k|x_i)}{\sum_{j=1}^{K} \max_{i=1}^{R} P(w_j|x_i)} \tag{5.9}$$

Cette approche permet de sélectionner la probabilité d'une classe qui sera maximale par rapport à celles proposées par l'ensemble des classifieurs. Elle correspond par ailleurs à une approximation d'une combinaison via une somme.

De même, la combinaison via le produit peut être approximée par la sélection de la probabilité minimale, ainsi définie :

$$P(w_k|x) = \frac{\min_{i=1}^{R} P(w_k|x_i)}{\sum_{j=1}^{K} \min_{i=1}^{R} P(w_j|x_i)} \tag{5.10}$$

Ces approximations lissent le résultat de la combinaison tout en conservant les approches initiales permettant de favoriser les probabilités fortes ou faibles.

5.1.5 Combinaisons basées sur la médiane

Quand une combinaison via une somme moyennée est utilisée, il est envisagé que l'ensemble des probabilités en compétition sont comparables. Cette hypothèse peut être faussée en présence de bruit, ou si l'un des classifieurs génère des scores extrêmes pour certaines classes. Dans ces cas, des maximaux ou minimaux locaux faussent la décision finale. Il est alors opportun de s'appuyer sur la médiane des classifieurs :

$$P(w_k|x) = \frac{med_{i=1}^{R} P(w_k|x_i)}{\sum_{j=1}^{K} med_{i=1}^{R} P(w_j|x_i)} \tag{5.11}$$

La médiane revient ici à sélectionner la probabilité séparant la distribution de probabilité en deux parties de cardinal identique.

5.1.6 Combinaison par vote majoritaire

Cette approche est utilisée par défaut dans la méthode *ROVER*. Cette combinaison consiste à sélectionner la classe pour laquelle le plus grand nombre de classifieurs aura donné une probabilité maximale. La classe ayant un maximum de votes sera choisie par consensus :

$$P(w_k|x) = \frac{\sum_{j=1}^{K} \delta_{ij}}{K} \tag{5.12}$$

avec

$$\delta_{ki} = \begin{cases} 1 & \text{si } P(w_k|x_i) = \max\limits_{j=1}^{K} P(w_j|x_i) \\ 0 & \text{sinon} \end{cases} \tag{5.13}$$

Étant donné que la décision se fait par un vote majoritaire, elle peut s'appliquer à des classifieurs ayant des scores disparates.

5.1.7 Combinaison par critère d'entropie

L'entropie peut s'interpréter comme une mesure du désordre ; plus la source est redondante, moins elle contiendra d'informations. En l'absence de contraintes particulières, l'entropie sera maximale pour une source dont tous les symboles sont équiprobables (ce qui correspond à un désordre maximal). Dans le cas d'une décision sur un vecteur de classes, il faudra sélectionner le classifieur présentant l'entropie minimale : celui où une classe se démarquera le plus. Ainsi, si un vecteur présente des probabilités semblables pour toutes les classes, la décision peut être considérée comme peu fiable. L'entropie se définit ainsi pour une observation x_i :

$$E = -\sum_{k=1}^{K} P(w_k|x_i) log P(w_k|x_i) \tag{5.14}$$

Le critère de décision consiste à sélectionner la sortie du classifieur dont le vecteur de probabilités propose la plus petite entropie. Ce type de combinaison [Misra *et al.*, 2003b] sélectionnera le classifieur apportant l'information la plus discriminante.

5.1.8 Synthèse sur les modèles de combinaisons

Dans cette section nous avons présenté quelques des techniques de combinaison théoriques. Chacune d'elle correspond à un champ d'application souhaité. Les combinaisons par somme et produit équivalent respectivement à des "et" & "ou" logiques dans l'espace cible de la combinaison et s'avèrent, assez comparables dans la pratique à une combinaison par critère maximum ou minimum. Les combinaisons basées sur la médiane tentent d'éliminer les classifieurs trop discriminants. Dans le même esprit, l'approche du vote est très répandue dans le cadre de la technique *ROVER* développée plus loin : les classifieurs n'ont pas besoin d'avoir des scores comparables. Les combinaisons linéaires, log-linéaires ou basées sur le critère d'entropie sont intéressantes car elles permettent de prendre en compte la qualité du classifieur qui se mesure par l'entropie ou par un estimateur de confiance α sur un classifieur donné.

5.2 Combinaison au niveau acoustique

Les combinaisons au niveau acoustique peuvent s'appliquer sur les paramètres ou sur la sortie des modèles. Nous présenterons rapidement les principales approches pour ces deux combinaisons au niveau acoustique.

5.2.1 Combinaison des paramètres acoustiques

Une approche simple est d'effectuer la combinaison au niveau des paramètres. Elle permet d'exploiter la complémentarité des différents paramètres. Ils seront alors tous intégrés en un paramètre unique, lequel sera exploité classiquement par les modèles. Cette technique présentée par [Ellis, 2000] nécessite cependant que les paramètres s'appliquent à la même portion de signal.

Dans d'autres travaux comme ceux de [Misra *et al.*, 2003a], les dérivées premières et secondes sont combinées aux paramètres pour leur apporter une information sur la dynamique acoustique. Une approche similaire, consiste en l'augmentation des vecteurs de paramètres, où des paramètres supplémentaires (voisement, prosodie...) sont incorporés dans les vecteurs.

Une approche différente consiste à combiner des paramètres, puis à réduire le vecteur obtenu afin de minimiser la complexité de modélisation et de sélectionner l'information pertinente. Dans ce cas, la sélection des paramètres essentiels se fait via une analyse en composantes principales (ACP) [Zolnay *et al.*, 2005].

5.2.2 Combinaison des modèles acoustiques

Dans les méthodes de [Kirchhoff, 1998, Zolnay *et al.*, 2005, Rong Zhang, 2006], plusieurs modèles fournissent des probabilités qui vont être combinées afin de prendre une décision globale. Un inconvénient de cette méthode est que les scores à combiner sont issus de différents modèles qui ont des dynamiques pouvant être très différentes et nécessitent des étapes de normalisation pour être combinés.

Un point essentiel est le synchronisme, lors de la combinaison des modèles acoustiques. Certaines approches, supposent que les modèles à combiner observent exactement le même évènement. Il est alors indispensable d'adapter la topologie des modèles pour introduire ce synchronisme [Bourlard *et al.*, 1996].

D'autres méthodes [Mirghafori et Morgan, 1998] tentent, au contraire, d'exploiter l'asynchronie. En effet, les modèles n'observent pas exactement les mêmes évènements : certains ont des transitions qui arrivent plus tôt ou plus tard. L'asynchronisme est pris en compte en particulier dans les systèmes multi-bande qui découpent le signal audio en plusieurs bandes de fréquences. À chaque bande de fréquence est associé un modèle. Puis, l'ensemble des modèles est combiné, permettant ainsi de mettre concurrence des informations réparties dans les différentes bandes.

5.3 Combinaison et adaptation des modèles de langage

Les qualités d'un modèle de langage dépendent énormément de son domaine d'application. Lorsque les données d'apprentissage et de test sont différentes, il est nécessaire d'adapter un modèle générique aux conditions d'utilisation. Cette amélioration

de la modélisation repose souvent sur la combinaison de différents modèles, qui peut également s'apparenter à l'adaptation d'un modèle générique. L'adaptation consiste à spécialiser un modèle généraliste à l'aide de données plus spécialisées ou adaptées (figure 5.2).

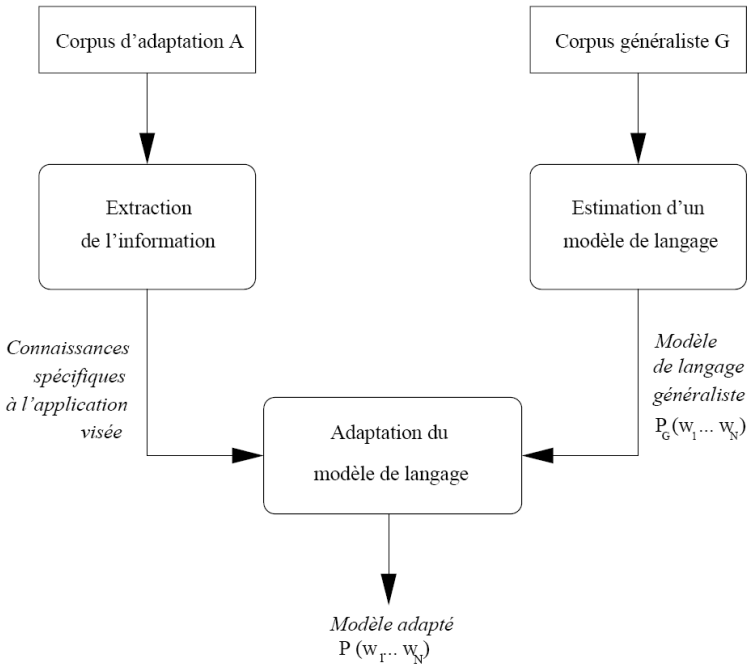

FIG. 5.2: *Principe général de l'adaptation des modèles [Estève, 2002]*

5.3.1 Interpolation de modèles

À partir de M corpus différents, il existe plusieurs approches pour construire des modèles de langage combinant l'ensemble des données. La technique la plus simple serait de fusionner l'ensemble des textes et d'estimer un modèle de langage n-gramme sur l'ensemble. Cependant, les différents corpus peuvent différer tant au niveau du volume que du contenu ; une concaténation écarterait les corpus sous-représentés. La combinaison linéaire introduite par [Jelinek et Mercer, 1980] reste la plus généralement utilisée. Elle permet de construire indépendamment différents modèles de langage, chacun avec ses propres propriétés. Une fois obtenues, les différentes distributions sont combinées par une combinaison linéaire convexe, selon laquelle $p(h|w)$ s'obtient comme :

$$P'(w|h) = \sum_{1}^{n} \alpha_n . P_n(w|h) \tag{5.15}$$

avec $\sum \alpha_n = 1$. Les valeurs α_n dépendant du poids que l'on veut donner au modèle de langage. Généralement α_n est une valeur estimée par rapport à la perplexité

du modèle de langage. Toutefois, l'interpolation linéaire peut induire une perte de précision au niveau des probabilités de replis. Pour cette raison [Stolcke, 2002] construit un nouveau modèle de langage à partir des *M-modèles* en (ré)évaluant les nouvelles probabilités de repli. Les travaux de [Bacchiani *et al.*, 2006] conduisent à une méthode alternative qui fusionne les comptes des n-gramme en lieu et place de la moyenne des probabilités de n-grammes :

$$P^{CM}(w|h) = \frac{\sum_i \beta_i tf_i^{disc}(hw)}{\sum_j \beta_j tf_j(h)} \tag{5.16}$$

Où β_i est le facteur d'échelle, tf_i^{disc} est le compte du n-gramme hw i et $tf_i(h)$ est le compte de l'historique h pour le modèle i. Les probabilités pour les n-grammes non observés sont calculées avec un repli :

$$P^{CM}(w|h) = \begin{cases} p^{CM}(w|h) & \text{si } \sum_i tf_i(hw) > 0 \\ \alpha(h) p_{bo}^{CM}(w|h') & sinon \end{cases} \tag{5.17}$$

Où $\alpha(h)$ est la probabilité de repli et h' est l'historique pour le repli. Cette méthode obtient une meilleure perplexité qu'une interpolation linéaire. D'autres travaux se sont focalisés sur des variantes, telles que l'interpolation log-linéaire [Klakow, 1998] ou exponentielle [Rosenfeld, 1996]. Mais les probabilités de repli sont difficiles à estimer sur ces combinaisons, ce qui rend leur intégration à un SRAP difficile.

[Hsu, 2007] montre que fusionner les comptes de n-grammes revient à une généralisation de l'interpolation linéaire où le coefficient d'interpolation $\lambda_i(h) = \frac{\beta_i tf_i(h)}{\sum_j \beta_j tf_j(h)}$ dépend de l'historique du n-gramme lors de son décompte :

$$P^{CM}(w|h) = \frac{\sum_i \beta_i tf_i^{disc}(hw)}{\sum_j \beta_j tf_j(h)} = \sum_i \lambda_i(h) p_i(w|h) \tag{5.18}$$

À partir de ce constat [Hsu, 2007] propose une généralisation des pondérations entre modèles de langages en fonction de l'historique courant. Le modèle se définit ainsi :

$$p^{GLI}(w|h) = \sum_i \lambda_i(h) p_i(w|h) \tag{5.19}$$

Où $\lambda_i(h) >= 0$ et $\sum_i \lambda_i(h) = 1$ pour l'ensemble des historiques observés. [Hsu, 2007] propose de modéliser la fonction de poids $\lambda_i(h)$ à l'aide d'une fonction $rel_i(h)$ dépendant du modèle i normalisée par les fonctions des autres modèles :

$$\lambda_i(h) = \frac{rel_i(h)}{\sum_j rel_j(h)} \tag{5.20}$$

La fonction $rel_i(h)$, quant à elle, dépendra du type d'interpolation souhaité. Cette méthode permet d'optimiser au mieux la combinaison des modèles via la fonction d'interpolation dynamique.

5.3.2 Combinaison par Maximum *a posteriori*

[Federico, 1996a] proposent d'adapter un modèle de langage général sur des corpus plus spécialisés en s'appuyant sur le critère de maximum *a posteriori* (MAP). Contrairement à l'interpolation linéaire qui combine des modèles de langage au niveau général, cette méthode interpole des modèles au niveau de la fréquence des mots. Cette méthode, très répandue, peut se définir de la manière suivante :

$$P^{MAP}(w_i|h_i) = \begin{cases} \dfrac{\epsilon tf_A(h_iw_i) + tf_G(h_iw_i)}{tf_A(h_i) + tf_G(h_i)} & \text{si } tf_A(h_iw_i) + tf_G(h_iw_i) > 0 \\ 0 & \text{sinon} \end{cases} \tag{5.21}$$

Où ϵ est un facteur estimé empiriquement ou dynamiquement [Chen et Huang, 1999] destiné à pondérer l'influence du corpus d'adaptation. $tf_A(h_iw_i)$ et $tf_G(h_iw_i)$ sont les nombres d'occurrences de la séquence de mots h_iw_i dans les corpus A et G.

5.3.3 Adaptation dynamique des modèles de langage

[Chen *et al.*, 2004a] proposent une adaptation dynamique des modèles de langage à partir de corpus de texte volumineux. Le principe repose sur deux étapes. La première extrait du corpus de texte les données nécessaires à l'adaptation, étape qui s'inspire des techniques de recherche d'information où l'hypothèse du système de reconnaissance automatique de la parole est considérée comme une requête. Celle-ci permet de retrouver le segment qui s'en rapproche le plus :

$$\frac{K(q,s_j)^\gamma}{N_j} \sum_{i=1}^{N} \sum_{k=1}^{N_j} idf(w_k) log \frac{Pr(keyword_i, w_k)}{Pr(keyword_i)Pr(w_k)} \tag{5.22}$$

Où $K(q,s_j)$ est le nombre de mots clefs différents qui sont communs à la requête et à l'article candidat.

La seconde étape consiste à sélectionner le segment maximisant la ressemblance pour adapter le modèle de langage sur celui-ci. Les auteurs obtiennent alors leurs meilleurs résultats avec une adaptation MDI [Federico, 1996b] présentée en 5.3.7.

Une approche similaire dans laquelle les modèles sont adaptés est présentée par [Federico et NicolaBertoldi, 2001], qui proposent d'exploiter des flash d'information quotidiens. [Whittaker, 2001] propose également d'adapter un modèle de langage à partir de données journalières. Ces méthodes se basent sur des interpolations classiques d'un modèle de langage générique avec les données contemporaines. Les deux travaux proposent de pallier le problème des mots hors-vocabulaire en augmentant les lexiques des modèles de langage.

5.3.4 Modèles caches et modèles "*triggers*"

Les modèles caches sont issus de l'observation suivante : un mot qui est apparu récemment dans un paragraphe a de grandes chances de réapparaître dans un futur très proche [Kuhn *et al.*, 1990] : les auteurs proposent de modifier dynamiquement le modèle de langage en fonction des évènements observés récemment. Le cache contient les mots observés dans un passé proche et l'algorithme augmente leur probabilité dans le modèle de langage :

$$P(w_i|w_1, w_2, ..., w_{i-1}) = \mu P_{cache}(w_i|w_1, w_2, ..., w_{i-1}) + (1-\mu)P(w_i|w_{i-2}, w_{i-1}) \quad (5.23)$$

Les différents travaux sur les modèles caches ont défini dans un premier temps :

$$P_{cache}(w_i|w_1, w_2, ..., w_{i-1}) = \frac{tf(w_i)}{K} \quad (5.24)$$

Où K est la taille du cache et $tf(w_i)$ le nombre d'occurrences de w_i dans le cache. Cependant cette formalisation présente quelques lacunes. En effet, la distance d'un mot du cache au mot courant n'est pas prise en compte. Or, les mots les plus récents devraient avoir plus d'impact sur la modification du modèle de langage. Dans leur article, [Clarkson et Robinson, 1997] proposent ainsi une amélioration du modèle cache :

$$P_{cache}(w_i|w_1, w_2, ..., w_{i-1}) = \beta \sum_{j=1}^{i-1} I_{w_i=w_j} e^{-\alpha(i-j)} \quad (5.25)$$

Où I est un indicateur booléen tel que $I_A = 1$ si A est vrai et 0 sinon. α correspond au décalage entre le mot du cache et le mot actuel. β est une constante de normalisation. Par ailleurs, [Rosenfeld, 1994] propose de mettre en cache uniquement les mots les plus

rares. Les modèles caches sont exploités lors de l'adaptation via une combinaison avec des modèles à base de classes :

$$P(w_i|h_i) = \sum_{\{g_i\}} P(w_i|g_i)P(g_i|h_i) \tag{5.26}$$

Où g_i est l'ensemble des classes associées au mot w_i, $P(w_i|g_i)$ la composante n-gramme statique dépendant du modèle de langage et $P(g_i|h_i)$ une composante dynamique dépendant à la fois du corpus d'adaptation et du modèle de langage, définie telle que :

$$P(g_i|h_i) = (1-\lambda)P_A(w_i|g_i) + \lambda P_G(w_i|g_i) \tag{5.27}$$

Le coefficient d'interpolation λ est estimé empiriquement à partir du corpus d'adaptation.

[Lau *et al.*, 1993, Singh-Miller et Collins, 2007] présentent une généralisation des modèles caches : les modèles *triggers* (déclencheurs). Les modèles *triggers* exploitent les dépendances existantes entre des couples de mots, lorsque les corpus d'apprentissage permettent de mettre en évidence de fortes corrélations. Par exemple, le couple de mots (*génétique ;éprouvette*) sera facilement associé au thème de la biologie.

Ces couples de mots peuvent être associés à des thèmes préalablement appris. Ainsi, lorsque ces mots sont détectés, ils déclenchent l'intégration de modèles plus spécialisés avec le modèle courant. Cette opération est alors réalisée via une interpolation classique entre modèles.

Plus généralement, les modèles *triggers* permettent d'augmenter la probabilité d'apparition de certains mots en fonction des occurrences trouvées dans l'historique de l'hypothèse courante.

5.3.5 Combinaison par mélange statique de modèles

Les modèles caches comportent cependant des lacunes : ils n'ont pas d'action au sein de la phrase elle-même, mais plutôt sur la globalité d'un texte [Iyer et Ostendorf, 1999].

La combinaison par mélange de modèles introduite par [Knesser et Steinbiss, 1993] puis améliorée par [Iyer *et al.*, 1994, Clarkson et Robinson, 1997] consiste à découper de grands corpus notés $\mathcal{M}$ en J sous-corpus $< \mathcal{M}_1, ..., \mathcal{M}_J >$; chaque sous-corpus correspondant à un sujet défini manuellement ou automatiquement. Un modèle est appris sur chaque sous-corpus puis les différents corpus sont mixés avec des poids dépendants du corpus d'apprentissage :

$$f(w_t|w_1^{t-1};\mathcal{M}) = \sum_{j=1}^{J} \alpha_j f(w_t|w_1^{t-1};\mathcal{M}_j) \tag{5.28}$$

Où $f(w_t|w_1^{t-1};\mathcal{M})$ est le n-gramme d'un mélange et $f(w_t|w_1^{t-1};\mathcal{M}_j)$ sa j^{ieme} composante. Les α_j représentent les coefficients d'interpolation, avec $\sum_{j=1}^{J} \alpha_j = 1$.

L'estimation des coefficients d'interpolation est réalisée à l'aide d'un algorithme d'Espérance-Maximisation (EM) [Dempster *et al.*, 1977, Gotoh et Renals, 1999] en maximisant la vraisemblance du corpus d'adaptation connaissant les mélanges de modèles. Dans ce type de combinaisons, les modèles sur les sous-corpus sont appris définitivement, et l'adaptation se fait via les coefficients d'interpolation qui dépendent du corpus d'adaptation. [Iyer et Ostendorf, 1999] combinent mélanges de modèles et modèles caches pour bénéficier des avantages de chaque méthode d'adaptation.

