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 536 « Sciences et Agrosciences »
Laboratoire Informatique d'Avignon (EA 4128)

Structuration de contenus audio-visuel pour le résumé automatique

par
Mickaël ROUVIER

Soutenue publiquement le 24 juillet 2012 devant un jury composé de :

M^{me}	Régine André-Obrecht	Professeur, IRIT, Toulouse	Rapporteur
M.	Guillaume Gravier	Chargé de recherche, IRISA, CNRS, Rennes	Rapporteur
M.	Yannick Estève	Professeur, LIUM, Le Mans	Examinateur
M.	Sylvain Meignier	Maître de Conférence, LIUM, Le Mans	Examinateur
M.	Juan-Manuel Torres-Moreno	Maître de Conférence (HDR), LIA, Avignon	Examinateur
M.	Georges Linarès	Professeur, LIA, Avignon	Directeur de thèse

---

Laboratoire Informatique d'Avignon

Remerciements

Résumé

Ces dernières années, avec l'apparition des sites tels que Youtube, Dailymotion ou encore Blip TV, le nombre de vidéos disponibles sur Internet a considérablement augmenté. Le volume des collections et leur absence de structure limite l'accès par le contenu à ces données. Le résumé automatique est un moyen de produire des synthèses qui extraient l'essentiel des contenus et les présentent de façon aussi concise que possible.

Dans ce travail, nous nous intéressons aux méthodes de résumé vidéo par extraction, basées sur l'analyse du canal audio. Nous traitons les différents verrous scientifiques liés à cet objectif : l'extraction des contenus, la structuration des documents, la définition et l'estimation des fonctions d'intérêts et des algorithmes de composition des résumés.

Sur chacun de ces aspects, nous faisons des propositions concrètes qui sont évaluées.

Sur l'extraction des contenus, nous présentons une méthode rapide de détection de termes. La principale originalité de cette méthode est qu'elle repose sur la construction d'un détecteur en fonction des termes cherchés. Nous montrons que cette stratégie d'auto-organisation du détecteur améliore la robustesse du système, qui dépasse sensiblement celle de l'approche classique basée sur la transcription automatique de la parole.

Nous présentons ensuite une méthode de filtrage qui repose sur les modèles à mixtures de Gaussiennes et l'analyse factorielle telle qu'elle a été utilisée récemment en identification du locuteur. L'originalité de notre contribution tient à l'utilisation des décompositions par analyse factorielle pour l'estimation supervisée de filtres opérants dans le domaine cepstral.

Nous abordons ensuite les questions de structuration de collections de vidéos. Nous montrons que l'utilisation de différents niveaux de représentation et de différentes sources d'informations permet de caractériser le style éditorial d'une vidéo en se basant principalement sur l'analyse de la source audio, alors que la plupart des travaux précédents suggéraient que l'essentiel de l'information relative au genre était contenu dans l'image. Une autre contribution concerne l'identification du type de discours ; nous proposons des modèles bas niveaux pour la détection de la parole spontanée qui améliorent sensiblement l'état de l'art sur ce type d'approches.

Le troisième axe de ce travail concerne le résumé lui-même. Dans le cadre du résumé automatique de vidéo, nous essayons, dans un premier temps, de définir

ce qu'est une vue synthétique. S'agit-il de ce qui le caractérise globalement ou de ce qu'un utilisateur en retiendra (par exemple un moment émouvant, drôle....) ? Cette question est discutée et nous faisons des propositions concrètes pour la définition de fonctions d'intérêts correspondants à 3 différents critères : la saillance, l'expressivité et la significativité. Nous proposons ensuite un algorithme de recherche du résumé d'intérêt maximal qui dérive de celui introduit dans des travaux précédents, basé sur la programmation linéaire en nombres entiers.

Mots-clefs : Résumé Automatique de Vidéo, Détection de Termes à la volée, Parole spontannée, Classification du Genre Vidéo, Factor Analysis

Abstract

These last years, with the advent of sites such as Youtube, Dailymotion or Blip TV, the number of videos available on the Internet has increased considerably. The size and their lack of structure of these collections limit access to the contents. Summarization is one way to produce snippets that extract the essential content and present it as concisely as possible.

In this work, we focus on extraction methods for video summary, based on audio analysis. We treat various scientific problems related to this objective : content extraction, document structuring, definition and estimation of objective function and algorithm extraction.

On each of these aspects, we make concrete proposals that are evaluated.

On content extraction, we present a fast spoken-term detection. The main novelty of this approach is that it relies on the construction of a detector based on search terms. We show that this strategy of self-organization of the detector improves system robustness, which significantly exceeds the classical approach based on automatic speech recogntion.

We then present an acoustic filtering method for automatic speech recognition based on Gaussian mixture models and factor analysis as it was used recently in speaker identification. The originality of our contribution is the use of decomposition by factor analysis for estimating supervised filters in the cepstral domain.

We then discuss the issues of structuring video collections. We show that the use of different levels of representation and different sources of information in order to characterize the editorial style of a video is principaly based on audio analysis, whereas most previous works suggested that the bulk of information on gender was contained in the image. Another contribution concerns the type of discourse identification ; we propose low-level models for detecting spontaneous speech that significantly improve the state of the art for this kind of approaches.

The third focus of this work concerns the summary itself. As part of video summarization, we first try, to define what a synthetic view is. Is that what characterizes the whole document, or what a user would remember (by example an emotional or funny moment) ? This issue is discussed and we make some concrete proposals for the definition of objective functions corresponding to three different criteria : salience, expressiveness and significance. We then propose an algorithm for finding the sum of the maximum interest that derives from the one introduced

in previous works, based on integer linear programming.

Keywords : Video summary extraction, Spoken term detection, Spontaneous speech classification, Video genre classification, Factor Analysis

Table des matières

Chapitre 1

Introduction

Sommaire

1.1 Introduction

Ces dernières années, avec l'apparition de sites tels que Youtube, Dailymotion ou encore Google Vidéo, le nombre de vidéos disponibles sur Internet a considérablement augmenté. C'est par exemple le cas pour Dailymotion où plus de 15 000 vidéos sont vues chaque jour. Ces vidéos ont permis de mettre en avant certains faits d'actualité parfois ignorés des médias traditionnels (par exemple le président Nicolas Sarkozy supposément éméché lors du G20 le 11 juin 2007), ou encore de propulser au rang de star des inconnus comme le rappeur Kamini. Cependant, le fait d'être "inondé" de vidéos peut empêcher l'utilisateur de trouver celles qui l'intéressent réellement. Pire encore, devant ce nombre important de vidéos, il devient de plus en plus difficile pour un utilisateur d'en trouver vraiment d'intéressantes pour lui ou encore de se faire une idée de l'actualité dans le monde. Les vidéos marquantes se retrouvent noyées parmi des centaines, voire des milliers d'autres. C'est le problème chez Dailymotion (ou d'autres services communautaires de vidéos) où des experts doivent évaluer en quelques secondes ce que les vidéos valent en termes d'audience potentielle, d'information contenue, artistique ou de créativité.

Devant cette quantité d'informations, la gestion de l'information est de plus en plus problématique pour notre société. Elle est devenue un enjeu industriel, scientifique et économique majeur. Dans la recherche de vidéos, les utilisateurs gardent un rôle actif ; c'est à dire qu'ils vont rechercher l'information dont ils ont besoin. Actuellement, les sociétés (dans le sens économique du terme) aimeraient

bien promouvoir un modèle d'information "push" où l'utilisateur entrerait dans un rôle passif de telle sorte que l'information vienne à lui. Les résumés sont la solution la plus logique à l'information pléthorique. Il faut arriver à fournir à l'utilisateur un éventail de vidéos disponibles où l'information serait condensée et la plus pertinente possible.

C'est ce à quoi s'est intéressé le projet Résumé Pluri-Média Multidocument (RPM2) [1]. Construit autour de 5 partenaires : Sinequa, Eurecom, Laboratoire Informatique d'Avignon (LIA), Syllabs et Wikio, ce projet a pour but la mise au point des méthodes de résumés multi-documents pour les médias texte, audio et vidéo sur des données issues du Web. Le projet se concentre sur la gestion de l'information multimédia, la génération de nouveaux documents à partir d'un flux existant, des collections de contenus présentant une cohérence éditoriale et une approche multimodale de l'indexation.

Nos travaux se sont focalisés sur la construction d'un résumé vidéo multi-documents par extraction. Ce résumé tentera de sélectionner les vidéos ayant un intérêt et en proposer un résumé, un modèle finalement assez similaire au principe de l'émission "le Zapping" diffusée quotidiennement sur la chaîne de télévision Canal +.

Le Zapping est une émission qui rediffuse les moments considérés par ses auteurs comme les plus drôles, les plus navrants, les plus émouvants ou les plus insolites des programmes de la veille, toutes chaînes confondues. Conçu par la chaîne Canal+ à l'initiative de son directeur des programmes d'alors (Alain de Greef), sur une idée de Michel Denisot, le Zapping est apparu dès septembre 1989. La réalisation d'un Zapping est un véritable challenge puisque c'est une équipe composée de 12 personnes (venant de divers horizons : art, danse, photographie, etc...) qui regarde toutes les émissions télévisées d'une journée pour isoler des séquences considérées comme "intéressantes". La sélection des vidéos marquantes n'est qu'une étape, puisqu'il faut ensuite les assembler afin de réaliser le documentaire. Comme le dit Patrick Menais (responsable du Zapping à Canal+), "le Zapping" est "un montage subjectif de la réalité objective" : montrer un extrait de reportage d'Arte où une rescapée des camps de la mort explique en pleurant qu'il ne faut "plus jamais ça", puis montrer des CRS arrachant des sans-papiers à leur squat, n'y ajouter aucun commentaire, tout y est dit.

1.2 Problématique

C'est du modèle du "Zapping" proposé par Canal+ dont nous avons essayé de nous rapprocher dans nos travaux. Le zapping est une forme de résumé qui agglomère des moments qui présentent un intérêt *particulier* dans lequel nous voulons sélectionner l'information *importante*. Contrairement au résumé texte, nous avons

1. http://www.rpm2.org/

ici une dimension supplémentaire à prendre en compte : la vidéo. De plus l'information à sélectionner est beaucoup plus souvent expressive, subjective que dans le résumé texte.

Nous désirons faire du résumé vidéo par extraction calqué sur le modèle du résumé texte : des segments sont extraits des différentes vidéos et aglomérés dans une vidéo "résumée". La création d'un document sous forme de zapping est un véritable challenge scientifique puisqu'afin de réaliser ce type de documents, plusieurs verrous scientifiques devront être levés. Ceux-ci sortent du cadre du résumé automatique classique et posent des problèmes plus généraux de caractérisation de vidéos et de structuration de base audio-visuelle.

Les étapes du processus de création d'un zapping sont :

1. Collection et sélection des vidéos
2. Sélection des séquences vidéos ayant un intérêt notable et évaluation de cet intérêt
3. Agrégation des différentes séquences

Pour collecter et sélectionner des vidéos, nous nous trouvons ici dans un contexte web qui soulève au moins deux problèmes : la taille des collections disponibles et la structuration de ces données.

La collection des vidéos disponibles sur le Web est gigantesque. Selon Youtube, 10 milliards de vidéos seraient hébergées par la plateforme. De plus, il s'agit d'un ensemble ouvert qui augmente considérablement chaque jour. Ce sont environ (toujours selon Youtube) plus de 65 000 vidéos postées quotidiennement, soit environ 20 heures de vidéos par minute ! Une recherche de vidéo efficace doit obligatoirement reposer sur une structuration des collections par le contenu et/ou par les métadonnées.

D'autre part, les vidéos disponibles sur ces plateformes sont, pour la plupart, très mal indexées et les collections très mal structurées, ce qui rend la recherche difficile sans des outils automatiques efficaces. Les moteurs de recherche classiques se basent sur des métadonnées laissées par l'utilisateur : titre de la vidéo, rubrique, commentaire, etc... Ceci peut poser un problème car, d'une part, la vision d'une information n'est pas la même d'un utilisateur à un autre, et d'autre part les informations laissées par un utilisateur peuvent être imparfaites. Par conséquent, la structuration des bases de données ne doit pas uniquement se faire sur des données laissées par un être humain, mais sur le contenu intrinsèque d'un document.

Une fois la vidéo sélectionnée, il faut détecter un segment qui a un intérêt notable, c'est à dire sélectionner une sous-séquence vidéo dans laquelle l'information est compréhensible et présente un intérêt. Par exemple : lors de l'interview politique d'un ministre ou d'un responsable politique, Jean-Jacques Bourdin (journaliste de RMC) pose traditionnellement une question qui met mal à l'aise ces invités et qui vise directement les compétences du poste qu'ils occupent. Ainsi, lors de l'interview de Luc Chatel (ministre de l'éducation d'alors), le journaliste pose un

problème de mathématique tiré d'un questionnaire d'évaluation pour des enfants de classe de CM2 : "10 objets identiques coûtent 22 euros, combien coûtent 15 de ces objets ?". Après lui avoir répété deux fois l'énoncé, le ministre un peu mal à l'aise, donne la réponse de 16,50 Euros. La vidéo se poursuit avec la solution du journaliste [2]. Nous sommes donc là au cœur de notre problème : comment réussir à détecter que la séquence ayant un intérêt notable dans cette interview est précisément la question posée ainsi que la réponse donnée par notre ministre ?

Une fois les sous-séquences obtenues, il faut les agréger afin de constituer notre document. Cette dernière étape devra respecter deux contraintes : les documents parlant d'un même sujet devront être agrégés et le contenu des sous-séquences devra être unique pour chaque document.

1.3 Organisation du document

Ce travail est organisé en quatre grandes parties qui regroupent différents chapitres. Dans la première partie, le chapitre 2 décrit l'état de l'art des méthodes de résumé automatique. L'étude inclut les états de l'art dans les domaines texte et audio mais également vidéo. Nous présenterons aussi les diverses mesures d'évaluation du résumé automatique. La seconde partie est centrée sur différentes méthodes d'extraction du contenu audio. Le chapitre 3 étudie une application pour extraire du contenu textuel dans un flux audio. Dans le chapitre 4, nous proposons une nouvelle méthode de normalisation des paramètres acoustiques dans un système de Reconnaissance Automatique de la Parole (RAP) permettant ainsi d'améliorer, dans des conditions acoustiques bruitées, la transcription automatique. Dans la troisième partie, nous étudierons la structuration et la catégorisation de larges bases de données. Le chapitre 5 traitera de la classification de larges bases de données selon le genre vidéo. Dans le chapitre 6, nous aborderons la structuration de bases de données audios selon le niveau de spontanéité. Dans la dernière partie, le chapitre 7 proposera diverses méthodes pour sélectionner les faits marquants d'une vidéo et les agréger pour proposer un document sous forme de Zapping. Le document se termine par une conclusion reprenant les différentes contributions et décrivant les perspectives de ce travail.

2. http://www.youtube.com/watch?v=W5SrTUQEngM

Première partie

Etat de l'art

Chapitre 2

Etat de l'art en résumé automatique

Sommaire

2.1 Introduction

Le but d'un système de résumé automatique est de produire une représentation condensée d'une source d'informations dans laquelle les informations "importantes" du contenu original sont préservées. Les sources d'informations pouvant être résumées sont nombreuses et hétérogènes : documents vidéos, audios ou textuels. Un résumé peut être produit à partir d'un ou plusieurs documents.

Historiquement, le résumé automatique a d'abord été appliqué au texte. La principale approche consistait à extraire des phrases d'un document. En effet, l'approche du résumé par extraction provient d'observations comme celles de (Lin, 2003) où environ 70% du contenu d'un panel de résumés textuels écrits à la main est extrait directement depuis les textes d'origine. Le résumé par extraction est une des approches les plus répandues. Plus récemment sont apparues des techniques essayant de compresser ou de régénérer les phrases (Lin and Hovy, 2003; Le Nguyen et al., 2004; Knight and Marcu, 2000).

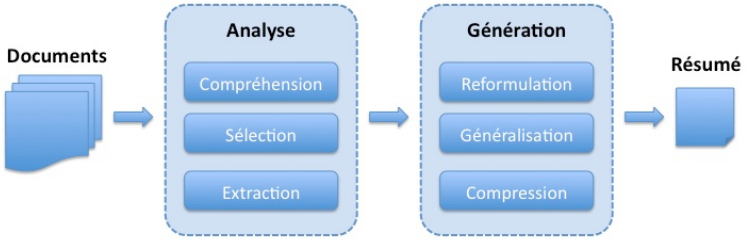

FIGURE 2.1 – *Présentation d'un système de résumé automatique audio.*

2.2 Résumé texte

2.2.1 Les approches classiques

Les travaux sur le résumé automatique ont commencé dans les années 50 avec Luhn (Luhn, 1958), qui propose d'utiliser la fréquence d'un terme pour mesurer la pertinence des phrases : l'idée étant qu'une personne aura tendance à répéter certains mots quand elle parle d'un même sujet. La pertinence du terme est considérée dans ces travaux comme étant proportionnelle à la fréquence du terme dans le document. De plus, l'auteur a proposé plusieurs idées clefs, comme la normalisation des mots (le regroupement de certains mots similaires du point de vue de l'orthographe aura pour but de s'affranchir des variantes des mots) mais également la suppression de certains mots outils à l'aide d'une *stop-liste*. Cette façon de procéder a eu un impact sur la grande majorité des systèmes d'aujourd'hui qui sont basés sur le même principe.

Cependant, la fréquence d'un terme n'est pas uniquement liée à la pertinence de ce terme. En effet, il est probable que les documents, dans un même domaine,

partagent des termes communs mais qu'ils n'apportent pas d'informations saillantes. (Jones, 1972) a montré que la pertinence d'un terme dans le document est inversement proportionnelle au nombre de documents dans le corpus contenant le terme. Le poids d'un terme est calculé ainsi :

$$w_{i,j} = tf_{i,j} * idf_i = \frac{n_{i,j}}{\sum_k n_{k,j}} * \log_2(\frac{N}{n_i}) \tag{2.1}$$

où $w_{i,j}$ est le poids du terme i dans le document j. $tf_{i,j}$ est la fréquence d'un terme i dans le document j. idf_i est la fréquence documentaire inverse, où N est le nombre total de documents dans le corpus et n_i le nombre de documents dans lesquels apparaît le terme i. Le score des phrases peut ensuite être calculé par différentes méthodes comme, par exemple, la somme des scores de termes présents dans la phrase.

D'autres indicateurs peuvent être utilisés pour juger de la pertinence d'une phrase comme sa position à l'intérieur du document (Baxendale, 1958), ou encore la présence de mots du titre, ou encore certains mots spécifiques (comme "plus encore" ou "pertinent") (Edmundson, 1969).

Dans (Hovy and Lin, 1998), l'auteur propose une autre manière de mesurer la pertinence des termes en considérant le concept qu'il évoque. Par exemple, l'occurrence du concept "bicycle" est comptée quand le mot "bicycle" apparait mais elle est aussi comptée pour les mots "vélo", "pédale", "guidon", etc... Les concepts peuvent être déterminés en utilisant les liens sémantiques de la base de données de *WordNet*.

2.2.2 Approche basée sur la cohésion

Les liens anaphoriques[1] peuvent poser des problèmes pour la création d'un résumé automatique. D'une part, l'extraction des phrases pour le résumé peut échouer à cause des relations entre les concepts dans un texte et d'autre part, le résumé peut devenir difficile à comprendre si la phrase contient des liens anaphoriques qui sont sortis du contexte.

Les propriétés de cohésion de texte ont été explorées par différentes approches de résumé. Dans (Barzilay and Elhadad, 1997), les auteurs introduisent une méthode appelée "chaîne lexicale" (*Lexical chains*). Ils utilisent la base de données WordNet pour déterminer les relations cohésives (répétition, synonymie, antonymie, hyperonymie et holonymie) entre les termes. La chaîne est ensuite composée des relations de termes et leurs scores sont déterminés sur la base du nombre de types de relations dans la chaîne. Les phrases ou les chaînes les plus concentrées sont sélectionnées pour produire le résumé.

1. Lien anaphorique : un mot ou une phrase qui se réfère à une expression ou un mot dit précédemment ; ce sont typiquement des pronoms tels que : lui, il, elle, etc...

2.2.3 Approche basée sur les graphes

(Mihalcea and Tarau, 2004) proposent de considérer le processus extractif comme une identification des segments les plus populaires dans un graphe. Les algorithmes de classement basés sur les graphes tel que *PageRank* ont été utilisés avec succès dans les réseaux sociaux. Ces algorithmes peuvent être vus comme les éléments clefs du paradigme introduit dans le domaine de la recherche sur Internet, à savoir le classement des pages Web par l'analyse de leurs positions dans le réseau et non sur leurs contenus (par exemple l'algorithme *Google PageRank* (Brin and Page, 1998)).

Cette propriété de relation a été explorée plus largement dans les approches basées sur le graphe qui mettent des phrases en relation. *TextRank* (Mihalcea and Tarau, 2004) proposent de transformer un document en un graphe dans lequel chaque phrase du document est modélisée par un nœud. Un arc entre deux nœuds est créé si les phrases sont lexicalement similaires. Une phrase S_i est représentée par un jeu de mots : $S_i = w_1^i, w_2^i, ..., w_n^i$, la similarité entre deux phrases S_i et S_j est définie comme suit :

$$Sim(S_i, S_j) = \frac{|\{w_k : w_k \in S_i \wedge w_k \in S_j\}|}{\log |S_i| + \log |S_j|} \tag{2.2}$$

Cette approche permet de décider de l'importance du sommet d'un graphe en se basant non pas sur l'analyse locale du sommet lui-même, mais sur l'information globale issue de l'analyse récursive du graphe complet. Appliqué au résumé automatique, cela signifie que le document est représenté par un graphe d'unités textuelles (phrases) liées entre elles par des relations issues de calculs de similarité. Les phrases sont ensuite sélectionnées selon des critères de centralité ou de prestige dans le graphe puis assemblées pour produire des extraits.

TextRank est assez efficace sur des documents structurés comme des articles où chaque phrase contient des informations utiles et où la redondance est faible. Cependant, le résultat est moins probant avec des phrases spontanées qui sont typiquement mal formées car les participants s'interrompent souvent et les informations sont souvent distillées.

Dans (Garg et al., 2009), les auteurs proposent une version modifiée de *TextRank* pour traiter des documents bruités et la redondance due à la parole spontanée. Cette méthode, appelée *ClusterRank*, propose dans une première étape de regrouper certaines phrases en classes selon leur score de similarité cosinus. Le graphe est construit selon ces classes.

2.2.4 Approche basée sur la rhétorique

La Théorie de la Structure Rhétorique (RST) est une théorie qui permet de décrire la structure d'un texte. Originellement, cette théorie a été développée pour

faire de la génération automatique de texte. Un texte peut être organisé en éléments reliés entre eux par des relations. Ces relations peuvent être de deux types : des "satellites" ou des "noyaux". Un satellite a besoin d'un "noyau" pour être compris, tandis que l'inverse n'est pas possible.

Par exemple, si l'on a une affirmation suivie de la démonstration l'étayant, la RST postule une relation de "démonstration" entre les deux segments. Elle considère également que l'affirmation est plus essentielle pour le texte que la démonstration particulière, et marque cette préséance en dénommant le segment d'affirmation un *noyau* et le segment de démonstration un *satellite*. L'ordre des segments n'est pas déterminé, mais pour toute relation, les ordres sont plus ou moins vraisemblables.

Pour un texte structuré et cohérent, la RST permet d'obtenir une analyse du document et indique pour chaque phrase, la raison pour laquelle elle a été retenue. Elle permet de rendre compte de la cohérence textuelle indépendamment des formes lexicales et grammaticales du texte. En postulant l'existence d'une structure reliant les phrases entre elles, la RST donne une base à l'étude des relations entre ces structures, discours et divers procédés de cohésion. Les représentations ainsi construites peuvent être utilisées pour déterminer les segments les plus importants du texte. Ces idées ont été utilisées par (Ono et al., 1994; Marcu, 1997) dans des systèmes visant à produire des résumés.

2.2.5 Approche basée sur les phrases

Dans les approches dites d'extraction, la sélection des phrases se faisait uniquement sur leur signification individuelle. Les phrases sélectionnées peuvent être soit complémentaires, soit redondantes entre elles. Cordonell et Goldstein ont proposé en 1998 de construire le résumé en prenant en compte l'anti-redondance des phrases ainsi que de la pertinence de celles-ci (Carbonell and Goldstein, 1998).

L'algorithme Maximal Marginal Relevance (MMR) est un algorithme glouton qui consiste à réordonner les phrases en fonction de deux critères : l'importance de la phrase et son niveau de redondance par rapport aux phrases déjà sélectionnées. A chaque itération, l'algorithme détermine la phrase (S_i) la plus proche du document tout en étant la plus éloignée des phrases (S_j) sélectionnées auparavant. Cette phrase est ajoutée à la sélection et l'algorithme s'arrête lorsqu'une condition est remplie comme par exemple un nombre de phrases, un nombre de mots ou un ratio de compression atteint.

$$MMR(S_i) = \lambda * Sim_1(S_i, D) - (1 - \lambda) * Sim_2(S_i, S_j) \tag{2.3}$$

Dans la formulation originelle de MMR, $Sim_1()$ et $Sim_2()$ sont des similarités $cosine()$ qui ont fait leur preuves en recherche documentaire. Cependant, n'importe quelle similarité entre phrases peut-être adaptée à ce problème. λ est un hyper-paramètre devant être ajusté empiriquement.

2.2.6 Approche basée sur les concepts

Jusqu'à présent, la plupart des modèles de résumé automatique s'appuient sur l'ajout d'une phrase pour l'inclure dans le résumé. La phrase la plus appropriée est sélectionnée puis agglomérée aux autres pour former le résumé. Ainsi, les phrases sont ajoutées les unes aux autres sans remettre en cause ce qui a déjà été sélectionné. C'est un des problèmes des algorithmes gloutons car durant la recherche, la sélection de la prochaine phrase dépend fortement de celle choisie précédemment et des phrases libres.

Dans (Gillick and Favre, 2009), les auteurs proposent une manière plus naturelle de créer un résumé automatique en estimant globalement la pertinence et la redondance dans un cadre basé sur la Programmation Linéaire en Nombres Entiers (PLNE). En effet la PLNE peut être utilisée pour maximiser le résultat de la fonction objective, lequel va essayer de chercher efficacement sur l'espace possible des résumés une solution optimale. Cette méthode considère, pour la sélection des phrases, que chaque phrase est constituée de concepts. Ces phrases sont définies de telle sorte que la qualité d'un résumé puisse être mesurée par la valeur des concepts uniques qu'il contient. La redondance est limitée implicitement par une contrainte de taille.

Les concepts sont représentés par des éléments d'informations comme pour un *meeting* : une décision prise à une réunion, ou l'opinion d'un participant sur un sujet. Mais l'abstraction de tels concepts est difficile à extraire automatiquement ; il faut ramener ces concepts à des mots plus simples, les n-grammes, qui peuvent être utilisés pour représenter la structure du document. Cependant, les n-grammes se recoupent souvent avec des marqueurs de discours ("en fait", "vous savez") lesquels peuvent rajouter du bruit. Un algorithme d'extraction de mot-clefs est proposé pour identifier les séquences caractéristiques ainsi que le contenu :

1. Extraction de tous les n-grammes pour $n = 1, 2, 3$
2. Suppression du bruit : suppression des n-grammes qui apparaissent seulement une fois
3. Réévaluation des poids des Bi-grammes et Tri-grammes : $w_i = frequence(g_i) \cdot n \cdot \arg\max_n idf(mot_n)$ où w_i est le poids final du n-gramme, n la taille du n-gramme et mot_n un mot du n-gramme.

Formellement, désignons la variable binaire c_i qui indique la présence d'un concept i dans le résumé et la variable binaire s_j qui indique la présence de la phrase j dans le résumé. Chaque concept peut apparaître dans des multiples phrases et les phrases peuvent contenir des concepts multiples. L'occurrence du concept i dans la phrase j est notée par la variable binaire o_{ij}. Le score du résumé est égal à la somme des poids (w_i) des concepts présents dans le résumé. l_j est la longueur de la phrase j, la taille du résumé est limitée par la constante L. Ainsi la recherche d'un résumé automatique peut être exprimée sous forme de problème PLNE :

$$
\begin{aligned}
\text{Maximize} \quad & \sum_i w_i c_i \\
\text{Subject To} \quad & \sum_j l_j n_j \\
& n_j o_{ij} \leq c_i, \qquad \forall i, j \\
& \sum_j n_j o_{ij} \geq c_i, \quad \forall i, j \\
& c_i \in \{0, 1\} \qquad \forall i \\
& n_j \in \{0, 1\} \qquad \forall j
\end{aligned}
$$

Dans ce cadre, la fonction objectif permet de maximiser la somme pondérée des concepts présents dans le résumé, compte tenu de la contrainte de longueur. Les contraintes de cohérence font en sorte que si une phrase est sélectionnée, tous les concepts contenus dans cette phrase sont aussi sélectionnés, et si c'est un concept qui est sélectionné, au moins une phrase qui contient ce concept est sélectionnée également.

2.3 Résumé audio

Le résumé de contenu parlé est un domaine de recherche relativement récent comparé au résumé automatique de texte. La nécessité de résumer le contenu parlé s'est faite ressentir lorsque les bases de données audio/vidéo ont commencé à fortement augmenter. Les principales problématiques (en plus des problématiques de résumé texte) sont : les disfluences de la parole, la détection des frontières de phrases, le maintien de la cohérence des locuteurs (lors de débats) ainsi que les erreurs issues des systèmes de transcription automatique de parole.

2.3.1 Approche basée sur la prosodie

Le résumé de texte par extraction sélectionne les segments les plus représentatifs pour former un résumé. Comparé au résumé automatique de texte qui repose sur le lexique, la syntaxe, la position et la structure de l'information, le résumé automatique de parole peut tirer parti des sources d'informations supplémentaires contenues dans le discours, tels que le locuteur et/ou information acoustique/prosodique. La prosodie joue un rôle important dans une communication verbale, car elle permet d'exprimer une information non-linguistique comme une intention, un changement de sujet, l'accent mis sur un mot ou sur une phrase importante. Ceci permet d'avoir des informations sur le contenu d'un document (sans avoir de transcription disponible *a priori*).

L'intégration d'un jeu de paramètres acoustiques/prosodiques a été principalement conduite dans des documents tels que les *meetings* et dans ceux où le style

de parole du locuteur varie. En effet dans (Kazemian et al., 2008), l'auteur a montré que dans le domaine journalistique où le style de parole du locuteur varie peu, l'intégration des jeux de paramètres acoustiques/prosodiques n'apporte rien en général et peut même dégrader les résultats.

De manière générale, les auteurs proposent d'extraire des paramètres prosodiques issus du F0 (maximum, minimum, moyenne, médiane et variance) et l'énergie du signal (maximum, minimum, moyenne, médiane et variance). D'autres jeux de paramètres peuvent être utilisés comme la durée de la phrase ou le nombre de mots (ou de lettres) dans une phrase. La plupart du temps, les paramètres prosodiques sont utilisés en complément de la transcription donnée par un système de RAP (Xie et al., 2009b; Maskey and Hirschberg, 2005) ; ils peuvent aussi être utilisés tout seuls (Zhang and Fung, 2007; Maskey and Hirschberg, 2006).

2.3.2 Approche basée sur les treillis

S'il n'existe pas de transcription générée par un humain pour un document audio, le résumé automatique doit compter sur des transcriptions générées automatiquement par un système de RAP. Selon le corpus, le taux d'erreur de mots peut osciller entre 10% et 50% (Fiscus et al., 1998). Ces taux peuvent être imputables aux langages utilisés dans le document, aux conditions acoustiques, etc...

Intuitivement, le taux d'erreur de mots a un impact négatif sur les performances du système de résumé (Murray et al., 2005). De précédentes recherches ont évalué le système de résumé utilisant la transcription humaine et la sortie d'un système de RAP. La plupart des travaux menés ont montré que les erreurs d'un système de RAP dégradent la qualité du résumé.

Pour résoudre le problème causé par des transcriptions imparfaites, les auteurs proposent d'utiliser les résultats étendus de la sortie d'un système de RAP pour créer le résumé (Liu et al., 2010). Les n-meilleures hypothèses, treillis de mots et réseaux de confusion ont été largement utilisés comme une interface entre un système de RAP et des modules de reconnaissance de langage tels que la traduction automatique, recherche de document parlé, et permettent d'améliorer les résultats en utilisant la meilleure hypothèse.

Jusqu'à présent, il y a eu peu de travaux qui utilisent plus que la meilleure hypothèse d'un système de RAP pour le résumé de parole. Plusieurs études utilisent les scores de confiance acoustique de la meilleure hypothèse de la sortie d'un système de RAP afin de réévaluer le poids des mots (Valenza et al., 1999; Zechner and Waibel, 2000; Hori and Furui, 2003). Dans (Lin and Chen, 2009), les auteurs utilisent le réseau de confusion et la position d'un mot dans un réseau selon les probabilités *a posteriori*, dans un cadre de génération de résumé automatique pour les émissions d'informations chinoises. Dans (Xie and Liu, 2010), les auteurs proposent aussi d'utiliser le réseau de confusion pour créer un résumé. L'approche est différente de celle de Lin, puisqu'elle se base sur l'impact des scores de confiance et sur une méthode d'élagage spécifique.

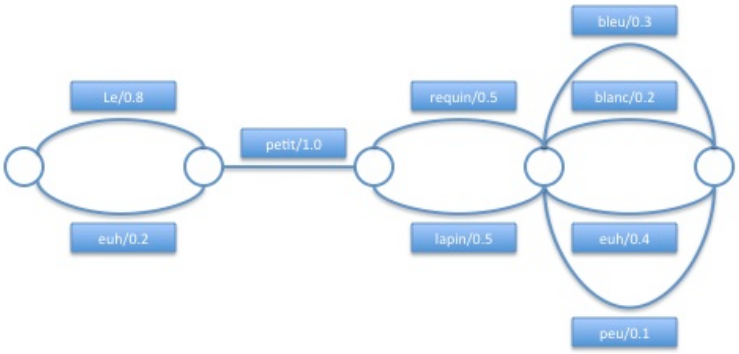

FIGURE 2.2 – *Exemple d'un treillis de mots.*

2.4 Résumé vidéo

Le résumé vidéo n'est pas le sujet principal de notre thèse. Dans ce travail nous évoquons certains objets, certaines techniques, venant du résumé vidéo. Nous proposons donc un bref aperçu de l'état de l'art du résumé vidéo.

La création d'un résumé vidéo permet de sélectionner les parties intéressantes d'une vidéo pour permettre d'avoir rapidement une idée sur le contenu de très grandes bases de données vidéos, sans visualisation et interprétation de l'ensemble de celles-ci. Les méthodes développées consistent à extraire un ensemble d'images fixes, appelées vignettes, qui ensemble forment le résumé vidéo.

2.4.1 Approche basée sur le changement de contenu

Cette méthode procède séquentiellement en sélectionnant une trame comme la trame clef, seulement si le contenu visuel est significativement différent des trames clefs précédemment extraites. La méthode basée sur le changement de contenu sélectionne la trame suivante f_{r+1} en fonction de la trame clef la plus récente f_r.

$$r_{i+1} = argmin_t\{C(f_t, f_{ri}) > \epsilon, i < t < n\} \tag{2.4}$$

Différentes métriques ont été proposées dans la littérature pour étudier le changement de contenu. La plus populaire se base sur la différence d'histogrammes (Yeung and Liu, 1995; Zhang, 1997). Initialement, la trame clef choisie est celle qui dépasse un certain seuil $C(f_t, f_{ri}) > \epsilon$, mais il se peut que cette trame ne soit pas représentative du contenu. D'autres travaux se sont concentrés sur le choix de la trame clef r_{i+1} entre f_t et f_{ri} en sélectionnant, par exemple, la trame de fin, du milieu...

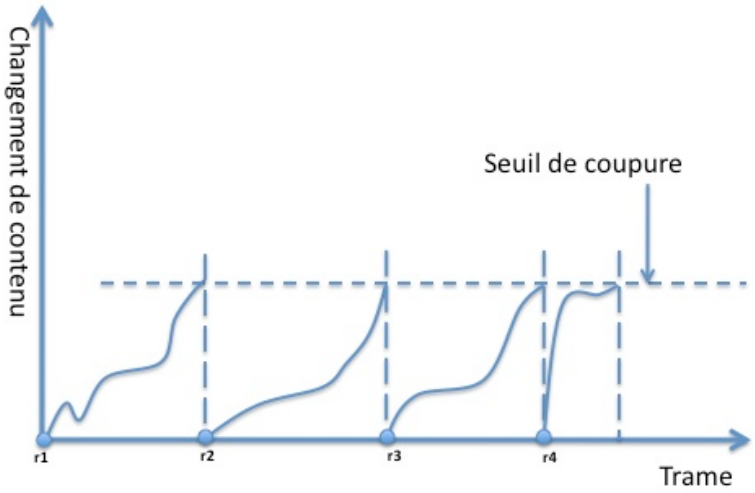

FIGURE 2.3 – *Illustration de la méthode de changement de contenu.*

2.4.2 Approche basée sur la classification

Cette approche représente les trames vidéos comme étant des points dans l'espace et fait l'hypothèse que le point représentatif d'une classe dans l'espace peut être utilisé comme une trame clef pour construire le résumé.

Typiquement, l'espace des paramètres utilisé est celui de l'histogramme des couleurs, mais cet espace est généralement trop grand et trop bruité pour être traité. Dans (Gibson et al., 2002), les auteurs proposent un prétraitement des trames vidéos en réduisant *la dimension de l'espace* et en retenant uniquement les variations significatives. Cette réduction de dimensions se fait via une Analyse en Composantes Principales (ACP).

Plusieurs approches de classification ont été proposées. La plus classique est une classification ascendante : à chaque itération, l'algorithme essaie de regrouper les points entre eux selon leur distance euclidienne (Zhang, 1997).

Certaines classes peuvent être bruitées ou ne contenir aucune information significative. Dans (Zhang, 1997), l'auteur considère les classes qui ont une taille plus grande que la moyenne des tailles des classes.

Finalement, l'extraction des points représentatifs d'une classe s'effectue en sélectionnant le point le plus proche du centre d'une classe. Ainsi, la trame peut être ajoutée à la séquence des trames du résumé (Yu et al., 2004b).

2.4.3 Video *Maximal Marginal Relevance*

Jusqu'à présent, les algorithmes de résumé automatique de vidéos sélectionnaient les trames uniquement par rapport à la similarité de leur contenu visuel.

Dans (Lie and Merialdo, 2010), les auteurs proposent de choisir une trame clef dans laquelle le contenu visuel est similaire au contenu de la vidéo, mais en même temps où la trame est différente des trames déjà sélectionnées dans le résumé. Cette technique assez récente est issue du domaine de résumé automatique de texte. Ainsi, par analogie avec l'algorithme MMR, l'algorithme *Video Marginal Relevance* (Video-MR) se définit ainsi :

$$\text{Video-MR}(f_i) = \lambda * Sim_1(f_i, V \backslash S) - (1 - \lambda) * maxSim_2(f_i, g) \tag{2.5}$$

où V contient toutes les trames de la vidéo, S contient les trames sélectionnées, g est une trame dans S et f_i est une trame candidate pour la sélection. Sim_2 permet de calculer la similarité entre les trames f_i et g. La similarité de $Sim_1(f_i, v \backslash S)$ peut être considérée comme suit :

- une somme arithmétique : $Sim_{AM}(f_i, V \backslash S) = \frac{1}{|v \backslash (S \cup f_i)|} \sum_{f_i \in V \backslash (S \cup f_i)} sim(f_i, f_j)$
- une somme géométrique : $Sim_{GM}(f_i, V \backslash S) = [\pi_{f_i \in V \backslash (S \cup f_i)} sim(f_i, f_j)]^{\frac{1}{|v \backslash (S \cup f_i)|}}$

Le contenu d'une trame peut être paramétrisé de différentes manières comme l'histogramme des couleurs de la trame, les objets présents dans la trame, etc... Les auteurs proposent d'utiliser comme paramètre le "sac de mots visuels" (*bag of visual word*). Un sac de mots visuels est défini par les Points d'Intérêts Locaux (LIP) dans l'image basés sur un DoG (*Difference of Gaussian*) et LoG (*Laplacian of Gaussian*). Ensuite, le descripteur SIFT est calculé sur ces points d'intérêts. Les descripteurs SIFT sont classifiés en k groupes par un algorithme de *k-moyennes*, où k représente le nombre de mots visuels dans le document.

2.5 Métrique d'évaluation

Évaluer un résumé est une tâche difficile notamment parce qu'il n'existe pas de résumé idéal pour un document donné ou un jeu de documents. La création d'un résumé par un être humain est une création subjective, elle peut différer d'une personne à l'autre, selon l'importance que la personne donne à certaines informations : celles que nous connaissons déjà, celles que nous voulons mettre en avant, celles qui nous plaisent ou celles que nous détestons... L'évaluation des résultats de résumé automatique est encore aujourd'hui un problème ouvert. Il existe de nombreuses mesures d'évaluation, allant des mesures automatiques à celles demandant à un être humain d'annoter le résumé selon des critères spécifiques pour l'évaluer (cohérence, concision, grammaticalité, lisibilité et contenu).

2.5.1 Précision, Rappel et F-Mesure

Le résumé par extraction revient parfois à ne sélectionner que les phrases clefs d'un document. Nous savons pour chaque phrase si elle peut être sélectionnée ou pas pour créer le résumé. On peut donc voir ce problème comme une tâche de classification binaire (acceptation/rejet d'une phrase) et donc utiliser les métriques d'évaluations comme la précision, le rappel et la F-Mesure, pour savoir à quel point nous sommes proches du résumé. La Précision (P) est définie par le rapport du nombre de phrases pertinentes trouvées au nombre total de phrases sélectionnées dans le résumé de référence. Le Rappel (R) est le rapport du nombre de phrases pertinentes trouvées au nombre total de phrases pertinentes dans le résumé de référence. La F-Mesure (F) est une mesure qui combine la précision et le rappel. La meilleure façon de calculer la F-Mesure est d'avoir une moyenne harmonique entre la précision et le rappel :

$$F = \frac{2 * P * R}{P + R} \tag{2.6}$$

L'équation 2.6 permet de pondérer le rappel et la précision de façon égale ; il s'agit d'un cas particulier de la F-Mesure. L'équation de la F-Mesure s'écrit ainsi :

$$F = \frac{(\beta^2 + 1) * P * R}{\beta^2 * P + R} \tag{2.7}$$

où β est un poids permettant de favoriser la précision quand $\beta \leq 1$ et de favoriser le rappel quand $\beta \geq 1$.

2.5.2 Utilité relative

Le principal problème de précision et rappel est que des juges humains sont souvent en désaccord avec le choix ainsi qu'avec l'ordre des phrases les plus importantes dans un document. Pour répondre à ce problème, la mesure d'utilité relative (*Relative Utility*, RU) a été introduite dans (Radev and Tam, 2003). Avec RU, le modèle de résumé représente toutes les phrases d'entrée du document avec des valeurs de confiance pour leur inclusion dans le résumé. Ces valeurs de confiance indiquent le degré avec lequel la phrase doit faire partie du résumé automatique selon un juge humain. Ce nombre est appelé "l'utilité de la phrase". Il dépend du document d'entrée, de la longueur de la phrase et du juge. Pour calculer le RU, il a été demandé à des juges d'assigner un score d'utilité à toutes les phrases d'un document. Nous pouvons définir le système de mesure suivant :

$$RU = \frac{\sum_{j=1}^{n} \lambda_j \sum_{i=1}^{N} u_{ij}}{\sum_{j=1}^{n} \varepsilon_j \sum_{i=1}^{N} u_{ij}} \tag{2.8}$$

u_{ij} est le score d'utilité de la phrase j de l'annotateur i, et e le nombre de phrases correspondant à la taille de résumé choisie. Les e phrases sont celles ayant obtenu le meilleur score. ε_j est égal à 1 pour les e phrases extraites par les annotateurs et 0 dans le cas contraire. λ_j est égal à 1 pour les e phrases extraites par le système et 0 dans le cas contraire.

2.5.3 Similarité cosinus

Les mesures présentées jusqu'à présent comptent le nombre de phrases qu'il y a en commun entre un résumé de référence et un résumé de test. Ces mesures ignorent le fait que 2 phrases peuvent contenir la même information même si elles sont écrites de manière différente. Une technique semi-automatique d'évaluation de résumé se fait au travers de mesures de similarité calculées entre un résumé candidat et un ou plusieurs résumés de référence. Une mesure basique de similarité est le cosinus :

$$cos(X,Y) = \frac{\sum_i x_i \cdot y_i}{\sqrt{\sum_i (x_i)^2} \cdot \sqrt{\sum_i (y_i)^2}} \tag{2.9}$$

où X représente le système de résumé candidat et Y celui de résumé de référence.

2.5.4 ROUGE : *Recall-Oriented Undestudy for Gisting Evaluation*

La mesure cosinus ne permet pas de prendre en compte la cohérence d'un résumé candidat. En mélangeant l'ordre des mots d'un résumé candidat, on obtiendra exactement le même score cosinus que le résumé candidat initial alors que le résumé obtenu sera complétement illisible. Dans (Lin, 2004), l'auteur propose la méthode Recall-Oriented Undestudy for Gisting Evaluation (ROUGE) : cette mesure permet de connaître la similarité de n-grammes entre un résumé candidat et un ou plusieurs résumés de référence.

Supposons qu'il y ait un nombre de résumés de référence (R_{ref}). Le score Rouge d'un résumé candidat se calcule ainsi :

$$ROUGE = \frac{\sum_{s \in R_{ref}} \sum_{N-grammes \in s} Co - occurences(N-grammes)}{\sum_{s \in R_{ref}} \sum_{N-grammes \in s} Nombre(N-grammes)} \tag{2.10}$$

où $Co - occurence(N_grammes)$ est le nombre maximum de n-grammes qui co-occurent entre un résumé candidat et le résumé de référence. $Nombre(N-grammes)$ est le nombre de n-grammes dans le résumé de référence. Deux variantes de ROUGE sont couramment utilisées dans les campagnes d'évaluation. ROUGE-N où N est la taille du n-gramme et ROUGE-SUX qui est une adaptation de ROUGE-2 utilisant

des bi-grammes à trous de taille maximum X et comptabilisant les uni-grammes. Le Tableau 2.1 regroupe quelques exemples d'unités utilisées pour ROUGE.

TABLE 2.1 – *Illustration des différents découpages d'une phrase pour le calcul ROUGE.*

Phrase	suit le lapin blanc néo
ROUGE-1	suit, le, lapin, blanc, néo
ROUGE-2	suit-le, le-lapin, lapin-blanc, blanc-néo
ROUGE-SU2	*ROUGE-1*, *ROUGE-2*, suit-lapin, suit-blanc, le-blanc, le-néo, lapin-néo
ROUGE-SU4	*ROUGE-SU2*, suit-néo

2.5.5 Pyramide

Les phrases peuvent être dites de manières différentes (e.g. "Madame Jouanno a pris jeudi des mesures", "La ministre a pris hier des mesures"), ce qui peut poser un problème lors de l'évaluation de résumés avec des méthodes automatiques. Dans (Nenkova et al., 2007), l'auteur propose de contourner ce problème avec une nouvelle méthode semi-automatique : Pyramid. L'idée est d'identifier des unités sémantiques (Summarization Content Units, SCU) à partir d'un ou plusieurs résumés de référence. Les SCU exprimant la même notion sont regroupées et pondérées en fonction du nombre de résumés de référence la contenant. Une pyramide est construite à partir de leurs pondérations. Au sommet de la pyramide se trouve la SCU qui a le plus grand poids et qui apparaît dans la plupart des résumés. Au bas de la pyramide apparaissent les SCU qui ont un poids faible. Le score Pyramide d'un résumé candidat dépend du nombre d'unités sémantiques qu'il contient et qui sont considérées comme importantes par les annotateurs. Cette méthode intéressante demande toutefois l'intervention d'un être humain pour annoter les corpus.

Ainsi, dans les exemples du Tableau 2.2, l'annotation commence en identifiant les phrases similaires (comme les 4 phrases soulignées). Les phrases sélectionnées nous permettent d'obtenir deux SCU. Chaque SCU a un poids correspondant au nombre de résumés dans lesquels elle apparait.

La première SCU parle des Libyens qui ont été officiellement accusés de l'attentat de Lockerbie. Cette SCU est présente dans les résumés A1 (two Lybyans, indicted), B1 (Two Libyans were indicted), C1 (Two Libyans, accused) et D2 (Two Libyan suspects were indicted). La SCU obtient un poids égal à 4.

La deuxième SCU parle de la date d'accusation des suspects de Lockerbie. Elle est présente uniquement dans 3 résumés de référence A1 (in 1991), B1 (in 1991) et D2 (in 1991). La SCU aura un poids égal à 3.

TABLE 2.2 – *Les phrases du résumé de référence sont indexées avec une lettre et un numéro : la lettre indique de quel résumé la phrase provient et le nombre, la position de la phrase dans celui-ci. Les résumés ont été pris dans le jeu de test de DUC 2003.*

	Phrase
A1	In 1998 two Libyans indicted in 1991 for the Lockerbie bombing were still in Libya.
B1	Two Libyans were indicted in 1991 for blowing up a Pan Am jumbo jet over Lockerbie, Scotland in 1988.
C1	Two Libyans, accused by the United States and Britain of bombing a New York bound Pan Am jet over Lockerbie, Scotland in 1988, killing 270 people, for 10 years were harbored by Libya who claimed the suspects could not get a fair trail in America or Britain.
D2	Two Libyan suspects were indicted in 1991.

2.6 Conclusion

Que ce soit dans le domaine de la vidéo, du texte ou de la parole, nous avons pu constater que la création d'un résumé automatique consistait à essayer de sélectionner les informations les plus importantes tout en essayant de minimiser leur redondance. Les travaux dans le domaine se sont généralement focalisés sur deux points :

- la manière d'extraire et de juger de la pertinence d'une information
- la proposition d'un algorithme qui permette à la fois de sélectionner et de minimiser la redondance d'informations

Le but dans un *zapping* est de sélectionner dans une collection de vidéos des sous-séquences ayant un intérêt notable pour les utilisateurs et de minimiser la redondance d'informations des sous-séquences sélectionnées. Nous constatons que le *zapping* est une forme particulière du résumé automatique. Nous retrouvons bien cette question de redondance et de fonction d'intérêt, mais cette dernière n'est pas la même que dans le résumé automatique. Nous verrons, dans le Chapitre 6, quelles peuvent être les fonctions d'intérêt liées aux zapping puis comment les intégrer à un algorithme de résumé par extraction et les évaluer.

Deuxième partie

Extraction du contenu

Les systèmes de résumés automatiques audios se décomposent en deux niveaux. Le premier niveau réalise une transcription automatique du signal de parole via un système de transcription de la parole. Cette transcription est fournie au deuxième niveau qui va lui appliquer une méthode de résumé texte. Malheureusement, la transcription automatique fournie par un système de RAP est souvent imparfaite, ce qui aura un impact négatif sur les performances du système. La transcription (et donc l'extraction de son contenu parlé) est très importante pour un système de résumé automatique.

Les performances d'un système de RAP sont liées aux documents qu'il décode et au contexte de la tâche. Généralement, les systèmes de transcription obtiennent sur des données de radios journalistiques un Taux d'Erreur de Mots – Word Error Rate (WER) de 10%, sur de la parole conversationnelle un WER compris entre 20% et 30% et sur des réunions un WER compris entre 30% et 40% (et parfois bien plus (Fiscus et al., 2007)). Les difficultés et les moyens à mettre en oeuvre pour décoder de manière robuste un document sont très divers. Dans le contexte de données web telles qu'on les trouve sur Youtube ou Dailymotion, nous observons deux principaux problèmes.

Le premier est lié au vocabulaire utilisé dans les vidéos. Le vocabulaire peut appartenir à un domaine scientifique, contenir des termes politiques, etc... et il n'est pas forcément bien couvert par le lexique d'un système de RAP à grand vocabulaire. On appelle cela des mots Hors Vocabulaires – Out-Of Vocabulary (OOV), ce sont des mots qui sont absents du vocabulaire d'un système de RAP. Un mot peut être OOV soit à cause de la taille limitée du vocabulaire, soit parce qu'il n'existait pas au moment de la création du lexique. Dans (Watson, 2003), l'auteur estime qu'il y a environ plus de 50 mots créés par jour. Ces nouveaux mots viennent de sources, de domaines différents incluant :

- Des termes scientifiques : comme les noms de nouveaux médicaments, nouveaux gènes, nouvelles espèces, nouvelles étoiles, nouvelles méthodes, nouveaux concepts...
- Des termes de la vie sociale : marques, nouveaux produits, nouveaux films, etc...
- Des termes politiques : noms de politiciens, noms de législations, etc...
- Des termes étrangers : ces nouveaux mots constituent la majeure partie des OOV, et le nombre de ces mots augmente considérablement.

Parce que les mots ne sont pas dans le lexique d'un système de RAP, les segments contenant des mots OOV par rapport au lexique sont toujours reconnus comme étant des mots Dans le Vocabulaire – In Vocabulary (IV), perturbant ainsi la transcription.

Le deuxième problème auquel peut faire face un système de RAP sur les vidéos disponibles depuis Youtube ou Dailymotion sont les conditions acoustiques. En effet, lors de l'enregistrement d'une vidéo, le signal audio ne véhicule pas seulement l'information sémantique (le message) mais aussi beaucoup d'autres informations relatives à la personne qui parle : sexe, âge, accent, santé, émotion, etc... ainsi que

des informations relatives aux canaux : micro, milieu bruité, écho, etc... Toutes ces informations présentes dans le signal audio peuvent perturber et dégrader le décodage.

Le système de RAP doit être assez robuste aux conditions acoustiques difficiles pour pouvoir fournir une transcription de qualité. La robustesse d'un système est définie par sa capacité à faire face à des événements nouveaux non prévus initialement. C'est un domaine de recherche très fertile et de nombreuses techniques ont été développées pour améliorer chaque composante du système. Les approches généralement suivies consistent à améliorer la tolérance aux variabilités diverses ou à les atténuer :

- paramètres acoustiques : des traitements spécifiques peuvent être mis en œuvre pour rendre les paramètres acoustiques plus robustes au bruit. L'objectif est de normaliser l'espace des vecteurs acoustiques.
- modèles acoustiques : une des principales contraintes pour le bon apprentissage des modèles acoustiques est la quantité de données disponibles pour l'estimation des paramètres du modèle. Chaque unité phonétique doit être suffisamment représentée dans le corpus d'apprentissage. En outre, un problème de modélisation se pose lorsque les données d'apprentissage sont très différentes des données de la tâche ciblée. Les modèles acoustiques peuvent alors être adaptés, afin de mieux faire correspondre leurs paramètres aux différentes prononciations des unités phonétiques pouvant être rencontrées.

Dans ce chapitre, nous allons présenter deux méthodes liées à l'extraction du contenu. Dans le chapitre 3, nous proposerons un système de détection de termes rapide dans des milieux bruités puis, dans le chapitre 4, un nouveau cadre de normalisation robuste de données acoustiques.

Chapitre 3

Détection de termes à la volée

Sommaire

3.1 Introduction

La recherche des mots clefs dans un flux audio (*Word Spotting*) est une des tâches historiques en reconnaissance de la parole. Elle a suscité un intérêt fort dès les années 90, non seulement parce qu'elle répondait à un besoin particulier, mais aussi parce que les limites des systèmes de RAP et celles des machines rendaient

cette tâche plus accessible que la reconnaissance de parole continue en grand vocabulaire.