5.3.6 Combinaison par mélange dynamique de modèles

La combinaison par mélange dynamique de modèles [Gotoh et Renals, 1999] est une extension de la combinaison par mélange de modèles. Dans la combinaison initiale, l'adaptation dépend essentiellement des coefficients d'interpolation et les composants (sous-corpus) d'adaptation sont indépendants de la tâche courante. Dans cette extension, les sous-corpus sont découpés en fonction des sujets définis par les hypothèses décodées. Ainsi, seuls des sous-corpus dépendant du sujet seront extraits puis interpolés (figure 5.4). Le protocole nécessite de découper les données d'apprentissage en deux sous-ensembles : l'un permet d'estimer les coefficients d'interpolation et l'autre d'extraire les sujets choisis. Cette approche obtient de meilleurs résultats qu'un mélange statique, les données étant plus spécifiques du sujet traité.

5.3.7 Combinaison par Information de Discrimination Minimale (MDI)

L'adaptation de modèles de langages par Minimal Discrimination Information (MDI) a été introduite par [Rao *et al.*, 1995], puis perfectionnée par [Federico, 1996b].

Étant donné un modèle générique $P_b(h,w)$ et un corpus d'apprentissage A l'adaptation par MDI a pour objectif de trouver un modèle $P(h,w)$ respectant une contrainte linéaire minimisant une divergence de Kullback-Leibler entre $P(h,w)$ et $P_b(h,w)$. La divergence de Kullback-Leibler se définit ainsi :

$$D_{KL}(P_b||P) = P_b(h,w) \sum_{h,w} log \frac{P_b(h,w)}{P(h,w)} \tag{5.29}$$

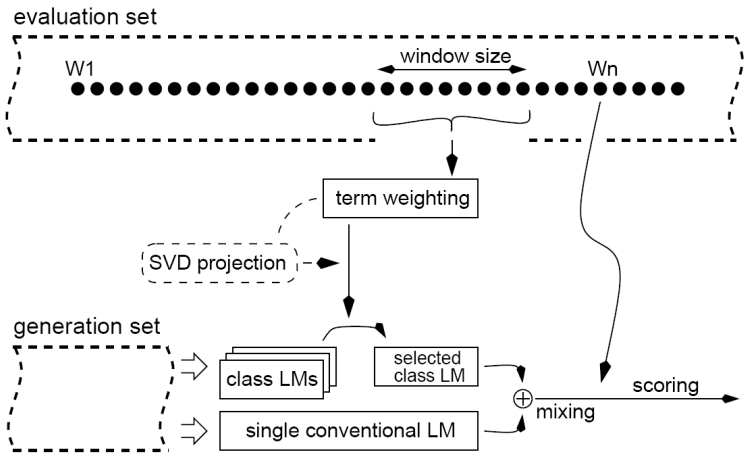

FIG. 5.3: *Combinaison par mélange dynamique de modèles [Gotoh et Renals, 1999]*

Dans leurs travaux, les auteurs de [Federico, 1996b] estiment leur modèle MDI en utilisant un algorithme GIS (*Generalized Iterative Scaling*) [Darrock et Ratcliff, 1972].

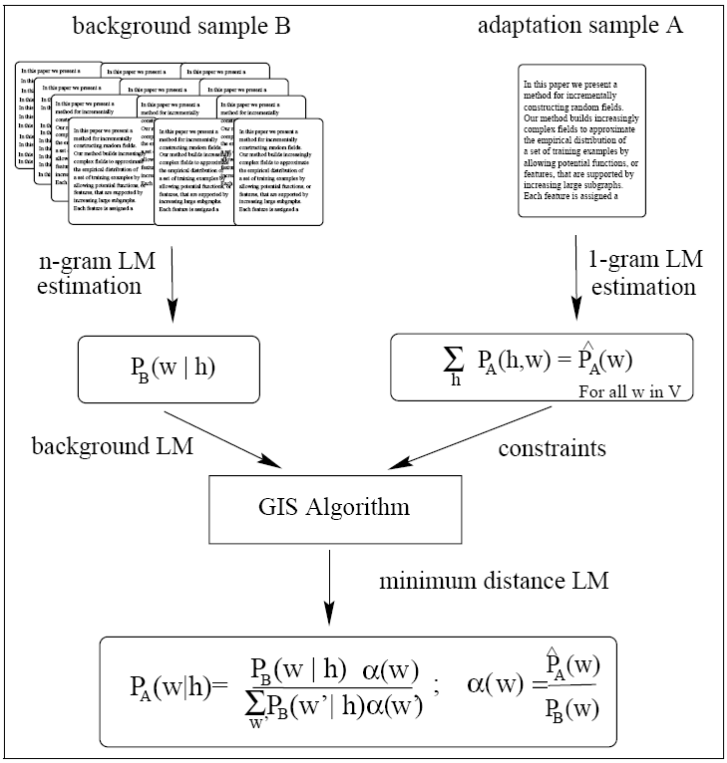

FIG. 5.4: *Adaptation de modèle de langage par MDI [Federico, 1996b]*

5.3.8 Adaptation par spécification de contraintes

L'adaptation par spécification de contrainte consiste à orienter l'apprentissage des modèles de langage sur des zones précises du corpus d'adaptation (estimées comme discriminantes) qui sont alors considérées comme des contraintes. Le critère maximum d'entropie, principalement utilisé pour sélectionner ces contraintes, oriente l'apprentissage sur les données qui n'ont pas encore été vues. Cependant, ce type d'apprentissage correspond à un cas particulier de l'adaptation par MDI.

5.4 Adaptation croisée

L'adaptation croisée (*cross-adaptation*) est une stratégie simple et efficace utilisée dans de nombreux systèmes à l'état de l'art. Il s'agit d'une stratégie d'adaptation non supervisée, pouvant être mise en œuvre par diverses techniques d'adaptation acoustique [Gales *et al.*, 2007] (figure 5.5).

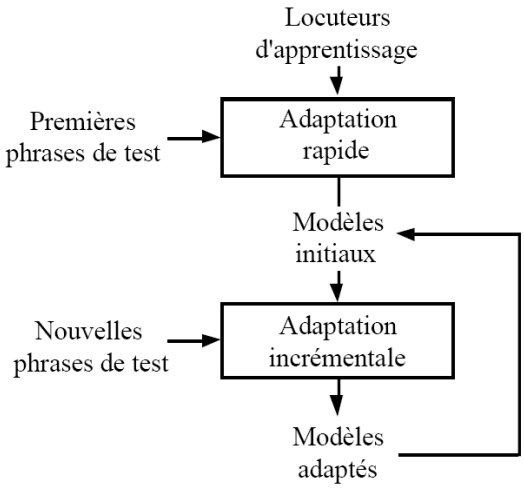

FIG. 5.5: *Principe de l'adaptation d'un modèle acoustique [Barras, 1996]*

Le principe est le suivant : la sortie d'un système S_1 est utilisée pour adapter les modèles acoustiques d'un modèle S_2. Le processus peut être itératif et symétrique en adaptant successivement les modèles de S_2 avec S_1 puis ceux de S_1 sur S_2. Cette méthode permet de propager les informations issues des autres systèmes via les modèles acoustiques (figure 5.6), en utilisant deux systèmes qui se fournissent réciproquement des données adaptées complémentaires. Il est important que les deux SRAP utilisés aient des performances comparables [Stuker *et al.*, 2006] pour que l'adaptation soit efficace.

Le système *SuperEARS* présente une stratégie de combinaison entre plusieurs SRAP issus de différents laboratoire et exploite l'adaptation croisée à plusieurs étages du

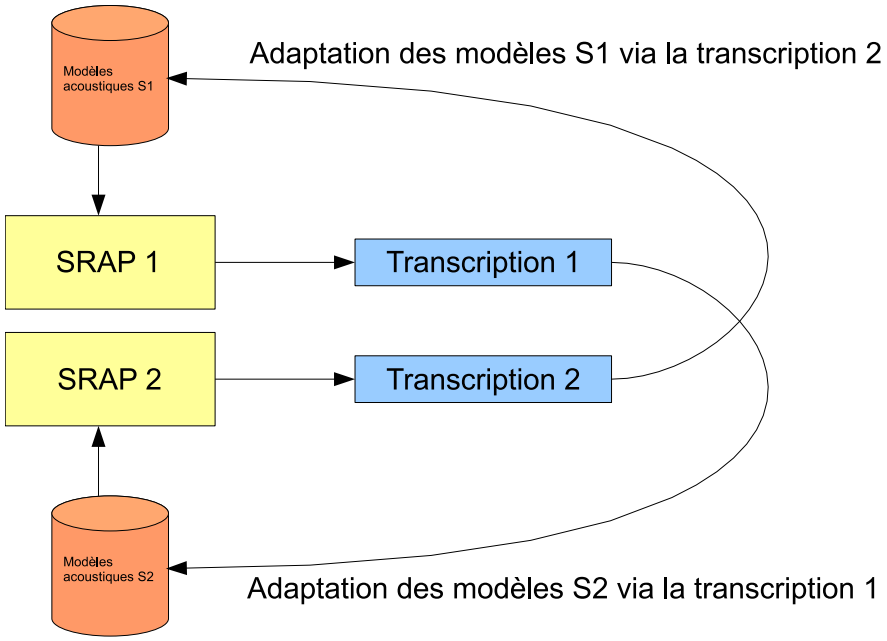

FIG. 5.6: *Principe de l'adaptation croisée entre deux systèmes*

système global [Woodland *et al.*, 2004]. Dans leur article [Zweig et Picheny, 2004], présentent aussi l'adaptation croisée entre modèles de langage. En effet, certains modèles sont spécifiques à des domaines : combiner des modèles issus du web et de domaines différents est équivalent à une adaptation croisée.

5.5 Combinaison *a posteriori*

La combinaison de multiples sorties de systèmes de reconnaissance automatique de la parole consiste à minimiser le taux d'erreur mot via une règle de Bayes associée à une fonction de coût (fonction de Levenshtein) :

$$\{w_1^N\}_{opt} = \arg\min_{w_1^N} \left\{ \sum_{v_1^M} \mathcal{L}(w_1^N, v_1^M) p(v_1^M | x_1^T) \right\} \tag{5.30}$$

avec une séquence de mots w_1^N, la probabilité *a posteriori* $p(v_1^M|x_1^T)$ pour la séquence de mots v_1^M sachant l'observation acoustique x_1^T.

5.5.1 Scores de confiance et combinaison

Les scores de confiance constituent un estimateur de la qualité de chacune des hypothèses. Leur mesure est essentielle lors de la combinaison d'hypothèses issues de

différents SRAP. Une combinaison sera d'autant plus optimale que les systèmes seront capables d'estimer leurs faiblesses ou les points forts de leurs hypothèses.

5.5.2 Combinaison par consensus : *ROVER*

Le *ROVER* (*Recognizer Output Voting Error Reduction*) [Fiscus et Fiscus, 1997] est destiné à réduire le taux d'erreur mot à partir des transcriptions de plusieurs systèmes. La méthode consiste en un vote sur l'ensemble des hypothèses alignées. *ROVER* s'appuie sur deux modules : le premier fusionne les transcriptions pour en faire un réseau de confusion et le second effectue un vote sur chaque nœud du réseau :

$$Score(w) = \alpha \frac{tf_i(w)}{N} + (1 - \alpha) S_i(w) \tag{5.31}$$

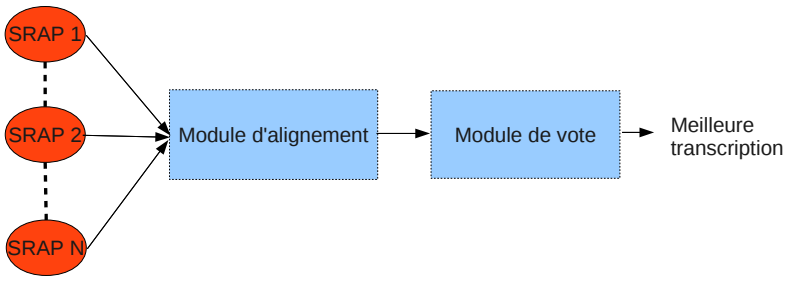

FIG. 5.7: *Principe de combinaison via ROVER*

Où N est le nombre de systèmes combinés, $S_i(w)$ est le score de confiance du mot w à la position i de l'hypothèse, et $tf_i(w)$ le nombre d'occurrences du mot w à la position i de l'hypothèse, α est calculé empiriquement sur un corpus d'apprentissage. Le *ROVER* (figure 5.7), dans sa première présentation offre trois types de vote : majoritaire, basé sur la moyenne des scores de confiance et un dernier s'accordant sur le score de confiance maximum.

5.5.3 *ROVER* assisté par un modèle de langage

[Schwenk et Gauvain, 2000] proposent une amélioration de *ROVER*, constatant que l'information linguistique n'est pas utilisée lors d'une application classique du *ROVER*. En effet, le vote s'effectue entre n transcriptions, et le mot choisi ne prend pas en considération l'historique des précédents élus. Ceci se vérifie d'autant plus lorsque le choix s'effectue entre plusieurs mots qui ont des scores similaires. Pour résoudre ce problème ils introduisent des informations linguistiques au sein de l'algorithme de décision. Ainsi pour chaque vote, l'aspect linguistique est pris en compte. Lorsque le *ROVER* ne peut pas prendre de décision, le modèle linguistique apporte sa contribution, améliorant nettement le résultat de la combinaison.

5.5.4 *ROVER* généralisé à des réseaux de confusion (*CNC*)

[Evermann et Woodland, 2000b] présentent un *ROVER* étendu à des réseaux de confusion, considérant que le *ROVER* sur les *one-best* des systèmes présente une limite. Ils modifient l'algorithme de décision pour l'étendre au choix entre plusieurs réseaux de confusion : une probabilité est alors associée à un mot connaissant les nœuds de réseaux de confusion où il apparaît. Le score de confiance de chaque mot est constitué de la somme des probabilités *a posteriori* sur le nœud courant du réseau de confusion. Le meilleur mot sera celui maximisant la somme des probabilités *a posteriori* sur l'ensemble des nœuds :

$$w = \arg\max_{w} \sum_{i=1}^{N} P(S_i)P(w|X, S_i) \tag{5.32}$$

Cette méthode permet de généraliser le *ROVER* à l'ensemble des n hypothèses des systèmes. Cependant, les gains par rapport à un *ROVER* avec score de confiance se révèlent relativement faibles.

5.5.5 *iROVER*

Récemment, le concept du *ROVER* a été généralisé à l'ensemble de la chaîne d'un système de reconnaissance automatique de la parole. [D. Hillard, 2007] présentent un *ROVER* s'appuyant sur de nombreux critères pour effectuer son vote. Six classes de paramètres sont exploitées :

1. Une information sur le nombre d'hypothèses identiques entre les systèmes. Cette information est également associée aux scores de confiance des hypothèses.
2. Des scores de confiance étendus : le score issu des réseaux de confusion, l'entropie liée au nœud dans le réseau de confusion, le nombre de choix sur le nœud du réseau de confusion, le score linguistique, le score acoustique et des scores étendus pour les modèles linguistiques et acoustiques.
3. Des informations relatives aux durées des hypothèses pour chaque système.
4. Les erreurs les plus fréquemment générées par le système en fonction de ses contextes gauche et droit.
5. Une distance de Levenshtein entre les mots des différents systèmes afin de prendre en compte les mots hors vocabulaires. Par ailleurs, dans cette classe les scores de confiance de chaque système sont réévalués par rapport aux autres.
6. Des scores basés sur le min-fWER.

À partir de ces paramètres, les auteurs utilisent un classifieur de type adaBoost [Schapire et Singer, 2000] pour estimer des poids dépendants des jeux de paramètres. À partir de ces poids, un *ROVER* classique est appliqué. Cette méthode s'avère plus efficace qu'un *ROVER* classique quand elle s'applique à deux systèmes, mais au-delà de deux systèmes les résultats ne sont guère améliorés.

5.5.6 Combinaison par SVM

Dans leur article [Utsuro *et al.*, 2004] les auteurs proposent une méthode similaire à *iROVER*, cependant le classifieur utilisé n'est pas le même. Un classifieur de type *Support Vector Machine* est employé : la méthode cherche un hyperplan qui sépare au mieux les données au sens de la marge ; le "kernel" trick permet (si on le souhaite) de faire cette recherche dans un espace transformé dont la dimension est en général plus grande voire infinie, sans surcoût calculatoire. Les auteurs remplacent le score de confiance du mot par sa distance à son hyper-plan. Cette méthode, bien que très similaire à la précédente, se comporte très différemment. La combinaison est comparable à un *ROVER* classique jusqu'à 3 systèmes. Mais au delà, le système de combinaison par SVM devient extrêmement efficace : les auteurs combinent ainsi jusqu'à 26 systèmes. Les systèmes mis en concurrence varient essentiellement au niveau de leurs modèles acoustiques. Leurs sorties sont alignées par un algorithme de type DTW puis le vote par les SVM est effectué.

5.5.7 *SuperEARS*

SuperEARS [Woodland *et al.*, 2004, Gales *et al.*, 2006] est un projet regroupant quatre laboratoires : le LIMSI, BBN, CU et le SRI. Leur objectif est de combiner l'ensemble des systèmes avec des combinaisons explicites et implicites. La méthode de combinaison se décompose ainsi (figure 5.8) :

- Un segmenteur fourni par le LIMSI
- Une génération des treillis par le système de Cambridge (CU)
- Les systèmes BBN et LIMSI qui re-décodent les treillis tandis le système SRI réestime le treillis
- Une combinaison de l'ensemble des résultats par un *ROVER*
- Un système CU qui réévalue la sortie du *ROVER*
- Un nouveau *ROVER* qui est appliqué sur l'ensemble des sorties.

Pour une combinaison optimale, le projet *SuperEARS* utilise plusieurs segmentations. Tous les systèmes sont adaptés avec les sorties du système CU. Quant au système CU, il est réadapté avec les sorties des autres systèmes et du *ROVER*. La méthode de combinaison proposée est robuste et présente une amélioration relative du taux d'erreurs mots de 10%.

5.5.8 *BAYCOM*

[Sankar, 2005] présente une approche Bayésienne de combinaison de sorties de différents SRAP qu'il nomme *BAYCOM* et qui permet de combiner de manière optimale les systèmes, en considérant au préalable qu'ils sont indépendants. Dans cette approche, *BAYCOM* exploite les transcriptions et des scores spécifiques à chaque système pour prendre une décision finale. Contrairement aux méthodes de combinaison par *ROVER*, *BAYCOM* ne nécessite pas une normalisation des scores de confiance entre

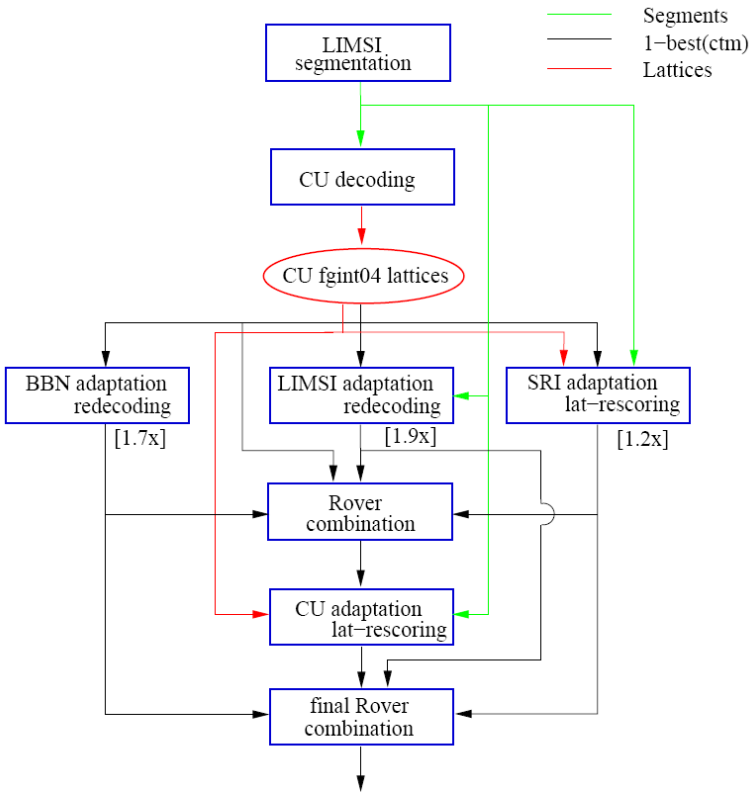

FIG. 5.8: *Schéma de fonctionnement des combinaisons au sein du projet SuperEARS [Woodland et al., 2004]*

les systèmes. La méthode de combinaison est issue de la théorie de la décision : soient M modèles qui produisent chacun des phrases x. L'hypothèse de reconnaissance de chaque modèle i est représentée par $h_i(x)$. À chaque hypothèse $h_i(x)$ est associé un ensemble de L scores désignés par $s_i^j(x)$ avec $j = 1, ..., L$. Ces scores peuvent êtres des probabilités *a posteriori*, des scores de confiance, des vraisemblances acoustiques... Le système combine l'ensemble des $h_i(x)$ représenté par $H = h1, ..., h_M$. L'évènement h est considéré comme une hypothèse correcte. La combinaison optimale sera donc :

$$h^* = \arg\max_{h \in H} P(h|h_1, ..., h_M, S_1, ..., S_M) \tag{5.33}$$

Où $S_i = s_i^1, ..., s_i^L$. En appliquant la règle de Bayes et en omettant le dénominateur, [Sankar, 2005] propose :

$$h^* = \arg\max_{h \in H} P(h) \prod_{i=1}^{M} P(S_i|h_i, h) P(h_i|h) \tag{5.34}$$

Il définit deux sous ensembles $I_C = i : h_i = h$ et $I_E = i : h_i \neq h$, et à partir de ces derniers :

$$P(h_i|h) = \begin{cases} P_i(Cor) & \text{si } i \in I_C \\ \dfrac{P_i(Err)}{N-1} & \text{si } i \in I_E \end{cases} \tag{5.35}$$

Où N est le nombre d'hypothèses en concurrence, $P_i(Cor)$ est la probabilité d'être correct indépendamment de l'hypothèse h_i et $P_i(Err) = 1 - P_i(Cor)$ la probabilité d'être incorrect répartie uniformément sur les $N-1$ hypothèses incorrectes. L'auteur définit également :

$$P(S_i|h_i, h) = \begin{cases} P(S_i|Cor) & \text{si } i \in I_C \\ P(S_i|Err) & \text{si } i \in I_E \end{cases} \tag{5.36}$$

Où $P(S_i|Cor)$ et $P(S_i|Err)$ sont les distributions conditionnelles sachant que h_i est une hypothèse correcte ou incorrecte. À partir de ces définitions, l'auteur développe l'équation 5.34 :

$$h^* = \arg\max_{h \in H} P(h) \prod_{i:h_i=h} (N-1) \frac{P_i(Cor)P_i(S_i|Cor)}{P_i(Err)P_i(S_i|Err)} \tag{5.37}$$

Et en prend le logarithme :

$$h^* = \arg\max_{h \in H} \left[\begin{array}{l} log(P(h)) \\ + \sum\limits_{i:h_i=h} log \dfrac{P_i(Cor)(N-1)}{P_i(Err)} \\ + \sum\limits_{i:h_i=h} log \dfrac{P_i(S_i|Cor)}{P_i(S_i|Err)} \end{array} \right] \tag{5.38}$$

Les paramètres $P_i(Cor)$, $P_i(S_i|Err)$, $P_i(S_i|Cor)$ doivent être estimés sur un corpus d'apprentissage. Cette approche permet de combiner de manière optimale selon un critère Bayésien ; et obtient de meilleurs résultats qu'avec un *ROVER*. Cependant l'approche décrite s'applique à des scores globaux sur les phrases et suppose que les systèmes mis en concurrence sont indépendants.