Plus récemment, cette tâche s'est étendue à la détection de termes dans des bases audios. Pour encourager la recherche et le développement de cette technologie, National Institute of Standards and Technology (NIST) organise une série d'évaluations pour la Détection de termes dans un document audio - Spoken Term Detection (STD). La première évaluation pilote réalisée en 2006 se fit sur trois conditions : radio journalistique, conversation téléphonique et conférence. Trois langues étaient associées à cette évaluation : anglais, arabe et mandarin.

Nous présenterons dans la section 3.2 un bref état de l'art du STD. Puis dans la section 3.3, nous présenterons notre contribution dans le domaine.

3.2 Etat de l'art

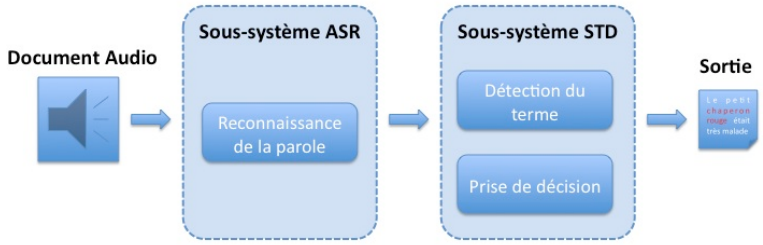

FIGURE 3.1 – *Présentation d'un système de détection de termes dans un document audio.*

La Figure 3.1 illustre le cadre de travail standard d'un système STD. Dans ce cadre de travail, le signal de la parole est d'abord transcrit automatiquement (mot, phonèmes, réseau de confusion...) puis une recherche du terme [1] est effectuée sur cette transcription.

Une façon simple et naturelle d'implémenter un STD est de se baser sur un système de RAP classique. Dans cette implémentation, le système de RAP est un système à grand vocabulaire qui transcrit l'audio en mots (ou treillis de mots). La détection du terme s'effectue en faisant une simple recherche dans la transcription. Dans (Miller et al., 2007), l'auteur a utilisé cette approche pour la campagne d'évaluation de NIST en 2006 et obtenu les meilleures performances dans la catégorie : recherche de termes en anglais sur de la parole téléphonique.

Les différentes approches de système STD proposées pour détecter des mots permettent d'obtenir une bonne précision car l'information lexicale est utilisée. Cependant, les systèmes STD souffrent d'une lacune concernant la détection des termes OOV. Les OOV n'apparaîtront jamais dans un treillis généré par le système de transcription de mots et, par conséquent, ne peuvent pas être détectés. Pour

1. Un terme est ici défini comme une suite de mots

résoudre ce problème, la plupart des systèmes font de la reconnaissance sur des unités sous-lexicales de mots (par exemple les phonèmes). Dans (Wechsler et al., 1998), les auteurs proposent un système STD basé sur la reconnaissance de phonèmes. Ainsi, le système génère un treillis de phonèmes et le terme (converti en séquence de phonèmes) est recherché dans le réseau de phonèmes. L'idée de cette représentation en unités phonétiques est de construire un système capable de représenter de nouveaux mots et de capturer des contraintes lexicales.

3.3 Contribution

Dans le cadre d'une utilisation pour le résumé automatique, les systèmes STD souffrent de nombreux problèmes. D'une part, le STD ne fait que rechercher dans un flux audio la requête, mais ne remet jamais en cause le contexte de la phrase contenant la requête. De plus, pour toutes ces tâches de détection, les performances reportées dans la littérature sont bonnes sur des conditions propres et spécialement sur les données de radio largement utilisées par les systèmes de RAP (Fiscus et al., 1998). Cependant, dans des conditions plus difficiles comme un enregistrement dans un contexte bruité ou un discours spontané, les performances sont dégradées (Pinto et al., 2008; Yu et al., 2004a; Saraclar and Sproat, 2004).

Nous proposons ici un système de détection de termes où le contexte est guidé par la requête. La détection des termes doit se faire sur de très grandes bases de données dans un temps raisonnable. Nous proposons un système de détection de termes en temps réel.

Notre système est basé sur une architecture à deux niveaux dans laquelle le premier niveau permet de faire un filtrage phonétique du flux de parole audio tandis que le second niveau implique un système de RAP à grand vocabulaire. Ces deux composantes en cascade sont optimisées afin qu'elles maximisent séquentiellement le rappel (au premier niveau) et la précision (au second niveau).

Au premier niveau, une recherche rapide est réalisée. Cette étape est vue comme une phase de filtrage qui a pour but d'accepter ou rejeter les segments selon leur probabilité de contenir le terme ciblé. Nous présentons un schéma général dans lequel le terme prononcé est projeté dans un graphe de filtres phonétiques. Le graphe résultant est ensuite élagué afin de minimiser sa complexité tout en maximisant sa capacité de détection.

Au second niveau, les segments de parole qui passent la première étape du filtre sont traités par un système de recherche de termes basé sur la RAP, avec l'objectif de raffiner la détection du terme. Nous proposons d'améliorer le taux de détection du terme en intégrant la requête dans le système. Cette intégration est basée sur l'algorithme de décodage guidé (DDA) qui a été précédemment proposé dans (Lecouteux et al., 2006).

Cette section est organisée comme suit : la section 3.3.1 présente l'architecture

globale de notre détection de termes, la section 3.3.2 décrit le premier niveau, qui a pour but d'identifier les segments de paroles dans lesquels la requête est probablement présente. Nous présentons le système de filtrage acoustique où différents classifieurs vont être testés. Dans la section 3.3.3, nous présentons le second niveau où une stratégie de décodage guidé est utilisée pour raffiner la détection du terme. Dans la section 3.3.4, nous présentons les expériences. Les résultats sur un corpus propre et sur un corpus de parole spontanée sont reportés et discutés dans la section 3.3.5.

3.3.1 Architecture du système

A partir d'une requête écrite formée d'une séquence de mots, le système de détection de termes à la volée est supposé rechercher dans un flux de parole toutes les occurrences et les notifier au fur et à mesure de l'analyse.

L'architecture du système est composée de deux niveaux dans lesquels la précision et le rappel sont séquentiellement optimisés. Le premier niveau, strictement acoustique, est composé d'un outil qui identifie les segments de parole susceptibles de contenir le terme recherché. Ces segments sont ensuite passés au deuxième niveau qui est basé sur un algorithme de reconnaissance de la parole guidé par la requête.

La requête écrite est d'abord transcrite phonétiquement en utilisant un lexique de prononciation et un phonétiseur à base de règles qui produit un graphe phonétique regroupant l'ensemble des variantes de prononciation. A partir de cette représentation phonétique, un filtre acoustique est construit. Celui-ci est composé d'un graphe de filtres phonétiques. Ces filtres phonétiques peuvent être basés sur des modèles statistiques comme le Modèle de Mélanges Gaussiens – Gaussian Mixture Model (GMM) ou le Multi-Layer Perceptron (MLP). Dans la suite, le graphe de filtres phonétiques est appelé *filtre acoustique*, tandis que les *filtres phonétiques* opèrent au niveau de chaque nœud.

Ici notre but est de maximiser la précision ainsi que les coûts de calcul sous la contrainte d'un rappel maximal.

Chaque segment de parole sélectionné par le premier niveau est passé au second niveau, comme montré dans la Figure 3.2. Le second niveau est un système de RAP basé sur l'algorithme de décodage guidé. A cette étape, le but du système de RAP est de raffiner la détection, en se focalisant sur l'amélioration de la précision.

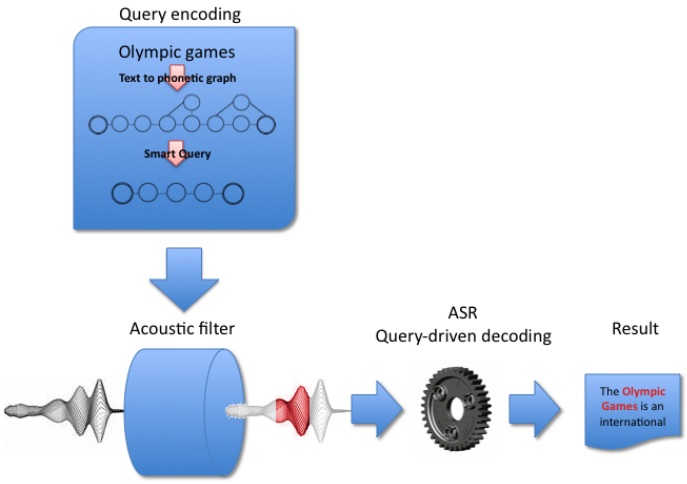

FIGURE 3.2 – *Architecture d'un système de détection de termes à la volée.*

3.3.2 Filtre Acoustique

3.3.2.1 Encodage de la requête

La première étape consiste à transcrire une requête écrite en chaîne phonétique. Toutes les variantes de prononciation d'un terme sont extraites à partir d'un dictionnaire. Dans le cas où le terme n'est pas présent dans un dictionnaire, la transcription phonétique est obtenue en utilisant les règles d'un système de transcription phonétique. Ensuite, toutes les transcriptions phonétiques sont compilées dans un graphe de phonèmes où chaque chemin représente une variante de prononciation, comme illustré dans la Figure 3.3.

L'approche classique utilisée pour détecter le terme est d'aligner le graphe et le signal dans une fenêtre glissante et prendre les décisions de détection en fonction de la probabilité d'émission obtenue. Cette approche est sous-optimale en terme de consommation de ressource CPU notamment parce que prendre la décision de détecter le terme uniquement sur la probabilité du chemin total peut être inutile. Les scores intermédiaires lors de l'alignement peuvent être une information suffisante pour stopper l'alignement. Nous proposons d'implémenter un élagage à chaque nœud du graphe où le filtre phonétique doit être capable d'arrêter ou de continuer l'exploration du graphe. Ce processus de filtrage est décrit de manière plus approfondie dans le prochain paragraphe où nous présenterons des filtres à base de GMM et de MLP.

Considérant cette stratégie d'élagage, il est clair que la partie la plus discriminate du graphe doit être évaluée en premier dans le but de réduire le temps de calcul tout en préservant la précision. Par conséquent, le graphe peut être réduit en

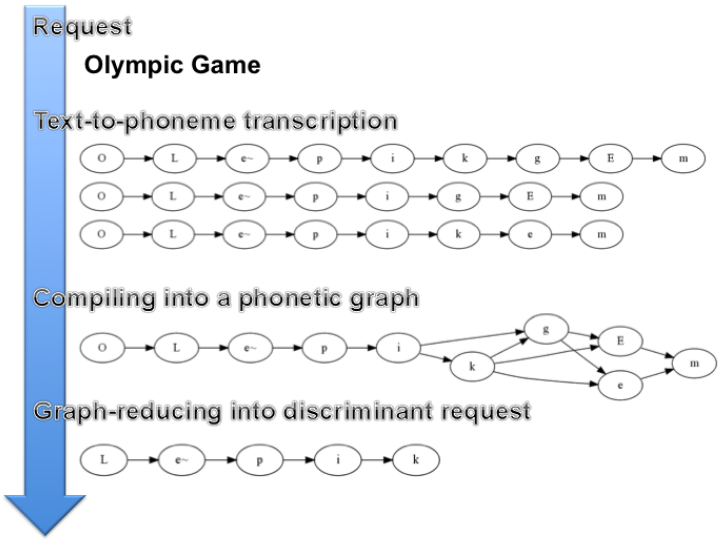

FIGURE 3.3 – *De la requête écrite à la requête minimale : La requête écrite est transcrite en graphe de prononciation. Le meilleur sous-graphe qui maximise la précision tout en minimisant les coûts CPU est extrait pour construire la requête minimale.*

fonction de la complexité et de la capacité discriminative des sous-graphes. Nous proposons un algorithme de réduction de graphes des variantes de prononciation, décrit dans le prochain paragraphe.

Finalement, la mise au point d'une requête minimale s'effectue en quatre étapes :

- Transcrire la requête écrite en phonèmes.
- Assembler l'ensemble des variantes de prononciation dans un graphe de prononciation.
- Étendre le graphe de phonèmes à un graphe d'états.
- Réduire le graphe en recherchant le meilleur sous-graphe selon le compromis précision et complexité.

Ce processus permet d'optimiser la détection du terme efficacement qui est un point critique durant la phase de détection de termes dans un flux.

3.3.2.2 Filtre phonétique basé sur des GMM

Les filtres à base de GMM utilisent les modèles acoustiques du système de RAP. Chaque filtre f_i est associé à un état émetteur S_i extrait depuis le jeu de Modèle de Markov Caché – Hidden Markov Model (HMM) du système de RAP. Le graphe phonétique est développé selon la topologie des HMM : chaque phonème dépendant du contexte est découpé suivant une séquence de n nœuds d'état dépendant du contexte.

Les filtres d'état dépendant du contexte doivent être capables d'arrêter l'explo-

ration du graphe quand l'observation X_t est en dehors du modèle. Ceci est réalisé en précisant, pour chaque filtre, un seuil minimal c_i pour la probabilité $P(X_t|S_i)$:

$$P(X_t|S_i) = \frac{ll(X_t|S_i)}{ll(X_t|UBM)} \tag{3.1}$$

où X_t est une trame de 39 coefficients : dont 12 coefficients cepstraux Perceptual Linear Predictive (PLP), l'énergie et les dérivés première et seconde. L'Universal Background Model (UBM) est un modèle générique permettant de structurer l'espace acoustique indépendamment du contexte phonétique. Ici, l'UBM est un GMM composé de 64 composantes, estimées en utilisant la procédure d'Expectation Maximization (EM) sur l'ensemble des données du corpus d'apprentissage.

Le seuil de coupure du filtre c_i est estimé sur le corpus d'entraînement en calculant la valeur supérieure c_i sous la contrainte $ll(X_t|S_i) > c_i, \forall X_t \in \Omega_i$, où Ω_i est la partie du corpus d'entraînement émis par l'état S_i. Lorsque ce seuil est atteint, l'algorithme Viterbi est stoppé.

Quand le dernier nœud du graphe est atteint (quand tous les filtres phonétiques ont été passés), une dernière règle est appliquée au niveau du segment. Cette règle repose sur la probabilité du chemin total du terme, normalisée par la durée du segment. Nous cherchons d'abord dans le corpus d'entraînement la probabilité la plus basse du terme. Nous utilisons la valeur la plus basse C comme un seuil de rejet. Ensuite, chaque segment de parole est accepté $X = \{X_t\}$ si elle satisfait la contrainte :

$$P(X|S) > C \tag{3.2}$$

où S correspond à la séquence de la chaîne phonétique.

3.3.2.3 Filtre phonétique basé sur un MLP

Des méthodes discriminatives pour la détection de mots clefs ont été récemment traitées par plusieurs auteurs dans (Keshet et al., 2009; Ezzat and Poggio, 2008; Benayed et al., 2004). Ces approches ont été motivées par le fait que la détection peut être vue comme une tâche de classification (en rejetant/acceptant les hypothèses). Le but du filtre acoustique est de rejeter les segments non-pertinents. Considérant cela, des approches discriminatives plus efficaces peuvent être utilisées pour le filtrage de segments. Nous proposons d'utiliser le MLP comme filtre phonétique discriminant.

Le filtrage à base de MLP repose sur le même principe de ce qui a été utilisé avec le filtre à base de GMM : les filtres phonétiques à base de GMM sont simplement substitués par un classifieur MLP pour estimer les probabilités.

Chaque sortie du MLP correspond à un état S_i, un jeu de phonèmes indépendant du contexte. Le filtre à base de MLP opère au niveau des trames. Le vecteur d'entrée est composé de 351 coefficients, résultant de la concaténation de 9 trames de 39 coefficients. La couche cachée est composée de 2 000 neurones et la couche de sortie de 108 neurones. Chaque neurone représente un état d'un phonème indépendant du contexte. Le MLP est entraîné sur un grand corpus en utilisant l'approche de rétropropagation du gradient (back-propagation).

Après normalisation, chaque neurone de la couche de sortie du MLP est supposé fournir une estimation de la probabilité $P(X_t|S_i)$ que la trame X_t appartienne à l'état S_i. Le filtre phonétique à base de MLP est ensuite intégré dans le graphe de filtres d'une façon similaire aux filtres GMM : un seuil de coupure c_i est associé à chacune des sorties du réseau de neurones permettant le rejet ou l'acceptation de l'hypothèse. La valeur c_i est calculée sur le corpus d'entraînement en estimant la valeur la plus basse obtenue par l'état S_i. La valeur du segment C est utilisée pour rejeter l'hypothèse quand la probabilité du chemin total $P(X|ph)$ est plus basse que C. ph représente la transcription phonétique de la requête.

La probabilité du chemin total est estimée par un alignement Viterbi basé sur les probabilités du MLP et normalisée selon la taille du chemin considéré.

Finalement, la stratégie du filtre est strictement similaire à celle utilisée dans le cas du GMM. Le MLP est utilisé pour estimer les probabilités et intégré dans le filtre phonétique en respectant le schéma de filtrage total désigné pour le filtrage à base de GMM.

3.3.2.4 Requête minimale

Nous partons de l'idée que, dans la requête phonétique il peut exister une sous-partie de la requête ayant une capacité significativement plus discriminante pour différentes raisons. Premièrement, plus la fréquence d'une séquence de phonèmes est basse, plus elle est spécifique à la requête. Deuxièmement, selon les performances du filtre phonétique, l'utilisation d'une partie de la requête peut fournir plus rapidement des coupures dans l'exploration du graphe. Par exemple, la recherche du terme "jeux olympiques" peut être réduite à la suite "eux oly" ; la recherche de cette sous-séquence devrait permettre d'obtenir un gain significatif en termes de temps de calcul sans avoir un impact sur la précision. Il est important de noter que le taux de rappel n'est pas influent sur la réduction de la requête. Une requête recherchée par la totalité de la chaîne phonétique est nécessairement notée par une sous-chaîne phonétique. Notre idée est de trouver la sous-chaîne phonétique optimale en termes de rappel et de complexité.

A ce point, la question est "comment trouver le meilleur sous-graphe ?". La première étape est de définir une fonction objective $F_{ob}(f)$ qui quantifie la complexité et la précision pour un filtre donné f associé à une requête W.

Pour simplifier, nous linéarisons le graphe en concaténant les modèles en com-

pétition en un filtre phonétique commun. Les filtres résultants $f = \{f_i\}_{i=0,...,n-1}$ sont composés en cascade de n filtres phonétiques f_i, correspondant à une séquence phonétique h et à la séquence d'état associé S_i. La pertinence de f est estimer via une fonction objective $F_{ob}(f)$ qui combine une fonction de précision $acc(f)$ et une fonction de complexité $cpx(f)$.

L'indice de complexité $cpx()$ permet d'estimer un nombre de trames qui peut être envoyé à chaque filtre phonétique f_k. La probabilité d'atteindre f_i dépend de la probabilité de passer tous les filtres précédents $f_{k,i>k>0}$ dans la cascade de filtres. En effet, pour estimer la probabilité de passer un filtre f_i, nous associons à chacun, une variabilité aléatoire $D_i(X_t)$ qui indique si une trame traverse le filtre ou pas. $D_i(X_t)$ est à 1 quand l'inégalité $ll(X_t|S_i) > c_i$ est vraie, et $D_i(X_t)$ est à 0 dans l'autre cas. La probabilité *a priori* de passer f_i est désignée par $P(D_i = 1)$. Les probabilités *a priori* sont estimées en sommant le nombre de trames qui passent le filtre dans le corpus d'entraînement sur le nombre total de trames.

La probabilité *a priori* d'atteindre le filtre phonétique i est le produit de la probabilité *a piori* $P(D_i = 1), k < i$ de passer les filtres précédents f_k.

Finalement, le coût de calcul de f est estimé en sommant toutes les probabilités *a priori* d'atteindre les filtres qui composent f :

$$cpx(f) = g * (1 + \sum_{k=0}^{n} \prod_{i=0}^{k} P(D_i = 1)) \tag{3.3}$$

où g est une constante de coût du calcul qui a été mise à 1 dans nos expériences.

La précision du filtre $f = \{f_0, ..., f_{n-1}\}$ peut être définie comme la probabilité *a priori* que f effectue une détection correcte. Cette valeur dépend de deux éléments. Premièrement, la requête minimale peut aboutir à une fausse détection même si les deux chaînes phonétiques sont identiques. Par exemple, la recherche de "Jeux Olympiques" en utilisant la sous-séquence "piqu" va probablement retourner des erreurs, acoustiquement proches, comme "piquer". Deuxièmement, le filtre phonétique peut faire des erreurs, et retourner de mauvaises réponses.

Le premier élément peut être évalué en estimant dans le corpus d'entraînement la probabilité du terme ciblé W, quand la séquence phonétique ph est rencontrée. Cette valeur est calculée :

$$P(W|ph) = \frac{|W|}{|ph|} \tag{3.4}$$

où $|W|$ est le nombre d'occurences du terme W dans le corpus d'entraînement et $|ph|$ est le nombre de segments de la séquence phonétique ph dans le même corpus.

D'une manière similaire, la précision du filtre phonétique $P(S_i|D_i = 1)$ représente la probabilité *a priori* que le filtre f_i trouve la requête. Cette valeur est estimée

sur le corpus d'apprentissage, en comptant le nombre de trames qui sont passées par le filtre et effectivement émises par l'état S_i.

Finalement, la précision globale du filtre f est estimée selon la précision de chaque filtre phonétique f_i. Celle-ci se calcule ainsi :

$$acc(f) = P(W|ph) * \prod_{i=0}^{n} P(S_i|D_i = 1) \tag{3.5}$$

La fonction objective est définie comme la différence de la précision et de la complexité :

$$F_{ob}(f) = acc(f) - \gamma \cdot cpx(f) \tag{3.6}$$

où γ est un facteur arbitraire déterminé empiriquement.

Cette fonction est utilisée pour déterminer le rang des sous-requêtes. La meilleure *sous-requête* est celle qui maximise F_{ob} :

$$F^{sq}(f) = \arg\max_{k} F_{ob}(f^k) \tag{3.7}$$

où (f^k) sont les sous-requêtes.

Pour chaque requête W, la sélection de la sous-requête est réalisée par une évaluation exhaustive de toutes les sous-requêtes possibles.

Cette technique de recherche de meilleure chaîne phonétique est utilisée pour les deux systèmes à base de GMM et de MLP. Cependant, la fonction F_{ob} repose sur la précision du filtre phonétique f_i qui est dépendant de la probabilité d'estimer l'état d'une trame.

3.3.3 Décodage guidé par la requête

Le but de cette étape est d'affiner la détection réalisée dans le premier niveau par le processus de filtrage. Le processus de filtrage détecte des segments de parole où le terme recherché peut avoir été prononcé. Ces segments de parole sont envoyés au système de RAP pour une passe de décodage. Afin d'être sûr que le segment de parole contienne le terme recherché, nous élargissons le segment avant et après la zone sélectionnée. Dans nos expérimentations, nous utilisons une valeur de 2 secondes sur les bords des segments.

Rechercher un terme en utilisant un système de RAP est connu pour avoir une bonne précision étant donné que la probabilité *a priori* d'avoir le terme recherché dans une transcription est basse. D'un autre côté, les erreurs de transcription peuvent introduire des fautes et tendent à laisser de côté des termes, en particulier

sur des grandes requêtes : plus le terme cherché comprend de mots, plus le risque de rencontrer une erreur dans la transcription est grande. Afin de limiter ce risque, la probabilité *a priori* de la requête est légèrement favorisée par l'algorithme de décodage : Algorithme de Décodage Guidé – Driven Decoding Algorithm (DDA) (Lecouteux et al., 2006).

Cet algorithme a pour but d'aligner une transcription *a priori* et l'hypothèse courante donnée par le moteur de reconnaissance de la parole, en utilisant un algorithme d'alignement qui minimise la distance d'édition entre l'hypothèse courante et la transcription *a priori*.

Une fois l'hypothèse synchronisée avec la transcription, l'algorithme estime un score de synchronie locale noté α. Ce score est basé sur le nombre de mots dans l'historique à court terme, lequel a été correctement aligné avec la transcription : seules trois valeurs sont utilisées, correspondant respectivement à un alignement complet du tri-gramme courant, un alignement complet du bi-gramme courant et un alignement du mot seulement. Les valeurs de α sont empiriquement déterminées sur un corpus de développement. Ensuite, les probabilités des tri-grammes sont réestimées :

$$\tilde{P}(w_i|w_{i-1}, w_{i-2}) = P^{1-\alpha}(w_i|w_{i-1}, w_{i-2}) \tag{3.8}$$

où $\tilde{P}(w_i|w_{i-1}, w_{i-2})$ est la probabilité du tri-gramme réestimée pour un mot w_i connaissant l'historique w_{i-1}, w_{i-2} et $P(w_i|w_{i-1}, w_{i-2})$ est la probabilité initiale du tri-gramme.

Ici, nous utilisons le DDA comme un outil permettant de valider les segments précédemment identifiés comme bons candidats par le filtre acoustique. Le terme ciblé utilisé comme une transcription *a priori* permet d'augmenter le score linguistique de l'hypothèse correspondant à la requête.

En utilisant cette méthode, il est possible de rechercher dans la transcription des termes contenant des OOV. Pour effectuer une telle recherche, il faut transcrire phonétiquement l'OOV et le rajouter dans le modèle de langage. La transcription phonétique du mot est obtenue avec l'application LIA-PHON [2]. Cette application est basée sur un ensemble de règles phonétiques, ce qui permet de traiter un vocabulaire ouvert. Pour modéliser les mots OOV dans le modèle de langage, nous proposons de regrouper les OOV dans une classe dont toutes les instances seront vues comme un seul mot, noté *inc*. Le mot *inc* correspond donc à l'ensemble des mots présents dans le corpus d'apprentissage qui sont absents du lexique du système de RAP. La probabilité du mot *inc* est estimée classiquement : nous remplaçons tous les mots dans le corpus d'entraînement qui ne sont pas dans le lexique par le mot *inc*. Les probabilités n-grammes de chaque mot mot OOV sont interpolées par la probabilité du mot *inc*.

2. http://lia.univ-avignon.fr/fileadmin/documents/Users/Intranet/chercheurs/bechet/download_fred.html

En effet, la probabilité d'un tri-gramme contenant un mot OOV w_{oov} peut être décomposée selon la probabilité conditionnelle du mot inconnu et la probabilité du w_{oov} :

$$P(w_{oov}|w_{i-1}, w_{i-2}) = P(w_{oov}|inc) * P(inc|w_{i-1}, w_{i-2}) \tag{3.9}$$

Ici, nous utilisons une valeur fixée *a priori* pour $P(w_{oov}|inc)$. Dans les expériences ci-dessous, cette probabilité est fixée à 10^{-4}.

3.3.4 Cadre de travail

3.3.4.1 Speeral

Les expériences sont réalisées en utilisant le système de RAP grand vocabulaire du LIA, Speech RAL (SPEERAL) (Linarès et al., 2007). Ce système utilise un algorithme A* pour le décodage et des HMM pour la modélisation acoustique. Le lexique contient 65 000 mots et le modèle de langage est un modèle tri-gramme estimé sur 200 millions de mots du journal Le Monde et sur environ 1 million de mots du corpus d'entraînement de la campagne d'évaluation Evaluation des Systèmes de Transcription Enrichie d'Emissions Radiodiffusées (ESTER).

Les paramètres acoustiques sont composés de 12 coefficients PLP, l'énergie et de leurs dérivées première et seconde, soit 39 dimensions pour la transcription. Deux configurations sont réalisées dans ces expériences selon leur vitesse de décodage, exprimée comme un facteur de temps réel. Nous utilisons le système en temps réel (noté 1xRT) et le système en trois fois le temps réel (noté 3xRT). Le système 1xRT utilise des modèles acoustiques composés de 24 gaussiennes, tandis que le système 3xRT repose sur des modèles de 64 gaussiennes par état.

3.3.4.2 Le corpus EPAC et ESTER

ESTER est un corpus développé pour la campagne d'évaluation ESTER-2005. Il est composé de 80 heures de radio d'actualité en français. Nous utilisons ces données comme un corpus d'entraînement pour estimer les GMM et le MLP. Les tests sont effectués sur le corpus Exploration de masse de documents audio pour l'extraction et le traitement de la parole conversationnelle (EPAC) fourni par le projet éponyme (Estève et al., 2010). Ce projet a pour but d'étudier des méthodes pour la reconnaissance et la compréhension de la parole spontanée. Environ 11 heures de parole spontanée ont été extraites de la base de données non transcrite d'ESTER et ont été manuellement annotées selon un niveau de spontanéité : le niveau 1 pour la parole lue et le niveau 10 pour la parole très spontanée. Ici, nous considérons 2 classes : moyenne (correspondant aux niveaux 1 à 4) et élevée (qui correspond aux niveaux 5 et plus).

Dans la suite, le corpus EPAC est utilisé seulement comme corpus de test. Les filtres acoustiques et les requêtes minimales sont calibrées sur les données du corpus d'apprentissage d'ESTER. Le jeu de test est composé de 270 requêtes incluant 130 requêtes IV, 70 OOV et 70 requêtes hybrides ; ces dernières incluant les requêtes IV et OOV. La taille des requêtes est composée de 1 à 4 mots, les requêtes hybrides sont composées bien entendu d'au moins 2 mots. Les performances de la baseline du système de RAP dans la configuration 1xRT sont de 40.3% WER. Dans ce corpus, on peut distinguer deux parties : la partie moyennement spontanée, qui obtient un WER de 33.2% et la partie hautement spontanée, qui obtient un WER de 47.2%. Dans la configuration 3xRT, ce taux décroit à 31.1% et 43.5%.

3.3.5 Résultat

3.3.5.1 Évaluation des graphes phonétiques

Le filtre acoustique est évalué dans différentes configurations. Notre système primaire consiste à trouver un terme dans lequel un alignement Viterbi entre le graphe phonétique et la fenêtre de signal est réalisé. Les modèles acoustiques sont des états du HMM (phonème indépendant du contexte) entraînés sur le corpus de données ESTER. Nous allons étudier d'abord l'impact des techniques de coupure (*A-GMM*) puis de la requête minimale (*A+SR-GMM*). Finalement, nous évaluerons le système à base de MLP avec le système de coupure et de requête minimale (*A+SR-MLP*). Dans le Tableau 3.1 nous montrons les résultats sur le corpus de parole spontanée EPAC en termes de taux de rappel, facteur de temps (temps passé par le filtre acoustique à détecter le terme, normalisé par la durée du signal audio) et taux de filtrage (calculé comme étant la durée cumulée de segments de parole, normalisé par la durée du segment en entier).

TABLE 3.1 – *Les performances du filtrage acoustique par un alignement Viterbi (Baseline), avec filtrage GMM et coupure (A-GMM), avec coupure et requête compressée qui sont couplées à un filtre GMM (A+SR-GMM) et un système à base de MLP (A+SR-ML). Les performances sont reportées en terme de rappel, taux de filtrage et facteur de temps.*

	Baseline	A-GMM	A+SR-GMM	A+SR-MLP
Rappel	0.99	0.97	0.97	0.97
Taux de filtrage	0.65	0.33	0.37	0.23
Facteur de temps	0.1	0.05	0.03	0.05

Les résultats montrent que les techniques de coupure permettent de réduire nettement le nombre de segments acceptés. Les requêtes compressées n'ont pas d'impact sur le taux de filtrage, mais permettent d'améliorer le facteur de temps (qui est pratiquement réduit par deux). Le MLP démontre l'intérêt d'utiliser des approches discriminantes dans une tâche de filtrage. Comme attendu, les performances MLP sont plus sélectives (de 37% à 23%) pour un rappel similaire.

3.3.5.2 Évaluation de la stratégie de décodage de la requête guidée

Ici, les performances des deux systèmes sont évaluées. Nous reportons dans la *baseline* les résultats obtenus avec le système de RAP du LIA en temps réel (*ASR-1xRT*), ainsi que le système en 3 fois le temps réel (*ASR-3xRT*). Pour ces deux systèmes, la recherche de termes est directement réalisée sur les sorties du système de RAP.

Ensuite, nous estimons le taux de détection en utilisant le DDA seulement, sans filtre acoustique (*DDA-1xRT*). Considérant le filtrage de flux de parole, seulement 37% de la durée totale de parole ont été passés au système de reconnaissance (et 23% pour le MLP). Nous proposons d'utiliser une configuration en 3xRT pour que le processus complet satisfasse la contrainte du temps réel.

Les performances obtenues avec les méthodes de filtrage complet basé sur le GMM (GMM+DDA-3xRT) et celui du MLP (MLP+DDA-3xRT) sont reportées dans le Tableau 3.2 en terme de F-mesure, qui est calculée selon la moyenne harmonique de rappel et de précision.

TABLE 3.2 – *F-Mesure sur le corpus de test EPAC.*

Système	IV	OOV	Hybrid	Total
ASR-3xRT	0.66	x	x	x
ASR-1xRT	0.56	x	x	x
DDA-1xRT	0.65	0.79	0.75	0.72
DDA-AF-GMM	0.78	0.86	0.76	0.77
DDA-AF-MLP	0.76	0.89	0.80	0.80

Les résultats montrent que le DDA apporte des améliorations significatives dans tous les cas. En utilisant l'algorithme de DDA en temps réel, la F-Mesure est similaire à celle obtenue avec le système *ASR-3xRT*, lequel est clairement en dehors des limites de temps réel requis pour un processus de détection de termes à la volée. Les deux systèmes bénéficient du filtrage acoustique et de l'algorithme de DDA. On observe pour le système *ASR-1xRT*, sur les requêtes IV, un gain de F-Mesure de 20%. Ceci montre qu'intégrer la requête dans le processus de RAP permet de réaliser un décodage guidé par la requête, ce qui limite les erreurs sur les énoncés cibles.

3.3.5.3 Détection des performances selon le niveau de spontanéité

Les expériences suivantes cherchent à évaluer l'impact du niveau de spontanéité sur le taux de détection. Nous utilisons la classification en niveau de spontanéité moyenne et élevée, en se basant sur le système de STD à deux niveaux.

Les résultats pour le système de RAP avec un décodage DDA (*DDA-1xRT*) sont reportés dans le Tableau 3.3. Comme attendu, les performances sont affectées par

TABLE 3.3 – *Le taux de détection (rappel, précision et f-mesure) selon le niveau de spontanéité avec un décodage guidé* **DDA-1xRT**. *Les tests sont conduits sur le corpus de tests EPAC, en utilisant 270 requêtes composées de 1 à 4 mots (70 requêtes OOV, 70 hybrides et 130 requêtes IV)*

Système	Niveau de spontanéité	Rappel	Précision	F-Mesure
DDA-1xRT	Moyenne	0.63	0.97	0.76
	Elevé	0.62	0.65	0.63
DDA-AF-GMM	Moyenne	0.65	0.97	0.78
	Elevé	0.74	0.81	0.77
DDA-AF-MLP	Moyenne	0.73	0.97	0.83
	Elevé	0.74	0.83	0.78

les disfluences, la F-Mesure passe de 0.76% à 0.63% ; le taux de rappel se stabilise, mais le taux de précision décroît d'environ 0.32 point. Le filtrage acoustique permet d'améliorer la détection de termes dans toutes les conditions mais le point le plus intéressant est qu'il semble être plus robuste à la parole spontanée. Le système à base de MLP améliore les performances du système GMM sur le niveau de spontanéité moyen (de 0.78 à 0.83 %) mais la F-Mesure reste affectée par la spontanéité. Pour un niveau de spontanéité élevé, le GMM et MLP ont des performances similaires.

3.4 Conclusion

Nous avons présenté une architecture à 2 niveaux pour la recherche rapide de termes dans laquelle le détecteur s'adapte automatiquement à la requête. Le premier niveau repose sur une optimisation de la représentation de la requête comme une cascade de filtres phonétiques. Le second niveau effectue un décodage guidé de la requête sur les segments de parole. Nous évaluons les performances de ces techniques sur de la parole spontanée. Les résultats démontrent que les coupures sur les filtres phonétiques et l'approche de requête minimale améliorent significativement l'efficacité de recherche de termes dans toutes les conditions. Plus encore, le décodage guidé par la requête permet d'améliorer significativement les résultats, comparé à un décodage non contraint. Les performances selon le niveau de spontanéité montrent que les méthodes proposées sont plus robustes aux disfluences qu'un système de RAP seul, tout en respectant le problème de contrainte temps réel.

Les vidéos sur Internet sont souvent accompagnées de tags, ou référencées sur des sites (ou des blogs) qui peuvent donner une idée *a priori* du sujet de la vidéo. On peut utiliser une stratégie de validation d'une hypothèse plutôt que d'extraire en "aveugle" le contenu de la vidéo, avec des perspectives d'amélioration de la transcription et de la vitesse de décodage. Les expériences que nous avons pré-

sentées confirment cette idée. Le système pourrait être utilisé conjointement avec un moteur de recherche pour collecter, filtrer les vidéos issues d'une base ouverte comme le web.

Chapitre 4

Normalisation des données acoustiques

Sommaire

4.1 Introduction

Le but d'un système de RAP est d'extraire le contenu linguistique d'un signal de parole enregistré. Cependant, le signal de parole n'inclut pas seulement l'information linguistique utile mais aussi des informations perturbantes (Benzeghiba

et al., 2007). Ces informations perturbantes sont très diverses : variabilités locuteurs (*vocal tract length* (Eide and Gish, 1996), niveau de spontanéité (Dufour et al., 2010b)...), condition d'enregistrement (environnement bruité (Sroka and Braida, 2005), configuration du microphone et canal de transmission). La parole observée est composée d'informations utiles (liées au contenu linguistique) mais aussi d'un ensemble d'informations inutiles appelées ici "variabilité session". Les variabilités des canaux, locuteurs et environnement sont les facteurs les plus importants qui affectent les performances du système de RAP.

On peut trouver dans la littérature de nombreuses méthodes pour réduire ces variabilités acoustiques. La compensation de ces variabilités peut être opérée à deux niveaux : sur les modèles acoustiques ou sur le signal de la parole (sur la paramétrisation acoustique).

Récemment, une approche à base de Analyse Factorielle – Factor Analysis (FA) a été appliquée dans le domaine de la reconnaissance de locuteur afin de modéliser la variabilité session comme une composante additive (Kenny et al., 2007). L'idée, derrière cette approche, est que la composante session est localisée dans un sous-espace acoustique de faible dimension.

Quelques auteurs ont proposé d'appliquer le paradigme FA dans les systèmes de RAP. Ces recherches se sont focalisées sur la modélisation de l'information utile : Subspace Gaussian Mixture Model (SGMM) (Bouallegue et al., 2011; Gales and Yu, 2010) et Canonical State Models (CSM) (Povey et al., 2010) mais pas sur la modélisation de l'information inutile. Nous proposons d'utiliser ici le paradigme FA pour modéliser la composante de la variabilité session afin de la supprimer directement des observations acoustiques. Une autre contribution est d'étendre le paradigme FA afin de modéliser explicitement plusieurs variabilités dans le signal audio.

Dans la section 4.2 nous présenterons le paradigme FA. Puis dans la section 4.3 nous présenterons notre modèle de normalisation de paramètre acoustique utilisant le paradigme FA.

4.2 Etat de l'art

La Classification de Forme Audio (CFA) inclut de nombreuses tâches telles que la reconnaissance de la parole, la vérification du locuteur, la détection de l'émotion, etc... En dépit des efforts faits dans les différents domaines sur la modélisation des paramètres audio, la CFA doit faire face à un problème de changement des conditions acoustiques qui varient de manière imprévisible d'un enregistrement à l'autre. Ce phénomène est généralement appelé *variabilité du bruit* et il est une des plus importantes sources de dégradation des performances de la CFA.

Le terme *variabilité du bruit* englobe un nombre de phénomènes importants comme les effets liées aux micros, l'environnement bruité, la position des micro-

phones etc...

L'approche classique en utilisant un classifieur statistique est d'estimer les paramètres qui modélisent la forme, tandis que la *variabilité du bruit* n'est pas explicitement modélisée. Récemment, dans le contexte de la tâche de vérification du locuteur basée sur un GMM-UBM, le paradigme FA a été introduit afin de modéliser l'information utile et inutile en même temps, mais dans des composantes différentes.

L'idée derrière le paradigme FA est que, pour un enregistrement donné, la composante liée à l'information inutile n'est pas estimée sur ces données seules, mais sur un large nombre d'enregistrements venant de plusieurs sessions différentes et de classes différentes. Soit θ_O un vecteur composé d'un jeu de paramètres estimé sur O (O est un jeu de trames composant un enregistrement audio), nous considérons le modèle suivant :

$$\theta_O = \theta_{utile} + Ux_o \tag{4.1}$$

où θ_{utile} est un vecteur de paramètre qui contient l'information intéressante (locuteur, sexe, langage, etc...). La composante inutile Ux_o est composée de deux termes. Le terme U est une matrice de faible dimension par rapport à la taille de θ. Cette matrice est estimée en utilisant une grande base de données correspondant aux différentes sessions. Le terme x_o est un vecteur caractérisant la session courante. En d'autres termes, la *variabilité du bruit* est censée se trouver dans un sous-espace de faible dimension.

Le succès de ce paradigme dépend principalement de l'hypothèse que la *variabilité du bruit* est localisée dans un espace de faible dimension et que les effets liés à l'information utile et inutile s'ajoutent.

Le super-vecteur m_s est obtenu par la concaténation de toutes les moyennes de l'UBM, s représentant la classe. Cette dernière a une distribution *a priori* normale avec une moyenne m et une variance $DD^t = (\Sigma/\tau)$. m et Σ sont des paramètres du modèle GMM-UBM. τ est un facteur lié à l'adaptation Maximum A Posteriori (MAP). La variable m_s s'écrit :

$$m_s = m + Dy_{(s)} \tag{4.2}$$

où y_s est un vecteur correspondant à une variable latente et ayant une distribution normale standard $\mathbf{N}(0, \mathbf{I})$. Actuellement, l'équation 4.2 est équivalente à celle obtenue par le MAP de Reynolds.

Compte tenu d'une collection d'enregistrements pour la classe s, désignons $m_{(h,s)}$ le super-vecteur correspondant à la classe s et à l'enregistrement $h(h = 1, 2, ...n)$. Pour une classe fixée s, supposons que toutes les moyennes du super-vecteur GMM $m_{(h,s)}$ sont statistiquement indépendantes. Ainsi $m_{(h,s)}$ peut s'écrire :

$$m_{(h,s)} = m_{(s)} + Ux_{(h,s)} \tag{4.3}$$

où $x_{(h,s)}$ est un vecteur correspondant à une variable latente et ayant une distribution normale standard $\mathbf{N}(0, \mathbf{I})$. Pour une adaptation à une classe s et à une session h, l'adaptation MAP consiste à une adaptation *a posteriori* de $x_{(h,s)}$.

Afin d'avoir dans le même cadre l'information utile et inutile, nous intégrons l'équation 4.2 dans l'équation 4.3. Ainsi le modèle final peut s'écrire :

$$m_{(h,s)} = m + Dy_{(s)} + Ux_{(h,s)} \tag{4.4}$$

où $m_{(h,s)}$ est le super-vecteur composé des moyennes de la session h et de la classe s (de dimension MN où M est le nombre de Gaussienne du GMM et N est la dimension du paramètre acoustique), D est une matrice diagonale (de dimension $MN \cdot MN$), y_s et le vecteur de la classe s (de dimension MN), U est la matrice de faible dimension représentant l'espace des dimensions de l'information inutile (de dimension $MN \cdot R$) et $x_{(h,s)}$ est un vecteur (de dimension R). Les vecteurs y_s et $x_{(h,s)}$ ont théoriquement une distribution normale standard $\mathbf{N}(0, \mathbf{I})$. $DD^t = (\Sigma/\tau)$ représente la variabilité du super-vecteur de la classe s. UU^t représente la variabilité session.

Le succès du modèle FA est lié à une bonne estimation de la *variabilité du bruit* de la matrice U dans le cas où plusieurs sessions différentes sont disponibles.

Soit $N_{(s)}$ et $N_{(h,s)}$ les vecteurs de la classe s et de la session h :

$$\begin{aligned} N_{g(s)} &= \sum_{t \in s} \gamma_g(t) \\ N_{g(h,s)} &= \sum_{t \in (h,s)} \gamma_g(t) \end{aligned} \tag{4.5}$$

où $\gamma_g(t)$ est la probabilité *a posteriori* d'une gaussienne g et d'une observation t. $\sum_{t \in s}$ correspond à la somme de toutes les trames de la même classe s et $\sum_{t \in (h,s)}$ à la somme de toutes les trames des sessions h et des classes s.

Soit $X_{(s)}$ et $X_{(h,s)}$ les vecteurs contenant les informations de la classe s et de la session h :

$$\begin{aligned} X_{g(s)} &= \sum_{t \in s} \gamma_g(t) \cdot t \\ X_{g(h,s)} &= \sum_{t \in (h,s)} \gamma_g(t) \cdot t \end{aligned} \tag{4.6}$$

Soit $\overline{X}_(s)$ et $\overline{X}_(h,s)$ les statistiques de l'information utile et inutile :

$$\begin{aligned} \overline{X}_{g(s)} &= X_{g(s)} - \sum_{h \in s} N_{g((h,s))} \cdot \{m + Ux_{(h,s)}\}_g \\ \overline{X}_{g(h,s)} &= X_{g(h,s)} - N_{g(h,s)} \cdot \{m + Dy_{(s)}\}_g \end{aligned} \tag{4.7}$$

où $\overline{X}_{(s)}$ est utilisé pour estimer le vecteur de la classe s, tandis que $\overline{X}_{(h,s)}$ est utilisé pour estimer l'information inutile.

Désignons $L_{(h,s)}$ une matrice de dimension $R \times R$ et $B_{(h,s)}$ un vecteur de dimension R, définis par :

$$\begin{aligned} B_{(h,s)} &= \sum_{g \in UBM} U_g^T \cdot \Sigma_g^{-1} \cdot \overline{X_{g(h,s)}} \\ L_{(h,s)} &= I + \sum_{g \in UBM} N_{g(h,s)} \cdot U_g^T \cdot \Sigma_g^{-1} \cdot U_g \end{aligned} \tag{4.8}$$

où Σ_g est la matrice de covariance de la composante g de l'UBM. En utilisant $L_{(h,s)}$ et $B_{(h,s)}$; nous pouvons obtenir $x_{h,s}$ et y_s depuis les équations suivantes :

$$\begin{aligned} x_{(h,s)} &= L_{(h,s)}^{-1} \cdot B_{(h,s)} \\ y_{(s)} &= \frac{\tau}{\tau + N_g} \cdot D_g \Sigma_g^{-1} \cdot \overline{X_{g(h,s)}} \end{aligned} \tag{4.9}$$

où $D_g = (1/\sqrt{\tau})\Sigma_g^{1/2}$ (τ est mis à 14.0 dans nos expériences).

Finalement la matrice U peut être estimée ligne par ligne, avec U_g^i comme étant la ligne i de U_g donc :

$$U_g^i = \mathcal{L}(g)^{-1} \cdot \mathcal{R}^i(g) \tag{4.10}$$

où $\mathcal{L}(g)$ et $\mathcal{R}^i(g)$ sont obtenus par :

$$\begin{aligned} \mathcal{L}(g) &= \sum_s \sum_{h \in s} N_{g(h,s)} \cdot (L_{(h,s)}^{-1} + \mathbf{x}_{(h,s)} \mathbf{x}_{(h,s)}^T) \\ \mathcal{R}^i(g) &= \sum_s \sum_{h \in s} \overline{X}_{g(h,s)}[i] \cdot \mathbf{x}_{(h,s)} \end{aligned} \tag{4.11}$$

La matrice U est estimée en utilisant l'algorithme 1.

Cet algorithme permet de trouver dans un sous-espace de faible dimension l'information inutile. Ce sous-espace est modélisé par la matrice U.

Algorithm 1: Estimation de la matrice U

Pour chaque phonème s et session h : $y_{(s)} \leftarrow 0$, $x_{(h,s)} \leftarrow 0$;
$U \leftarrow random$ (U est initialisé aléatoirement) ;
Les statistiques : $N_{(s)}$, $N_{(h,s)}$, $X_{(s)}$, $X_{(h,s)}$;
for $i = 1$ *to nb_iterations* **do**
  for *tous les h et s* **do**
    Les statistiques sont centrées : $\overline{X}_{(h,s)}$;
    Estimer $L^{-1}_{(h,s)}$ et $B_{(h,s)}$;
    Estimer $x_{(h,s)}$;
    Les statistiques sont centrées : $\overline{X}_{(s)}$;
    Estimer $y_{(s)}$;
  end
  Estimer la matrice U ;
end

4.3 Contributions

4.3.1 Modélisation de la variabilité session

Dans un système de RAP, le signal de parole véhicule non seulement des informations linguistiques utiles mais aussi des informations inutiles. Ces informations inutiles sont de natures différentes et peuvent être liées aux environnements variables (bruit de fond...), variabilité locuteur (genre, âge, émotion...), variabilité canal (microphone...)... Ces informations inutiles sont présentes dans le signal de la parole et affectent le HMM d'un système de RAP. Afin de modéliser seulement l'information phonétique dans le HMM, une solution serait de supprimer l'information inutile des trames de la parole.

Le paradigme FA donne la possibilité de modéliser l'information inutile afin de la supprimer des trames acoustiques. Désignons G un jeu de gaussiennes structurant l'espace acoustique du signal de la parole. Désignons m le super-vecteur obtenu par la concaténation de toutes les moyennes dans G. Nous désignons i l'information utile et h l'information session (qui représente la variabilité locuteur ou canal). En utilisant le paradigme FA, le super-vecteur $m_{i,h}$ peut être décomposé en trois composantes différentes :

$$m_{i,h} = m + Dy_i + Ux_h \tag{4.12}$$

ici m est la composante du super-vecteur des moyennes de gaussiennes venant de G. G est entrainé sur une large quantité de données contenant les informations utiles et inutiles. y_i est l'information utile à modéliser. Elle peut correspondre à l'information linguistique d'un enregistrement donné, à un phonème ou à un état d'un HMM. Ux est la composante de variabilité session. U est composée des vec-

teurs propres associés à la variabilité session. y_i et x_h sont tous les deux normalement distribués selon $\mathbf{N}(0, \mathbf{I})$. D est une matrice diagonale de sorte que DD^t soit la matrice de covariance *a priori* de la composante liée aux phonèmes. U est une matrice rectangulaire de sorte que UU^t soit la matrice de covariance de la composante session du vecteur aléatoire.

Comme montré dans l'Equation 4.12, le succès du paradigme FA dépend de l'hypothèse selon laquelle la variabilité nuisible est située dans un sous-espace vectoriel de faible dimension et l'effet session est additif.

Afin d'avoir un compromis entre la précision de la modélisation et la quantité de données conduisant à estimer les paramètres, nous avons choisi de modéliser i comme étant un phonème indépendant du contexte. En fait, si nous prenons i comme étant une partie d'un phonème, par exemple un état d'un HMM, pour plusieurs états nous n'aurions pas assez de trames pour estimer le facteur de session s_h. Dans cette section nous considérerons les variabilités locuteur ou canal comme une session. En prenant i comme un phonème indépendant du contexte, l'équation du modèle Equation 4.12 peut être écrite plus explicitement :

$$m_{phoneme,session} = m + Dy_{phoneme} + Ux_{session} \tag{4.13}$$

La matrice U est globale et commune à tous les phonèmes. Elle est estimée en utilisant une large quantité de phonèmes produits par différents locuteurs et une diversité de conditions acoustiques. De cette manière, nous pouvons isoler la variabilité session (locuteur ou canal). Il est important de noter que le modèle de l'Equation 4.13 n'est pas utilisé pour le modèle de la reconnaissance de la parole, nous verrons dans le prochain paragraphe comment nous utilisons ce modèle afin de compenser les trames de la parole.

4.3.1.1 Estimation du sous-espace de la variabilité session

La matrice U est un paramètre global. Elle est estimé en utilisant un grand nombre de données contenant la variabilité session. La matrice est itérativement estimée en utilisant l'algorithme d'EM. Pour chaque étape, $x_{session}$ est estimé, puis $y_{phoneme}$ est estimé pour chaque phonème (utilisant le nouveau x) et finalement U est estimée globalement en se basant sur ces $x_{session}$ et $y_{phoneme}$. Les étapes de l'algorithme sont décrites plus en profondeur dans (Matrouf et al., 2007).

4.3.1.2 Suppression de la variabilité sur les trames acoustiques

Chaque segment dans le corpus de test est d'abord normalisé en respectant la variabilité de la session et en utilisant les équations suivantes :

$$\hat{t} = t - \sum_{g=1}^{M} \gamma_g(t) \cdot \{U \cdot x_{session}\}_{[g]} \tag{4.14}$$

où M est le nombre de gaussiennes dans l'UBM, $\gamma_g(t)$ est la probabilité *a posteriori* de la gaussienne g donnée par la trame t. Ces probabilités sont estimées en utilisant l'UBM, et $U \cdot x_{session}$ est la composante de la variabilité session estimée sur le segment enregistré.

Après avoir normalisé tous les segments en utilisant l'Equation 4.14, les HMM du système de RAP sont entraînés en utilisant les données de parole normalisées. Théoriquement, pour chaque segment nous devons estimer la composante de la variabilité session sur chaque phonème et normaliser celle-ci avec la composante de variabilité session. En pratique, ceci n'est pas réalisable en raison du manque de données pour un phonème et un segment. Nous estimons donc la composante de la variabilité session globalement sur les segments et nous appliquons la normalisation des paramètres.

4.3.2 Modélisation des multiples variabilités sessions

Dans les précédentes sections (Equation 4.13), la matrice U modélise une variabilité session spécifique (variabilité locuteur ou canal). Cependant, les variabilités dans un système de RAP sont multiples. Nous proposons une version modifiée du paradigme FA afin de faire face aux multiples facteurs de variabilité. Nous étendons le paradigme FA en considérant que chaque matrice peut modéliser une variabilité spécifique. Ici, la matrice U modélise la variabilité locuteur et la matrice V modélise la variabilité canal. La version modifiée du paradigme FA peut-être formulée ainsi :

$$m_{observed} = m_{ubm} + Dy_{phoneme} + Ux_{speaker} + Vz_{channel} \tag{4.15}$$

où, comme précédemment, m sont les moyennes du super-vecteur, y est la partie spécifique du phonème indépendant du contexte, pondéré par D. Dans cette section, Ux est la composante de variabilité locuteur et Vz est la composante de variabilité canal.

Précédemment, la matrice U est obtenue d'un corpus représentant une seule variabilité spécifique. Ici, nous utilisons deux corpus, chacun modélisant une variabilité spécifique. La matrice U est estimée sur le corpus de variabilité locuteur où chaque session représente un couple phonème-locuteur. La matrice V est estimée sur le corpus de variabilité canal où chaque session représente un couple phonème-canal.

En vérification du locuteur, les auteurs ont proposé un cadre théorique (Kenny, 2006) basé sur la décomposition décrite par l'Equation 4.15. Cette approche appelée Joint Factor Analysis (JFA) modélise sur le même corpus, deux variabilités

sessions : effet locuteur et effet canal. Cependant, nous proposons une approche différente du JFA, dans laquelle la variabilité session est estimée itérativement et est modélisée sur deux corpus différents permettant d'étendre le cadre de travail à d'autres variabilités.

4.3.2.1 Estimation du sous-espace de variabilité locuteur et canal

Les matrices U et V sont communes à tous les phonèmes. Elles sont optimisées conjointement. La procédure d'estimation est présentée dans l'algorithme 2. Dans une première étape, la matrice U est optimisée sur les données du corpus locuteur. Les vecteurs $x_{speaker}$ et $z_{channel}$ sont estimés, ensuite $y_{phoneme}$ est estimé pour chaque phonème (utilisant les nouveaux x et z) et finalement U est estimée globalement, basée sur ces x, z et y. Dans une seconde étape, la matrice V est optimisée sur le corpus de données canal. Les vecteurs $x_{speaker}$ et $z_{channel}$ sont estimés, ensuite $y_{phoneme}$ est estimé pour chaque phonème (utilisant les nouveaux x et z) et finalement, V est estimée globalement utilisant ces x, z et y variables.