5.6 Combinaison intégrée

5.6.1 Combinaison par augmentation de l'espace de recherche

[Chen et Lee, 2006] proposent une méthode de combinaison basée sur l'espace de recherche. L'étape de combinaison s'effectue au niveau des treillis ainsi que des phonèmes. La méthode peut s'appliquer à N systèmes dont les treillis et réseaux de phonèmes sont fusionnés avec des règles relativement simples. L'avantage de cette approche est de prendre en compte l'ensemble de l'espace de recherche de tous les systèmes mis en concurrence. La formalisation de cette méthode est la suivante : soient $G_1, G_2, ..., G_N$ les graphes issus des N SRAP. En considérant deux treillis G_a et G_{a+1}, où q_a et q_{a+1} représentent un arc entre deux mots dans les treillis respectifs G_a et G_{a+1}. q_a peut s'écrire $q_a = [w_i; t_{start}, t_{end}]$ pour le mot w_i et $q_{a+1} = [w_j; \tau_{start}, \tau_{end}]$ pour un mot w_j. Par ailleurs, on peut également noter $G_a = q_a$ et $G_{a+1} = q_{a+1}$, où les deux treillis sont composés respectivement de l'ensemble des arcs q_a et q_{a+1}. À chaque arc est associé un score $s(q)$. L'égalité entre deux arcs issus de systèmes différents est alors définie :

$$q_a = q_{a+1} \text{ si } \begin{cases} w_i = w_j \\ t_{start} = \tau_{start} \\ t_{end} = \tau_{end} \end{cases} \tag{5.39}$$

Si deux arcs sont égaux, leurs scores peuvent être combinés :

$$s(q = q_a + q_{a+1}) = comb(s(q_a), s(q_{a+1})) \text{ si } q_a = q_{a+1} \tag{5.40}$$

À partir de ces équations, la combinaison de deux graphes G_a et G_{a+1} est alors définie par :

$$G_a + G_{a+1} = \{q = q_a + q_{a+1} | q_a = q_{a+1}\} \cup \{q_a | q_a \notin G_{a+1}\} \cup \{q_{a+1} | q_{a+1} \notin G_a\} \tag{5.41}$$

De la combinaison des deux treillis résulte dans un treillis où tous les arcs sont additionnés ou combinés s'ils sont égaux. L'application à N systèmes se note donc :

$$G_{final} = \sum_{i=1}^{N} G_i \tag{5.42}$$

L'étape de fusion des treillis ne pose pas de difficulté. Le principal problème est la normalisation des scores entre les treillis ou leur ré-estimation. Les auteurs proposent quatre approches pour ré-estimer les treillis ainsi combinés :

La première s'appuie sur une combinaison par consensus. Ainsi, si deux arcs q_a et q_{a+1} sont égaux : $s(q = q_a + q_{a+1}) = s(q_a) + s(q_{a+1})$ où les scores de chaque arc correspondent à la probabilité *a posteriori* $P_i(q)$ calculée par un algorithme *forward-backward*. Dans le cas où un arc q a été généré par un seul système i : $s(q) = P_i(q)$. Si cet arc a été généré par plusieurs systèmes, son score est alors : $s_{cons}(q) = \sum_{sys=i}^{I} P_{sys}(q)$. L'équation utilisée pour trouver le meilleur chemin dans le nouveau graphe est alors :

$$w^* = (q^1, q^2, ..., q^M) = \arg_{w \in G} \max_{q^k \in w} \prod_{k=1}^{M} s(q^k) \tag{5.43}$$

Cette méthode, permet de focaliser l'espace de recherche au niveau du mot et aura tendance à favoriser les meilleurs arcs par consensus entre les systèmes.

La seconde approche proposée se base sur une granularité au niveau phonétique. Elle s'appuie sur le concept MPE. Le critère EPA (*Expected Phone Accuracy*) est employé pour combiner les scores. Le critère EPA se définit ainsi : Étant donné un phonème p à un instant t, on considère tous les chemins de w dans le graphe G comme séquence de référence. p' est un phonème de la séquence de référence w qui se recouvre dans le temps avec p. Ce recouvrement est noté $r(p, p')$. Le score relatif aux recouvrements de phonèmes est alors :

$$S(p) = \sum_{w \in G} P(w|X) \max_{p' \in w} \begin{cases} -1 + 2r(p, p') \text{ si p et p' sont identiques} \\ -1 + r(p, p') \text{ si p et p' sont différents} \end{cases} \tag{5.44}$$

Ainsi pour un arc q contenant les phonèmes $p_1, p_2, ..., p_k$ le critère EPA pour cet arc sera : $S(q) = \sum_{i=1; p_i \in q}^{K} S(p_i)$. À partir de ce critère, chaque arc est ré-estimé en combinant le critère EPA avec les probabilités *a posteriori* :

$$s_{epa}(q) = S(q).P(q) \tag{5.45}$$

L'équation pour trouver le meilleur chemin reste identique à la précédente.

La troisième méthode est une combinaison des deux premières, destinée à prendre en compte l'information aussi bien au niveau phonétique qu'au niveau lexical. Le score devient :

$$s(q) = s_{cons}(q).s_{epa}(q)^{\beta} \tag{5.46}$$

Où β permet de pondérer la combinaison.

La dernière méthode proposée s'appuie sur le TFE (*Time Frame Error*). Les auteurs s'inspirent du décodage par fWER 1.5.2. Ils définissent une nouvelle estimation du score de chaque arc :

$$s_{tfe}(q) = \frac{(t_{end} - t_{start} + 1) - \sum_{q'=[w_i;t'_{start},t'_{end}]} r(q,q').P(q')}{1 + \alpha(t_{end} - t_{start})} \tag{5.47}$$

Où comme défini précédemment $q = [w_i; t'_{start}, t'_{end}]$, α est un paramètre de normalisation, q' est l'arc d'un autre treillis correspondant au même mot w_i, $r(q,q')$ est le nombre de trames qui sont communes aux deux arcs et $P(q')$ représente la probabilité *a posteriori* de q'. L'équation permettant de trouver le meilleur chemin sur le nouveau treillis devient :

$$w^* = (q^1.q^2...q^M) = \arg_{w \in W} \min_{q^k \in w} \sum_{k=1}^{M} s_{tfe}(q^k) \tag{5.48}$$

L'ensemble des expériences menées par les auteurs montre une amélioration par rapport à un *ROVER*. Les méthodes, par ordre croissant d'efficacité, sont :
- Combinaison par consensus au niveau du mot
- Combinaison par consensus au niveau des phonèmes
- Combinaison au niveau phonétique+mot
- Combinaison par minimisation des erreurs à la trame

5.6.2 Combinaison par fWER

Dans des travaux similaires à ceux de [Chen et Lee, 2006], [Hoffmeister *et al.*, 2006] proposent une méthode de combinaison qui se base sur le décodage par fWER. Ils généralisent le décodage à T graphes issus de N systèmes de reconnaissance différents. Les auteurs mettent en avant les avantages de ce type de combinaison : la règle de décision ne prend pas en compte le contexte, car elle se base uniquement sur les probabilités *a posteriori* et la combinaison est indépendante de la segmentation. Au final, ils obtiennent des résultats qui sont très similaires à ceux obtenus avec *ROVER* ou une combinaison de réseaux de confusion.

5.7 Complémentarité des systèmes et WER

Différents travaux, dont ceux de [Gales *et al.*, 2006], ont montré qu'il est nécessaire d'utiliser des systèmes qui sont complémentaires avec des taux d'erreurs mots similaires pour obtenir une combinaison efficace. Partant de ce constat, il est envisageable

de concevoir des systèmes complémentaires. [Breslin et Gales, 2006] entraînent différents modèles acoustiques afin que les systèmes les exploitant soient complémentaires. Pour estimer les modèles acoustiques, une méthode d'apprentissage discriminante est utilisée : MBR (*Minimum Bayes Risk training*). L'expression pour l'apprentissage par MBR est la suivante :

$$\mathcal{F}(\mathcal{M}) = \sum_{r=1}^{R} \sum_{H_m \in \mathcal{H}} P(H_m|\mathcal{X}_r;\mathcal{M})\mathcal{L}(H_m,\tilde{H}) \tag{5.49}$$

Où $\tilde{H}$ est l'hypothèse correcte pour la donnée $\mathcal{X}_r$, $\mathcal{H}$ est l'ensemble de toutes les hypothèses possibles et $\mathcal{M}$ est le modèle courant. $\mathcal{F}(\mathcal{M})$ est une fonction objective qui doit être minimisée et $\mathcal{L}$ représente une fonction de perte. Les auteurs proposent par exemple :

$$\mathcal{L}(H_m,\tilde{H}) = \begin{cases} 1 \text{ si } P(H|X;S) > \alpha, H_m \neq \tilde{H} \\ 0 \text{ sinon} \end{cases} \tag{5.50}$$

Les auteurs perfectionnent la méthode MBR afin de générer S modèles complémentaires. Ils réalisent cela en intégrant chaque nouveau modèle estimé au sein de la fonction objective :

$$\mathcal{F}(\mathcal{M}) = \sum_{r=1}^{R} \sum_{H_m \in \mathcal{H}} P(H_m|\mathcal{X}_r;\mathcal{M}^{(0)},...,\mathcal{M}^{(S-1)},\mathcal{M})\mathcal{L}(H_m,\tilde{H}) \tag{5.51}$$

Les auteurs proposent une modélisation considérant qu'une erreur est plus importante si un système se trompe au même endroit que les autres et moins grave dans le cas contraire. La fonction objective devient :

$$\mathcal{F}(\mathcal{M}) \sum_{r=1}^{R} \sum_{H_m \in \mathcal{H}} P(H_m|\mathcal{X}_r;\mathcal{M})\mathcal{L}(H_m,\tilde{H}|\mathcal{M}^{(0)},...,\mathcal{M}^{(S-1)}) \tag{5.52}$$

La fonction objective $\mathcal{F}(\mathcal{M})$ est estimée à partir des réseaux de confusion des différents modèles : chaque modèle génère son réseau de confusion, puis l'ensemble des réseaux est fusionné. Il est alors possible d'estimer pour chaque nœud quelles sont les hypothèses mal apprises : le modèle sera alors surentraîné sur les zones mal modélisées. Ce type de méthode permet de générer de multiples systèmes complémentaires, qui indépendamment les uns des autres sont plus mauvais qu'un modèle estimé classiquement.

Une méthode pour générer des modèles complémentaires a également été présentée par [Siohan *et al.*, 2005] qui proposent de partager les états ou paramètres entre l'ensemble des modèles acoustiques. Le principe énoncé consiste à regrouper au maximum les unités acoustiquement proches plutôt que d'essayer d'apprendre des modèles séparés sur moins de données. Cette combinaison entre modèles est réalisée en utilisant un arbre de décision qui regroupe les unités acoustiques proches via un critère MAP ou MMIE. Une fois l'arbre de décision élaboré, les décisions lors de chaque séparation sont sélectionnées aléatoirement parmi les N meilleures pour constituer des modèles acoustiques. Cette approche permet d'estimer divers modèles acoustiques dépendant du contexte et basés sur des données d'apprentissage variables. Cependant, les arbres de décision aléatoires ne garantissent pas la complémentarité des systèmes mais ils augmentent la probabilité de répartition des états ambigus dans les différents modèles. En tout état de cause, la combinaison de modèles obtenus par des arbres de décision aléatoires permet d'obtenir des gains. Par ailleurs les arbres de décision aléatoires souffrent d'une autre insuffisance : l'ordre des combinaisons optimal ne peut être prédit, ce qui implique que toutes les combinaisons doivent être testées pour trouver la meilleure.

Une alternative à cette méthode est celle proposée par [Breslin et Gales, 2007] où les décisions des arbres sont dirigées par des fonctions de poids, afin de combler les lacunes de la méthode proposée par [Siohan *et al.*, 2005]. Les auteurs ajoutent, dans cet objectif, une donnée supplémentaire au sein des arbres de décision : une statistique obtenue via un algorithme de *forward-backward* qui met en avant l'ordre des questions au sein de l'arbre de décision pour mieux classer les données. Cette méthode s'applique récursivement jusqu'à ce que les poids se stabilisent. Par ailleurs, au fur et à mesure des passes, un poids plus important est attribué aux données mal estimées. Ce poids est estimé par une fonction qui reflète la qualité du classement de la donnée. Dans leurs travaux, cette fonction s'applique sur chaque mot, et se base sur la moyenne des probabilités *a posteriori* des mots du nœud courant dans un réseau de confusion.

5.8 Conclusion sur la combinaison

Dans ce chapitre nous avons survolé les différentes approches de combinaison de SRAP. Les combinaisons peuvent s'effectuer à tous les niveaux. Elles peuvent être implicites : l'information se propage d'un système à l'autre via une adaptation croisée, où les treillis peuvent être réévalués. Elles peuvent être explicites : en modifiant directement les probabilités de certains éléments du SRAP, comme les *CNC*, le fWER ou le *ROVER*. [Hoffmeister *et al.*, 2007] présentent les principales formes de combinaisons actuelles post-acoustiques : le *ROVER*, les *CNC*, le fWER. Les travaux mettent clairement en avant la nécessité de complémentarité des systèmes, ainsi que la qualité nécessaire des scores de confiance : une combinaison efficace se base sur une sélection pertinente des informations. La combinaison pose également des problèmes de normalisation, les systèmes n'ayant pas des hypothèses évaluées uniformément, il est nécessaire de les accorder correctement entre eux. Les approches basées sur l'ensemble de l'espace de recherche des différents systèmes sont plus efficaces que celles axées uniquement sur

les sorties. Partant de ces observations, nous présentons, dans le chapitre suivant, une approche originale qui permet d'intégrer des systèmes auxiliaires au cœur de l'algorithme d'exploration.

Chapitre 6

Combinaison par décodage guidé

Sommaire

Ce chapitre présente une méthode de combinaison originale basée sur le décodage guidé. Nous proposons d'utiliser des transcriptions issues de SRAP auxiliaires pour guider un système principal. Nous présentons un ensemble de stratégies de combinaisons impliquant le décodage guidé. Les expériences montrent que DDA permet une amélioration des performances supérieures à celles obtenues par *ROVER*.

6.1 Introduction

Les stratégies généralement retenues dans le domaine de la combinaison sont celles qui sont faciles à mettre en œuvre et qui demeurent efficaces : principalement l'adaptation croisée où des systèmes se partagent leurs sorties, les méthodes basées sur des votes *a posteriori* telles que *ROVER* ou son extension à des réseaux de confusion. Dans le chapitre précédent nous avons vu que les techniques de combinaison les plus efficaces sont celles qui opèrent au niveau de l'espace de recherche. Cependant elles ne sont pas toujours évidentes à mettre en œuvre et posent des problèmes de normalisation des scores entre les systèmes.

Dans ce chapitre, nous proposons une stratégie de combinaison basée sur le décodage guidé (DDA). Cette méthode consiste à diriger l'algorithme d'exploration d'un SRAP via les transcriptions fournies par un système auxiliaire.

Nous présenterons d'abord le principe du décodage guidé par des transcriptions fournies par des systèmes auxiliaires. Nous élaborerons ensuite divers schémas de combinaison basés sur DDA ainsi que leurs évaluations. Nous conclurons en présentant une analyse de ce type d'approche.

6.2 Combinaison par décodage guidé : présentation et principe de DDA

La combinaison proposée consiste à effectuer une première passe de reconnaissance automatique de la parole, en utilisant un système auxiliaire qui propose ses meilleures hypothèses (one-best) $h_{aux} = \{w_i\}$. À chaque mot $\{w_i\}$ de h_{aux}, un score de confiance $\phi_{aux}(w_i)$ est associé. Cette information est intégrée au sein de l'algorithme de recherche du système primaire, qui aura la possibilité de réévaluer dynamiquement les probabilités linguistiques en fonction de l'hypothèse h_{aux} associée à ses scores de confiance $\phi_{aux}(w)$. Nous détaillons dans les sections suivantes les différents composants intervenant dans la combinaison DDA (figure 6.1).

6.2.1 Réévaluation à la volée du score linguistique

Afin de prendre en compte l'information provenant du système auxiliaire, la partie linguistique de la fonction d'estimation F est réévaluée par un score $\alpha(w)$ mixant :

- La similitude entre l'hypothèse courante et la transcription du système auxiliaire
- Les scores de confiance issus du système auxiliaire

Le système de reconnaissance SPEERAL génère ses hypothèses de mots au fur et à mesure que le treillis de phonèmes est exploré. La meilleure hypothèse à un temps t est étendue en fonction de la probabilité de l'hypothèse courante et de la sonde.

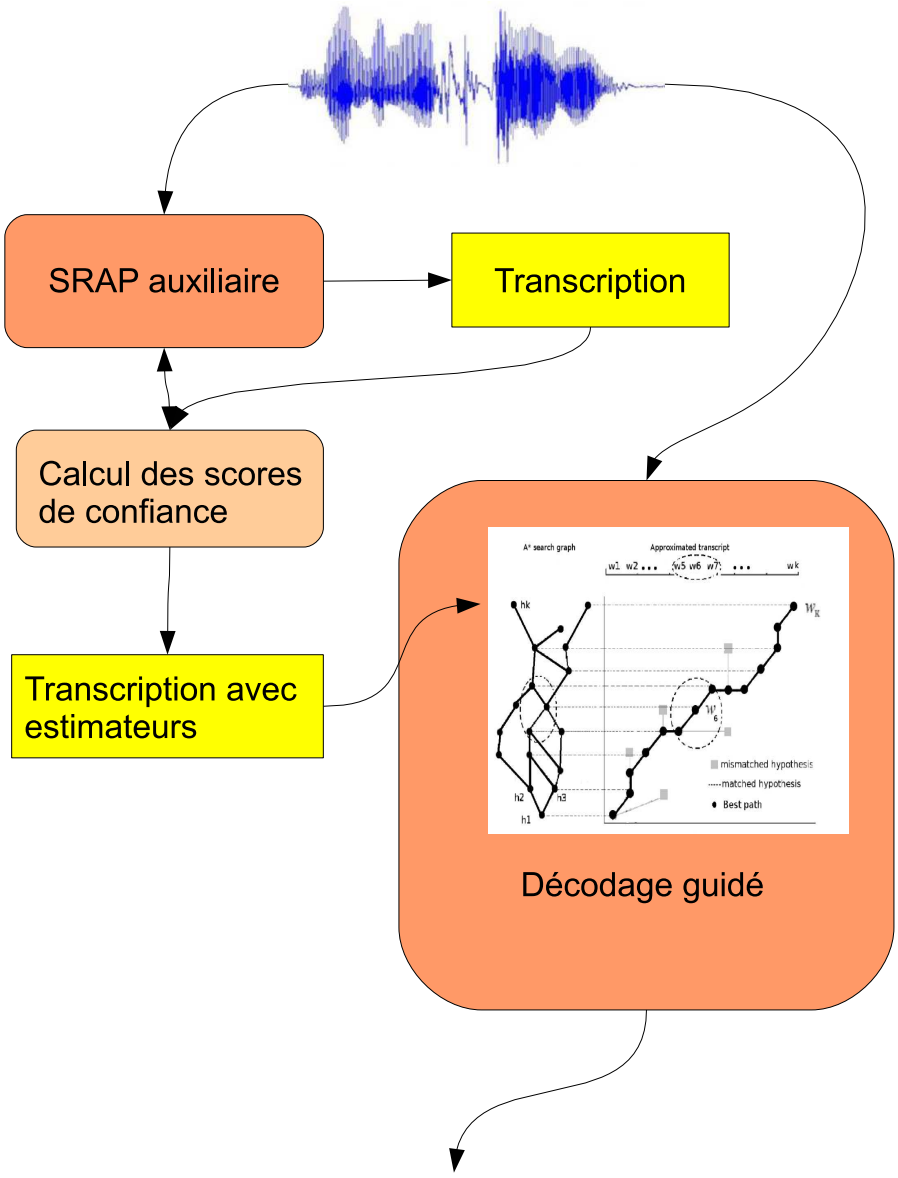

FIG. 6.1: *Principe de la combinaison par décodage guidé : La combinaison proposée consiste à effectuer une première passe de reconnaissance automatique de la parole, en utilisant un système auxiliaire qui propose ses meilleures hypothèses (one-best) $h_{aux} = \{w_i\}$. Pour chaque mot $\{w_i\}$ de h_{aux}, un score de confiance $\phi_{aux}(w_i)$ est associé. Le système auxiliaire guide le système principal en s'alignant dynamiquement et ré-estimant la fonction de coût.*

Les points de synchronisation entre l'hypothèse courante h_{cur} et la transcription auxiliaire h_{aux} sont déterminés en utilisant un algorithme d'alignement dynamique présenté dans la section 3.4.1. Lorsque l'hypothèse h_{cur} est synchronisée avec la transcrip-

tion auxiliaire h_{aux}, l'algorithme estime un score de confiance $\phi_{aux}(w_i)$. Ce score est basé à la fois sur un score de confiance local et sur le nombre de mots similaires dans un historique déterminé.