Les statistiques sont calculées en tenant compte des matrices U et V :

$$\begin{aligned}
\overline{X}_{g(s)} &= X_{g(s)} - \sum_{h \in s} N_{g(h,s)} \cdot \{m + Ux_{(h,s)} + Vz_{(h,s)}\}_g \\
\overline{X}_{g(h,s)} &= X_{g(h,s)} - N_{g(h,s)} \cdot \{m + Dy_s + Vz_s\}_g \\
\overline{Z}_{g(s)} &= Z_{g(s)} - \sum_{h \in s} M_{g(h,s)} \cdot \{m + Ux_{(h,s)} + Vz_{(h,s)}\}_g \\
\overline{Z}_{g(h,s)} &= Z_{g(h,s)} - M_{g(h,s)} \cdot \{m + Dy_s + Ux_s\}_g
\end{aligned} \tag{4.16}$$

où $N_{(s)}$, $N_{(h,s)}$, $X_{(s)}$, $X_{(h,s)}$ sont les statistiques calculées sur le corpus de variabilité locuteur et $M_{(s)}$, $M_{(h,s)}$, $Z_{(s)}$, $Z_{(h,s)}$ sont les statistiques calculées sur le corpus de variabilité canal.

Désignons $L_{(h,s)}$ et $P_{(h,s)}$ une matrice de dimension $R \times R$ et $B_{(h,s)}$, $Q_{(h,s)}$ un vecteur de dimension R, définis par :

$$\begin{aligned}
B_{(h,s)} &= \sum_{g \in UBM} U_g^T \cdot \Sigma_g^{-1} \cdot \overline{X_{g(h,s)}} \\
L_{(h,s)} &= I + \sum_{g \in UBM} N_{g(h,s)} \cdot U_g^T \cdot \Sigma_g^{-1} \cdot U_g \\
Q_{(h,s)} &= \sum_{g \in UBM} V_g^T \cdot \Sigma_g^{-1} \cdot \overline{Z_{g(h,s)}} \\
P_{(h,s)} &= I + \sum_{g \in UBM} M_{g(h,s)} \cdot V_g^T \cdot \Sigma_g^{-1} \cdot V_g
\end{aligned} \tag{4.17}$$

Algorithm 2: Algorithme d'estimation des matrices U et V

Pour chaque phonème s et session h : $y_{(s)} \leftarrow 0$, $x_{(h,s)} \leftarrow 0$, $z_{(h,s)} \leftarrow 0$;
$U \leftarrow random$ (U est initialisé aléatoirement);
$V \leftarrow random$ (V est initialisé aléatoirement);
Les statistiques : $N_{(s)}$, $N_{(h,s)}$, $X_{(s)}$, $X_{(h,s)}$ sont estimées sur le corpus de variabilité locuteur;
Les statistiques : $M_{(s)}$, $M_{(h,s)}$, $Z_{(s)}$, $Z_{(h,s)}$ sont estimées sur le corpus de variabilité canal;
for $i = 1$ *to nb_iterations* **do**
  for *tous les h et s du corpus de variabilité locuteur* **do**
    Les statistiques sont centrées : $\overline{Z}_{(h,s)}$;
    Les statistiques sont centrées : $\overline{X}_{(h,s)}$;
    Estimer $L^{-1}_{(h,s)}$ et $B_{(h,s)}$;
    Estimer $z_{(h,s)}$;
    Estimer $x_{(h,s)}$;
    Les statistiques sont centrées : $\overline{Z}_{(s)}$;
    Les statistiques sont centrées : $\overline{X}_{(s)}$;
    Estimer $y_{(s)}$;
  end
  Estimer la matrice U ;
  for *tous les h et s du corpus de variabilité canal* **do**
    Les statistiques sont centrées : $\overline{Z}_{(h,s)}$;
    Les statistiques sont centrées : $\overline{X}_{(h,s)}$;
    Estimer $L^{-1}_{(h,s)}$ et $B_{(h,s)}$;
    Estimer $z_{(h,s)}$;
    Estimer $x_{(h,s)}$;
    Les statistiques sont centrées : $\overline{Z}_{(s)}$;
    Les statistiques sont centrées : $\overline{X}_{(s)}$;
    Estimer $y_{(s)}$;
  end
  Estimer la matrice Z ;
end

où Σ_g est la matrice de covariance de la composante g de l'UBM. En utilisant $L_{(h,s)}$, $B_{(h,s)}$, $P_{(h,s)}$ et $Q_{(h,s)}$ nous pouvons obtenir $x_{(h,s)}$, $z_{(h,s)}$ et $y_{(s)}$ depuis les équations suivantes :

$$\begin{aligned} z_{(h,s)} &= P^{-1}_{(h,s)} \cdot Q_{(h,s)} \\ x_{(h,s)} &= L^{-1}_{(h,s)} \cdot B_{(h,s)} \\ y_{(s)} &= \frac{\tau}{\tau + N_g} \cdot D_g \Sigma_g^{-1} \cdot \overline{X_{g(h,s)}} \end{aligned} \tag{4.18}$$

où $D_g = (1/\sqrt{\tau})\Sigma_g^{1/2}$ (τ est mis à 14.0 dans nos expériences).

Finalement les matrices U et V peuvent être estimées ligne par ligne, avec U_g^i et V_g^i étant la i^{th} ligne de U_g et V_g ; donc :

$$\begin{aligned} U_g^i &= \mathcal{L}(g)^{-1} \cdot \mathcal{R}^i(g) \\ V_g^i &= \mathcal{P}(g)^{-1} \cdot \mathcal{Q}^i(g) \end{aligned} \tag{4.19}$$

où $\mathcal{L}(g)$, $\mathcal{R}^i(g)$, $\mathcal{P}(g)$ et $\mathcal{Q}^i(g)$ sont obtenus par :

$$\begin{aligned} \mathcal{L}(g) &= \sum_s \sum_{h \in s} N_{g(h,s)} \cdot (L_{(h,s)}^{-1} + \mathbf{x}_{(h,s)} \mathbf{x}_{(h,s)}^T) \\ \mathcal{R}^i(g) &= \sum_s \sum_{h \in s} \overline{X}_{g(h,s)}[i] \cdot \mathbf{x}_{(h,s)} \\ \mathcal{P}(g) &= \sum_s \sum_{h \in s} M_{g(h,s)} \cdot (P_{(h,s)}^{-1} + \mathbf{x}_{(h,s)} \mathbf{x}_{(h,s)}^T) \\ \mathcal{Q}^i(g) &= \sum_s \sum_{h \in s} \overline{Z}_{g(h,s)}[i] \cdot \mathbf{x}_{(h,s)} \end{aligned} \tag{4.20}$$

4.3.2.2 Modèle acoustique normalisé sur des variabilités multiples

Une fois les matrices U et V obtenues, les paramètres sont normalisés afin de supprimer les effets locuteur et canal. Comme précédemment, l'adaptation de chaque vecteur est obtenue en soustrayant du paramètre d'observation les composantes variabilités locuteur et canal :

$$\hat{t} = t - \sum_{g=1}^{M} \gamma_g(t) \cdot (\{U \cdot x_{speaker}\}_{[g]} + \{V \cdot z_{channel}\}_{[g]}) \tag{4.21}$$

où $U \cdot x_{speaker}$ et $V \cdot z_{channel}$ sont les composantes canal et locuteur estimées sur l'enregistrement. Les variables latentes estimées $x_{speaker}$, $z_{channel}$ sont respectivement calculées depuis U et V.

Comme précédemment, après normalisation de tous les segments utilisant l'Equation 4.21, les HMM du système de RAP sont entraînés en utilisant les segments de parole normalisés.

4.3.3 Description du système et résultats

4.3.3.1 Système et corpus

Pour ces expériences nous utilisons le système SPEERAL, décrit dans la Section 3.3.4.1. Le processus de transcription se compose de deux passes :

- La première passe (*PASS-1*) utilise les modèles acoustiques correspondant aux sexe et bande passante détectés par le processus de segmentation et utilisant un modèle de langage tri-gramme.
- La seconde passe (*PASS-2*) applique une transformation de type Maximum Likelihood Linear Regression (MLLR) par locuteur ou par segment et utilise le même modèle de langage tri-gramme que la *PASS-1*.

Les performances du système sont évaluées sur le corpus d'évaluation ESTER. Les données sont composées de 18 fichiers audios pour une durée totale de 10h.

4.3.3.2 Entraînement du modèle acoustique

Le modèle acoustique a été appris sur un corpus d'entraînement où toutes les trames sont normalisées par l'Equation 4.14 ou 4.21. La normalisation est aussi appliquée aux trames de test qui doivent être décodées. Pour tous ces résultats, le rang des matrices U et V est fixé à 60. Le GMM-UBM utilisé dans le paradigme FA est composé de 600 gaussiennes.

4.3.4 Modèle acoustique sur une variabilité spécifique

Dans une première étape, nous comparons les résultats de notre *baseline* avec un système entraîné sur une variabilité spécifique. *Norm-speaker* et *Norm-channel* sont les systèmes où les modèles acoustiques sont entraînés sur une variabilité spécifique utilisant les Equations 4.14.

TABLE 4.1 – *Les résultats sont exprimées en % de WER sur le corpus ESTER*

	PASS-1	PASS-2
Baseline	29.6	27.5
Norm-speaker	28.5	26.9
Norm-channel	28.6	26.7

Le Tableau 4.1 montre les résultats obtenus sur le corpus ESTER. Dans la *PASS-1*, nous observons que la baseline obtient un WER de 29.6% et que les systèmes *Norm-speaker* et *Norm-channel* obtiennent un WER respectivement de 28.5% et 28.6% (une amélioration absolue respectivement de 1.1% et 1.0%). Dans la *PASS-2*, nous obtenons une amélioration absolue du WER pour *Norm-channel* et *Norm-speaker*

respectivement de 0.8% et 0.6%. Si les gains sont moins importants que la *PASS-1*, ceci peut être expliqué par l'adaptation MLLR. En effet, la technique MLLR adapte le modèle acoustique à un locuteur particulier, capturant les relations entre le modèle original et le locuteur courant ou l'environnement acoustique. Le nouveau modèle dépendant du locuteur permet de réduire la variabilité intra-locuteur.

4.3.5 Modèle acoustique entraîné sur des variabilités multiples

Le Tableau 4.2 montre les résultats utilisant le paradigme FA étendu. *Norm-speaker-channel* est le système qui modélise le modèle acoustique entraîné sur des variabilités multiples utilisant Equation 4.21. Comparé à notre *Baseline*, le système *Norm-speaker-channel* obtient en *PASS-2* un gain absolu de WER de 1.3%. Dans la précédente section, le meilleur système (*Norm-channel*) obtenait en *PASS-2* une amélioration absolue de WER de 0.8%.

Ces résultats confirment que le paradigme FA peut modéliser différentes variabilités et permet de les supprimer dans l'espace acoustique. Dans ces expériences, nous nous limitons aux variabilités locuteur et canal mais il est tout à fait possible d'étendre le paradigme FA pour supprimer d'autres variabilités.

TABLE 4.2 – *Les résultats sont exprimés en % de WER sur le corpus ESTER*

	PASS-1	PASS-2
Baseline	29.6	27.5
Norm-speaker-channel	28.0	26.2

Dans le Tableau 4.3, nous observons la robustesse du système *Norm-speaker-channel* en évaluant chaque phrase. Nous avons trié les phrases du système de la *Baseline* en 11 intervalles suivant leur WER. Les phrases utilisées dans le système *Norm-speaker-channel* sont ordonnées de la même façon que dans le système *Baseline*. Les différences entre les deux systèmes, reportées dans la colonne gain WER, sont principalement basées sur la normalisation des trames. Ce tableau permet de comparer le WER des phrases en fonction du système utilisé.

Nous pouvons observer, pour l'intervalle 0-10, que le WER entre *Norm-speaker-channel* et *Baseline* est augmenté. Nous obtenons sur l'intervalle 0-5 une augmentation du WER absolue de 2.11%. Sur les rangées 30 à 100, nous observons quelques gains pour le système *Norm-speaker-channel*. Ce gain est particulièrement important sur les rangées 50 à 100 (une réduction absolue du WER de 5.74%). Plus encore, nous observons sur le corpus ESTER une réduction absolue du WER de 1.3%. La normalisation est particulièrement importante sur les phrases avec des difficultés acoustiques nombreuses. Cependant, sur les phrases avec un WER bas, la normalisation n'apporte aucune amélioration et peut détériorer les résultats.

TABLE 4.3 – *Les résultats pour chaque rangée de WER*

Intervalle WER %	Baseline	Norm-spk-cha	Gain WER
0-5	0.35	2.46	-2.11
5-10	7.19	8.52	-1.33
10-15	12.73	13.44	-0.70
15-20	17.60	18.36	-0.75
20-25	21.52	20.91	0.61
25-30	26.71	25.80	0.91
30-35	32.18	29.79	2.39
35-40	37.01	35.79	1.22
40-45	41.85	39.45	2.39
45-50	46.36	44.00	2.36
50-100	68.28	62.54	5.74

4.4 Conclusion

Dans ces travaux, nous proposons un cadre de travail pour la normalisation de données basée sur le paradigme FA. Nous avons aussi présenté une extension pour faire face aux multiples et différents types de variabilités. Cette extension peut être utilisée pour d'autres variabilités qui pourront être étudiées dans le futur.

Ce nouveau cadre nous permet d'améliorer la robustesse des systèmes de RAP face aux différentes variabilités liées aux locuteurs et aux canaux. C'est une robustesse particulièrement intéressante dans des conditions adverses et imprévisibles, ce qui est typiquement le cas des données Web (principalement les données issues de Youtube et/ou Dailymotion) qui sont enregistrées dans de mauvaises conditions (enregistrement depuis les téléphones portables, matériel d'acquisition, etc...).

Nous avons observé que le système *Norm-speaker-channel* pouvait détériorer les résultats dans l'intervalle 0-20. Nous avons vu que l'estimation des paramètres FA est principalement basée sur la probabilité *a posteriori* d'un modèle UBM. Il est possible que cette détérioration soit due à l'utilisation d'un UBM qui calculerait cette probabilité de manière grossière. Peut-être qu'une manière plus robuste d'obtenir ces probabilités serait d'utiliser un système de RAP complet. En fait, en utilisant un système de RAP, nous n'utilisons pas seulement l'information acoustique mais aussi l'information linguistique contenue dans le modèle de langage. Dans ce cas, le super-vecteur m de l'Equation 4.13 est la concaténation de toutes les moyennes des gaussiennes contenues dans le HMM. Dans cette étude, nous avons utilisé un GMM-UBM au lieu d'un HMM qui probablement tend à être moins précis pour estimer les probabilités.

Troisième partie

Structuration et catégorisation de collections multimédia

La banalisation des moyens de numérisation et de diffusion de données audiovisuelles a permis ces dernières années, de constituer de très grandes bases de données dans des domaines très variés : bases de vidéos générées par l'utilisateur, archives télévisuelles ou cinématographiques, archivage de dialogues en centre d'appel...

Ces collections multimédia sont souvent de natures très diverses. Les documents peuvent être hétérogènes par la forme, comme par exemple le genre vidéo ; ou sur le fond, comme par exemple les thèmes. Cette hétérogènéité est un phénomène particulièrement important lorsque les bases sont générées par les utilisateurs. De plus, ces collections sont aussi peu ou trés mal structurées, soit parce qu'une annotation manuelle serait très coûteuse à cette échelle, soit parce que la production des données elle-même échappe à toute structure claire. Un dernier facteur favorisant l'hétérogénéité est liée à la diversité des publics et des acteurs du Web.

Le succès croissant de sites comme YouTube ou Dailymotion constitue une illustration de l'essor de ces collections de documents multimédia qui viennent s'ajouter aux collections audiovisuelles plus traditionnelles comme celles archivées par l'INA (Institut National de l'Audiovisuel).

L'exploitation de ces collections ne peut se faire que par une caractérisation riche des contenus qui doit ouvrir l'accès aux bases et permettre leur analyse.

Dans ce chapitre, nous allons présenter nos travaux sur la structuration des bases de données. Tenant compte du fait qu'il existe une multitude d'éléments qui peuvent permettrent de structurer de grandes bases de données, nous nous sommes focalisés dans le chapitre 5 sur la classification des vidéos selon leur genre et, dans le chapitre 6 à caractériser et à détecter le niveau de spontanéité du discours.

Chapitre 5

Catégorisation selon le genre vidéo

Sommaire

5.1 Introduction

Les vidéos disponibles sur les services de vidéos communautaires sont produites dans des contextes très variés, comme par exemple des vidéos générées par l'utilisateur, des archives télévisuelles (publicité, actualité...) ou cinématographiques, etc... La structuration de ces vidéos a pour but d'interpréter automatiquement le contenu d'une vidéo afin de fournir une représentation utilisable de ce contenu à d'autres processus (comme par exemple la navigation, la recherche d'information, le résumé, etc...).

Il existe potentiellement une multitude d'éléments structurants qui peuvent être liés au contenu des vidéos, à leur type, etc... La structuration automatique de telles collections requière une catégorisation de haut niveau par des descripteurs qui ne sont pas liés seulement au contenu mais aussi à la forme du document. Le genre est une de ces métadonnées, qui peut permettre l'organisation des vidéos dans de grandes catégories. Le genre se réfère aux styles éditoriaux d'une vidéo. L'identification automatique du genre vidéo est un challenge motivé par de récentes recherches comme le Google Challenge [1] et les campagnes d'évaluation TREC Video Retrieval Evaluation [2].

Nous présenterons dans la section 5.2.1 une taxonomie du genre vidéo puis, dans la section 5.2, nous dressons un état de l'art de la classification de genre vidéo dans les domaines textuels (Section 5.2.2), audio (Section 5.2.3) et vidéo (Section 5.2.4). Enfin, nous terminerons ce chapitre par la section 5.3 où nous présenterons notre contribution dans le domaine.

1. Google Challenge : http://comminfo.rutgers.edu/conferences/mmchallenge/2010/02/10/google-challenge/
2. TREC Video Retrieval Evaluation : http://trecvid.nist.gov/

5.2 Etat de l'art

5.2.1 Taxonomie et Historique

Historiquement, la classification du genre vidéo a commencé en 1995 avec les travaux de Fischer (Fischer et al., 1995). A l'époque, il prenait en compte 3 genres vidéos : bulletin météo, sport (courses de voitures et tennis) et publicité. L'approche était principalement focalisée sur des descripteurs vidéos. Puis, dans (Dimitrova et al., 2000), l'auteur proposa de détecter 4 genres : actualité, publicité, feuilleton télévisé (*soap*) et série télévisée (*sitcom*). L'approche s'est focalisée sur la détection des visages et du texte inscrit sur la vidéo. C'est en 2000 que Truong (Truong and Dorai, 2000) proposa une détection du genre à 5 classes : sport, actualité, publicité, bande dessinée et clip vidéo.

Comme on peut le constater au fil du temps, le nombre de classes ainsi que le type de classes à détecter ont changé. La classification du genre vidéo a commencé en 1995 avec 3 classes principalement focalisées sur les bulletins météorologiques et la publicité, pour passer en 2011 à 7 classes.

Dans (Snoek and Worring, 2005), l'auteur propose une taxonomie complète de la classification du genre vidéo. On peut constater qu'elle est constituée de 9 classes : *publicité*, *actualité*, *documentaire*, *série télévisée*, *feuilleton télévisé*, *cartoon*, *film*, *sport*, *sport*, *musique* et *débat télévisé*. Il n'est pas impossible que ce nombre augmente encore avec l'apparition de nouveaux types de programmes télévisés.

5.2.2 Approche basée sur le texte

Les méthodes d'extraction du texte à partir d'une vidéo peuvent être classées en deux catégories. La première consiste à extraire les informations textuelles présentes à l'écran ; cela concerne un texte présent sur des objets, des personnes, etc... comme par exemple : le nom d'un athlète, l'adresse d'un bâtiment, le score d'un match, etc... (Kobla et al., 2000). Le texte est capturé puis extrait en utilisant un système de Reconnaissance Optique de Caractères – Optical Character Recognition (OCR) (Hauptmann et al., 2002).

La seconde catégorie utilise la transcription des contenus parlés de la vidéo, qui peut être obtenue à partir des sous-titres ou à partir de l'audio par transcription automatique. Il existe différentes manières de récupérer les sous-titres : soit le sous-titre est disponible en fichier texte (c'est notamment le cas sur les DVD, Blu-Ray (BR), etc...), soit le sous-titre fait partie intégrante de la vidéo et peut être extrait en utilisant une détection de texte avec un OCR. Dans le cas où il n'y a aucun sous-titre disponible, on peut obtenir la transcription audio en utilisant un système de transcription automatique de la parole (Wang et al., 2003).

Un des avantages des approches basées sur le texte est que l'on peut utiliser un large éventail de techniques développées pour la classification de documents

de textes (Sebastiani, 2002). De plus, la relation entre les paramètres (les mots) et le genre est souvent évidente. Par exemple, on ne sera pas surpris de trouver les mots "stade", "ballon" et "arbitre" dans une transcription d'émissions sportives.

Cependant, utiliser des approches basées sur le texte pose des problèmes difficiles à résoudre et comporte quelques inconvénients. Les sous-titres ne sont pas toujours disponibles avec la vidéo et les obtenir revient souvent à faire de la transcription manuelle qui est très onéreuse. Ces approches sont inutilisables lorsque les sous-titres ne sont absolument pas disponibles, par exemple sur les plateformes internet d'échanges de contenus vidéos. Une manière peu coûteuse d'obtenir la transcription des vidéos est d'utiliser un système de reconnaissance automatique de la parole. Les performances des systèmes de RAP sur les vidéos issues de la télévision ou de données web ont un taux d'erreurs-mots assez élevé, ce qui affecte les méthodes utilisées pour faire de la classification de textes.

L'approche classique pour représenter le texte est de construire un vecteur utilisant le modèle de "sac-de-mot" (Forman, 2003). Dans ce modèle, chaque terme du vecteur représente le nombre de fois où le mot apparait dans le document. Un des inconvénients de ce modèle est que l'information sur l'ordre des mots n'est pas gardée.

Représenter une transcription dans laquelle chaque mot est inclus requiert un vecteur de paramètres avec une dimension assez élevée. Pour réduire la dimension de l'espace de représentation, une *stop-liste* et un *stemming* de mot (lemmatisation) sont souvent appliqués. La *stop-liste* contient un ensemble de mots discriminant comme "le", "je", etc... Le *stemming* supprime le suffixe et le préfixe des mots, laissant ainsi la racine du mot ; par exemple les mots : "construction" et "construirons" ont tous les deux, avec le *stemming*, la même racine "constru". Ces techniques permettent notamment de réduire la dimensionnalité de la représentation du document et ainsi d'améliorer la classification des documents.

Une autre approche est de pondérer les mots du document en utilisant la fréquence du terme et la fréquence inverse de document (Fréquence de Terme (TF)-Fréquence Inverse de Document (IDF)). Cette technique permet d'évaluer l'importance d'un terme contenu dans un document, relativement à un corpus. Le poids augmente proportionnellement au nombre d'occurrences du mot dans le document. Il varie également en fonction de la fréquence du mot dans le corpus.

$$w_{i,j} = tf_{i,j} * idf_i \tag{5.1}$$

5.2.3 Approches basées sur l'audio

Les paramètres audios ont été assez peu utilisés pour faire de la classification en genre. Ils peuvent être subdivisés en 2 classes : le domaine temporel et le domaine fréquentiel.

5.2.3.1 Domaine temporel

L'énergie du signal permet d'avoir une approximation du volume sonore de la vidéo (Wold et al., 1996). Dans (Liu et al., 1998), l'auteur a montré que les sports ont un niveau constant de bruit, lequel peut être détecté en utilisant l'énergie du signal audio.

Le Zero Crossing Rate (ZCR) est le nombre de fois où le signal change de signe dans une fenêtre. C'est un paramètre qui a été utilisé en reconnaissance de la parole. La parole a un taux élevé de ZCR par rapport à la musique.

5.2.3.2 Domaine fréquentiel

Le signal numérisé est trop variable pour servir directement dans une application de classification de genre vidéo. Il doit être traité de manière à extraire au mieux l'information nécessaire et suffisante à la caractérisation de son contenu. Une représentation traditionnelle pour le traitement et l'interprétation du signal est la représentation temps-fréquence.

Les Mel Frequency Cepstral Coefficients (MFCC) sont des coefficients cepstraux calculés par une transformée en cosinus discrète appliquée au spectre de puissance d'un signal. Les bandes de fréquence de ce cepstre sont espacées logarithmiquement selon l'échelle de Mel. C'est le jeu de paramètres le plus utilisé dans la détection du genre vidéo (Roach and Mason, 2001).

5.2.4 Approches basées sur la vidéo

Les approches basées sur la vidéo, comme la couleur, le mouvement, l'interprétation du contenu visuel, ont été très largement étudiées. Les paramètres de mouvement et d'interprétation du contenu visuel sont des paramètres de haut niveau. Ils permettent d'apporter des informations nouvelles et complémentaires par rapport aux paramètres de bas niveau et peuvent donc améliorer la détection du genre vidéo.

5.2.4.1 Paramètres basés sur la couleur

Une trame vidéo est composée d'une matrice de points appelés pixels. La couleur de chaque pixel est représentée par un jeu de couleurs représenté dans un espace de couleurs. Il existe plusieurs espaces de couleurs ; deux des plus populaires sont le Rouge Vert Bleu (RVB) et Teinte Saturation Valeur – Hue Saturation Value (HSV) (Gupta et al., 1997).

La distribution des couleurs dans une trame vidéo est souvent représentée en utilisant un histogramme ; mais cette représentation ne permet pas de donner l'in-

formation sur la localisation des pixels. Elle permet cependant de différencier facilement les cartoons des autres genres (Ianeva et al., 2003).

5.2.4.2 Paramètres basés sur la détection et l'identification des objets

Les paramètres basés sur la détection et l'identification des objets semblent être rares peut-être en raison de la difficulté à détecter et identifier des objets ainsi que la quantité de calculs à faire. Quand ces méthodes sont utilisées, les auteurs tentent de se focaliser sur la détection d'objets spécifiques, tels que les visages (Yuan et al., 2006; Wang et al., 2003).

Dimitrova (Dimitrova et al., 2000) et Wei (Wei et al., 2000) utilisent une approche proposée initialement dans (Wei and Sethi, 1999) pour détecter les visages : ils utilisent un modèle qui détecte dans l'image les pixels proches de la couleur de la peau.

5.2.4.3 Paramètres basés sur le mouvement et les transitions de scènes

Le mouvement à l'intérieur d'une vidéo est un descripteur qui est assez caractéristique du genre vidéo. On peut distinguer deux types de mouvement : celui des objets filmés et celui dû à la caméra. Dans quelques cas particuliers, il peut y avoir aussi des mouvements liés au défilement de texte pendant l'actualité. Les méthodes basées sur le mouvement consistent à calculer le flux optique.

Une autre façon de classifier le genre vidéo est de détecter les différentes transitions effectuées dans une vidéo (Wei et al., 2000). Ainsi, la plupart des types de transition de scènes tombe dans les catégories suivantes : rupture (*hard cuts*), ouverture/fermeture (*fades*) et fondu enchaîné (*dissolves*).

5.3 Contribution

Dans la littérature, la plupart des approches pour l'identification du genre sont basées sur la vidéo. Les auteurs ont proposé d'extraire des paramètres de bas niveau comme la couleur (Section 5.2.4.1) mais aussi des paramètres de plus haut niveau comme la détection et l'identification des objets (Section 5.2.4.2) ou le mouvement et les transitions de scènes (Section 5.2.4.3). Combinés, l'ensemble de ces paramètres ont ainsi permis d'obtenir une classification du genre vidéo plus robuste.

Malheureusement, peu d'études ont été faites sur l'extraction de paramètres haut niveau dans le domaine audio. Nos contributions ont porté sur deux domaines : la catégorisation dans le domaine cepstral, qui est l'approche la plus populaire en audio pour la classification de genre vidéo, et l'extraction de descripteurs audios de plus haut niveau.

5.3.1 Tâche et corpus

Les expériences sont conduites sur un corpus vidéo composé de 7 classes communément utilisées pour l'évaluation des méthodes de classification de genre vidéo : *publicité*, *sport*, *actualité*, *cartoon*, *documentaire*, *musique* et *film* (bandes annonces). La base de données contient 1 840 vidéos collectées depuis des plateformes de partage de vidéos. Les documents sont relativement courts : de 1 à 5 minutes avec une durée moyenne de 2 min 15 s. Le contenu de la parole est principalement en français, la classe *musique* contient des chansons en français et en anglais.

La collection est découpée en 2 parties : 1 260 vidéos sont utilisées pour l'apprentissage et 560 composent le test en équilibrant les vidéos sur chaque classe : 180 vidéos pour chaque classe dans le corpus d'apprentissage et 80 pour le corpus de test.

5.3.2 Coefficients cepstraux

Les coefficients cepstraux sont les plus fréquemment utilisés dans le domaine de la parole et de l'acoustique. Nous partons de l'idée que le flux audio peut être représenté comme une séquence de forme acoustique, chaque forme pouvant être estimée dans une fenêtre qui est la plus petite possible pour considérer l'état stationnaire du signal à l'intérieur. Des classifieurs statistiques estiment la probabilité de chaque hypothèse de classification en cumulant les statistiques à court terme (probabilités ou log-probabilités) sur la totalité de la séquence d'observation.

Classifier les documents en analysant les vecteurs acoustiques présente deux difficultés majeures. La première est la variabilité intra-classe qui peut être élevée en raison de la diversité des documents du même genre, par exemple : les publicités peuvent être composées de musique ou de parole exclusivement, quelques séquences de films peuvent être filmées dans un environnement bruité ou dans une pièce silencieuse, etc... La seconde difficulté est que les classes ne sont potentiellement pas séparables, les documents pouvant appartenir à différentes classes parce qu'ils sont vraiment proches : une publicité peut être vue comme un film, les musiques et les cartoons peuvent être difficilement distinguables par l'audio, etc...

Nous proposons d'utiliser le cadre de travail GMM-UBM où l'UBM modélise la totalité de l'espace acoustique du genre. Pour chaque genre, nous adaptons l'UBM pour obtenir un GMM spécifique au genre. La technique d'adaptation utilisée est le MAP utilisé de la même façon que dans le domaine de vérification du locuteur. Seules les moyennes sont adaptées. Les poids et la matrice de covariance restent inchangés.

Dans l'identification du locuteur, quelques techniques ont été proposées pour réduire la variabilité intra-classe. La FA a démontré une grande efficacité. Nous proposons ici d'évaluer la méthode pour réduire la variabilité intra-classe.

La FA permet de décomposer le modèle d'un genre en trois composantes différentes : une composante genre-session-indépendant, une composante genre-dépendant et une composante session-dépendante.

Un modèle peut être exprimé suivant son genre GE et suivant sa session h. Ainsi, le modèle FA peut être écrit :

$$\mathbf{m}_{(h,GE)} = m + D\mathbf{y}_{GE} + U\mathbf{x}_{(h,GE)} \tag{5.2}$$

où m est un super-vecteur (de dimension $M \cdot N$) défini comme la concaténation des moyennes du GMM, N la dimension de l'espace acoustique (39 dans notre cas), M le nombre de gaussiennes dans l'UBM, D la matrice diagonale $MN \times MN$, $\mathbf{y}_{GE}$ un vecteur de genre aléatoire (un vecteur de dimension MN), U est la matrice de variabilité session de rang R (une matrice de taille $MN \times R$) et $\mathbf{x}_{(h,GE)}$ une variable aléatoire. $\mathbf{y}_{GE}$ et $\mathbf{x}_{(h,GE)}$ sont normalement distribués autour de $\mathcal{N}(0, I)$. D satisfait l'équation suivante $\mathbf{I} = \tau D^t \mathbf{\Sigma}^{-1} D$ où τ est le facteur de pertinence requis pour l'adaptation MAP et DD^t représente la matrice de covariance *a priori* de $\mathbf{y}_{GE}$.

5.3.2.1 Tâche de classification

Cette section détaille la stratégie employée pour effectuer la compensation de variabilité inutile. La tâche de classification est définie comme suit. Un genre GE_{tar} est inscrit par le système avec ses données d'apprentissage $Y_{GE_{tar}}$. Le modèle retenu pour le genre GE_{tar} est :

$$m_{(\mathbf{h}_{tar}, GE_{tar})} = m + \mathbf{D}y_{GE_{tar}}. \tag{5.3}$$

La tâche de classification de genre consiste à déterminer si une trame du test $\mathcal{Y}$ appartient à GE_{tar} ou pas. Utilisant la décomposition par FA dans les données de test, nous pouvons écrire :

$$m_{(\mathbf{h}_{test}, GE_{test})} = m + \mathbf{D}y_{GE_{test}} + \mathbf{U}x_{\mathbf{h}_{test}}. \tag{5.4}$$

Les genres GE_{tar} dans les données d'apprentissage ainsi que GE_{test} dans les données de test ont été distingués. Dans ce travail, une stratégie hybride est utilisée dans le but de retirer la composante inutile dans les données du test au niveau des paramètres acoustiques. Une trame x est modifiée comme suit :

$$\hat{x} = x - \sum_{g=1}^{M} \gamma_g(x) \cdot \{\mathbf{U} \cdot x_{\mathbf{h}_{test}}\}_{[g]}. \tag{5.5}$$

où $\gamma_g(x)$ est la probabilité *a posteriori* de la gaussienne g donnée par la trame x. Ces probabilités sont estimées en utilisant l'UBM. $\mathbf{U} \cdot x_{\mathbf{h}_{test}}$ est le super-vecteur avec $M \times N$ composantes.

5.3.2.2 Score

La fonction de score est donnée par :

$$LLK(\mathcal{Y}|m + \mathbf{D}y_{GE_{tar}}) - LLK(\mathcal{Y}|m) \tag{5.6}$$

où $LLK(\cdot|\cdot)$ indique la moyenne de la fonction de log-vraisemblance de toutes les trames. Ici, les GMM partagent leurs matrices de covariance ainsi que le poids des mixtures (les deux ont été supprimés de l'équation pour plus de clarté). La soustraction de l'information inutile dans le test est effectuée au niveau des trames (domaine cepstral).

5.3.2.3 SVM

En utilisant l'équation 5.7, le modèle FA estime le super-vecteur contenant seulement l'information du genre. Dans (Campbell et al., 2006), les auteurs proposent un noyau qui calcule une distance entre les super-vecteurs. Désignons $\mathcal{X}_{\mathbf{s}}$ et $\mathcal{X}_{\mathbf{s}'}$ deux séquences de données audios correspondant aux genres GE et GE' ; l'équation du noyau peut s'écrire ainsi :

$$K(\mathcal{X}_{GE}, \mathcal{X}_{GE'}) = \sum_{g=1}^{M} \left(\sqrt{\alpha_g}\boldsymbol{\Sigma}_g^{-\frac{1}{2}} m_{GE}^g\right)^t \left(\sqrt{\alpha_g}\boldsymbol{\Sigma}_g^{-\frac{1}{2}} m_{GE'}^g\right). \tag{5.7}$$

Ce noyau est valide seulement si les moyennes du modèle GMM varient (poids et matrices de covariance sont ceux de l'UBM). m_{GE} est pris ici du modèle 5.3, *i.e.* $m_{GE} = m + \mathbf{D}y_{GE}$.

5.3.2.4 Protocole et résultats

Toutes les expériences ont été réalisées en utilisant le toolkit ALIZE et LIA_SpkDet (Bonastre et al., 2005; Charton et al., 2008) et LaRank SVM (Bordes et al., 2007). Dans nos expériences, nous utilisons les paramètres acoustiques MFCC en utilisant une fenêtre Hamming de 25 ms. Chaque trame est composée de 39 coefficients (MFCC 13, δ MFCC 13 et $\delta\delta$ MFCC 13) toutes les 10 ms. La prochaine section décrit tous les différents systèmes que nous avons testés dans nos expériences.

5.3.2.5 GMM-UBM-FA

Les GMM-UBM sont entrainés avec l'algorithme d'EM. Pour un genre donné, le GMM est obtenu par un MAP où on adapte uniquement les moyennes.

Pour un UBM et un genre donné, la décomposition par FA est réalisée avec l'Equation 5.2. Le modèle retenu pour le genre GE_{tar} est donné par $m_{GE_{tar}} = m +$

$\mathbf{D}y_{GE_{tar}}$. Les scores de classification sont estimés comme expliqué dans la section 5.3.2.1.

5.3.2.6 SVM-UBM et FA

Un Support Vector Machine (SVM) est un classifieur à deux classes. Afin d'utiliser un SVM sur un problème multi-classe, nous proposons d'utiliser le toolkit SVM LaRank (Bordes et al., 2007). L'algorithme LaRank est inspiré de la méthode de descente de gradient.

5.3.2.7 Résultats

Dans les expériences ci-dessous, nous étudions l'impact de la FA dans la classification de genre vidéo. Nous utilisons un GMM doté de 256 gaussiennes et une matrice U de rang 40 (rang ayant obtenu les meilleurs résultats).

TABLE 5.1 – *Paramètres cepstraux : taux d'erreurs de classification (%) par genre.*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport	Total
GMM-UBM	66	64	66	71	58	76	28	61
GMM-UBM-FA	4	14	16	18	24	55	13	18
SVM-UBM-FA	4	18	14	14	23	9	17	11

La première ligne montre les résultats obtenus avec une approche GMM-UBM, la plus couramment utilisé dans le domaine (*baseline*). Pour les résultats de l'approche GMM-UBM-FA à la seconde ligne, on peut constater que les performances sont grandement améliorées grâce à la FA avec une réduction du taux d'erreurs de 43 points. Pour le système SVM-UBM-FA, nous observons une réduction du taux d'erreurs de 50 points par rapport au système GMM-UBM.

Le Tableau 5.2 reporte la matrice de confusion du système SVM-UBM-FA. Nous observons que le système a correctement classifié les classes *documentaire, cartoon* et *publicité* avec respectivement un Taux d'Erreur de Classification – Correct Error Rate (CER) de 4%, 2% et 9%. Cependant, les classes *actualité, film, musique* et *sport* obtiennent les plus mauvais résultats. Néanmoins, le fossé entre toutes les classes est significativement réduit par rapport à la baseline : tous les scores sont dans l'intervalle [82-98].

La FA pour l'identification de locuteur est utilisée comme état de l'art dans les systèmes. Néanmoins, la tâche de Identification du Genre Vidéo (IGV) est vraiment différente : au contraire de l'identification de locuteur, le nombre de classes est vraiment petit (de 5 à 10 classes) et la variabilité intra-classe est vraiment grande.

Nos expériences démontrent que la décomposition par FA peut correspondre

TABLE 5.2 – *Matrice de confusion (%) pour les coefficients cepstraux avec un SVM-UBM et une méthode de FA (SVM-UBM-FA)*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport
Documentaire	96	2	0	0	1	1	0
Actualité	13	82	0	1	0	4	0
Film	2	0	86	0	1	11	0
Music	0	0	11	86	0	2	0
Cartoon	0	0	0	2	98	0	0
Publicité	2	0	5	0	2	91	0
Sport	2	5	0	2	4	4	83

au problème IGV : le taux de classification est réduit d'environ 81% en erreur relative.

5.3.3 Paramètres acoustiques de haut niveau

La première partie démontre que les descripteurs cepstraux contiennent des informations pertinentes sur le genre vidéo. Cependant, cela reste une approche bas niveau et des descripteurs de haut niveau pourraient apporter différents points de vue sur le document. Les paramètres que nous proposons tentent de modéliser la structure du document ou son contenu. Les prochaines sections étudient les paramètres reliés à la morphologie du document, spécialement les paramètres de l'interactivité du locuteur et de la qualité du contenu vidéo. Pour chacun de ces paramètres, nous étudierons les performances seules et nous les combinerons à celles évaluées précédemment.

5.3.4 Paramètres d'interactivité

Le nombre de personnes et la façon dont elles communiquent peuvent différer selon le genre. Par exemple, il y a généralement un seul locuteur dans le genre *actualité*, au contraire des *cartoons* ou *film* qui contiennent généralement beaucoup de locuteurs avec un temps de parole très variable ainsi que plusieurs tours de parole. L'interactivité a pour but de représenter le profil d'interaction, elle est composée de 3 paramètres : le nombre de tours de parole, le nombre de locuteurs et le temps de parole du locuteur principal.

Ces données sont extraites en utilisant un système de segmentation et regroupement en locuteurs. La première étape effectue une segmentation en Viterbi basée sur les classes suivantes : "parole", "parole sur de la musique" et "musique". Chacun de ces modèles est un GMM de 64 mixtures. Les vecteurs acoustiques sont composés de 12 coefficients MFCC, de l'énergie ainsi que de leurs dérivées première et seconde. Ensuite, les deux dernières étapes effectuent une détection du

tour de locuteurs et un regroupement en locuteurs. Nous utilisons le système décrit dans (Dan Istrate, 2005) basé sur un Bayesian Information Criterion (BIC). Ces techniques permettent d'estimer le nombre de locuteurs et le nombre de tours de parole pour chaque vidéo.

Les trois paramètres d'interactivité composent un vecteur qui est envoyé à un classifieur SVM pour l'identification du genre.

TABLE 5.3 – *Les paramètres d'interactivité : taux d'erreurs de classification (%) par genre.*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport	Total
CER -Int.	28	28	23	31	72	14	87	38

Les résultats reportés dans le Tableau 5.3 montrent que l'interactivité est clairement moins précise que les paramètres acoustiques.

TABLE 5.4 – *La matrice de confusion (%) pour les paramètres d'interactivité et un classifieur SVM.*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport
Documentaire	72	5	1	1	1	0	0
Actualité	3	72	0	3	11	5	6
Film	0	3	77	0	9	11	0
Musique	22	0	0	69	9	0	0
Cartoon	0	3	46	0	28	23	0
Publicité	0	4	9	0	0	86	1
Sport	0	31	20	0	7	29	13

La distribution des erreurs par genre est tout à fait différente de celle obtenue pour les paramètres acoustiques (Tableau 5.2). Les erreurs les plus fréquentes concernent le *sport* et le *cartoon* tandis que pour les paramètres acoustiques, il s'agit de l'*actualité*. Ces différences qualitatives correspondent à nos attentes ; l'information structurée est liée à l'organisation globale du document qui est clairement spécifique à l'*actualité* mais probablement sans importance pour le style éditorial qui est faiblement défini.

5.3.5 Paramètres de qualité de la parole

Nous partons de l'idée que la qualité de la parole pourrait fournir des informations pertinentes sur le genre. Par exemple, la parole est claire dans l'actualité où le domaine linguistique est bien couvert par les systèmes de reconnaissance de la parole contrairement à la publicité où le domaine linguistique peut être inattendu en raison des spécifications du produit et du type de locuteurs.

Nous utilisons 3 paramètres dans ce groupe, tous basés sur le système de transcription du LIA, SPEERAL. Le premier descripteur est la probabilité *a posteriori* de la première hypothèse. Nous utilisons comme mesure de confiance les scores linguistiques et acoustiques. Le second est la probabilité linguistique de la meilleure hypothèse. Le dernier paramètre est basé sur l'entropie phonétique. Ce descripteur a été introduit par (Jitendra Ajmera and Bourlard, 2002) pour la séparation musique/parole. Il est calculé comme l'entropie de la probabilité acoustique :

$$H(n) = -\frac{1}{N}\sum_{m=1}^{N}\sum_{k=1}^{K} P(q_k|x_m)\log_2 P(q_k|x_m) \tag{5.8}$$

où les valeurs des trames sont calculées sur une fenêtre glissante de taille N, K représente un modèle phonétique et $P(q_k|x_m)$ représente la probabilité d'avoir un phonéme connaissant une trame. Cette mesure est supposée être grande sur une qualité de parole moyenne et décroître sur une qualité de parole propre.

TABLE 5.5 – *Qualité de la parole : taux d'erreurs de classification (%) par genre.*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport	Total
Q	22	21	52	30	50	71	24	39

Les paramètres de qualité de parole sont proches de ceux obtenus à l'interactivité en termes de taux d'erreurs : nous obtenons 39% CER tandis que les paramètres d'interactivité ont un taux d'erreurs de 38%. La distribution des erreurs est très différente comme montré dans le Tableau 5.6 : les meilleures classes sont l'*actualité* et le *documentaire* qui contiennent normalement de la parole correspondant aux conditions d'entraînement d'un système de RAP.

TABLE 5.6 – *La matrice de confusion en (%) sur les paramètres de qualité de parole.*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport
Documentaire	78	8	1	1	9	2	1
Actualité	12	79	1	0	1	7	0
Film	4	0	48	21	21	6	0
Musique	0	0	7	70	0	21	2
Cartoon	17	3	14	16	50	0	0
Publicité	7	5	5	41	8	29	5
Sport	0	0	2	11	0	11	76

5.3.6 Paramètres linguistiques

5.3.6.1 Introduction

Les travaux sur l'analyse linguistique ont été réalisés avec Stanislas Oger. L'analyse du contenu linguistique des vidéos que nous proposons repose sur l'utilisation d'un système de RAP pour obtenir les transcriptions des vidéos. Ce système utilise un lexique fermé et un modèle de langage qui est estimé sur un corpus textuel de grande taille. Entraîner un tel modèle pour chaque genre vidéo n'est pas réalisable car nous ne disposons pas du volume de données textuelles nécessaire. Nous proposons donc d'utiliser un modèle de langage standard, malgré le décalage entre le modèle générique et les particularités des genres, qui causera la plupart du temps un fort taux d'erreurs dans les transcriptions.

Le principe du "sac-de-mots" est utilisé pour la modélisation des documents. Selon ce modèle, chaque dimension de l'espace des paramètres représente un terme et chaque document est représenté par un vecteur de fréquence de terme dans cet espace.

Pour les problèmes de catégorisation automatique de texte, les approches généralement proposées reposent sur l'extraction de mots porteurs de sens des documents à classer. Pour la classification du genre vidéo, les études s'appuient sur la modalité textuelle utilisant en général cette approche. Soit les mots-outils de la langue sont filtrés par une *stop-liste*, soit une métrique de type Term Frequency-Inverse Document Frequency (TF-IDF) est utilisée pour ne sélectionner que les mots porteurs de sens des documents (Takenobu and Makoto, 1994). Cette approche sera notre système de base. En effet, nous proposons ici une approche différente et inhabituelle, dans laquelle les fréquences des mots-outils peuvent être utilisées pour identifier le genre écrit.

5.3.6.2 TF-IDF

Pour un terme t et un document d, TF-IDF est défini comme suit :

$$w_{i,j} = tf_{i,j} * idf_i \tag{5.9}$$

avec $tf_{i,j}$ la fréquence normalisée du terme i dans le document j et idf_i une métrique représentant le pouvoir discriminant du terme i. Ainsi avec le $tf \cdot idf$, plus la valeur d'un mot est élevée, plus le mot considéré est représentatif du document et porteur de la thématique qu'il aborde.

Pour chaque genre, nous construisons un vecteur de paramètres avec les n termes ayant les meilleurs $tf \cdot idf$ de chaque document. Ces vecteurs sont ensuite regroupés dans un super-vecteur qui est fourni au classifieur.

5.3.6.3 Les mots-outils

Les méthodes précédentes permettent d'identifier des termes discriminants pour un document ou un genre. Ces termes sont souvent des mots porteurs de sens et plutôt rares en général, ils auront donc une forte probabilité d'être victimes du décalage entre le lexique du système de RAP et celui du document. Nous pensons que les mots-outils peuvent tout aussi bien être porteurs d'information pour détecter le genre vidéo. Contrairement à l'approche TF-IDF, celle-ci est indépendante des thématiques des documents et est donc plus robuste pour classifier des genres comme les classes *actualité*, *documentaire* et *cartoon*, qui abordent des thématiques très variées. De plus, les mots outils sont caractérisés par leurs fréquences très élevées et sont donc robustes à la couverture lexicale incomplète d'un système de RAP.

Les n termes les plus fréquents des transcriptions automatiques des documents du corpus d'entraînement servent ainsi de paramètres au classifieur bas-niveau.

5.3.6.4 Evaluation

Nous avons choisi de tester deux types de classifieurs : Boosting et Artificial Neural Network (ANN). Concernant l'extraction des paramètres TF-IDF, la taille du vecteur n est déterminé empiriquement sur un corpus de développement. Il est constitué de 6 000 mots. La fréquence des mots dans le vecteur des paramètres de chaque document est normalisée en respectant la taille du document. En outre, le nombre total de mots dans le document est ajouté dans le vecteur comme un paramètre. Le classifieur Boosting obtient les meilleurs résultats et il permet d'obtenir un CER de 27.9%. Ce résultat constituera notre système de base et sera reporté dans la Figure 5.1.

Pour les paramètres des mots-outils, le taux correct de classification des 2 classifieurs est présenté dans la Figure 5.1 en fonction du nombre de mots présents dans le vecteur. Nous notons que l'ANN est un MLP doté d'une seule couche cachée ; la taille de la couche cachée est optimisée sur le corpus d'entraînement. Dans le vecteur des paramètres, les fréquences sont brutes. Nous avons observé que les résultats sont meilleurs lorsque nous normalisons les données.

Les performances obtenues sont comparables à celles de notre système de base des 6 000 paramètres, alors qu'elles sont ici obtenues avec seulement 23 paramètres. Le meilleur CER obtenu est de 29.6% avec un classifieur ANN en utilisant les 100 mots les plus fréquents.

Avec seulement le mot le plus fréquent, <*sil*>, lequel représente un silence, le meilleur classifieur obtient un CER à 51.4% et en ajoutant le second mot cela donne un CER à 46.1%. Nous observons qu'en ajoutant un mot, les gains suivent une loi inverse du logarithme. Nous pouvons conclure que plus la fréquence du mot est élevée, plus le mot est saillant et porteur d'information pour l'identification selon

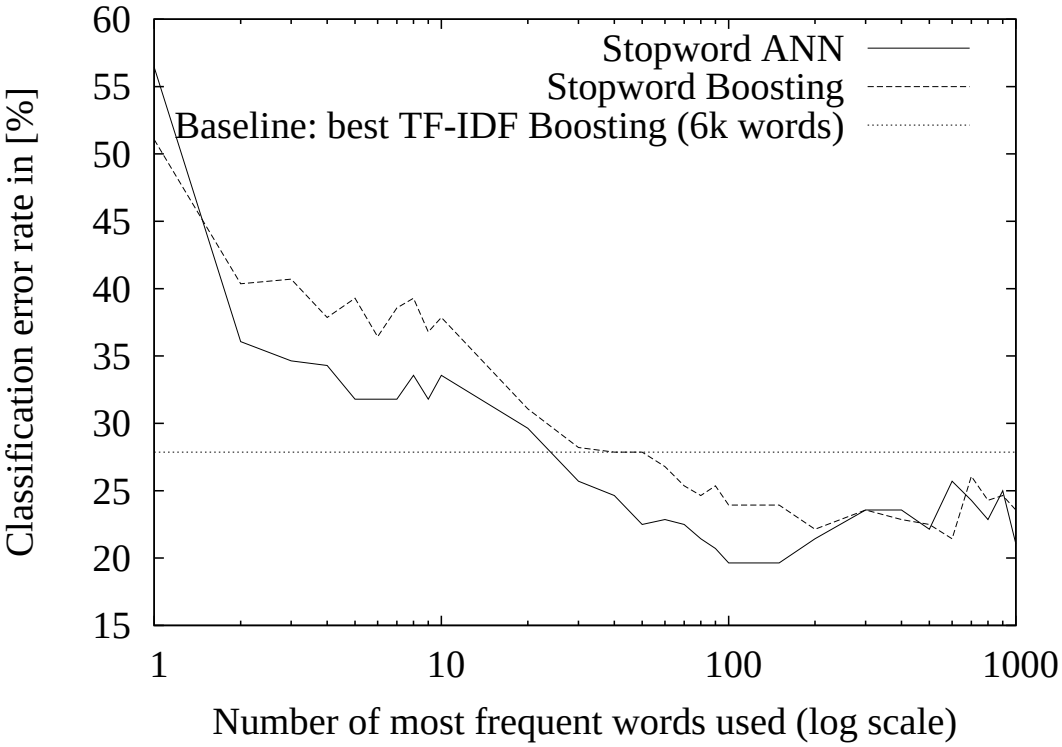

FIGURE 5.1 – *CER (%) du classifieur ANN et Boosting en utilisant les paramètres mots-outils selon le nombre de mots utilisés (la figure est extraite de l'article : "Transcription-based video genre classification").*

le genre. Le Tableau 5.7 contient les neufs mots les plus fréquents dans le corpus d'apprentissage, associés à leur fréquence.

Ces performances valident notre hypothèse initiale que la fréquence des mots-outils contient l'information qui est caractéristique au genre vidéo. Plus encore, l'approche proposée permet d'obtenir un gain absolu de 8% par rapport à notre baseline TF-IDF, tandis que l'espace de représentation est réduit de 98%.

TABLE 5.7 – *Fréquence des 9 mots les plus fréquents trouvés dans une transcription automatique sur le corpus d'apprentissage.*

Mot	Fréquence	Mot	Fréquence	Mot	Fréquence
<sil>	146100	et	12236	est	9385
de	20093	le	10961	des	8682
les	12526	la	10819	il	7628

Les résultats reportés dans le Tableau 5.8 montrent que les paramètres linguistiques sont relativement pertinents pour l'identification du genre vidéo.

TABLE 5.8 – *Paramètres linguistiques : taux d'erreurs de classification (%) par genre.*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport	Total
L	07	15	43	4	07	25	3	24

TABLE 5.9 – *La matrice de confusion en (%) sur les paramètres linguistiques.*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport
Documentaire	93	01	01	01	01	03	0
Actualité	08	85	04	01	0	02	0
Film	0	01	57	19	01	20	02
Musique	0	0	10	60	02	22	06
Cartoon	01	0	01	05	93	0	0
Publicité	0	0	14	05	0	75	06
Sport	0	04	04	04	02	16	70

5.3.6.5 Transcription de référence

Afin de déterminer si ces résultats sont liés à la nature du corpus issu de la RAP, nous avons mené les mêmes expériences sur un corpus de sous-titres qui contient les transcriptions exactes de vidéos de quatre genres : *cartoon*, *documentaire*, *actualité* et *film*. Ce corpus n'est pas la cible principale de nos travaux, mais il va nous servir de référence pour identifier les spécificités de l'identification de genre sur les sorties de la RAP. Il contient 1960 documents dont 1400 servent pour l'entraînement et 560 pour le test. Les vidéos auxquelles sont associés ces sous-titres durent de 25 minutes à 2 heures.

Les résultats sont présentés dans le tableau 5.10 (colonnes intitulées ST), aux côtés de ceux obtenus sur le corpus issu de la RAP (colonnes intitulées RAP) en ne prenant en compte que les quatre genres considérés. On constate que la méthode TF-IDF est la meilleure approche pour les sous-titres, suivie de près par TF. Sur le corpus issu de RAP, ces résultats sont inversés, ce qui indique que TF est plus robuste aux erreurs de reconnaissance que TF-IDF. Ce résultat s'explique probablement par le fait que les mots fréquents sont moins sensibles aux erreurs dues à la couverture lexicale du système de RAP. De plus, la méthode TF fonctionne très bien sur le corpus de sous-titres, ce qui indique que les fréquences d'utilisation des mots-outils de la langue contiennent une information caractéristique du genre vidéo qui n'est pas liée au système de RAP : c'est un phénomène linguistique.