Quand ces informations ont été quantifiées, le score linguistique de l'hypothèse courante est réactualisé selon :

$$L(w_i|w_{i-2}, w_{i-1}) = P(w_i|w_{i-2}, w_{i-1})^{1-\alpha(w_i)} \tag{6.1}$$

Où $L(w_i|w_{i-2}, w_{i-1})$ est le score ré-estimée du tri-gramme (w_{i-2}, w_{i-1}, w_i) et $P(w_i|w_{i-2}, w_{i-1})$ est la probabilité initiale du tri-gramme. $\alpha(w_i)$ est le score de confiance associé au mot w_i dans la transcription fournie par le système auxiliaire.

6.2.2 Score d'alignement et transcription auxiliaire

Dans l'objectif de combinaison, nous introduisons le score de confiance issu du système auxiliaire. Ce score dépend de deux paramètres : un score d'alignement et le score de confiance du système auxiliaire. Le score d'alignement permet d'estimer la confiance entre l'hypothèse auxiliaire et l'hypothèse courante. La combinaison ne s'appuiera que sur les scores de confiance correctement alignés.

Le score de confiance α peut être défini comme une mesure de similarité entre l'hypothèse courante h_{cur} et l'hypothèse auxiliaire h_{aux}. Ce score est évalué dynamiquement lors de l'exploration du treillis de mots, en combinant le score de confiance $\phi_{aux}(w_i)$ et le nombre de mots de h_{aux} qui sont similaires à l'hypothèse courante. Le calcul de $\alpha(w)$ est effectué selon l'équation :

$$\alpha(w) = \begin{cases} \dfrac{\phi_{aux}(w_1) + \phi_{aux}(w_2) + \phi_{aux}(w_3)}{3} & \text{si } (w_1..w_3) = (hw_1..hw_3) \\ \dfrac{\phi_{aux}(w_1) + \phi_{aux}(w_2)}{2} & \text{si } (w_1, w_2) = (hw_1, hw_2) \\ \phi_{aux}(w_1) - \gamma & \text{si } w_1 = hw_1 \text{ et } \phi_{aux}(w_1) \geq \gamma \\ 0 & \text{si } w_1 \neq hw_1 \text{ ou } \phi aux(w_1) \leq \gamma \end{cases} \tag{6.2}$$

Où γ est un seuil de confiance défini *a priori*, (w_1, w_2, w_3) le trigramme de l'hypothèse courante et (hw_1, hw_2, hw_3) les trois mots de la transcription alignés sur le trigramme (w_1, w_2, w_3). Ce seuil permet de filtrer et élaguer les segments de l'hypothèse auxiliaire h_{aux} qui sont probablement de mauvaise qualité.

Lorsque le tri-gramme ou le bi-gramme est aligné, la confiance de l'hypothèse courante est estimée sur la moyenne de l'ensemble des scores de confiance : la solution est estimée suffisamment fiable pour servir d'estimateur. Par contre, si un seul mot est aligné, la confiance est plus réduite : il ne sera pris en compte que dans le cas où le système auxiliaire lui accorde une confiance supérieure au seuil γ.

6.2.3 Mesure de confiances de la transcription

Notre stratégie de combinaison s'appuie sur la transcription d'un système auxiliaire où chaque mot w_i de l'hypothèse h_{aux} est associé à un score de confiance local $\phi(w_i)$. Nos premiers travaux sur la combinaison avec DDA s'appuient sur la meilleure hypothèse fournie par le système de reconnaissance automatique du LIUM.

Le système du LIUM utilise des scores de confiance WP/LMBB. Cette mesure est une combinaison des classiques probabilités *a posteriori* avec une mesure basée sur le repli du modèle de langage (LMBB). En utilisant une entropie croisée normalisée (*Normalized Cross Entropy* : NCE) comme évaluation des mesures de confiance, les scores WP :LMBB obtiennent 0.266 sur les données utilisées dans le cadre expérimental présenté dans les sections suivantes. Ce score montre la qualité/fiabilité des scores de confiance vis-à-vis des mots générés par le système de reconnaissance.

6.2.4 Fusion des segmentations

Nous avons remarqué que les différences entre les segmentations pouvaient être sources d'erreur. Notamment lorsque des segments de parole n'ont pas été identifiés en tant que tels. Lors de nos expérimentations sur la combinaison, nous avons combiné *a posteriori* les segmentations des deux systèmes.

Cette fusion se base essentiellement sur les segments oubliés par le système primaire par rapport au système auxiliaire : dans ce cas les segments sont extraits et les scores de confiance analysés. Si les scores de confiance sont supérieurs à un seuil défini, le segment est fusionné avec l'hypothèse finale. Cette heuristique simple permet de réduire l'impact de segments manquants, et permettra de mettre en évidence les gains apportés par la combinaison DDA.

6.3 Cadre expérimental

Dans l'ensemble de nos expérimentations, le système principal est basé sur le décodeur SPEERAL et la transcription auxiliaire associée à ses scores de confiance est fournie par le laboratoire du LIUM. Ce dernier est brièvement décrit dans la section qui suit.

6.3.1 Le système du LIUM

Le système de transcription automatique du LIUM se base sur le décodeur du laboratoire CMU : Sphinx 3.3 (*fast*). Cette version du décodeur est issue de la branche du décodeur Sphinx III, elle a été développée avec l'objectif d'accroître la vitesse de calcul des algorithmes d'exploration. Le décodeur Sphinx utilise des modèles acoustiques continus basés sur des MMC à trois ou cinq états gauche-droite.

Le laboratoire du LIUM a rajouté un module d'adaptation au locuteur, un processus de ré-estimation du treillis via des quadri-grammes et ses outils de segmentation automatique. Le processus complet de décodage se décompose en deux passes : la première utilise des modèles acoustiques dépendant du genre et de la bande de fréquence de l'audio, ainsi qu'un modèle de langage tri-gramme. La seconde utilise des modèles acoustiques adaptés via la première passe et un treillis de phonèmes basé sur un modèle de langage quadri-gramme. Le processus complet de décodage s'effectue en 12x le temps réel sur une machine standard. Le système du LIUM est présenté plus en détails par [Deléglise *et al.*, 2005].

Pour l'ensemble des expériences qui suivent, les modèles acoustiques et linguistiques ont été estimés sur les même données que le système du LIA : le corpus d'apprentissage d'ESTER ainsi que le corpus issu du journal "Le Monde".

6.4 Évaluation de la combinaison par DDA

Les deux SRAP ont été évalués sur trois heures issues du corpus de développement d'ESTER : une heure de France Inter, une heure de France Info et une heure de Radio France International.

La transcription auxiliaire fournie par le système du LIUM est issue d'un processus de décodage complet tel que défini dans la section 6.3.1 et les scores de confiance associés calculés selon la méthode présentée à la section 6.2.3. Cette transcription est alors directement intégrée en utilisant le principe du décodage guidé présenté en détail en 6.2.

Nous rappelons qu'en appliquant l'équation 1.26 présentée dans le chapitre 1.6, nous estimons les améliorations significatives dès lors qu'elles sont supérieures à 0.2%.

6.4.1 Résultats expérimentaux

Les résultats de référence présentés correspondent aux transcriptions initiales des deux SRAP, dans leurs meilleures configurations. LIA-P2 et LIUM correspondent aux processus complet de décodage en deux passes pour chaque système.

Le décodage guidé est utilisé ici au cours de la seconde passe du SRAP SPEERAL (une adaptation non supervisée des modèles acoustiques étant appliquée en première passe du décodage de SPEERAL). La première passe est identique à celle utilisée lors des expériences de référence. Le résultat de ce processus - première passe normale puis DDA - a été noté "LIA-P1 DDA-P2". Le seuil de confiance γ a été fixé empiriquement à 0.4.

Le tableau 6.1 montre que la combinaison par décodage guidé permet d'obtenir une réduction significative du taux d'erreur mot (*Word Error Rate* : WER), en comparaison des résultats de référence des deux systèmes. La réduction du WER observée montre des gains absolus compris entre 0.4% et 1.9% de WER. La réduction moyenne

sur l'ensemble des heures est de 1.7% en comparaison avec le meilleur des systèmes de référence : 21.1% de WER pour le système du LIUM contre 19.4% pour la combinaison DDA.

	F. Inter	F. Info	RFI
LIA-P2 (base. LIA)	21.1	22.2	24.6
LIUM (base. LIUM)	19.5	18.8	25.4
LIA-P1 DDA-P2	**18.1** *(-1.4)*	**18.4** *(-0.4)*	**22.7** *(-1.9)*

TAB. 6.1: *Évaluation de la combinaison par décodage guidé (Driven Decoding Algorithm : DDA) (LIA-P1 DDA-P2) en terme de WER. Les résultats sont comparés à ceux obtenus par le SRAP DU LIA (LIA-P2) et le SRAP du LIUM (LIUM). Les expériences sont effectuées sur trois heures issues du corpus de développement d'ESTER*

6.4.2 Qualité de la combinaison DDA

Afin d'estimer la qualité de la combinaison des hypothèses issues des deux systèmes de reconnaissance de référence, nous avons effectué une combinaison des hypothèses, connaissant *a priori* la bonne hypothèse. Ceci permet de déterminer le WER *Oracle* en utilisant une méthode de type *ROVER* afin de fusionner les résultats des deux systèmes. De plus, un *Oracle* est calculé entre les trois systèmes : les deux systèmes de référence et le système combiné.

	F. Inter	F. Info	RFI
LIA-P2⊕LIUM	*14.9*	*13.8*	*19.5*
LIA-P2⊕LIUM⊕DDA	*13.0 (-1.9)*	*12.1 (-1.7)*	*18.8* (-0.7)

TAB. 6.2: *WER obtenu via une combinaison ROVER Oracle des sorties des systèmes de référence et ROVER Oracle de la combinaison de ces sorties avec celle de DDA.*

L'ensemble des résultats est reporté dans le tableau 6.2. Ils permettent d'évaluer le gain maximal potentiel, en admettant que l'information existe pour prendre la décision correcte. Le résultat du *ROVER Oracle* entre les trois systèmes est également très intéressant : il est plus bas que celui entre les deux systèmes de référence. Cette baisse signifie que l'approche de combinaison par DDA permet de proposer de nouvelles hypothèses de mots qui ne sont présentes dans aucune des sorties initiales des systèmes de référence (figure 6.2).

6.5 Adaptation croisée et DDA

L'adaptation croisée se révèle comme une méthode simple et efficace pour la combinaison de systèmes de reconnaissance [Prasad *et al.*, 2005]. L'utilisation la plus répandue consiste à adapter les modèles acoustiques d'un SRAP sur les transcriptions fournies par un SRAP auxiliaire. L'adaptation croisée permet d'obtenir des gains

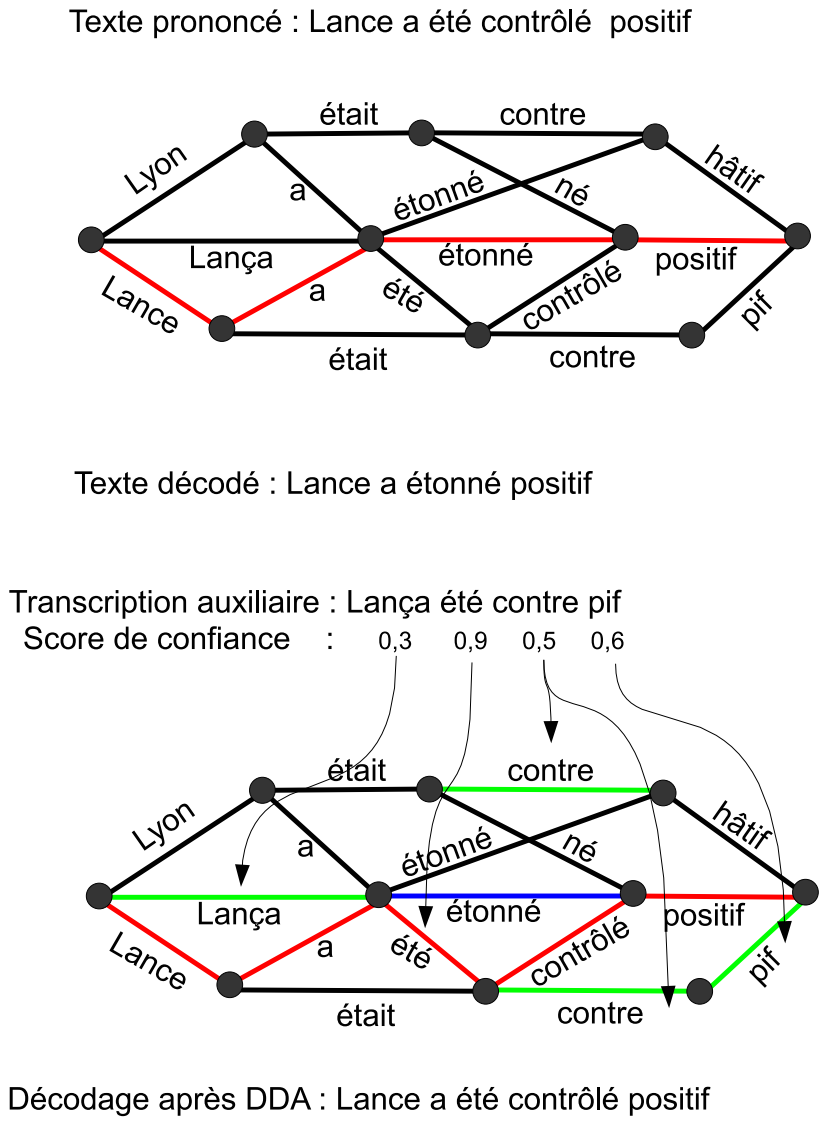

FIG. 6.2: *Exemple de combinaison par DDA : de nouvelles hypothèses sont générées. Chaque nœud est ré-estimé par le système auxiliaire et modifie l'exploration du graphe, permettant ainsi de faire apparaître des hypothèses.*

significatifs, en tirant parti de la complémentarité des deux systèmes au niveau de leur modélisation acoustique.

Nous avons donc combiné l'adaptation croisée et DDA, car ces deux méthodes opèrent à des niveaux différents et sont susceptibles d'interagir et améliorer leurs performances réciproques. Dans cette section, nous expérimentons plusieurs approches d'adaptation croisée, en exploitant les transcriptions intermédiaires fournies par le système auxiliaire (LIUM) ou par la combinaison DDA elle-même.

6.5.1 Adaptation croisée entre les systèmes de référence

Nous avons expérimenté trois configurations de référence :
- Un décodage du système primaire SPEERAL sans adaptation (LIA-P1)
- Un décodage guidé sans adaptation (DDA-P1)
- Une adaptation croisée basée sur les transcriptions h_{aux} fournies par le LIUM et le décodage en seconde passe par SPEERAL (LIUM-P1 LIA-P2)

Le tableau 6.3 récapitule les trois configurations que nous utiliserons comme références.

	F. Inter	F. Info	RFI
Meilleur système seul	19.5	18.8	24.6
LIA-P1	22.5	23.3	26.3
LIUM-P1 LIA-P2	20.4	21.8	24.1
DDA-P1	18.1	18.7	23.6

TAB. 6.3: *Adaptation croisée de référence : DDA en première passe (DDA-P1), décodage de SPEERAL après adaptation acoustique sur les transcriptions h_{aux} du LIUM (LIUM-P1 LIA P2) sans intervention du DDA.*

Sur ces premiers résultats, il apparaît clairement que le décodage opéré par DDA sans adaptation des modèles acoustiques (DDA-P1) est largement meilleur en terme de WER que les résultats obtenus par le système primaire de référence SPEERAL (-1.0% de WER). La première passe DDA offre également un taux d'erreur mot plus bas par rapport au meilleur des deux systèmes (le LIUM sur les deux premières heures, et SPEERAL sur la dernière) : -0.7%. Nous avons donc tenté d'exploiter au mieux l'information disponible entre les systèmes.

6.5.2 Adaptation croisée en première passe

Nous avons dans un premier temps testé deux stratégies d'adaptation des modèles acoustiques pour la combinaison par décodage guidé :

- Les modèles acoustiques estimés sur la transcription h_{aux} du LIUM (LIUM-P1 DDA-P2), puis un décodage avec DDA (figure 6.3)
- Une adaptation utilisant la première passe du système principal (LIA-P1 DDA-P2), puis un décodage avec DDA (figure 6.4)

Les résultats de ces adaptations sont reportés dans le tableau 6.4.

Dans le cadre de ces deux adaptations, les taux d'erreur mot sont significativement améliorés aussi bien par rapport à une adaptation croisée entre les systèmes de référence que par rapport à une combinaison DDA en une seule passe. La meilleure configuration consiste à adapter les modèles sur le système auxiliaire (LIUM) : les modèles acoustiques permettent de faire ressortir des hypothèses présentes dans la transcription qui guide le décodage, d'où ce gain important (figure 6.5).

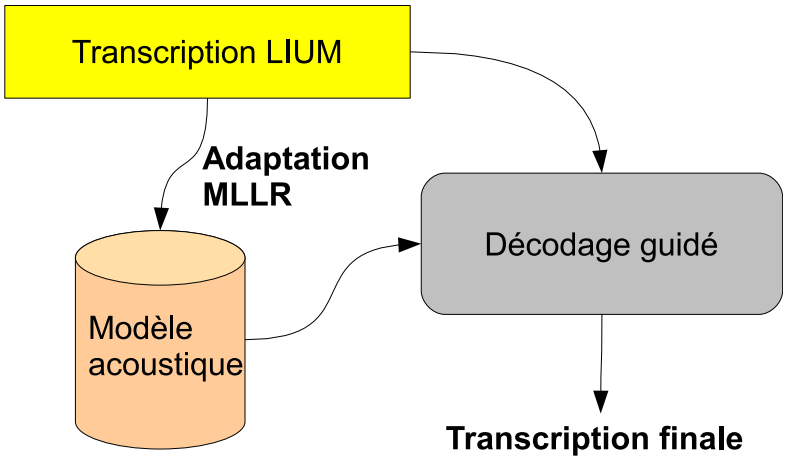

FIG. 6.3: *Adaptation croisée LIUM-P1 DDA-P2 : : les modèles du SRAP principal sont adaptés sur la transcription du LIUM. La transcription du LIUM guide également le décodage*

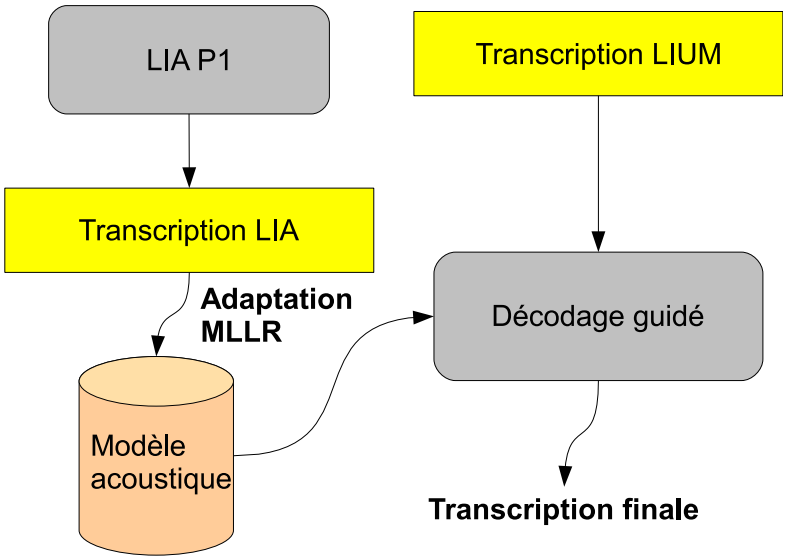

FIG. 6.4: *Adaptation croisée LIA-P1 DDA-P2 : les modèles du SRAP principal sont adaptés sur la transcription du LIA (SPEERAL). La transcription du LIUM guide le décodage en seconde passe*

Cependant, l'adaptation croisée en utilisant les transcriptions de SPEERAL en première passe, montre que les améliorations ne se cumulent pas : le gain obtenu n'est que de 0.27% de WER absolu par rapport à une combinaison DDA de base.

En combinant la combinaison DDA avec une simple adaptation croisée, nous avons réussi à obtenir un gain absolu de WER de 2.9% en comparaison du décodage de SPEERAL, et 1.9% par rapport au système du LIUM.

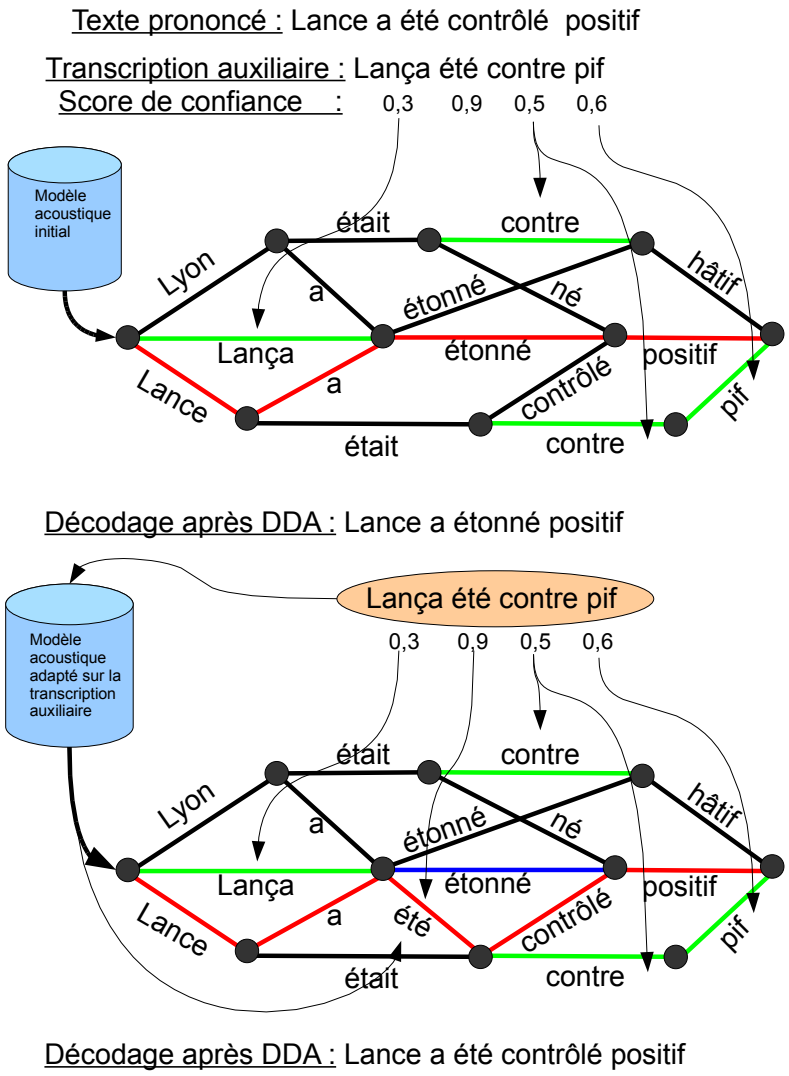

FIG. 6.5: *Influence de l'adaptation croisée sur la combinaison DDA : l'adaptation des modèles acoustiques permet au décodage guidé d'augmenter son espace de recherche.*

6.5.3 Double adaptation croisée

Finalement, nous avons testé une adaptation croisée avec deux passes basées sur une combinaison DDA. Les résultats obtenus sont similaires à une adaptation croisée

	F. Inter	F. Info	RFI
LIA-P1	22.5	23.3	26.3
LIUM-P1 LIA-P2	20.4	21.8	24.1
DDA-P1	18.1	18.7	23.6
LIA-P1 DDA-P2	18.1	18.4	22.7
LIUM-P1 DDA-P2	17.9	18.1	22.7

TAB. 6.4: *Stratégies d'adaptation croisée : adaptation sur les transcriptions du LIUM (LIUM-P1 DDA-P2), adaptation sur les transcriptions de SPEERAL (LIA-P1 DDA-P2). Les résultats sont comparés à la première passe de SPEERAL (LIA-P1), à la première passe de DDA (DDA-P1), et à une adaptation sur les transcriptions du LIUM h_{aux} sans DDA (LIUM-P1 LIA P2).*

basée sur la transcription du LIUM (tableau 6.5, figure 6.6). Ce résultat montre que la combinaison par décodage guidé est capable de modifier l'espace de recherche sans dégrader le résultat final par rapport à une adaptation directe sur la transcription originale.

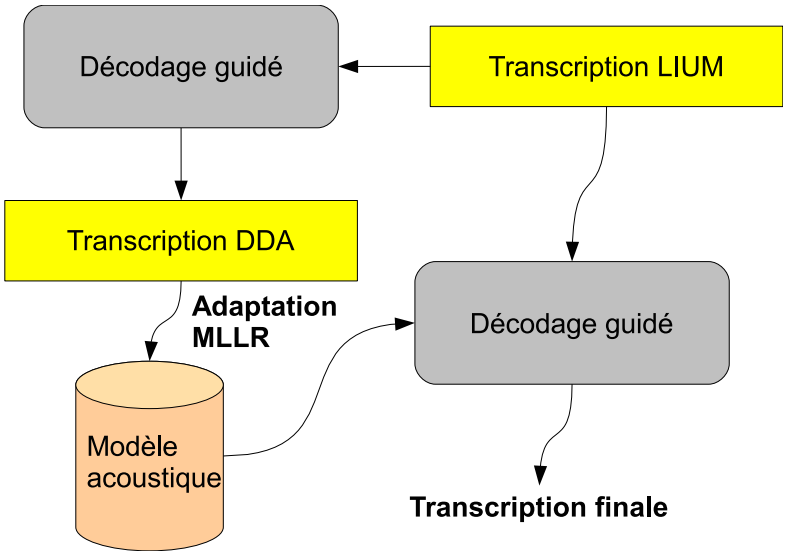

FIG. 6.6: *Adaptation croisée DDA-P1 DDA-P2 : les modèles du SRAP principal sont adaptés sur la transcription d'une première passe DDA guidée par le LIUM. La transcription du LIUM guide le décodage en seconde passe*

	F. Inter	F. Info	RFI
LIUM (base. LIUM)	19.5	18.8	25.4
DDA-P1	18.1	18.7	23.6
LIUM-P1 DDA-P2	17.9	18.1	22.7
DDA-P1 DDA-P2	17.9	18.1	22.7

TAB. 6.5: *Stratégies d'adaptation croisée : adaptation sur les transcriptions du LIUM (LIUM-P1 DDA-P2), adaptation sur les transcriptions de DDA-P1 (DDA-P1 DDA-P2)*

Cependant, ce résultat confirme les précédents : les améliorations ne se cumulent pas. En effet, la transcription fournie par DDA-P1 est de meilleure qualité que celle du LIUM, mais lors de la seconde passe les erreurs convergent vers le même résultat.

6.6 Conclusions sur la combinaison par décodage guidé

Dans ce chapitre, nous avons présenté une combinaison par décodage guidé. Cette méthode inspirée par le guidage de transcriptions imparfaites, exploite les transcriptions d'un système auxiliaire associées à leurs scores de confiance. L'exploration du graphe de recherche est guidée en réévaluant à la volée les scores linguistiques.

Les résultats expérimentaux montrent que cette approche de combinaison intégrée obtient des gains significatifs par rapport à une combinaison par adaptation croisée.

Nous avons obtenu les meilleures performances en réalisant deux passes, toutes deux basées sur DDA. Notre combinaison obtient un gain de -1.3% de WER en absolu par rapport à une adaptation croisée. En combinant une ré-estimation dynamique des probabilités linguistiques et une adaptation croisée des modèles acoustiques, cette stratégie présente un gain en WER absolu de 1.9% par rapport aux deux systèmes de référence. De plus, l'analyse d'un *ROVER Oracle* entre le DDA et les systèmes de référence montre que la combinaison DDA génère de nouvelles hypothèses qui ne sont présentes dans aucun des systèmes de référence. L'ensemble de ces travaux a été présenté dans [Lecouteux *et al.*, 2007b].

Dans le chapitre suivant, nous proposons d'étendre cette approche en exploitant des sorties de systèmes auxiliaires plus riches qu'une simple *one-best*, et en généralisant la combinaison à n SRAP.

Chapitre 7

Généralisation du décodage guidé

Sommaire

Ce chapitre introduit la suite de nos travaux sur la combinaison. Nous proposons une méthode de combinaison log-linéaire, puis nous généralisons la combinaison à n SRAP. Nous étendons la combinaison par décodage guidé par des réseaux de confusion. Nous présentons ensuite plusieurs expériences permettant d'évaluer nos stratégies.

7.1 Introduction

Précédemment, nous avons proposé et évalué une méthode permettant de combiner deux systèmes de reconnaissance. Dans ce chapitre, nous proposons une extension de cette combinaison à n systèmes ainsi qu'à un décodage guidé par des réseaux de confusion, car l'information contenue dans une seule hypothèse semble réduite. Nous tentons d'intégrer un maximum d'informations issues d'un système auxiliaire, à savoir le réseau de confusion qu'il aura produit. Nous proposons d'abord une autre approche de combinaison, puis nous investirons les diverses stratégies de combinaison inter-systèmes énoncées.

7.2 Stratégies de combinaisons linéaires et log-linéaires

Dans l'état de l'art, nous avons vu que différentes stratégies de combinaison de classifieurs pouvaient être envisagées. Nous proposons donc une combinaison log-linéaire. La fonction d'estimation a été remodelée afin de comparer deux types de combinaisons classiques : linéaire et log-linéaire. Dans ce paragraphe, la nouvelle combinaison est proposée en conservant un principe similaire ; la meilleure hypothèse à un temps t est étendue en fonction de la probabilité de l'hypothèse courante et du score de la sonde. La combinaison de l'information issue de la transcription auxiliaire H_{aux} durant le processus de recherche se fait via un alignement dynamique avec l'hypothèse courante, pour chaque mot exploré. Ce processus permet d'identifier dans la transcription auxiliaire le sous-ensemble correspondant à l'hypothèse courante h_{cur} explorée. Cette sous-séquence sera notée h_{aux} et les probabilités *a posteriori* associées à chaque mot w_i de cette sous-séquence $\phi(w_i)$.

$tf(w_i)$ est le compte du nombre de mots qui sont similaires entre h_{cur} et h_{aux}. Ce score est combiné avec les probabilités *a posteriori* pour réévaluer la probabilité linguistique en fonction de la règle suivante :

$$L(w_i|w_{i-2}, w_{i-1}) = P(w_i|w_{i-2}, w_{i-1})^{1-\beta}.\alpha(w_i)^{\beta} \tag{7.1}$$

Où $L(w_i|w_{i-2}, w_{i-1})$ est le score linguistique résultant. $P(w_i|w_{i-2}, w_{i-1})$ est la probabilité initiale du tri-gramme, β est un facteur d'échelle estimé empiriquement et $\alpha(w_i)$ est le score de confiance du mot w_i, singularisé par :

$$\begin{aligned} \text{Si } tf(w_i) > 0 \text{ alors } \alpha(w_i) = \phi(w_i).\frac{tf(w_i)}{\gamma} \quad & \text{et } \beta = \text{facteur d'échelle} \\ & \text{sinon } \beta = 0 \end{aligned} \tag{7.2}$$

Où γ est la taille de la fenêtre d'analyse où est calculée la distance d'édition (dans nos expériences, $\gamma = 4$) et $\phi(w_i)$ le score de confiance initial accordé au mot w_i du système auxiliaire.

Les résultats obtenus avec cette fonction de ré-estimation basée sur une combinaison log-linéaire sont présentés dans le tableau 7.1.

	F. Inter	F. Info	RFI
Meilleur système	19.5	18.8	24.6
DDA-lin-P1	18.1	18.7	23.6
DDA-lin-P2	17.9	18.1	22.7
DDA-log-P1	17.8	18.1	22.4
DDA-log-P2	17.2	17.8	21.5
DDA-log-P3	17.2	17.8	21.5

TAB. 7.1: *Combinaison log-linéaire : DDA en première passe (DDA-*-P1), DDA en seconde passe (DDA-*-P2) avec la version linéaire (lin) et la version log-linéaire (log) de DDA, DDA en troisième passe*

La fonction de ré-estimation basée sur une combinaison log-linéaire est beaucoup plus performante que la précédente. L'équation Bayésienne du SRAP devient donc :

$$\hat{W} = \arg\max_{w} P(W|X, Tr) \tag{7.3}$$

$$= \arg\max_{w} P(X|W)P(W)P(Tr|W) \tag{7.4}$$

Où $P(W)$ est la probabilité donnée par le modèle de langage, $P(X|W)$ est la probabilité acoustique de la suite de mots W et $P(Tr|W)$ correspond à la probabilité de la transcription connaissant l'hypothèse W. Dans cette équation, X et Tr sont considérés comme indépendants.

Différents travaux présentés dans la littérature, vis à vis de la combinaison de SRAP proposent de combiner une multitude de systèmes. Les approches de combinaison les plus efficaces sont basées sur l'intégration des espaces de recherche issus des différents systèmes ainsi que sur la quantité/complémentarité de ces systèmes. Dans ce chapitre, nous proposons plusieurs évolutions de notre stratégie de décodage guidé :

- La première consiste à étendre le décodage guidé à n systèmes
- La seconde permet d'intégrer au sein du décodage guidé, l'espace de recherche complet du système auxiliaire

Nous présenterons le cadre expérimental, puis les améliorations apportées à la combinaison DDA ainsi que sa généralisation à n systèmes. Puis dans une seconde partie, nous explorerons l'intégration de réseaux de confusion au sein de DDA.

7.3 Évolutions de DDA

Dans les travaux qui suivent, nous exploitons un décodage guidé basé sur une combinaison log-linéaire, qui permet d'obtenir des résultats plus intéressants.

7.3.1 Extension à n systèmes

À partir de l'approche de combinaison log-linéaire, nous proposons d'intégrer n sorties de systèmes auxiliaires au sein du décodage guidé. Pour chacun des mots issus des transcriptions auxiliaires h_{aux_i}, est associé un score de confiance. Chacune des sorties auxiliaires est synchronisée avec l'hypothèse courante via un alignement dynamique, puis l'ensemble des scores de confiance est fusionné par une combinaison log-linéaire :

$$L(w_i|w_{i-2}, w_{i-1}) = P(w_i|w_{i-2}, w_{i-1})^{1-\beta}.\frac{1}{n}\sum_{k=0}^{n}\alpha_k(w_i)^{\beta_k} \tag{7.5}$$

Où $\beta = \sum_{k=0}^{n}\beta_k$ correspond au facteur d'échelle, $\alpha_k(w_i)$ est le score de confiance associé au mot w_i par le système k, n est le nombre de systèmes auxiliaires. $L(_i|w_{i-2}, w_{i-1})$ est la nouvelle probabilité associée au tri-gramme (w_{i-2}, w_{i-1}, w_i)

Dans le premier chapitre nous avons présenté un décodage guidé se basant sur un seul système auxiliaire. Dans cette section nous proposons une extension qui permet de généraliser le DDA à plusieurs systèmes auxiliaires.

En reprenant le principe général de la combinaison par décodage guidé, combiner plusieurs systèmes peut s'envisager de deux manières :

- La première consiste à fusionner l'ensemble des hypothèses auxiliaires en utilisant un vote basé sur un *ROVER*. L'hypothèse résultante guidera alors le décodage.
- La seconde consiste à conserver indépendamment les hypothèses auxiliaires. Ces hypothèses sont alors intégrées, chacune séparément au sein du décodage guidé. Dans cette approche, chaque transcription auxiliaire est synchronisée avec l'hypothèse courante du décodeur. Les scores d'alignement sont indépendants, tout comme les probabilités *a posteriori* relatives aux transcriptions auxiliaires. Finalement, l'ensemble des scores est combiné pour réévaluer la partie linguistique : aucune information n'est perdue.

Nous avons testé et comparé ces deux approches sur trois configurations différentes. Finalement nous testons une dernière approche, où tous les systèmes ainsi que la sortie du DDA sont combinés par un *ROVER*.

7.3.2 Extension aux réseaux de confusion

L'information utilisée par le décodage guidé et basée sur les transcriptions de systèmes auxiliaires peut sembler restreinte. Nous abordons dans cette partie l'utilisation d'une information plus complète que la transcription des systèmes auxiliaires. Nous avons étendu l'idée en intégrant les réseaux de confusion générés par les systèmes auxiliaires (figure 7.1).

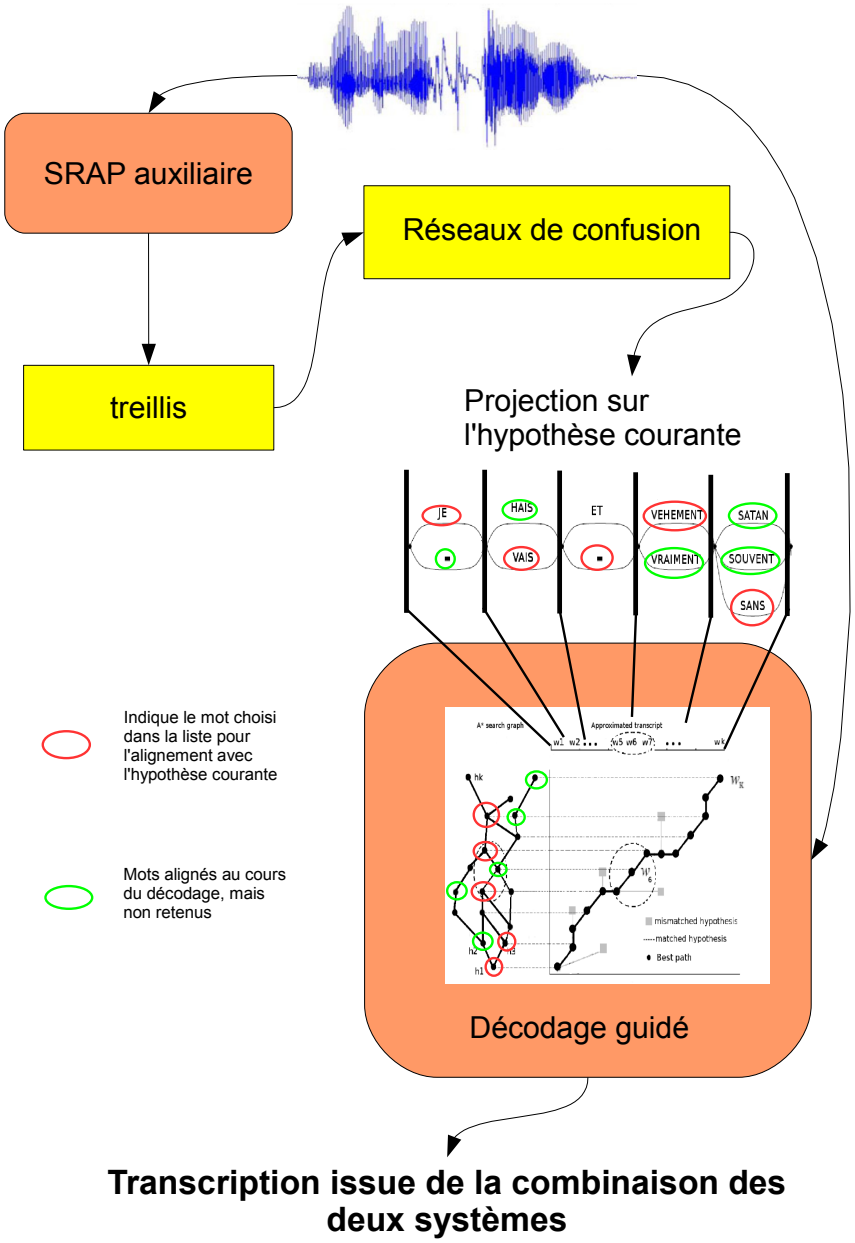

FIG. 7.1: *Décodage guidé par des réseaux de confusion : le réseau de confusion est projeté sur l'hypothèse, afin d'en extraire les probabilités a posteriori qui ré-estimeront l'hypothèse courante. Après projection, le problème revient à aligner les deux hypothèses et de ré-estimer la fonction de coût.*

Le principe est similaire à celui utilisé avec une seule transcription. La combinaison s'effectue au niveau de l'algorithme d'exploration du graphe en alignant dynamiquement l'hypothèse courante avec le réseau de confusion. Ceci est effectué en minimisant la distance d'édition entre l'hypothèse et le réseau de confusion. Cette opération est réalisée par de multiples alignements projetant le réseau de confusion sur l'hypothèse courante. Le nombre d'hypothèses de projection est réduit en supprimant temporairement du réseau de confusion tous les mots qui n'appartiennent pas à l'hypothèse courante. Après cette simplification, dans la pratique, les chemins sont encore très nombreux. Il est nécessaire que les chemins partiels soient sauvegardés et restaurés à la demande en fonction de l'historique exploré. Le temps requis par cette étape devient alors négligeable en comparaison de l'ensemble du décodage. Par ailleurs, nous avons extrait l'intervalle de temps correspondant à chacun des nœuds du réseau de confusion afin de limiter la complexité de l'alignement.