Ces résultats valident notre hypothèse initiale : les fréquences des mots-outils contiennent une information permettant de caractériser le genre vidéo. De plus, le modèle TF permet une réduction de l'espace de représentation de 99.7%.

TABLE 5.10 – *Taux d'erreurs de classification (%cer) et nombre de paramètres optimaux (#p) obtenus sur le corpus de vidéos issu de la RAP (RAP) et sur le corpus de sous-titres (ST) en fonction de la métrique utilisée (TF ou TF-IDF)*

System		ST	RAP
TF	%cer	21	11
	#p	700	80
TF-IDF	%cer	17	13
	#p	50k	34k

5.3.7 Combinaison de paramètres audios

Cette section présente le système qui intègre tous les paramètres audios précédemment décrits en groupant ceux-ci dans un large vecteur de 17 coefficients.

Les 7 premiers coefficients sont les sorties des 7 scores données par le classifieur SVM sur les paramètres cepstraux. Les 7 coefficients suivants sont les sorties du classifieur linguistique (les 7 sorties du réseau de neurones). Les 6 derniers coefficients sont respectivement les 3 paramètres d'interactivité et les 3 paramètres de qualité de la parole. Ensuite, nous entrainons un SVM à noyau linéaire sur ces super-vecteurs.

Etant donné le manque de données d'apprentissage, les modèles SVM sont entraînés par une stratégie de *leave-one-out*. Le corpus d'apprentissage a été découpé en 6 parties : 5 parties utilisées pour entraîner les différents modèles (modèle cepstral, linguistique) et la dernière pour entraîner le méta-modèle.

Afin d'estimer la complémentarité des paramètres, la combinaison est réalisée étape par étape : nous commençons par le meilleur groupe de paramètres (descripteur cepstral) et nous ajoutons successivement les meilleurs descripteurs restants : linguistique, interactivité et qualité de la parole.

5.3.7.1 Résultats

Les résultats de la combinaison globale sont reportés dans le Tableau 5.11. Nous observons un gain absolu de 3% par rapport au meilleur descripteur (descripteur cepstral). Ces résultats montrent que tous les paramètres proposés sont globalement complémentaires pour la classification de genre.

Nous pouvons observer que le système a correctement classifié les classes *documentaire*, *film*, *cartoon*, *musique* et *publicité* ; mais les classes *actualité* et *sport* obtiennent les plus mauvais résultats. Le Tableau 5.12 montre que la classe *acutalité* est fréquemment substituée à la classe *documentaire*. Les résultats pour la classe *sport* sont probablement affectés par une large variabilité intra-classe, groupant des sources variables (course de voitures, football...).

TABLE 5.11 – *CER (%) sur la combinaison des paramètres audios combinés.*

System	Doc.	Actu.	Film	Cartoon	Musique	Pub.	Sport	Total
AS	04	18	14	14	02	09	17	11
AS+L	03	13	11	07	02	08	19	09
AS+L+Int	03	12	11	07	03	08	19	09
AS+L+Int+Q	04	14	07	02	05	06	18	08

TABLE 5.12 – *La matrice de confusion en (%) sur la combinaison des paramètres audios.*

System	Doc.	Actu.	Film	Cartoon	Musique	Pub.	Sport
Doc.	96	0	0	02	01	0	0
Actualité	10	86	0	0	0	04	0
Film	0	0	93	04	0	03	0
Musique	0	0	02	98	0	0	0
Cartoon	0	0	02	03	95	0	0
Publicité	0	0	02	04	0	94	0
Sport	0	05	0	03	0	10	82

5.4 MediaEval 2011 - *Genre Tagging*

En juillet 2011, le LIA a participé à la campagne d'évaluation MediaEval 2011 [3] sur la tâche de détection de genre (*Genre Tagging*). La campagne proposait d'assigner automatiquement pour chaque vidéo un et un seul tag parmi les 26 tags proposés [4].

Le corpus était constitué de vidéos issues de *blip.tv*. Il contenait 1 974 vidéos (247 pour le corpus de développement et 1 727 pour le corpus de test), correspondant à environ 350 heures de données. Pour chaque vidéo, étaient associés : des métadonnées (titre, description, tags, utilisateur), une transcription automatique issue d'un système de RAP ainsi que les commentaires des vidéos postés sur Twitter. Les participants pouvaient envoyer jusqu'à 5 soumissions. Les résultats soumis étaient évalués selon la métrique Mean Average Precision (MAP).

Le genre vidéo tel que proposé dans la campagne d'évaluation associait des étiquettes liées au style éditorial et à la thématique de la parole. Cette catégorisation nous a obligés à proposer un système différent qui garde les descripteurs acoustiques et linguistiques.

3. MediaEval : `http://www.multimediaeval.org/mediaeval2011/`

4. art, autos and vehicles, business, citizen journalism, comedy, conferences and other events, default category, documentary, educational food and drink, gaming, health, literature, movies and television, music and entertainment, personal or auto-biographical, politics, religion, school and education, sports, technology, the environment, the mainstream media, travel, videoblogging, web development and sites

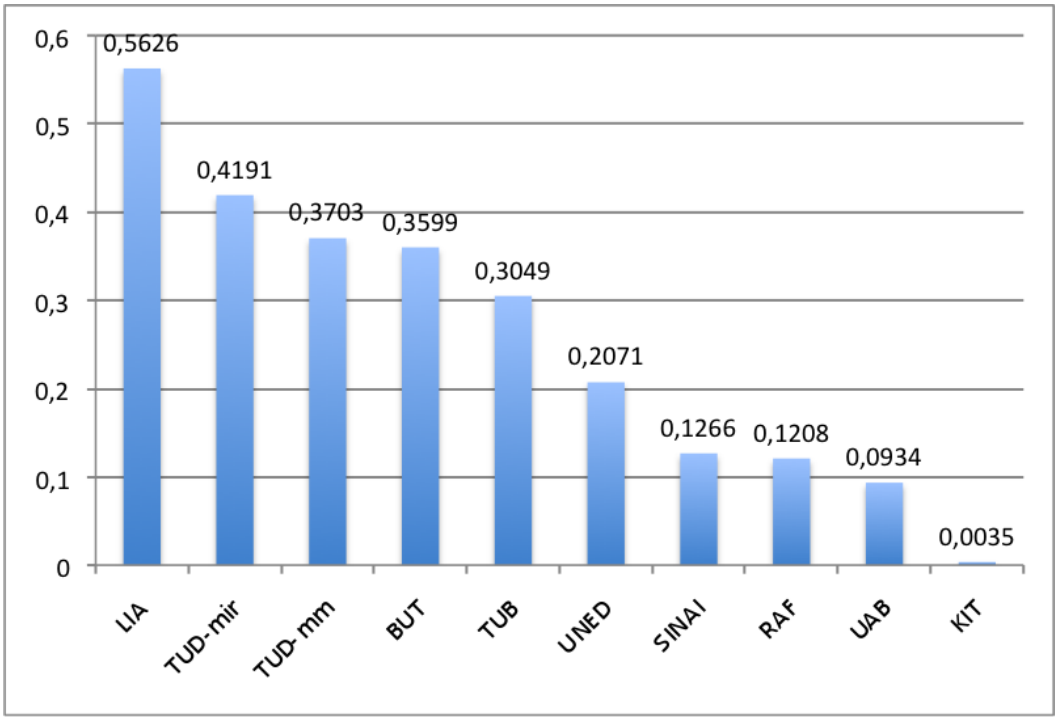

FIGURE 5.2 – *Résultat de la campagne d'évaluation MediaEval 2011 sur la tâche détection du genre (Genre Tagging). Les résultats sont exprimés en MAP (%).*

Lors de cette campagne, nous avions la possibilité d'utiliser les métadonnées associées à chaque vidéo. Nous avons observé que le nom de la personne qui a posté la vidéo (présent dans les métadonnées) peut donner des informations intéressantes sur le genre de la vidéo. En effet, un utilisateur va souvent envoyer des vidéos dans le même genre, par exemple : les utilisateurs *Anglicantv* ou *Aabbey1* (utilisateur de *bilp.tv*) vont souvent envoyer des vidéos dans le genre *Religion*. L'information sur l'utilisateur nous a permis d'améliorer notre système est de remporter cette campagne d'évaluation (cf. Figure 5.2).

Les informations sur l'utilisateur sont des informations importantes pour la détection du genre. A notre connaissance, nous n'avons pas encore trouvé dans la littérature de travaux traitant cette information.

5.5 Conclusion

Nous avons présenté nos recherches dans le domaine de l'identification de genre. La première contribution concerne la catégorisation dans le domaine cepstral qui est l'approche la plus populaire en audio pour la classification de genre vidéo. Nous démontrons que la réduction de variabilité par FA améliore sensiblement la précision du classifieur.

L'extraction automatique de paramètres linguistiques est fortement dépendante des performances du système de RAP, spécialement sur la couverture du lexique qui peut être critique dans ces domaines ouverts. Nous proposons de caractériser le genre linguistique en utilisant un classifieur statistique sur les mots les plus fréquents du langage, lequel est supposé être spécifique au style éditorial plutôt qu'au sujet. Les expériences confirment cette idée.

Nous avons observé pendant la campagne d'évaluation MediaEval 2011 que l'information sur l'utilisateur pouvait être utilisée comme paramètre afin d'améliorer notre système.

La classification du genre vidéo est un des descripteurs qui nous permet de structurer les vidéos issues du Web. Cette structuration, nous permet selon la finalité de notre zapping d'obtenir des vidéos selon un genre précis (par exemple un zapping focalisé sur l'actualité) ou d'obtenir un zapping sur un sujet précis, mais dans lequel les genres alternent (afin de donner une certaine dynamique à notre document).

Chapitre 6

Structuration de document : détection du niveau de spontanéité

Sommaire

6.1 Introduction

La détection du niveau de spontanéité a été étudiée ces dernières années par plusieurs auteurs dans le domaine de la reconnaissance de la parole ou de l'interprétation. La détection du niveau de spontanéité peut être utilisée comme un descripteur dans différentes applications comme par exemple, le traitement de la parole spontanée dans un système de RAP (Dufour et al., 2010a) ou encore dans les systèmes de résumé automatique, pour soit re-travailler certaines phrases (en supprimant les faux départs, etc...), soit supprimer certaines phrases trop spontanées et donc trop bruitées pour le système (Zhu and Penn, 2006).

Dans le résumé sous forme de zapping, le niveau de spontanéité peut nous être utile pour détecter les sous-segments d'une vidéo ayant un intérêt notable. En effet

lors d'un débat, quand une question embarrassante ou inattendue est posée à une personne, l'interlocuteur choqué par cette question peut hésiter, douter, bégayer...

Une des premières difficultés de l'identification du niveau de spontanéité tient au fait que les structures acoustiques et linguistiques de la parole spontanée sont complètement différentes de celles de la parole lue ou préparée : les locuteurs hésitent fréquemment, s'interrompent, changent leur débit, etc... Certaines études décrivent les disfluences comme étant la plus grande caractéristique de la parole spontanée (Bazillon et al., 2008; Adda-Decker et al., 2004).

Dans (Dufour et al., 2009) pour détecter la parole spontanée, les auteurs proposent d'utiliser des descripteurs prosodiques et linguistiques, ces derniers étant extraits à partir d'un système de RAP. Dans (Jousse et al., 2008) nous constatons très clairement que le WER est fortement corrélé au niveau de la spontanéité. Ainsi on peut voir que les systèmes de transcription obtiennent sur de la parole préparée un WER aux alentours de 20%, sur de la parole faiblement spontanée un WER se situant aux alentours de 45% et sur de la parole spontanée les taux d'erreurs oscillent entre 45% et 60%. Le degré de spontanéité a un impact considérable sur les performances d'un système de transcription. Dans le cadre de la détection du niveau de spontanéité, celui-ci pourra perturber fortement l'extraction des descripteurs linguistiques. Afin de ne pas être dépendant des performances d'un système de transcription de la parole, nous proposons de nous focaliser uniquement sur l'acoustique.

Nous proposons de combiner des paramètres acoustiques différents et complémentaires, où chaque paramètre détecte une disfluence caractéristique de la parole spontanée. Ici, notre objectif est d'évaluer le niveau de spontanéité d'un segment de parole, qui pourrait être candidat à l'intégration dans les résumés.

6.2 Contribution

6.2.1 Tâche et corpus

Les expériences ont été conduites sur le corpus français EPAC composé de parties spontanées issues de la radio (Estève et al., 2010). Chaque tour de parole est annotée avec un jeu de 10 étiquettes, chacune correspondant à un niveau de spontanéité : le niveau 1 correspond à de la parole préparée (souvent similaire à de la parole lue) et le niveau 10 correspond à de très grandes disfluences dans la parole (parfois même incompréhensible). Dans nos expériences, 3 classes sont considérées : parole préparée (E1) correspondant au niveau 1, parole faiblement spontanée (E2) correspondant aux niveaux 2 et 4 et parole fortement spontanée (E3) correspondant aux niveaux 5 et plus.

Cet étiquetage en niveau de spontanéité a été effectué par deux annotateurs. Ce corpus a, au préalable, été segmenté automatiquement au moyen du système de segmentation du Laboratoire d'Informatique de l'Université du Maine (LIUM).

Pour pouvoir évaluer l'accord inter-annotateurs sur cette tâche, le coefficient Kappa (Cohen, 1960) de cet accord a été calculé sur une heure d'émission radiophonique. Le score obtenu pour les trois classes de spontanéité était de 0.85, un score supérieur à 0.8 étant considéré comme excellent (Di Eugenio and Glass, 2004).

La durée totale du corpus est de 11 h 37 pour 3 322 segments de paroles : 1 142 de ces segments sont étiquetés comme parole préparée, 1 175 comme parole faiblement spontanée et 1 005 comme parole fortement spontanée.

6.2.2 Principe et architecture du système

L'architecture du système proposé est composée de 2 niveaux : chaque niveau, permet d'évaluer le niveau de spontanéité des segments localement puis globalement.

Le premier niveau consiste à extraire les paramètres acoustiques. Nous identifions 3 paramètres acoustiques différents. Les paramètres acoustiques vont essayer de se focaliser respectivement sur la détection des disfluences liées aux pauses, aux émotions et aux variations du débit. Afin de tirer parti des 3 jeux de paramétrisation et d'améliorer les résultats, nous proposons de fusionner les scores obtenus de l'ensemble des classifieurs.

Jusqu'à présent pour détecter le niveau de spontanéité d'un segment nous nous servons de ses informations. L'estimation était faite de manière locale. Dans le second niveau, nous proposons un modèle qui permet de détecter le niveau de spontanéité du segment en fonction des autres segments. Ce modèle global re-score la probabilité du niveau de spontanéité par segments selon le contexte des segments co-occurents.

6.2.3 Paramètres acoustiques

6.2.3.1 Les pauses

Dans un discours, les pauses apparaissent comme des marqueurs de la parole spontanée. Ces pauses peuvent être classées en deux catégories : les pauses silencieuses et les pauses sonores.

Dans le cadre de la parole spontanée, les pauses silencieuses marquent souvent la rupture au niveau d'une idée. Ces pauses permettent de structurer le discours (Campione and Véronis, 2004). De plus, dans (Bazillon et al., 2008), les auteurs affirment que les pauses de respiration dans ce type de parole étaient beaucoup plus nombreuses et plus longues que celles que l'on pouvait retrouver en parole préparée. Cette différence est due au fait que ce type de parole est conçu à l'instant où le locuteur parle, il lui arrive donc de devoir s'arrêter pour continuer à construire son discours.

Les pauses sonores sont des phénomènes typiques de l'oral. En effet, les pauses remplies regroupent les morphèmes tels que "euh", "hum" ou encore "ben". Dans (Bazillon et al., 2008), les auteurs montrent la difficulté de définir la fonction de ces pauses remplies, appelées aussi morphèmes. Le morphème "euh" est alors catégorisé en tant qu'hésitation mais pour les autres morphèmes, une catégorisation reste plus délicate. Les auteurs donnent alors l'exemple des emplois de "ben", qui peut être adverbe, conjonction de coordination... Les pauses nous paraissent un marqueur discriminant pour la détection de la parole spontanée.

Dans le domaine du traitement de la parole, l'approche classique en ce qui concerne la classification est d'utiliser le couple MFCC/GMM. Pour détecter le niveau de spontanéité, on cherche à se focaliser sur des informations extra-linguistiques telles que : le débit, la prosodie, les pauses (silencieuses ou remplies), etc... Malencontreusement, les trames MFCC, obtenues lors de la paramétrisation du signal acoustique, contiennent une part importante d'information linguistique.

Afin de détecter plus facilement les informations extra-linguistiques, nous avons besoin d'une paramétrisation acoustique qui capture la dynamique à long terme du cepstre, ce qui ne peut être réalisé que par une analyse de fenêtres temporelles assez larges. Une des façons d'avoir un aperçu de la trame à plus long terme est d'utiliser les paramètres Shifted Delta Cepstra (SDC) qui ont été proposés initialement pour l'identification de langage (Kohler and Kennedy, 2002).

Le calcul des paramètres SDC est illustré dans la figure 6.1. Les paramètres SDC sont déterminés par 4 paramètres : N, d, P et k, où N est le nombre de coefficients cepstraux calculé à chaque trame, d représente le temps pour le calcul des deltas, k est le nombre de blocs dont les coefficients deltas sont concaténés pour former le paramètre final, et P est le décalage de temps entre les blocs consécutifs. Par conséquent, kN paramètres sont utilisés pour chaque paramètre SDC. Par exemple, le vecteur de la trame t est donné pour la concaténation de tous les $c(t+iP)$, où :

$$\Delta c(t) = c(t+iP+d) - c(t+iP-d) \tag{6.1}$$

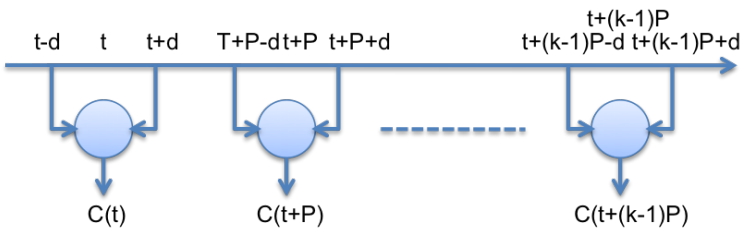

FIGURE 6.1 – *Calcul du Shift Delta Cepstrum.*

Nous comparons les MFCC et SDC-MFCC sur un classifieur GMM, puis nous utilisons la FA pour essayer de supprimer la variabilité inutile dans l'acoustique. Pour les trois systèmes, nous utilisons un modèle de mixtures composé de 256

gaussiennes, entrainé par maximum de vraisemblance avec l'algorithme EM. Le SDC est calculé avec les paramètres 11-1-3-5 (N-d-P-k).

Pour l'ensemble des expériences, nous utilisons une méthode de LeaveOneOut : 10 fichiers sont utilisés pour l'entraînement et 1 pour l'évaluation. Ce processus est répété jusqu'à ce que tous les fichiers soient évalués.

Les résultats sont reportés dans le Tableau 6.1 :

TABLE 6.1 – *F-Mesure, Rappel et Précision selon la classe de spontanéité par classification GMM dans le domaine cepstral. TCC représente le Taux Correct de Classification.*

	E1	E2	E3	TCC
MFCC	0.46 *(0.52/0.42)*	0.28 *(0.25/0.33)*	0.42 *(0.42/0.42)*	**0.39**
SDC-MFCC	0.47 *(0.50/0.44)*	0.28 *(0.24/0.33)*	0.48 *(0.52/0.44)*	**0.41**
FA-SDC-MFCC	0.62 *(0.68/0.56)*	0.43 *(0.41/0.47)*	0.62 *(0.59/0.65)*	**0.56**

Les résultats montrent que le SDC-MFCC dépasse légèrement les MFCC, le taux correct de classification passe de 39% à 41%. Nous notons que le SDC-MFCC semble particulièrement efficace sur les niveaux fortement spontanés (E3) : nous obtenons une F-Mesure de 42% et 48% respectivement pour les MFCC et SDC-MFCC. Nous constatons aussi que la FA permet d'améliorer significativement les résultats, puisqu'en utilisant la paramétrisation SDC-MFCC avec et sans FA, le taux correct de classification passe de 41% à 56% .

6.2.3.2 Les émotions

Les émotions dans la parole spontanée semblent jouer un rôle beaucoup plus important que dans la parole lue. Dans (Caelen-Haumont, 2002), les expériences semblent montrer que l'état émotionnel d'un locuteur est parfois beaucoup plus marqué en parole spontanée. En effet, en parole préparée, si le locuteur est en état de stress, il pourra toujours s'appuyer sur son texte préparé et son état émotionnel aura donc une influence assez faible sur les idées et les suites de mots prononcés. Or, dans un contexte spontané cet état émotionnel peut rendre complexe la construction et l'organisation des idées du locuteur : difficultés à parler, construction de phrases plus difficile et phrases moins compréhensibles (disfluences, hésitations, pauses, élisions...).

Étant donné qu'il existe un lien entre état émotionnel et niveau de spontanéité, l'idée est que le niveau de spontanéité peut être distingué de la même façon que la détection d'un état émotionnel. La plupart des approches pour détecter les émotions d'un locuteur sont basées sur les paramètres prosodiques (pitch, énergie et

débit d'élocution) et cepstraux (MFCC). Dans (Neiberg et al., 2006), l'auteur propose d'utiliser comme paramètres cepstraux le MFCC-Low. Les paramètres acoustiques sont calculés de la même façon que le MFCC mais les bancs de filtres sont placés entre 20 et 300 Hz, au lieu de 300 à 3400 Hz. L'auteur explique qu'en mettant les bancs de filtres plus bas, il arrive à mieux modéliser les variations F0 et donc à mieux capter les informations liées à l'émotion.

Tout comme les précédentes expériences, nous proposons d'utiliser la paramétrisation MFCC-Low avec les paramètres SDC, puis nous essayons d'utiliser la FA pour supprimer la variabilité inutile. Les expériences sont reportées dans le Tableau 6.2 :

TABLE 6.2 – *F-Mesure, Rappel et Précision selon la classe de spontanéité par classification GMM dans le domaine cepstral. TCC représente le taux correct de classification.*

	E1	E2	E3	TCC
MFCC-Low	0.49 *(0.52/0.46)*	0.21 *(0.17/0.30)*	0.48 *(0.50/0.46)*	**0.39**
SDC-MFCC-Low	0.49 *(0.55/0.44)*	0.22 *(0.18/0.31)*	0.52 *(0.57/0.47)*	**0.43**
FA-SDC-MFCC-Low	0.58 *(0.61/0.55)*	0.44 *(0.42/0.46)*	0.61 *(0.61/0.62)*	**0.54**

On constate que la paramétrisation SDC-MFCC-Low avec la FA permet d'obtenir de meilleurs résultats, avec 54% de taux correct de classification. Le résultat obtenu est certes moins bien que celui obtenu pour la paramétrisation SDC-MFCC avec la FA, mais nous espérons que cette nouvelle paramétrisation combinée avec les autres améliore les résultats.

6.2.3.3 Débit de la parole

Le débit de parole, défini comme la variation de la vitesse de production des sons par un locuteur, peut être une caractéristique de la parole spontanée. Des analyses menées sur de la parole lue ont permis de constater que le débit varie peu dans le cadre d'une parole préparée. Mais dans le cadre de la parole spontanée, ce débit aura tendance à varier au cours de l'énonciation. La raison essentielle est que les changements de débit (ainsi que les pauses) sont inévitables dans un niveau de spontanéité et qu'elles sont prononcées lorsque le processus de réflexion n'arrive pas à suivre le processus de production orale. Lorsque la vitesse de la parole devient plus rapide que la vitesse de préparation de son contenu, un locuteur varie son débit (ou utilise des pauses) jusqu'à ce que le prochain discours dont le contenu a été réfléchi soit prêt. Le changement de débit peut être un excellent moyen de catégoriser les différents niveaux de spontanéité.

Calculer les variations de débits dans un flux audio consiste, à partir d'une transcription, à calculer la vitesse d'articulation mesurée au sein des macro-unités de segmentation (par exemple les phonèmes). Dans (Jousse et al., 2008), des études ont été menées sur la durée des voyelles et l'allongement des syllabes à la fin d'un mot en utilisant des transcriptions. Cette méthode a donné des résultats concluants uniquement sur les transcriptions de référence.

Une autre façon de mesurer la variation de débit d'un locuteur est de mesurer sur l'ensemble d'un segment la régularité du débit, et ceci sans transcription. Nous proposons d'utiliser le noyau de Fisher qui permet de modéliser dans un vecteur les variations qu'il y a entre un modèle et les trames d'un segment.

Le noyau de Fisher a été utilisé dans le domaine de l'identification du locuteur par C. Longworth dans (Longworth and Gales, 2008). Il propose un contraste intéressant avec les autres approches puisqu'au lieu d'utiliser les paramètres d'un modèle acoustique (GMM), le noyau de Fisher utilise les probabilités de vraisemblance d'un modèle acoustique. Le noyau de Fisher se calcule ainsi :

$$\phi \bigtriangledown (O;\lambda) = \frac{1}{T} \begin{pmatrix} \bigtriangledown \log p(O;\lambda_1) \\ ... \\ ... \\ ... \\ \bigtriangledown \log p(O;\lambda_N) \end{pmatrix} \tag{6.2}$$

où λ_N correspond aux paramètres de la gaussienne N du GMM λ et O représente les trames d'un segment.

$$\bigtriangledown \log p(O;\lambda_N) = \sum_{t=1}^{T} \gamma(t)\Sigma^{-1}(o_t - \mu) \tag{6.3}$$

où $\gamma(t)$ est la probabilité *a posteriori* que la trame t appartienne à la gaussienne λ_N, μ et Σ^{-1} corresponde respectivement à la moyenne et à la matrice de covariance de la gaussienne λ_N.

En prenant comme modèle acoustique le GMM du niveau de spontanéité préparée, le vecteur obtenu par le noyau de Fisher permet d'obtenir un vecteur qui modélise les variations entre un modèle de parole préparée et chaque trame de notre segment. Les vecteurs obtenus avec le noyau de Fisher sont utilisés dans un classifieur SVM. Les résultats sont reportés dans le Tableau 6.3 :

Nous constatons qu'en utilisant la paramétrisation SDC-MFCC, on obtient de meilleurs résultats avec le noyau de Fisher.

TABLE 6.3 – *F-Mesure, Rappel et Précision selon la classe de spontanéité par classification GMM dans le domaine cepstral. TCC représente le taux correct de classification.*

	E1	E2	E3	TCC
MFCC-kd	0.56 *(0.60/0.53)*	0.35 *(0.30/0.42)*	0.63 *(0.68/0.58)*	**0.51**
SDC-MFCC-kd	0.57 *(0.60/0.54)*	0.36 *(0.31/0.42)*	0.63 *(0.68/0.59)*	**0.53**

6.2.4 Combinaison acoustique

Le but de la combinaison acoustique est d'exploiter l'information complémentaire apportée par différents paramètres acoustiques. Nous proposons ici de combiner les paramètres au niveau des scores avec l'idée d'estimer la probabilité *a posteriori* d'un niveau de spontanéité en combinant les scores fournis par différents jeux de paramètres. Cette combinaison est effectuée par une combinaison linéaire, le choix étant motivé par des expériences empiriques où la combinaison linéaire émerge comme le meilleur des classifieurs. La combinaison linéaire s'écrit :

$$s = \sum_{i=0}^{j} \lambda_i \cdot score_i \tag{6.4}$$

où $score_i$ est le score obtenu par le classifieur sur le paramètre acoustique i (les scores sont normalisés entre 0 et 1), et λ_i correspond au poids attribué au score i. Dans cette combinaison linéaire nous nous assurons que : $\sum_i \lambda_i = 1$. Les valeurs de λ sont calculées par une méthode de descente de gradient.

Dans le Tableau 6.4, nous rappelons les taux corrects de classification obtenus par les différentes paramétrisations acoustiques.