L'étape d'alignement permet d'extraire la meilleure projection de l'hypothèse sur le réseau de confusion. Cet alignement effectué, le problème est similaire à un décodage guidé par une transcription : les probabilités linguistiques sont ré-estimée en fonction d'un score d'alignement avec le réseau de confusion, ainsi qu'avec les probabilités *a posteriori* du réseau de confusion. Les probabilités *a posteriori* du réseau de confusion sont de qualité inférieure à celle des scores de confiance et nous avons observé que souvent l'hypothèse la plus probable a un score très élevé, et toutes les alternatives se partagent la probabilité restante.

7.4 Cadre expérimental

L'ensemble des travaux qui suivent ont été effectués avec trois systèmes différents : le SRAP du LIUM, le SRAP de l'IRISA et celui du LIA.

7.4.1 Corpus d'évaluation

L'ensemble des expériences a été réalisé sur trois heures extraites du corpus de développement de la campagne ESTER [Galliano *et al.*, 2005] : une heure de France Inter, une heure de France Info et une de RFI.

Le SRAP du LIA a été utilisé comme système primaire, tandis que ceux du LIUM et de l'IRISA ont fait office de systèmes auxiliaires. L'ensemble des systèmes ont été entraînés sur les mêmes ressources : le corpus d'apprentissage officiel de la campagne ESTER. Les données d'apprentissages consistent en 80 heures de parole annotée (environ 1 million de mots) et les modèles de langages ont été estimés sur 200 millions de mots extraits de sept années du journal Le Monde.

Nous décrivons brièvement le système de l'IRISA

7.4.2 Le système de transcription de l'IRISA

Irène utilise des modèles acoustiques de type MMC et un modèle de langage trigramme comprenant un vocabulaire de 64000 mots. Le système fonctionne en trois étapes auxquelles s'ajoute un processus de ré-estimation linguistique. La première étape utilise des modèles acoustiques non-contextuels avec un modèle de langage trigramme pour générer un treillis de mots. Ce dernier est réévalué avec un modèle de langage quadri-gramme et des modèles acoustiques contextuels. Un treillis est généré dans une troisième passe après une adaptation MLLR des modèles acoustiques sur les différents locuteurs. Finalement, un nouveau décodage est appliqué sur les 1000 meilleures hypothèses en combinant les scores acoustiques, linguistiques et morpho-syntaxiques [Huet *et al.*, 2007].

7.4.3 Résultats individuels

Les résultats sont reportés dans le tableau 7.2 pour chaque système auxiliaire combiné avec SPEERAL, avant et après l'adaptation des modèles acoustiques. Une stratégie en deux passes est testée après l'adaptation, basée sur la transcription du premier décodage guidé. Nous présentons également les WER (*Word Error Rate*) pour chaque système.

	F. Inter	F. Info	RFI
LIA	21.1	22.2	**24.6**
LIUM	**18.5**	**18.9**	25.6
IRISA	21.4	21.8	25.6
DDA-IRISA-P1	19.6	19.3	23.5
DDA-IRISA-P2	18.7	18.7	22.2
DDA-LIUM-P1	17.8	18.1	22.4
DDA-LIUM-P2	**17.2**	**17.8**	**21.5**

TAB. 7.2: *WER pour la combinaison DDA de SPEERAL avec le système du LIUM (DDA-LIUM) et celui de l'IRISA (DDA-IRISA) avec (P1) et sans (P2) adaptation acoustique.*

Ces résultats montrent une amélioration significative avec la combinaison des systèmes par rapport aux systèmes seuls. La combinaison avec le système du LIUM est meilleure que celle obtenue avec Irène (environ 1% de WER en absolu), qui est le système présentant le taux d'erreur le plus élevé par rapport aux deux autres. Cependant, la combinaison avec le système de l'IRISA aboutit à une réelle amélioration par rapport à la performance initiale de SPEERAL.

7.5 Résultats avec les réseaux de confusion

Nous avons utilisé le décodage guidé par les réseaux de confusion du LIUM. Deux approches ont été expérimentées pour intégrer les réseaux de confusion :

– La première n'utilise que les réseaux de confusion
– La seconde utilise à la fois la meilleure hypothèse du système auxiliaire et ses réseaux de confusion

Les résultats sont reportés dans les tableaux 7.3 et 7.5. Le tableau 7.3 présente le système primaire guidé par les réseaux de confusion seuls.

	F.Inter	F.Info	RFI
LIUM	18.5	18.9	25.6
DDA-LIUM-P1	17.8	18.1	22.4
DDA-LIUM-P2	**17.2**	**17.8**	**21.5**
DDA-WCN-LIUM-P1	17.9	18.5	23.3
DDA-WCN-LIUM-P2	**17.6**	**18.2**	**22.5**

TAB. 7.3: *WER pour le décodage guidé par les réseaux de confusion (DDA-WCN) et la one-best. Les résultats sont comparés aux systèmes seuls (LIUM) et au décodage guidé par une one-best (DDA-LIUM).*

Nous observons une amélioration significative en comparaison des deux systèmes seuls. Mais dans le cas d'un guidage par les réseaux de confusion seuls, le DDA-WCN s'avère moins efficace qu'une combinaison guidée par la meilleure hypothèse du système auxiliaire.

Dans le cas d'une combinaison guidée (tableau 7.5) à la fois par les réseaux de confusion et la *one-best*, le gain comparé à celui obtenu avec la *one-best* est négligeable (environ -0.15% de WER) pour la première passe et nul après l'adaptation locuteur.

	F.Inter	F.Info	RFI
LIUM	18.5	18.9	25.6
DDA-LIUM-P1	17.8	18.1	22.4
DDA-LIUM-P2	**17.2**	**17.8**	**21.5**
DDA-WCN-LIUM-P1	17.7	18.1	22.3
DDA-WCN-LIUM-P2	**17.2**	**17.8**	**21.5**

TAB. 7.4: *WER pour le décodage guidé par les réseaux de confusion (DDA-WCN) seuls. Les résultats sont comparés aux systèmes seuls (LIUM) et au décodage guidé par une one-best (DDA-LIUM).*

Plusieurs raisons peuvent expliquer ces résultats :

– Le décodage guidé par la *one-best* utilise à la fois les mesures de confiance et la décision prise par le système auxiliaire. Ce dernier guidant la recherche parmi les meilleures hypothèses, c'est probablement une stratégie qui écarte les mauvaises hypothèses et qui rend la combinaison plus robuste.
– Les mesures de confiance utilisées dans la *one-best* sont plus fiables que les probabilités *a posteriori* utilisées dans le réseau de confusion. Ceci est en particulier dû au fait qu'ils ont été ré-estimés avec un modèle de langage quadri-gramme. Étant

donné que le score de confiance est crucial pour ré-estimer la partie linguistique, ce point peut impacter significativement les résultats finaux.
– Les réseaux de confusion bruitent la meilleure hypothèse : la complémentarité entre les systèmes est réduite.

7.5.1 Conclusions sur l'utilisation de réseaux de confusion

L'utilisation de réseaux de confusion pour guider un système de reconnaissance, semble *a priori* être une idée séduisante augmentant l'espace de recherche. Les réseaux de confusion ont la capacité d'améliorer la qualité du décodage, mais ils aboutissent à des résultats limités à ceux d'une combinaison DDA classique. Les réseaux de confusion génèrent sans doute trop de bruit, qui oriente le système sur des hypothèses parfois incorrectes. De plus, l'information directement issue des réseaux de confusion est plus pauvre que celle contenue dans les scores de confiance. En effet, ces derniers combinent de nombreux estimateurs basés sur les probabilités *a posteriori*, le modèle de langage et ses replis ...

7.6 Résultats avec un décodage guidé généralisé

Dans cette section, nous proposons plusieurs stratégies de combinaisons basées sur DDA : la première consiste à combiner d'abord les transcriptions auxiliaires par consensus, puis introduire le résultat dans le SRAP principal. La seconde approche propose de guider le SRAP principal avec l'ensemble des transcriptions auxiliaires.

7.6.1 Combinaison à deux niveaux : ROVER-DDA

Dans cette approche nous fusionnons, dans une première étape, l'ensemble des transcriptions auxiliaires. Nous avons utilisé *ROVER* pour fusionner les systèmes du LIUM et de l'IRISA. Les scores de confiance finaux sont composés de la moyenne des scores de chacun des systèmes (figure 7.2).

	F.Inter	F.Info	RFI
LIUM	18.5	18.9	25.6
ROVER-3	17.1	18.2	22.5
2-Level DDA-ROVER	16.8	17.3	21.3

TAB. 7.5: *WER en fonction de la combinaison utilisée : la référence ROVER avec les trois systèmes (ROVER-3), la méthode DDA-ROVER (2-Level DDA-ROVER)*

La transcription obtenue est alors utilisée comme hypothèse auxiliaire, de la même façon qu'un simple décodage guidé :

$$\hat{W} = \arg\max_{W} P(X|W)P(W)\frac{1}{N}\sum_{k=0}^{N} P(Tr_k|W) \tag{7.6}$$

Les résultats obtenus via cette méthode sont présentés dans le tableau 7.6.1. Ils montrent une combinaison plus efficace qu'un *ROVER* classique ou qu'un DDA individuel.

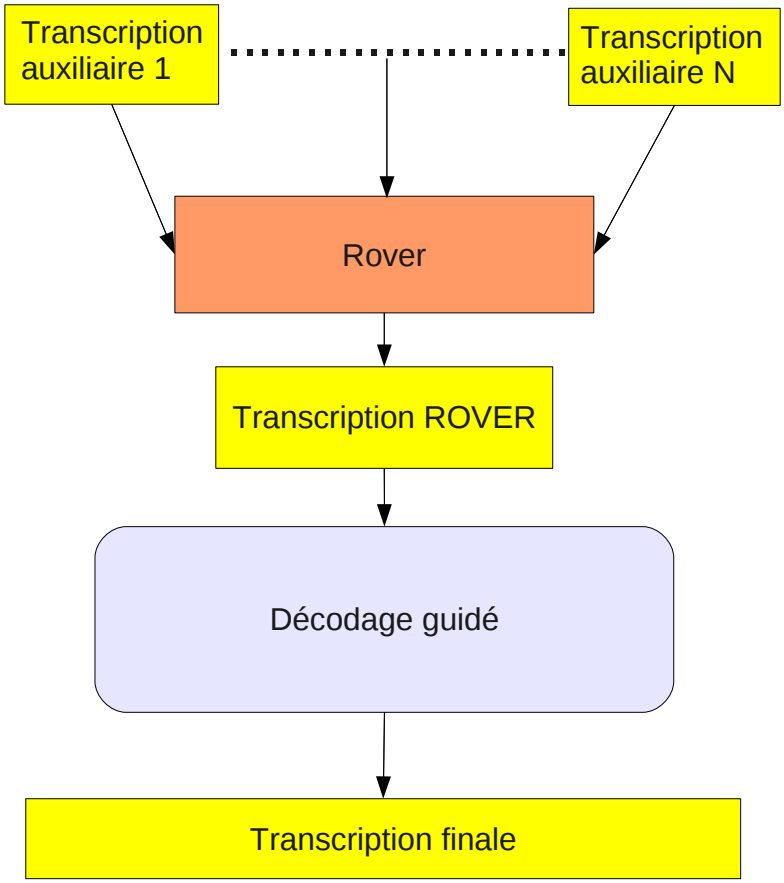

FIG. 7.2: *L'ensemble des transcriptions auxiliaires est fusionné via ROVER. La transcription obtenue est alors utilisée comme hypothèse auxiliaire*

7.6.2 Combinaison basée sur l'intégration de DDA

Dans cette méthode toutes les transcriptions auxiliaires sont soumises indépendamment à l'algorithme de recherche (figure 7.3). Un score d'alignement est calculé pour chacune et les scores linguistiques sont fusionnés via une combinaison log-linéaire étendue à n systèmes :

$$L(w_i|w_{i-2}, w_{i-1}) = P(w_i|w_{i-2}, w_{i-1})^{1-\beta}.\prod_{k=0}^{n} \alpha_k(w_i)^{\beta_k} \tag{7.7}$$

où β est la moyenne des β_k comme défini dans l'équation 7.7, α_k sont les probabilités *a posteriori* fournies par les autres systèmes k et n est le nombre de systèmes auxiliaires.

Le tableau 7.6.2 montre sur sa dernière ligne les résultats d'une combinaison DDA intégrant directement l'ensemble des trois systèmes auxiliaires : cette stratégie s'avère être la plus efficace.

	F.Inter	F.Info	RFI
LIUM	18.5	18.9	25.6
ROVER-3	17.1	18.2	22.5
2-Level DDA-ROVER	16.8	17.3	21.3
DDA-3	16.7	17.0	**20.6**

TAB. 7.6: *WER en fonction de la combinaison utilisée : la référence ROVER avec les trois systèmes (ROVER-3), la méthode DDA-ROVER (2-Level DDA-ROVER), la combinaison des 3 systèmes par DDA (DDA-3).*

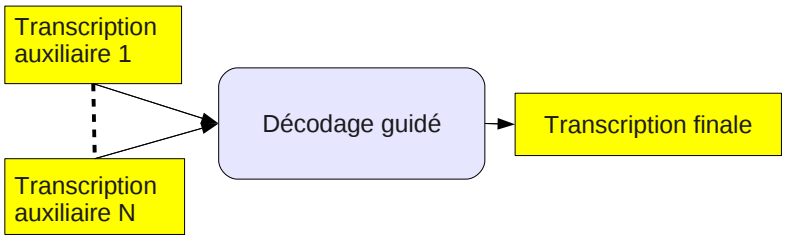

FIG. 7.3: *Combinaison intégrée : les transcriptions auxiliaires sont soumises indépendamment à l'algorithme de recherche, Un score d'alignement est calculé pour chacune et les scores linguistiques sont fusionnés via une combinaison log-linéaire.*

L'approche Bayésienne de cette combinaison devient alors :

$$\hat{W} = \arg\max_{W} P(X|W)P(W)\prod_{k=0}^{n} P(Tr_k|W) \tag{7.8}$$

Où Tr_k correspond à la transcription auxiliaire du SRAP k, $P(X|W)$ est la probabilité acoustique, $P(W)$ la partie linguistique et n le nombre de systèmes auxiliaires.

7.6.3 Analyses des résultats de DDA

Nous observons que l'adjonction d'un troisième système améliore systématiquement les performances. Cependant, le *ROVER* à 3 systèmes aboutit à un résultat identique à celui de la meilleure combinaison à 2 systèmes (-0.2% de WER en absolu). La méthode en deux passes permet d'obtenir une baisse du WER plus significative (1.1% de mieux que le DDA-LIUM), mais cette approche reste cependant moins performante que la combinaison intégrant directement les trois systèmes (un gain de 0.4% de WER).

La dernière méthode de combinaison consiste à fusionner toutes les sorties disponibles (celle du DDA comprise). Cette méthode améliore encore légèrement le système d'environ 0.3% de WER absolu.

Au final, notre meilleure configuration de combinaison permet d'améliorer le meilleur des systèmes d'environ 3.3% de WER en absolu, bien plus que la combinaison *ROVER* classique (-1.6% de WER absolu).

Afin de compléter notre analyse de la combinaison par décodage guidé nous avons réalisé quelques expériences supplémentaires.

Nous avons calculé le taux de nouvelles hypothèses, qui n'étaient présentes dans aucun des systèmes auxiliaires. Ceci a été réalisé en comparant les résultats obtenus à ceux d'un Oracle (*ORACLE DDA+ROVER*) entre tous les systèmes (*ORACLE-3*). Les résultats reportés dans le tableau 7.6.3 montrent que ré-estimer les scores linguistiques permet de guider la recherche sur d'autres chemins. Ceci confirme que le DDA n'est pas seulement un vote en ligne, mais une approche intégrée apportant une information supplémentaire à la fonction de coût qui explore le graphe.

	F.Inter	F.Info	RFI
LIUM	18.5	18.9	25.6
DDA-3	16.7	17.0	20.6
ORACLE-3	10.3	10.5	14.5
DDA-3+ROVER	16.0	16.4	20.7
ORACLE DDA+ROVER	9.8	10.0	13.6

TAB. 7.7: *Comparaisons entre le DDA, l'Oracle et le Rover.*

De plus, il est important de noter qu'une combinaison *ROVER* du DDA (tableau 7.6.3) avec tous les autres systèmes améliore encore le résultat du DDA : Ce dernier améliore de 15.7% relatifs le WER par rapport au meilleur des systèmes initiaux (LIUM). Ceci montre que le DDA trouve de nouveaux chemins corrects, mais aussi en supprime certains qui étaient présents dans les systèmes initiaux. Cette constatation suggère qu'il est encore possible d'améliorer le DDA pour qu'il prenne plus systématiquement les bonnes hypothèses trouvées dans les systèmes auxiliaires.

7.6.4 Conclusions sur décodage guidé généralisé

Dans cette section nous avons expérimenté deux approches pour généraliser le décodage guidé à n systèmes. La première consiste à effectuer un vote préliminaire entre les transcriptions auxiliaires, puis de soumettre l'hypothèse résultante au décodage guidé. La seconde approche consiste à intégrer indépendamment les unes des autres, les hypothèses issues des systèmes auxiliaires.

Les deux approches permettent d'améliorer significativement le taux d'erreur par rapport à une combinaison DDA sur chacun des systèmes. Cependant, la seconde stratégie s'avère plus efficace : en effet, lorsqu'un vote préalable est effectué entre les trans-

criptions auxiliaires, certaines hypothèses sont perdues définitivement et elles ne pourront donc pas influencer le décodage.

Par ailleurs, malgré le gain élevé de WER, les solutions sélectionnées par la combinaison DDA ne sont pas toujours optimales. En effet, un *ROVER* entre les transcriptions auxiliaires et le résultat du DDA montre un nouveau gain : des hypothèses correctes ont été écartées. Cependant, globalement le décodage guidé et généralisé à n systèmes permet d'améliorer significativement le WER. Les plus fortes améliorations sont observées sur l'heure qui était la plus mauvaise (RFI) initialement. Cet aspect montre que le décodage guidé exploite bien la complémentarité des systèmes, et qu'il est robuste. Ces travaux ont été présentés dans [Lecouteux *et al.*, 2008b] et [Lecouteux *et al.*, 2008a].

7.7 Conclusion et perspectives sur la combinaison par DDA

Dans la partie précédente, nous avons décrit un décodage guidé par des transcriptions *a priori*. Nous proposons d'étendre ce concept à une approche intégrée destinée à la combinaison de systèmes de reconnaissance de la parole. Ce modèle de combinaison est fondé sur l'intégration, dans le moteur de reconnaissance d'un système primaire, des sorties de systèmes auxiliaires.

Les transcriptions *a priori* guidant le système sont fournies par les SRAP auxiliaires et associées à leurs scores de confiance. Chacune des transcriptions auxiliaires est synchronisée avec l'hypothèse explorée courante. Une fois synchronisée, l'hypothèse courante est réévaluée en fonction des scores de confiance des transcriptions auxiliaires.

Différentes configurations ont été évaluées sur la base ESTER-2005. Les différentes stratégies envisagées ont été une combinaison simple en une passe, une combinaison simple par *ROVER*, une combinaison en deux passes avec décodage guidé et une combinaison deux passes suivie d'un *ROVER* entre tous les systèmes. Nous avons également expérimenté plusieurs adaptations croisées entre l'ensemble des systèmes. Les résultats montrent que le décodage guidé permet une réduction très sensible du taux d'erreur mot. La meilleure configuration consiste à réaliser deux passes avec décodage guidé suivies d'un *ROVER* : l'adaptation des modèles acoustiques sur la première passe permet d'introduire de nouvelles hypothèses au cours de la seconde passe. Par ailleurs, nos expériences montrent que la combinaison par DDA utilisant la meilleure hypothèse auxiliaire obtient de meilleurs résultats que le guidage par réseau de confusion : ces derniers apportent trop de bruit dans le système. Enfin, l'intégration de plusieurs systèmes auxiliaires (au lieu d'un seul) apporte un gain additionnel très substantiel et dépasse significativement la combinaison *ROVER* des 3 systèmes. Finalement en utilisant le DDA avec une dernière passe en *ROVER* nous obtenons un gain global d'environ 3.3% de WER (15.7% relatifs) par rapport au meilleur des systèmes initiaux. Ce gain observé via un *ROVER* montre donc que DDA n'est pas toujours optimal lors de sa combinaison : il est sans doute possible d'optimiser la qualité du guidage. Malgré tout, ce gain est supérieur à ceux cités dans la littérature que ce soit par des combinaisons *CNC* ou *ROVER*.

Conclusion et perspectives

Les systèmes de reconnaissance automatique de la parole souffrent d'un manque de robustesse qui limite leur champ d'application. Une solution efficace et économique à ce problème consiste à exploiter l'ensemble des informations disponibles pour améliorer le décodage, ou, plus généralement, pour augmenter la qualité des systèmes.

Nous nous sommes intéressé à l'information issue de transcriptions *a priori*, avec l'objectif de l'exploiter au mieux dans le cadre de l'amélioration de systèmes de reconnaissance. En raison de coûts grandissants liés à l'élaboration des données d'apprentissages, des techniques utilisant des transcriptions *a priori* existantes et imparfaites se sont développées ces dernières années. Les méthodes proposées n'exploitent que rarement l'ensemble des données disponibles. Les principales stratégies exploitent cette information indirectement en adaptant des modèles de langage ou des modèles acoustiques sur des segments jugés pertinents. D'autres, minoritaires, introduisent directement l'information dans le SRAP en influençant l'algorithme de recherche. Les méthodes indirectes permettent d'améliorer les modèles, quant aux directes elles améliorent la qualité du décodage. Nous avons présenté une méthode permettant d'améliorer la qualité du décodage, en s'appuyant sur des transcriptions imparfaites.

Nous proposons une formalisation qui rajoute une source d'information, comme un canal supplémentaire dans le SRAP. Nos résultats confirment que cette approche est la plus efficace : elle améliore les qualités respectives du décodage et de la transcription *a priori*. La dispersion de l'information exploitable dans la masse de données représente une difficulté supplémentaire des transcriptions *a priori*. Nous avons élaboré un algorithme, inspiré de la recherche d'information, permettant de cibler des îlots de transcription susceptibles d'être utilisés à la demande par le SRAP. L'association de cet algorithme au décodage guidé a permis d'obtenir une méthode générant des corpus de qualité correcte en utilisant des prompts et un SRAP. Contrairement aux méthodes existantes, notre approche permet de corriger les corpus disponibles imparfaits et d'augmenter ainsi la quantité de corpus correct. Cette approche apporte une solution au coût grandissant des SRAP actuels qui demandent des quantités de données annotées de plus en plus grandes pour apprendre leurs modèles. Les transcriptions *a priori* présentent un intérêt certain dans ce cadre d'utilisation et il serait intéressant d'étendre l'approche à l'exploitation de résumés ainsi qu'à des règles permettant de moduler les transcriptions *a priori* : synonymes, souplesse syntaxique...

La seconde partie de nos travaux présente une extension du décodage guidé au do-

maine de la combinaison de SRAP. La plupart des systèmes état de l'art pratiquent une combinaison opérant en aval (sur les modèles) ou en amont des systèmes, par combinaison des sorties des décodeurs. Pourtant, les moteurs de reconnaissance statistiques reposent tous sur le principe d'une évaluation des hypothèses de reconnaissance par une fonction de coût qui intègre l'ensemble des connaissances disponibles. Le développement du graphe et les heuristiques du décodage opèrent à ce niveau, et on peut penser que c'est à ce niveau que l'intégration d'une source de connaissance auxiliaire serait la plus efficace. Quelques travaux récemment présentés dans la littérature semblent confirmer cette hypothèse. Disposant des outils nécessaires pour intégrer une transcription directement dans la fonction d'estimation du SRAP, nous avons envisagé de guider ce dernier par les transcriptions issues de SRAP auxiliaires.

Nous proposons une mise en œuvre du principe du décodage guidé dans laquelle la transcription *a priori* et les scores de confiance associés sont intégrés à l'algorithme de recherche comme un canal d'information supplémentaire. Nos expériences ont montré que non seulement, le résultat de la combinaison était meilleur que les systèmes initiaux, mais que de nouvelles hypothèses apparaissent après combinaison. En modifiant l'exploration du graphe de recherche, des hypothèses correctes sont sélectionnées alors qu'elles avaient été écartées par chacun des systèmes. Ceci confirme que le principe proposé n'est pas une combinaison vote intégrée, mais bien une ré-exploration du graphe intégrant les connaissances issues des systèmes auxiliaires. Les meilleurs résultats ont été obtenus en réalisant deux passes par décodage guidé : l'information issue du premier décodage guidé biaise les modèles acoustiques, permettant lors de la seconde passe, de faire apparaître de nouvelles hypothèses dans le SRAP. Nos expériences utilisant des réseaux de confusion pour guider la reconnaissance se sont avérées décevantes mais intéressantes : l'information issue de la meilleure hypothèse *a priori* retenue est suffisante pour la combinaison, les réseaux de confusion introduisant sans doute trop de bruit. Cependant, l'intégration de réseaux ouvre des perspectives pour l'intégration d'informations auxiliaires dont la *one-best* est de moindre qualité.

Notre stratégie est simple, efficace et se mesure à des approches plus complexes. Cependant, nous avons observé que certaines hypothèses correctes proposées par les SRAP initiaux disparaissent après combinaison : la combinaison est globalement efficace, mais introduit quelques erreurs. Cet aspect montre que malgré son efficacité, la combinaison n'est pas toujours optimale. Une extension à explorer serait un DDA où chaque SRAP serait guidé par les autres, le tout dynamiquement. Il serait également intéressant d'introduire comme étape préalable, la combinaison des modèles acoustiques issus des différents systèmes afin d'augmenter l'espace de recherche du SRAP qui explore le graphe.

En conclusion, nous avons présenté une méthode permettant d'exploiter tout type de transcription *a priori*. L'originalité de notre stratégie réside principalement dans l'intégration, au niveau de la fonction de coût, des informations issues des transcriptions *a priori*. La stratégie s'intègre dans de nombreux contextes tels que l'amélioration du décodage, la fabrication automatique de corpus et la combinaison de systèmes.

Perspectives d'applications

Dans ces perspectives, nous présentons l'ensemble des applications pratiques résultant de nos travaux. Le décodage guidé est extrêmement facile à mettre en œuvre, et s'adapte à de nombreux domaines.

Le DDA a été utilisé dans le cadre du projet AVISON. L'un des objectifs de ce projet est d'indexer une base de données audio spécialisée dans le milieu médical. Des annotations sont associées à l'ensemble des données audio. Elles sont dépourvues d'informations temporelles et très proches du signal audio. Le décodage guidé est parfaitement adapté à ces conditions, et a permis de transcrire efficacement le contenu audio.

Le décodage guidé a également été intégré dans le cadre d'un système de détection de termes/expressions dans de larges corpus audio. L'application présentée dans [Rouvier *et al.*, 2008] fonctionne sur niveaux :

- La première étape consiste en un filtre phonétique basé sur la requête, qui élimine un tiers du document.
- La seconde utilise le décodage guidé par les termes et expressions pour identifier la requête. Le décodage guidé permet de propager efficacement le contexte lié à la requête et de faire ressortir les mots plus facilement.

Cette application montre que DDA est une alternative pour introduire des expressions régulières au sein d'un SRAP.

Le décodage guidé a également été utilisé dans le cadre de corrections de transcriptions générées par un SRAP. Le SRAP génère ses transcriptions et l'utilisateur y localise des erreurs. La correction est introduite via le décodage guidé dans le SRAP : ainsi une grande partie des corrections s'effectue par la modification directe de la fonction d'estimation du décodeur.

Il est envisageable d'utiliser nos travaux pour créer des systèmes d'aide à la transcription : sous-titrage automatique dans le cadre du théâtre, transcription automatique de pièces ou de cours pour des personnes présentant des problèmes d'audition...

À plus long terme, nous souhaiterions généraliser le décodage guidé à un cadre plus général, permettant d'intégrer des transcriptions issues de sources très hétérogènes. Par ailleurs, un projet est en perspective, pour élaborer un décodage guidé appliqué et synchronisé simultanément sur n systèmes. Cette approche permettrait de combiner des méthodes d'exploration opposées (en largeur et en profondeur).

Liste des illustrations

Liste des tableaux

Bibliographie

[Ahmed, 2005] Farhan AHMED (2005). *Pruning algorithm to reduce the search space of the Smith-Waterman algorithm & Kernel extensions to the uC/OS-II Real-Time Operating System*. Thèse de doctorat, Department of Electrical and Computer Engineering Lafayette College, Easton, PA.

[Anni R. Coden, 2002] Savitha Srinivasan ANNI R. CODEN, Eric W. Brown (2002). « Clustering of imperfect transcripts using a novel similarity measure ». *Information Retrieval Techniques for Speech Applications*, 1:23–34.

[Aubert, 2002] Xavier L. AUBERT (2002). *An overview of decoding techniques for large vocabulary continuous speech recognition*, volume vol. 16 no 1. Academic Press.

[Bacchiani *et al.*, 2006] M. BACCHIANI, M. RILEY, B. ROARK et R. SPROAT (2006). « Map adaptation of stochastic grammars ». *Computer speech & language*, 20(1):41–68.

[Bahl *et al.*, 1986] L. BAHL, L. BAHL, P. BROWN, P. SOUZA, de et R. MERCER (1986). « Maximum mutual information estimation of hidden markov model parameters for speech recognition ». *In* P. BROWN, éditeur : *Proc. IEEE International Conference on ICASSP '86. Acoustics, Speech, and Signal Processing*, volume 11, pages 49–52.

[Barras, 1996] Claude BARRAS (1996). *Reconnaissance de la parole continue : adaptation au locuteur et controle temporel dans les modèles de Markov cachés*. Thèse de doctorat, Université de Paris VI.

[Barras *et al.*, 2004] C. BARRAS, G. ADDA, M. ADDA-DECKER, B. HABERT, P. Boula MAREUIL, de et P. PAROUBEK (2004). « Automatic audio and manual transcripts alignment, timecode transfer and selection of exact transcripts ». *In LREC*.

[Bechet, 2001] Frederic BECHET (2001). *LIA_PHON : un système complet de phonétisation de textes*. TAL volume 42 numero 1.

[Bonastre *et al.*, 2005] J.-F. BONASTRE, F. WILS et S. MEIGNIER (2005). « Alize, a free toolkit for speaker recognition ». *In ICASSP'05*, Philadelphia, USA.

[Bourlard *et al.*, 1996] H. BOURLARD, S. DUPONT et C. RIS (1996). « Multistream speech recognition ». Rapport technique, IDIAP.

[Breslin et Gales, 2006] C. BRESLIN et M.J.F. GALES (2006). « Generating complementary systems for speech recognition ». *In ICSLP*.

[Breslin et Gales, 2007] C. BRESLIN et M.J.F. GALES (2007). « Complementary system generation using directed decision trees ». *In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, volume 4, pages IV–337–IV–340.

[Caillet *et al.*, 2007] Marc CAILLET, Jean CARRIVE, Cécile BOISIN et Francois YVON (2007). « Engineering multimedia applications on the basis of multi-structured descriptions of audiovisual contents ». *In International Workshop On Semantically Aware Document Processing And Indexing (SADPI), Montpellier, France*.

[Cardinal *et al.*, 2005] P. CARDINAL, G. BOULIANNE et M. COMEAU (2005). « Segmentation of recordings based on partial transcriptions ». *In INTERSPEECH*.

[Chan et Woodland, 2004] H.Y. CHAN et P.C. WOODLAND (2004). « Improving broadcast news transcription by lightly supervised discriminative training ». *In International Conference of Speech and Language processing*. Cambridge University Engineering Dept.

[Chen et Lee, 2006] I-Fan CHEN et Lin-Shan LEE (2006). « A new framework for system combination based on integrated hypothesis space ». *In Interspeech'06-ICSLP*, Pittburgh, Pensylvania, USA.

[Chen *et al.*, 2004a] Langzhou CHEN, Jean-Luc GAUVAIN, Lori LAMEL et Gilles ADDA (2004a). « Dynamic language modeling for broadcast news ». *In ISCA*. CNRS-LIMSI.

[Chen et Huang, 1999] L. CHEN et T. HUANG (1999). « An improved map method for language model adaptation ». *In Proceedings of European Conference on Speech Communication and Technology. Budapest, Hongrie*, volume vol. 5, pages 1923–1926.

[Chen *et al.*, 2004b] Langzhou CHEN, L. LAMEL et J.L. GAUVAIN (2004b). « Lightly supervised acoustic model training using consensus networks ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, volume 1, pages I–189–92.

[Chih-wei, 2003] Huang CHIH-WEI (2003). « Automatic closed caption alignment based on speech recognition transcripts ». Rapport technique, Columbia.

[Clarkson et Robinson, 1997] P.R. CLARKSON et A.J. ROBINSON (1997). « Language model adaptation using mixtures and an exponentially decaying cache ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP-97*, volume 2, pages 799–802 vol.2.

[Cormen *et al.*, 2001] T. CORMEN, C. LEISERSON, R. RIVEST et C. STEIN (2001). *Introduction to algorithms*. MIT Press.

[D. Hillard, 2007] M. Ostendorf R. Schluter H. Ney D. HILLARD, B. Hoffmeister (2007). « irover : Improving system combination with classification ». *In HLT*.

[Darrock et Ratcliff, 1972] J.N. DARROCK et D. RATCLIFF (1972). « Generalized iterative scaling for log-linear models ». *The Annals of Mathematical Statistics*, 5:1470–1480.

[Deléglise *et al.*, 2005] P. DELÉGLISE, Y. ESTÉVE, S. MEIGNIER et T. MERLIN (2005). « The lium speech transcription system : a cmu sphinx iii-based system for french broadcast news ». *In Interspeech'05-Eurospeech*, Lisbon, Portugal.

[Dempster *et al.*, 1977] A. P. DEMPSTER, N. M. LAIRD et D. B. RUBIN (1977). « Maximum likelihood from incomplete data via the em algorithm ». *Journal of the Royal Statistical Society. Series B (Methodological)*, 39:1–38.

[Ellis, 2000] D. ELLIS (2000). « Feature stream combination before and or after the acoustic model ». Rapport technique, ICSI.

[Estève, 2002] Yannick ESTÈVE (2002). *Intégration de sources de connaissances pour la modélisation stochastique du langage appliquée à la parole continue dans un contexte de dialogue oral homme-machine*. Thèse de doctorat, Laboratoire Informatique d'Avignon.

[Evermann et Woodland, 2000a] G. EVERMANN et P.C. WOODLAND (2000a). « Large vocabulary decoding and confidence estimation using word posterior probabilities ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP '00*, volume 3, pages 1655–1658 vol.3.

[Evermann et Woodland, 2000b] G. EVERMANN et P.C. WOODLAND (2000b). « Posterior probability decoding, confidence estimation, and system combination ». *In In Proceedings NIST Speech Transcription Workshop,College Park P. (2000a).*

[Federico, 1996a] Marcello FEDERICO (1996a). « Bayesian estimation methods for n-gram language model adaptation ». *In ICSLP*, pages 240–243.

[Federico, 1996b] Marcello FEDERICO (1996b). « Efficient language model adaptation through mdi estimation ». *In ICSLP*.

[Federico et NicolaBertoldi, 2001] Marcello FEDERICO et NICOLABERTOLDI (2001). « Broadcast news lm adaptation using comtemporary texts ». *In Eurospeech - Scandinavia*. ITC-irst - Centro per la Ricerca Scientifica e Tecnologica, Italy.

[Fiscus et Fiscus, 1997] J.G. FISCUS et J.G. FISCUS (1997). « A post-processing system to yield reduced word error rates : Recognizer output voting error reduction (rover) ». *In Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 347–354.

[Fu et Du, 2005] Yuewen FU et Limin DU (2005). « Combination of multiple predictors to improve confidence measure based on local posterior probabilities ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, volume 1, pages 93–96.

[Gales, 1997] M.J.F. GALES (1997). « Maximum likelihood linear transformations for hmm-based speech recognition ». *In CUED*.

[Gales *et al.*, 2006] M.J.F. GALES, Do Yeong KIM, P.C. WOODLAND, Ho Yin CHAN, D. MRVA, R. SINHA et S.E. TRANTER (2006). « Progress in the cu-htk broadcast news transcription system ». *IEEE Transactions on Audio, Speech, and Language Processing*, 14(5):1513–1525.

[Gales *et al.*, 2007] M.J.F. GALES, X. LIU, R. SINHA, P.C. WOODLAND, K. YU, S. MATSOUKAS, T. NG, K. NGUYEN, L. NGUYEN, J.-L. GAUVAIN, L. LAMEL et A. MESSAOUDI (2007). « Speech recognition system combination for machine translation ». *In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, volume 4, pages IV–1277–IV–1280.

[Galliano *et al.*, 2005] S. GALLIANO, E. GEOFFROIS, D. MOSTEFA, K. CHOUKRI, J.-F. BONASTRE et G. GRAVIER (2005). « The ester phase ii evaluation campaign for the rich transcription of french broadcast news ». *In Proc. of the European Conf. on Speech Communication and Technology*.

[Gao *et al.*, 2000] Yuqing GAO, Bhuvana RAMABHADRAN et Michael PICHENY (2000). « New adaptation techniques for large vocabulary continuous speech recognition ». *In ASR2000*.

[Gauvain et Lee, 1994] J.-L. GAUVAIN et Chin-Hui LEE (1994). « Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains ». 2(2): 291–298.

[Gotoh et Renals, 1999] Yoshihiko GOTOH et Steve RENALS (1999). « Topic-based mixture language modelling ». *Natural Language Engineering*, 5:355–375.

[Haubold *et al.*, 2007] Alexander HAUBOLD, Alexander HAUBOLD et John R. KENDER (2007). « Alignment of speech to highly imperfect text transcriptions ». *In* John R. KENDER, éditeur : *Proc. IEEE International Conference on Multimedia and Expo*, pages 224–227.

[Hazen, 2006] Timothy J. HAZEN (2006). « Automatic alignment and error correction of human generated transcripts for long speech recordings ». *In INTERSPEECH, paper 1258-Wed1CaP.2.*

[Heigold *et al.*, 2005] G. HEIGOLD, W. MACHEREY, R. SCHLUTER et H. NEY (2005). « Minimum exact word error training ». *In Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 186–190.

[Hermansky et Cox, 1991] H. HERMANSKY et Jr. COX, L.A. (1991). « Perceptual linear predictive (plp) analysis resynthesis technique ». *In IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics Final Program and Paper Summaries*, pages 037–038.

[Hoffmeister *et al.*, 2007] B. HOFFMEISTER, D. HILLARD, S. HAHN, R. SCHLUTER, M. OSTENDORF et H. NEY (2007). « Cross-site and intra-site asr system combination : Comparisons on lattice and 1-best methods ». *In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, volume 4, pages IV–1145–IV–1148.

[Hoffmeister *et al.*, 2006] B. HOFFMEISTER, T. KLEIN, R SCHLUTER et H. NEY (2006). « Frame based system combination and a comparison with weighted rover and cnc ». *In ICSLP*.

[Hsu, 2007] Bo-June (Paul) HSU (2007). « Generalized linear interpolation of language models ». *In ASRU*.

[Huet *et al.*, 2007] Stéphane HUET, Guillaume GRAVIER et Pascale SÉBILLOT (2007). « Morphosyntactic processing of n-best lists for improved recognition and confidence measure computation ». *In Interspeech*.

[Iyer et Ostendorf, 1999] R.M. IYER et M. OSTENDORF (1999). « Modeling long distance dependence in language : topic mixtures versus dynamic cache models ». 7(1):30–39.

[Iyer *et al.*, 1994] R IYER, M OSTENDORF et R ROHLICEK (1994). « An improved language model using a mixture of markov components ». *In Proceedings of the ARPA Workshop on Human Language Technology*.

[James, 1995] David Anthony JAMES (1995). *The application of Classical Information Retrieval Techniques to Spoken Documents*. Thèse de doctorat, Cambridge.

[Jang *et al.*, 1999] Photina Jaeyun JANG, Photina Jaeyun JANG et A.G. HAUPTMANN (1999). « Improving acoustic models with captioned multimedia speech ». *In* A.G. HAUPTMANN, éditeur : *Proc. IEEE International Conference on Multimedia Computing and Systems*, volume 2, pages 767–771 vol.2.

[Jelinek, 1969] F. JELINEK (1969). « A fast sequential decoding algorithm using a stack ». *IBM J. Res. Develop.*, 13.

[Jelinek, 1976] F. JELINEK (1976). « Continuous speech recognition by statistical methods ». 64(4):532–556.

[Jelinek et Mercer, 1980] F. JELINEK et R.L. MERCER (1980). « Interpolated estimation of markov source parameters from sparse data ». *In Workshop on Pattern Recognitionin Practice*, page 381.

[Jiang, 2005] H. JIANG (2005). « Confidence measures for speech recognition : A survey ». *In Speech Communication 45*, pages 455–470.

[Kemp Thomas, 1998] Waibel Alex KEMP THOMAS (1998). « Unsupervised training of a speech recognizer using tv broadcasts ». *In Fifth International Conference on Spoken Language Processing - ISCA*.

[Keogh et Pazzani, 2001] E. KEOGH et M. PAZZANI (2001). « Derivative dynamic time warping ». *In SIAM International Conference on Data Mining, Chicago*.

[Kirchhoff, 1998] K. KIRCHHOFF (1998). « Combining articulatory and acoustic information for speech recognition in noise and reverberant environments ». *In International Conference on Spoken Language Processing, Interspeech, Sydney, Australia*, pages 891–894.

[Kittler *et al.*, 1998] J. KITTLER, M. HATEF, R.P.W. DUIN et J. MATAS (1998). « On combining classifiers ». 20(3):226–239.

[Klakow, 1998] Dietrich KLAKOW (1998). « Log-linear interpolation of language models ». *In ICSLP*.

[Knesser et Steinbiss, 1993] R. KNESSER et V. STEINBISS (1993). « On the dynamic adaptation of stochastic language models ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP-93*, volume 2, pages 586–589 vol.2.

[Kuhn *et al.*, 1990] R. KUHN, R. KUHN et R. DE MORI (1990). « A cache-based natural language model for speech recognition ». 12(6):570–583.

[Lamel *et al.*, 2002] L. LAMEL, J.L. GAUVAIN et G. ADDA (2002). « Lightly supervised and unsupervised acoustic models training ». *Computer Speech and Language*, 16:115–229.

[Lamel *et al.*, 2001] L. LAMEL, L. LAMEL, J.L. GAUVAIN et G. ADDA (2001). « Investigating lightly supervised acoustic model training ». *In* J.L. GAUVAIN, éditeur : *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)*, volume 1, pages 477–480 vol.1.

[Lau *et al.*, 1993] R. LAU, R. ROSENFELD et S. ROUKOS (1993). « Trigger-based language models : a maximum entropy approach ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP-93*, volume 2, pages 45–48.

[Lecouteux *et al.*, 2006] Benjamin LECOUTEUX, Georges LINARES, J.F. BONASTRE et Pascal NOCERA (2006). « Imperfect transcript driven speech recognition ». *In InterSpeech'06*.

[Linarès *et al.*, 2005a] G. LINARÈS, D. MASSONIÉ, P. NOCÉRA et C. LÉVY (2005a). « A scalable system for embedded large vocabulary continuous speech recognition ». *In DSP Workshop*.

[Linarès *et al.*, 2007] G. LINARÈS, D. MASSONIÉ, P. NOCÉRA et C. LÉVY (2007). « A scalable system for embeded large vocabulary continuous speech recognition ». *In IEEE Workshop on DSP in Mobile and vehicular systems*.

[Linarès *et al.*, 2005b] G. LINARÈS, P. NOCÉRA, D. MATROUF, F. BÉCHET, D. MASSONIÉ et C. FREDOUILLE (2005b). « Le système de transcription du lia pour ester-2005 ». *In Workshop ESTER-2005*, Avignon.

[Lo et Soong, 2005] Wai Kit LO et F.K. SOONG (2005). « Generalized posterior probability for minimum error verification of recognized sentences ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, volume 1, pages 85–88.

[Lo *et al.*, 2004] Wai Kit LO, F.K. SOONG et S. NAKAMURA (2004). « Generalized posterior probability for minimizing verification errors at subword, word and sentence levels ». *In Proc. International Symposium on Chinese Spoken Language Processing*, pages 13–16.

[Mangu Lidia, 2000] STOLCKE Andreas MANGU LIDIA, BRILL Eric (2000). « Finding consensus in speech recognition : word error minimization and other applications of confusion networks ». *Computer speech & language (Comput. speech lang.) ISSN 0885-2308 Computer speech and language*, 14 n4:373–400.

[Markel et Jr., 1976] J. D. MARKEL et A. H. Gray JR. (1976). « Linear prediction of speech ». *In ommunication and Cybernetics. Berlin Heidelberg New York : Springer-Verlag*.

[Massonié *et al.*, 2005] D. MASSONIÉ, P. NOCÉRA et G. LINARÈS (2005). « Scalable language model look-ahead for lvcsr ». *In InterSpeech'05, Lisboa, Portugal*.

[Mauclair, 2006] Julie MAUCLAIR (2006). *Mesures de confiance en traitement automatique de la parole et applications*. Thèse de doctorat, LIUM.

[Mauclair *et al.*, 2006] Julie MAUCLAIR, Yannick ESTÈVE, S. PETIT-RENAUD et Paul DELÉGLISE (2006). « Automatic detection of well recognized words in automatic speech transcription ». *In LREC 2006*, Genoa, Italy.

[Mirghafori et Morgan, 1998] N. MIRGHAFORI et N. MORGAN (1998). « Transmissions and transitions : a study of two common assumptions in multi-band asr ». *In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 2, pages 713–716.

[Misra *et al.*, 2003a] H. MISRA, H. BOURLARD et V. TYAGI (2003a). « New entropy based combination rules in hmm/ann multi-stream asr ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, volume 2, pages II–741–4.

[Misra *et al.*, 2003b] Hemant MISRA, Herv E BOURLARD et Vivek TYAGI (2003b). « New entropy based combination rules in hmm/ann multi-stream asr ». *In in Proc. ICASSP 2003, Hong Kong*, pages 741–744.

[Mohri, 2002] M. MOHRI (2002). « Edit-distance of weighted automata ». *In CIAA*.

[Moissinac *et al.*, 2004] Jean-Claude MOISSINAC, Francois YVON et Slim Ben HAZEZ (2004). « Automating indexing of classes and conferences ». *In RIAO, Avignon*.

[Moreno *et al.*, 2001] P. MORENO, B. LOGAN et B. RAJ (2001). « A boosting approach for confidence scoring ». *In Interspeech, Aalborg, Denmark*, pages 2109–2112.

[Moreno *et al.*, 1998] Pedro J. MORENO, Chris JOERG, Jean-Manuel Van THONG et Oren GLICKMAN (1998). « A recursive algorithm for the forced alignment of very long audio segments ». *In International Conference on Spoken Language Processing*. Cambridge Research Laboratory and Compaq Computer Corporation.

[Ney *et al.*, 1992] H. NEY, R. HAEB-UMBACH, B.-H. TRAN et M. OERDER (1992). « Improvements in beam search for 10000-word continuous speech recognition ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP-92*, volume 1, pages 9–12 vol.1.

[Nocera *et al.*, 2004] P. NOCERA, C. FREDOUILLE, G. LINARES, D. MATROUF, S. MEIGNIER, J.-F. BONASTRE, D. MASSONIÉ et F. BÉCHET (2004). « The lia's french broadcast news transcription system ». *In SWIM : Lectures by Masters in Speech Processing*, Maui, Hawaii.

[Nocera *et al.*, 2002a] Pascal NOCERA, Georges LINARES et Dominique MASSONIÉ (2002a). « Principes et performances du décodeur parole continue speeral ». *In XXI-Vées journées d'étude sur la parole*. Laboratoire Informatique d'Avignon.

[Nocera *et al.*, 2002b] Pascal NOCERA, Georges LINARES et Dominique MASSONIÉ (2002b). « Phoneme lattice based a* search algorithm for speech recognition ». *In TSD*. Laboratoire Informatique d'Avignon.

[Ortmanns et Ney, 2000] S. ORTMANNS et H. NEY (2000). « The time-conditioned approach in dynamic programming search for ». 8(6):676–687.

[Paul, 1991] D.B. PAUL (1991). « Algorithms for an optimal a* search and linearizing the search in the stack decoder ». *In Proc. International Conference on Acoustics, Speech, and Signal Processing ICASSP-91*, pages 693–696 vol. 1.

[Paulik *et al.*, 2005] M. PAULIK, C. FUGEN, S. STUKER, T. SCHULTZ, T. SCHAAF et A. WAIBEL (2005). « Document driven machine translation enhanced asr ». *In ICSLP*.

[Placeway et Lafferty, 1996] P. PLACEWAY et J. LAFFERTY (1996). « Cheating with imperfect transcripts ». *In* J. LAFFERTY, éditeur : *Proc. Fourth International Conference on Spoken Language ICSLP 96*, volume 4, pages 2115–2118 vol.4.

[Povey et Woodland, 2002] D. POVEY et P.C. WOODLAND (2002). « Minimum phone error and i-smoothing for improved discriminative training ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '02)*, volume 1, pages I–105–I–108.

[Prasad *et al.*, 2005] R. PRASAD, S. MATSOUKAS, C.-L. KAO, J.Z. MA, D.-X. XU, T. COLTHURST, O. KIMBALL, R. SCHWARTZ, J.L. GAUVAIN, L. LAMEL, H. SCHWENK, G. ADDA et F. LEFEVRE (2005). « The 2004 bbn/limsi 20xrt english conversational telephone speech recognition system ». *In InterSpeech 2005*, Lisbon.

[Rahim *et al.*, 1997] M.G. RAHIM, Chin-Hui LEE et Biing-Hwang JUANG (1997). « Discriminative utterance verification for connected digits recognition ». *IEEE J SAP*, 5(3):266–277.

[Rao *et al.*, 1995] P.S. RAO, M.D. MONKOWSKI et S. ROUKOS (1995). « Language model adaptation via minimum discrimination information ». *In Proc. International Conference on Acoustics, Speech, and Signal Processing ICASSP-95*, volume 1, pages 161–164.

[Robert-Ribes et Mukhtar, 1997] J. ROBERT-RIBES et R.G. MUKHTAR (1997). « Automatic generation of hyperlinks between audio and transcript ». *In Eurospeech'97*.

[Robertson et Jones, 1994] S. E. ROBERTSON et K. Spark JONES (1994). « Simple, proven approaches to text retrieval ». Rapport technique, Cambridge University Engineering Department.

[Rong Zhang, 2006] Alexander Rudnicky RONG ZHANG (2006). « Investigations of issues for using multiple acoustic models to improve continuous speech recognition ». *In INTERSPEECH*.

[Rosenfeld, 1994] Ronald ROSENFELD (1994). « A hybrid approach to adaptive statistical language modeling ». *In Proc. ARPA Workshop on Human Language Technology*, pages 76–87.

[Rosenfeld, 1996] R. ROSENFELD (1996). « A maximum entropy approach to adaptive statistical language modeling ». *Computer Speech and Language*, 10(3):187–228.

[Salton, 1989] G. SALTON (1989). *Automatic Text Processing*. Addison Wesley.

[San-Segundo *et al.*, 2001] R. SAN-SEGUNDO, B. PELLOM, K. HACIOGLU, W. WARD et J.M. PARDO (2001). « Confidence measures for spoken dialogue systems ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)*, volume 1, pages 393–396 vol.1.

[Sankar, 2005] A. SANKAR (2005). « Bayesian model combination (baycom) for improved recognition ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, volume 1, pages 845–848.

[SAPORTA, 1990] G. SAPORTA (1990). *Probabilités analyse des donnés et statistique*. Editions Technip.

[Sarikaya *et al.*, 2005] R. SARIKAYA, Yuqing GAO, M. PICHENY et H. ERDOGAN (2005). « Semantic confidence measurement for spoken dialog systems ». 13(4):534–545.

[Schapire et Singer, 2000] Robert E. SCHAPIRE et Yoram SINGER (May 2000). « Boostexter : A boosting-based system for text categorization ». *Machine Learning*, Volume 39:2–3.

[Schwenk, 2007] Holger SCHWENK (2007). « Continuous space language models ». *Comput. Speech Lang.*, 21:492–518.

[Schwenk et Gauvain, 2000] Holger SCHWENK et Jean-Luc GAUVAIN (2000). « Combining multiple speech recognizers using voting and language model information ». *In ICSLP*.

[Schwenk et Gauvain, 2002] H. SCHWENK et J.-L. GAUVAIN (2002). « Connectionist language modeling for large vocabulary continuous speech recognition ». *In Acoustics, Speech, and Signal Processing*.

[Singh-Miller et Collins, 2007] N. SINGH-MILLER et C. COLLINS (2007). « Trigger-based language modeling using a loss-sensitive perceptron algorithm ». *In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, volume 4, pages IV–25–IV–28.

[Siohan *et al.*, 2005] O. SIOHAN, O. SIOHAN, B. RAMABHADRAN et B. KINGSBURY (2005). « Constructing ensembles of asr systems using randomized decision trees ». *In* B. RAMABHADRAN, éditeur : *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, volume 1, pages 197–200.

[Siu et Gish, 1999] Manhung SIU et Herbert GISH (1999). « Evaluation of word confidence for speech recognition systems ». *Computer Speech and Language*, 13:299–319.

[Smith et Waterman, 1981] T. F. SMITH et M. S. WATERMAN (1981). « Identification of common molecular subsequences ». *J. Mol. Biol. 147*, 147:195–197.

[Son *et al.*, 2000] Jongmok SON, Jinwoong KIM, Kyungok KANG et Keunsung BAE (2000). « Application of speech recognition with closed caption for concept-based video segmentation ». *In Ninth DSP Workshop*.

[Souvignier et Wendemuth, 1999] Bernd SOUVIGNIER et Andreas WENDEMUTH (1999). « Combination of confidence measures for phrases ». *In Proceedings Automatic Speech Recognition and Understanding Workshop 1999, Keystone, CO, USA*.

[Stolcke, 2002] A. STOLCKE (2002). « Srilm an extensible language modeling toolkit ». *In ICSLP*.

[Stuker *et al.*, 2006] S. STUKER, C. FUGEN, S. BURGER et M. WOFEL (2006). « Cross-system adaptation and combination for continuous speech recognition : The influence of phoneme set and acoustic front-end ». *In InterSpeech 2006*.

[Sukkar *et al.*, 1996] R. SUKKAR, A. SETLUR, M. RAHIM et C.-H. LEE (1996). « Utterrance verification of keywords strings using word-based minimum verification error(wb-mve) training ». *In ICASSP*.

[Sundaram *et al.*, 2004] R. SUNDARAM, R. SUNDARAM et J. PICONE (2004). « Effects on transcription errors on supervised learning in speech recognition ». *In* J. PICONE, éditeur : *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '04)*, volume 1, pages I–169–72 vol.1.

[Tshibasu-Kabeya *et al.*, 2006] B. TSHIBASU-KABEYA, G. BONTEMPI, F. BEAUGENDRE et G. MARECHAL (2006). « Aidar : Une architecture pour l'indexation de documents audio numériques ». *In VSST 2006*.

[Utsuro *et al.*, 2004] Takehito UTSURO, Yasuhiro KODAMA, Tomohiro WATANABEL, Hiromitsu NISHIZAKIL et Seiichi NAKAGAWA (2004). « Combining outputs of multiple lvcsr models by machine learning ». *Systems and Computers in Japan*, 36:1428–1440.

[Valtchev *et al.*, 1997] V. VALTCHEV, J. ODELL et S. J. WOODLAND, P. C.and Youngs (1997). « Mmie training of large vocabulary recognition system ». *In Speech Communication 22, pp303-314*.

[VAPNIK, 1982] Vladimir N. VAPNIK (1982). « Estimation of dependences based on empirical data ». *In Springer-Verlag*.

[VAPNIK, 1995] Vladimir N. VAPNIK (1995). « The nature of statistical learning theory ». *In Springer-Verlag New York, Inc.*

[Wagner et Fisher, 1974] R.A. WAGNER et M.J. FISHER (1974). « The string-to-string correction problem ». *The journal of the ACM*, 1:168–173.

[Wessel, 2002] Frank WESSEL (2002). *Word Posterior Probabilities for Large Vocabulary Continuous Speech Recognition*. Thèse de doctorat, Fakultat fur Mathematik, Informatik.

[Wessel *et al.*, 1998] F. WESSEL, F. WESSEL, K. MACHEREY et R. SCHLUTER (1998). « Using word probabilities as confidence measures ». *In* K. MACHEREY, éditeur : *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 1, pages 225–228 vol.1.

[Wessel *et al.*, 2000] F. WESSEL, F. WESSEL, R. SCHLUTER et H. NEY (2000). « Using posterior word probabilities for improved speech recognition ». *In* R. SCHLUTER, éditeur : *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP '00*, volume 3, pages 1587–1590 vol.3.

[Wessel *et al.*, 2001] F. WESSEL, F. WESSEL, R. SCHLUTER et H. NEY (2001). « Explicit word error minimization using word hypothesis posterior probabilities ». *In* R. SCHLUTER, éditeur : *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)*, volume 1, pages 33–36 vol.1.

[Whittaker, 2001] E. W. D. WHITTAKER (2001). « Temporal adaptation of language models ». *In ITRW on Adaptation Methodes for Speech Recognition - Sophia Antipolis*.

[Wiggers et Rothkrantz, 2003] Pascal WIGGERS et Leon J. M. ROTHKRANTZ (2003). « Using confidence measures and domain knowledge to improve speech recognition ». *In* ALLEMAGNE SPRINGER, Berlin, éditeur : *EUROSPEECH 2003*, volume 2807, pages 237–244. Springer, Berlin, ALLEMAGNE.

[Witbrock et Hauptmann, 1997] Michael J. WITBROCK et Alexander G. HAUPTMANN (1997). « Using words and phonetic strings for efficient information retrieval from imperfectly transcribed spoken documents ». *In DL '97 : Proceedings of the second ACM international conference on Digital libraries*, pages 30–35, New York, NY, USA. ACM.

[Witbrock et Hauptmann, 1998] M. J. WITBROCK et A G. HAUPTMANN (1998). « Improving acoustic models by watching television ». *In AAAI*.

[Woodland *et al.*, 2004] P. C. WOODLAND, H. Y. CHAN, G. EVERMANN, M. J. F. GALES, D. Y. KIM, X. A. LIU, D. MRVA, K. C. SIM, L. WANG, K. YU, J. MAKHOUL, R. SCHWARTZ, L. NGUYEN, S. MATSOUKAS, B. XIANG, M. AFIFY, S. ABDOU, J.-L. GAUVAIN, L. LAMEL, H. SCHWENK, G. ADDA, F. LEFEVRE, D. VERGYRI, W. WANG, J. ZHENG, A. VENKATARAMAN, R. R. GADDE et A. STOLCKE (2004). « Superears : Multi-site broadcast news system ». *In Rich Transcription Workshop (RT-04), Palisades, NY, November 2004*.

[Xue *et al.*, 2005] Jian XUE, Jian XUE et Yunxin ZHAO (2005). « Improved confusion network algorithm and shortest path search from word lattice ». *In* Yunxin ZHAO, éditeur : *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, volume 1, pages 853–856.

[Yan *et al.*, 2008] Zhi-Jie YAN, Bo ZHU, Yu HU et Ren-Hua WANG (2008). « Minimum word classification error training of hmms for automatic speech recognition ». *In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2008*, pages 4521–4524.

[Zhan et Waibel, 1997] Puming ZHAN et Alex WAIBEL (1997). « Vocal tract length normalization for large vocabulary continuous speech recognition ». *In CMU-CS-97*.

[Zhang et Rudnicky, 2001] R. ZHANG et A. RUDNICKY (2001). « Word level confidence annotation using combinations of features ». *In Conference on Speech Communication and Technology, Interspeech, Aalborg, Denmark*, pages 2105–2108.

[Zolnay *et al.*, 2005] A. ZOLNAY, R. SCHLUTER et H. NEY (2005). « Acoustic feature combination for robust speech recognition ». *In Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*, volume 1, pages 457–460.

[Zweig et Picheny, 2004] Geoffrey ZWEIG et Michael PICHENY (2004). « Advances in large vocabulary continuous speech recognition ». *Advances in Computers : Information Security*, 1:1.

Bibliographie personnelle

[Lecouteux et Linarès, 2008] Benjamin LECOUTEUX et Georges LINARÈS (2008). « Using prompts to produce quality corpus for training automatic speech recognition systems ». *In Proc. 14th IEEE Mediterranean Electrotechnical Conference MELECON 2008*, pages 841–846.

[Lecouteux *et al.*, 2007a] B. LECOUTEUX, G. LINARÈS, Frédéric BEAUGENDRE et Pascal NOCÉRA (2007a). « Text island spotting in large speech databases ». *In Interspeech (Anvers, Belgium)*.

[Lecouteux *et al.*, 2006a] Benjamin LECOUTEUX, Georges LINARÈS, J.F. BONASTRE et Pascal NOCÉRA (2006a). « Imperfect transcript driven speech recognition ». *In InterSpeech'06*.

[Lecouteux *et al.*, 2008a] B. LECOUTEUX, G. LINARÈS, Y. ESTÈVE et G. GRAVIER (2008a). « Combinaison de systèmes par décodage guidé ». *In JEP*.

[Lecouteux *et al.*, 2008b] B. LECOUTEUX, G. LINARÈS, Y. ESTÈVE et G. GRAVIER (2008b). « Generalized driven decoding for speech recognition system combination ». *In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2008*, pages 1549–1552.

[Lecouteux *et al.*, 2007b] B. LECOUTEUX, G. LINARÈS, Y. ESTÈVE et J. MAUCLAIR (2007b). « System combination by driven decoding ». *In Proc. IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2007*, volume 4, pages IV–341–IV–344.

[Lecouteux *et al.*, 2006b] B. LECOUTEUX, G. LINARÈS, P. NOCÉRA et J.F. BONASTRE (2006b). « Reconnaissance de la parole guidée par des transcriptions approchées ». *In JEP*.

[Rouvier *et al.*, 2008] Mickael ROUVIER, Georges LINARÈS et Benjamin LECOUTEUX (2008). « On-the-fly term spotting by phonetic filtering and request-driven decoding ». *In IEEE Workshop Spoken Language Technology*.

Glossaire

BAYCOM : Bayesian Combination

CMU : Carnegy Mellon University

CU : Cambridge University

DTW : Dynamic Time Warping

EM : Expectation Maximisation

fMLLR : Feature Maximum Likelihood Linear Regression

FST : Finite State Transducers

fWER : Frame Word Error Rate

GMM : Gaussian Mixture Model

HMM : Hidden Markov Model

IRENE : Système de reconnaissance de l'IRISA

IRISA : Institut de Recherche en Informatique et Systèmes Aléatoires

iROVER : Improved Recognizer Output Voting Error Reduction

IDF : Inverse Document Frequency

KLD : Kullback-Liebler Divergence

LIA : Laboratoire Informatique d'Avignon

LIMSI : Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur

LIUM : Laboratoire d'Informatique de l'Université du Maine

LMBB : Language Model Back-off Behavior

LPCC : Linear Prediction Cepstral Coefficients

MAP : Maximum A Posteriori

MFCC : Mel Frequency Cepstral Coefficients

MHV : Mots Hors Vocabulaire

ML : Maximum Likelihood

MLLR : Maximum Likelihood Linear Regression

MMC : Modèle de Markov Caché

MPE : Minimum Phone Error

MWE : Minimum Word Error

MCE : Minimum Classification Error

MMI : Maximal Mutual Information

MMIE : Maximal Mutual Information Estimation

PLP : Perceptual Linear Prediction

ROVER : Recognizer Output Voting Error Reduction

RPLP : RelAtive SpecTrAl - Perceptual Linear Prediction

RTBF : Radio Télévision Belge Francophone

SPEERAL : Système de reconnaissance du LIA

SRI : Stanford Research Institute (University)

TF : Term Frequency

TV : TéléVision

VSM : Vector Space Model

WP : Word Posterior