TABLE 6.4 – *Rappel des scores exprimés en taux correct de classification sur le niveau de spontanéité selon les paramètres acoustiques.*

MFCC	MFCC-Low	MFCC-kd
0.56	0.54	0.53

Nous pouvons observer dans le Tableau 6.4, que les performances des 3 paramétrisations acoustiques sont très proches : taux correct de classification d'environ 55%. Dans le Tableau 6.5, nous combinons les différentes paramétrisations acoustiques pour estimer leur complémentarité.

En étudiant les résultats obtenus dans le Tableau 6.5 et en combinant deux paramètres acoustiques pour n'importe quel ensemble de paramètres (MFCC, MFCC-

TABLE 6.5 – *F-Mesure, Rappel et Précision selon la classe de spontanéité par classification GMM dans le domaine cepstral. TCC représente le taux correct de classification.*

	E1	E2	E3	TCC
MFCC - MFCC-Low	0.62 *(0.68/0.57)*	0.44 *(0.41/0.46)*	0.65 *(0.63/0.68)*	**0.57**
MFCC-Low - MFCC-kd	0.61 *(0.66/0.57)*	0.42 *(0.39/0.47)*	0.64 *(0.65/0.63)*	**0.56**
MFCC - MFCC-kd	0.63 *(0.71/0.57)*	0.41 *(0.36/0.47)*	0.66 *(0.67/0.65)*	**0.57**
MFCC - MFCC-kd - MFCC-Low	0.65 *(0.72/0.60)*	0.45 *(0.41/0.50)*	0.68 *(0.68/0.68)*	**0.59**

Low et MFCC-kd), nous observons une amélioration des résultats d'environ 2 points en valeur absolue (55% à 57%). En combinant tous les paramètres acoustiques, nous observons une autre amélioration des résultats d'environ 2 points en valeur absolue (57% à 59%). La combinaison des paramètres acoustiques permet bien d'améliorer le système de classification du niveau de spontanéité.

6.2.5 Processus de décision globale

Les approches précédentes prennent seulement en considération les descripteurs qui ont été extraits depuis le segment sans prendre en compte les informations autour des segments voisins. Dans les travaux de (Dufour, 2010), afin d'améliorer les résultats, il est proposé de prendre en compte la nature des segments de parole contigus, ce qui implique que la catégorisation de chaque segment de parole a un impact sur la catégorisation des autres segments : le processus de décision devient un processus de décision globale.

Désignons s_i un tag du segment i et définissons $P(s_i|s_{i-1}, s_{i+1})$ comme la probabilité d'observation du segment i associé au tag s_i quand le segment précédent est associé au tag s_{i-1} et le segment suivant est associé au tag s_{i+1}. Désignons $c(s_i)$ la mesure de confiance donnée par notre modèle pour choisir le tag s_i pour le segment i. S est une séquence de tag s_i associée à la séquence de tous les segments de parole i (seulement un tag par segment). Le processus de décision globale consiste à choisir la séquence de tag $\hat{S}$ qui maximise le score global obtenu en combinant $c(si)$ et $P(s_i|s_{i-1}, s_{i+1})$ pour chaque segment de parole i détecté sur le fichier audio. La séquence $\hat{S}$ est calculée en utilisant la formule suivante :

$$\bar{S} = \underset{S}{\operatorname{argmax}}\, c(s_1) \times c(s_n) \times \prod_{i=2}^{n-1} \times c(s_i) \times P(s_i|s_{i-1}, s_{i+1}) \tag{6.5}$$

où n correspond au nombre de segments de parole automatiquement détectés dans le fichier d'enregistrement. L'auteur propose de résoudre ce problème au moyen de machines à états-finis.

Le tableau 6.6 montre les résultats avec et sans la prise de décision globale. Nous nous apercevons qu'avec cette méthode le taux correct de classification augmente, puisqu'il passe de 59% pour une décision locale à 62% pour une décision globale.

TABLE 6.6 – *F-Mesure, Rappel et Précision selon la classe de spontanéité par classification GMM dans le domaine cepstral. TCC représente le taux correct de classification.*

	E1	E2	E3	TCC
Local	0.65 *(0.72/0.60)*	0.45 *(0.41/0.50)*	0.68 *(0.68/0.68)*	**0.59**
Global	0.70 *(0.86/0.58)*	0.37 *(0.27/0.56)*	0.73 *(0.75/0.70)*	**0.62**

6.2.6 Conclusion

L'architecture que nous proposons permet de détecter le niveau de spontanéité d'un segment de parole selon 3 classes : parole préparée, faiblement spontanée et fortement spontanée (E1, E2 et E3). Contrairement aux précédentes approches dans le domaine, cette architecture permet de faire abstraction de toute transcription, puisqu'elle se focalise sur des paramètres acoustiques bas niveau. Nous avons proposé 3 différents jeux de paramètres tous centrés sur la détection d'une disfluence particulière.

Pour le zapping, structurer un document par niveaux de spontanéité nous semble l'un des descripteurs les plus importants. En effet, nous espérons pour la création de notre zapping que les sous-séquences vidéos comportant un fort taux de spontanéité soient d'un intérêt notable pour les utilisateurs.

Quatrième partie

Résumé vidéo par extraction

Chapitre 7

Résumé vidéo par extraction : le zapping

Sommaire

7.1 Introduction

Le résumé automatique est un des outils pouvant être utilisé pour obtenir des vues synthétiques d'une masse de documents disponibles ou pour accéder rapidement à des contenus relatifs à un sujet ou à une période donnée (Favre, 2007).

Ici, nous nous focalisons sur les méthodes de résumé automatique qui permettent de produire des vues synthétiques d'une collection de documents. D'une

façon générale, ces résumés doivent extraire l'essentiel de l'information contenue sous une forme aussi concise que possible.

Un premier problème de fond est la définition de ce qui est *essentiel* dans un document : s'agit-il de ce qui le caractérise globalement ou de ce qu'un utilisateur en retiendra (par exemple un moment émouvant, drôle, effrayant, ect.) ? La réponse à cette question déterminera ce que la vue synthétique doit montrer à l'utilisateur : soit une image "moyenne" quantitativement caractéristique de l'ensemble des contenus, soit une vue qui regroupe les moments présentants un intérêt particulier pour l'utilisateur et qui sont plutôt des moments atypiques par rapport à un document ou à une collection de documents.

Il est difficile de fournir une réponse générale à cette question. Notre approche consiste à chercher à reproduire le comportement subjectif d'un utilisateur en définissant 3 critères qui pourraient le conduire à identifier un moment d'intérêt :

- *saillance :* un moment saillant est un moment atypique, non-prévisible, en rupture avec l'ensemble du document.
- *expressivité :* qui caractérise la forme du document plutôt que le fond, par exemple l'émotion avec laquelle le discours est prononcé, le type de contexte, etc...
- *significativité :* un moment significatif caractérise le fond, c'est-à-dire le contenu sémantique du document.

Bien que les raisons qui guideraient le choix d'un utilisateur peuvent relever de plusieurs de ces critères, nous considérons ici qu'un moment d'intérêt correspond majoritairement à un seul d'entre eux.

Nous tentons donc de produire des résumés automatiques tels que la vision synthétique mais subjective de l'utilisateur les aurait produits. Ces résumés par extraction sont obtenus par regroupement des moments des vidéos présentant un intérêt particulier, selon l'un des 3 critères expliqués précédemment.

Pour arriver à créer ce résumé automatique vidéo, trois problèmes doivent être résolus :

- segmentation : les vidéos doivent être segmentées en parties compréhensibles hors de leur contexte et donc potentiellement intégrables à un résumé.
- évaluation de l'intérêt des segments : il faut être en mesure de choisir les segments qui composent le résumé et d'évaluer l'intérêt de chacun, afin de pouvoir évaluer l'intérêt global du résumé.
- sélection et agglomération des segments : une fois les segments extraits et leur intérêt évalué, il faut les sélectionner et les assembler dans le résumé vidéo.

L'application visée est la construction du résumé de l'actualité d'une journée, dans le cadre du résumé vidéo par extraction. Cependant, notre démarche est de proposer des solutions qui sont généralisables et qui pourront être appliquées à d'autres types de problèmes.

Nous présentons dans la section 7.2 l'architecture du système et les différentes

étapes nécessaires à la construction du zapping. Dans la section 7.3, nous présenterons le corpus et les mesures d'évaluations utilisées pour le zapping. Dans la section 7.4, nous proposons une méthode de segmentation de vidéo. Enfin dans la section 7.5, nous proposons un modèle pour sélectionner et assembler les moments d'intérêt des vidéos.

7.2 Architecture du système

La construction du résumé vidéo sous forme de zapping consiste à sélectionner les segments d'intérêt dans une collection de vidéos et à les agglomérer dans une vidéo-résumé. Un système idéal mettrait en compétition tous les segments de l'ensemble des vidéos. Pour chaque vidéo, le système sélectionnerait en fonction de leur intérêt les segments pertinents. Ainsi, à partir d'une collection de vidéos, le système extrairait les n meilleurs segments d'intérêt.

Ce système idéal est pour l'instant irréalisable pour deux raisons :
- détecter les segments d'intérêt dans de telles collections vidéos pose un problème de complexité.
- il n'y a pas de fonction d'intérêt universelle : l'intérêt est une notion intrinsèquement subjective, dépendante des individus, des moments, des contextes, etc...

Pour pallier ces deux problèmes, nous proposons une approche "pas à pas" dans laquelle nous affinons progressivement la sélection des vidéos en fonction des critères d'intérêts. Cette approche est théoriquement sous-optimale mais techniquement réaliste ; elle va nous permettre de réduire la complexité du problème et de mettre en place une stratégie pour l'évaluation efficace et évaluable.

Le processus de composition de vidéo-zapping enchaine les 5 étapes suivantes :

- Collection des vidéos : sélection par recherche des documents correspondant à une requête.
- Pré-sélection des vidéos d'intérêt : sélection des vidéos ayant un intérêt présumé pour l'utilisateur.
- Pré-segmentation : découpage des vidéos en segments.
- Point de vue : sélection pour chaque vidéo d'un mode (expressivité, significativité ou saillance) et de la fonction objective associée.
- Sélection et Agrégation : évaluation de l'intérêt des segments, recherche du groupe de segments qui maximise l'intérêt global du résumé (au sens de la fonction d'intérêt) et minimise sa redondance.

Ces 5 étapes sont décrites brièvement dans les 5 pararagraphes suivants.

La collecte des vidéos selon une requête sur des services communautaires est un réel problème. Les vidéos disponibles sur ces plateformes sont pour la plupart très mal indexées et très mal structurées, rendant la recherche difficile sans des outils automatiques efficaces. Actuellement, la recherche se base sur des métadonnées

laissées par l'utilisateur : titre de la vidéo, rubrique, commentaire etc... ce qui peut poser un problème car d'une part, la vision d'une information n'est pas la même d'un utilisateur à un autre, d'autre part les informations laissées par un utilisateur peuvent être partielles et/ou imprécises. Dans le cadre du résumé vidéo sous forme de zapping d'actualités, nous avons besoin de sélectionner les vidéos dont le genre est l'actualité. Nous proposons d'utiliser le système proposé dans le chapitre III. C'est une version légèrement modifiée que nous utilisons, puisqu'au lieu de détecter 7 genres nous proposons de détecter 2 classes : *actualité* et *non-actualité*.

A cette étape, les vidéos sélectionnées ne contiennent pas toutes un segment ayant un intérêt particulier. Il faut donc les filtrer et les sélectionner. Nous avons fait l'hypothèse que la popularité d'une vidéo peut être un critère de pré-sélection. Bien que cette hypothèse soit discutable, nous considérons qu'il y a une corrélation entre le nombre de visiteurs et l'intérêt d'une vidéo.

Le but d'un résumé vidéo est de sélectionner un sous-segment ayant un intérêt particulier. Nous proposons de segmenter la vidéo de manière à ce que chaque segment sélectionné hors contexte soit compréhensible. Nous détaillons cette étape plus en profondeur dans la section 7.4.

Un moment d'intérêt peut être un moment expressif, significatif ou saillant. Cette étape consiste à identifier le point de vue pertinent pour une vidéo donnée. Ceci est réalisé en détectant de manière automatique celui des trois critères d'intérêt qui est le plus pertinent pour la vidéo considérée. Cette détection est effectuée en utilisant un classifieur et des paramètres issus de la structure d'un document, par exemple la proximité de la vidéo à un fait d'actualité, son contenu linguistique, etc... qui peuvent nous donner une idée de la meilleure fonction d'intérêt à utiliser. Cette étape est décrite plus en détail dans la section 7.5.1.5.

La dernière étape consiste à sélectionner les segments pour composer un résumé d'intérêt maximal et de redondance minimale. Elle est basée sur un algorithme d'optimisation et une fonction objective intégrant ces deux critères et mettant en compétition l'ensemble des solutions respectant des contraintes fixées *a priori*. Cet algorithme est décrit dans la section 7.5.

7.3 Corpus et Evaluation

7.3.1 Corpus

Les vidéos présentes dans notre corpus sont issues du site Dailymotion [1] entre le 06/09/2010 et le 14/09/2010. Nous avons pris chaque jour les 15 meilleures d'entre elles (les vidéos les plus vues, selon l'indication de visite de Dailymotion), soit 120 vidéos. Toutes les vidéos ont une durée comprise entre 3 et 5 minutes.

1. http://www.dailymotion.com/fr

FIGURE 7.1 – *Plateforme permettant de sélectionner les moments d'intérêt des vidéos.*

Pour évaluer le moment d'intérêt des vidéos, nous avons mis en place un site Internet [2] (cf. Figure 7.1), pour demander à des personnes de visionner des vidéos et de sélectionner, selon elles, le meilleur moment d'intérêt de chacune.

Les utilisateurs devaient donc visionner une première fois une vidéo prise aléatoirement dans le corpus, puis sélectionner à l'aide d'une barre contenant deux curseurs le début et la fin du moment d'intérêt de la vidéo. L'utilisateur ne pouvait sélectionner qu'un seul moment d'intérêt par vidéo (soit une sous-séquence obtenue par vidéo). Une fois le moment d'intérêt sélectionné, l'utilisateur devait cliquer sur "Sauvegarder" afin d'enregistrer la sélection et passer à la vidéo suivante. Si aucun moment d'intérêt n'apparaissait dans la vidéo, l'utilisateur pouvait cliquer sur le bouton : "Aucun moment d'intérêt".

35 utilisateurs ont participé à l'évaluation. Chacun a évalué en moyenne un peu moins de 6 vidéos. Il y a eu 40 réponses pour lesquelles les utilisateurs n'ont trouvé aucun moment d'intérêt, parmi celles-ci, 22 ont été attribuées par 4 utilisateurs.

Le corpus (les annotations et les pointeurs des vidéos) est librement téléchargeable depuis cette adresse [3].

2. http://zapping.mickael-rouvier.fr/
3. http://zapping.mickael-rouvier.fr/corpus.zip

7.3.2 Evaluation

Les performances du système sont évaluées en termes de taux de rappel :

$$R = \frac{\text{Nombre de trames de la sous-séquence d'intérêt correctement détectées}}{\text{Nombre de trames de la sous-séquence d'intérêt}} \tag{7.1}$$

La sous-séquence d'intérêt correctement détectée correspond à l'intersection entre la sous-séquence d'intérêt issue de la référence et celle obtenue par le système. Cette mesure permet de connaître le ratio entre le nombre de trames correctement trouvées et celui de la référence. Une trame est extraite toutes les 0.01 secondes.

7.4 Segmentation audio et vidéo

La segmentation d'un document est la première des étapes de la construction du résumé vidéo. Elle consiste à identifier les segments de tailles minimales qui sont intelligibles hors contexte. En effet, ces segments sont destinés à être extraits et visionnés hors de leur contexte intial.

Cette taille minimale peut être différente suivant la tâche et le document : par exemple, dans un document audio la taille minimale peut correspondre à un tour de locuteur, une phrase, un sujet abordé, etc... La segmentation demande une attention particulière puisqu'une mauvaise segmentation peut dégrader fortement la qualité du résumé automatique produit.

Traditionnellement dans la vidéo, la segmentation se fait par scène (Sundaram and Chang, 2000). Les techniques de segmentation en scènes sont basées sur une combinaison des segmentations issues de l'audio et de l'image (Sundaram and Chang, 2000; Chen et al., 2002). Cette combinaison permet d'une part, de rattraper les erreurs faites par les segmentations automatiques qui seraient basées sur une seule modalité, d'autre part de regrouper plusieurs segments de locuteur ou de plans dans une même scène. C'est pourquoi l'intégration des informations audio et image pour segmenter les vidéos permet d'améliorer la segmentation en scènes.

La scène est en général un moment pour lequel il y a une unité de contexte qui peut être relativement longue, mais de laquelle pourraient être extraits des segments intelligibles. La scène n'est donc pas un segment minimal tel que nous l'avons défini. Ces segments peuvent dépendre du type de document, par exemple si la sous-séquence qui nous intéresse est l'intervention d'une personne, alors le segment minimal correspond à un tour de locuteur ; dans d'autres contextes, il peut correspondre à un plan particulier, à une expression ou une phrase prononcée... Afin de tenir compte de cette diversité des unités minimales, nous proposons

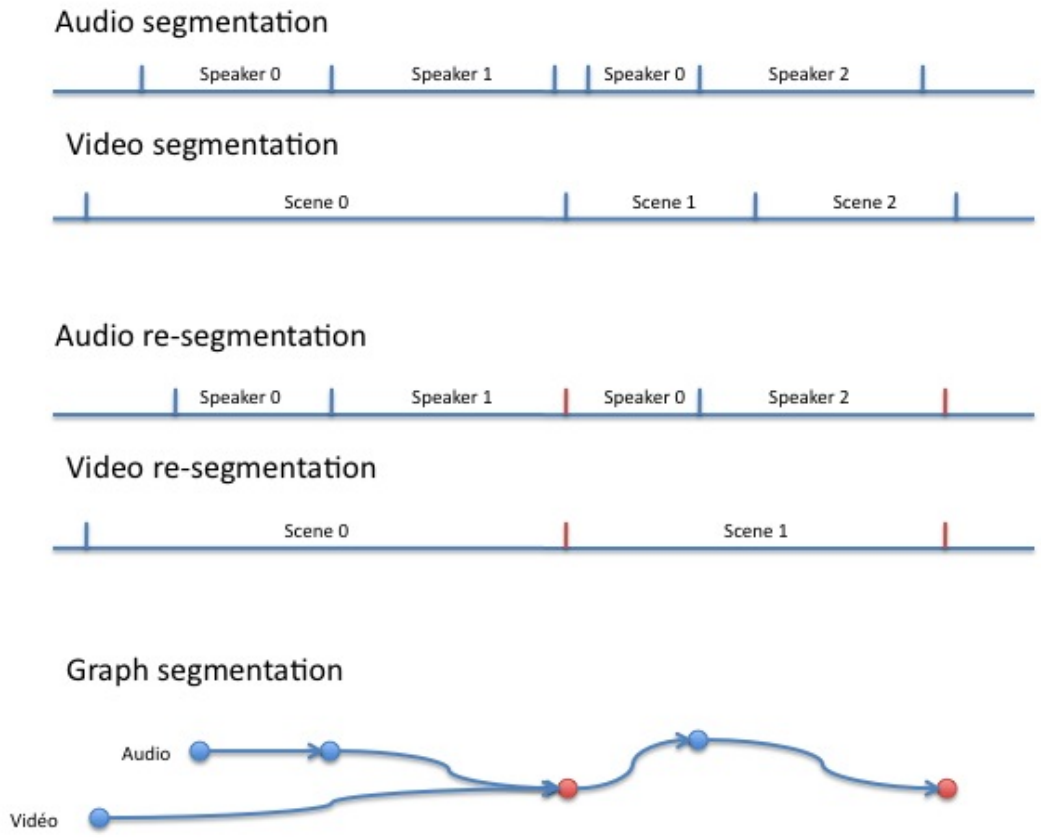

FIGURE 7.2 – *Processus de création du graphe de segmentation.*

de créer un graphe issu des segmentations audios et vidéos. L'algorithme de recherche de la meilleure sous-séquence vidéo parcourra ce graphe et y cherchera la meilleure sous-séquence qui pourra être composée de plusieurs segments atomiques contigüs.

Ici, la segmentation audio est une segmentation en locuteur. L'algorithme utlisé repose sur une modélisation GMM, une classification ascendante et un critère d'arrêt BIC (Meignier and Merlin, 2010). La segmentation vidéo est une segmentation en plans fournie par Eurecom.

Le processus de création de graphe de segmentation, illustré par la Figure 7.2, se déroule en 3 étapes :

1. Segmentation de la vidéo en locuteurs et en plans.
2. Détection des points d'accroche : lorsqu'un segment de locuteur et un segment de plan se terminent *presque* en même temps (espacement inférieur à 0.5 s), nous les regroupons. Un même point marque alors la fin du segment de locuteur et du segment de plan : le point d'accroche.

3. Fusion de deux segments de plan s'ils coupent un segment de locuteur

Le graphe ainsi construit repésente l'ensemble des hypothèses de segmentations possibles pour une vidéo.

7.5 Sélection des sous-séquences par programmation linéaire en nombres entiers

L'algorithme permettant de composer le zapping devra maximiser l'intérêt global (sélectionner pour chaque vidéo le groupement de segments contigüs ayant un intérêt maximal pour l'utilisateur) et minimiser la redondance (sélectionner les segments qui ont le moins de redondances entre eux).

L'algorithme MMR utilisé notamment dans le résumé automatique texte, essaie de résoudre ces problèmes de manière itérative. Malheureusement, le MMR est un algorithme glouton : il prend à chaque itération une décision localement optimale et ne remet jamais en cause les segments précédemment sélectionnés. De plus, le moment d'intérêt dans une vidéo peut correspondre à l'agglomération de plusieurs segments contigüs. Individuellement, les segments n'ont aucun intérêt, mais ensemble ils forment un moment d'intérêt. Ceci pose un problème car les algorithmes gloutons auront du mal à détecter ces moments résultant d'une agglomération. Nous avons besoin d'un algorithme qui prenne une décision de manière plus globale sur l'ensemble des segments constituant nos vidéos. Dans (Gillick and Favre, 2009), les auteurs proposent de modéliser le problème de résumé automatique texte comme un problème de programmation linéaire en nombres entiers. La résolution du problème se fait sur un large éventail de solutions possibles. Nous proposons d'utiliser cet algorithme pour composer le zapping.

La composition du résumé par extraction se fait en maximisant l'intérêt global des segments et en minimisant la redondance des segments inter-vidéo. Dans la suite de nos travaux, nous avons séparé le problème en deux parties. La première consistera à trouver pour chaque vidéo le segment d'intérêt maximal. La seconde consistera à trouver l'ensemble des segments de la collection de vidéos qui est d'intérêt maximal et de redondance minimale. Nous commençons par présenter le problème d'extraction du meilleur segment pour une vidéo. Nous étendons ensuite notre problème au multi-document.

7.5.1 Sélection de la sous-séquence d'intérêt dans une vidéo

Sélectionner une sous-séquence d'intérêt dans une vidéo consiste à rechercher dans le graphe de segments la sous-séquence maximisant une fonction objective (fonction d'intérêt). Un moment d'intérêt peut être défini comme une sous-séquence qui peut être significative, expressive ou saillante.

Dans la section 7.5.1.1, nous proposons dans un premier temps de modéliser dans notre problème de PLNE, les contraintes liées au zapping. Puis dans les sections 7.5.1.2, 7.5.1.3, 7.5.1.4, nous verrons les différentes fonctions objectives proposées pour résoudre notre problème. Dans la section 7.5.1.5, nous proposons une méthode pour choisir sur chaque vidéo l'une des fonctions objectives.

7.5.1.1 Algorithme d'optimisation

L'objectif de la méthode de programmation linéaire en nombres entiers est de minimiser (ou maximiser) une fonction linéaire de n variables entières non négatives sur un ensemble d'inégalités linéaires. Une fois le problème posé, l'algorithme de résolution recherche parmi un ensemble de solutions respectant les contraintes celle qui minimise (ou maximise) la fonction linéaire.

Notre problème de création de zapping revient à rechercher une sous-séquence vidéo qui maximise une fonction objective (fonction d'intérêt) soumise à une série de contraintes. Dans le résumé vidéo par extraction, les deux contraintes sont :

- les segments sélectionnés doivent être contigüs
- la sous-séquence extraite ne doit pas excéder un certain temps

Pour simplifier la présentation de notre modèle de sélection de sous-séquences d'intérêt dans une vidéo, nous nous plaçons dans le cadre d'un graphe linéaire. Le modèle peut alors s'écrire ainsi :

$$
\begin{array}{lll}
\text{Maximize} & f^{obj} & \\
\text{Subject To} & \sum_x l_x n_x \leq \delta & (1) \\
& n_x - n_{x+1} - o_x <= 0 & \forall x (2) \\
& \sum_j o_x <= 1 & (3) \\
& n_x \in \{0,1\} & \forall x \\
& o_x \in \{0,1\} & \forall x
\end{array}
$$

où f^{obj} est notre fonction objective, s_x dénote la présence du segment x dans la sous-séquence vidéo, l_x est une constante qui permet d'exprimer la durée du segment x. L'équation 1 limite l'extraction d'une sous-séquence vidéo à δ secondes.

Les équations 2 et 3 permettent de sélectionner des segments contigüs. Dans l'équation 2, o_x est un critère d'arrêt pour la sélection des segments contigüs. Dans le cas où n_x est sélectionnée ($n_x = 1$), soit le segment suivant est sélectionné (et donc $n_{x+1} = 1$), soit le critère d'arrêt est sélectionné (et donc $o_x = 1$). Il y a autant de critères d'arrêt que de segments. Nous nous assurons dans l'équation 3 du nombre de sous-segments à sélectionner dans la vidéo, en faisant la somme des critères d'arrêt.

Dans le cas d'un graphe de segmentation non-linéaire, le modèle peut s'écrire ainsi :

$$
\begin{array}{lll}
\text{Maximize} & f^{obj} & \\
\text{Subject To} & \sum_x \sum_j l_x n_{x,j} \leq \delta & \forall x \\
& (\sum_j n_{x,j}) - (\sum_j n_{x+1,j}) - o_x <= 0 & \\
& \sum_j n_{x,j} = 1 & \forall x \\
& \sum_x o_x <= 1 & \\
& n_{x,j} \in \{0,1\} & \forall x \forall j \\
& o_x \in \{0,1\} & \forall x
\end{array}
$$

où $n_{x,j}$ correspond au segment x issu de la segmentation j (audio ou vidéo). Dans le cas où nous nous trouvons sur un point d'accroche alors $n_{x,audio} = n_{x,video}$.

7.5.1.2 Fonction d'intérêt : significativité

Dans un zapping, un moment intéressant peut être une sous-séquence qui est significative des contenus sémantiques parlés de la vidéo ; par exemple, une séquence courte évoquant un fait d'actualité qui est le sujet principal de la vidéo.

L'extraction d'une telle sous-séquence peut reposer sur les méthodes développées pour le résumé automatique de texte. Le principe général de ce type d'approches est que le résumé doit synthétiser le contenu sémantique des documents sources. Ici, nous suivons l'approche proposée initialement par (Gillick and Favre, 2009) qui cherche à extraire les concepts dominants pour les intégrer de façon aussi concise que possible dans le résumé.

Nous appelons "concept" des éléments d'information comme par exemple : une décision prise lors d'une réunion, l'opinion d'un participant sur un sujet, etc... Mais le niveau d'abstraction de tels concepts rend difficile une extraction automatique. Nous avons choisi de représenter ces concepts par des unités linguistiques simples : les n-grammes. Cependant, ces n-grammes se recoupent souvent avec des marqueurs de discours ("en fait", "vous savez") lesquels peuvent rajouter du bruit.

L'extraction de concepts ainsi représentés est un problème d'extraction de mots clefs. Pour extraire ces concepts, nous proposons donc d'utiliser une version modifiée de l'algorithme d'extraction de mots-clefs proposé initialement dans (Xie et al., 2009a).

Les modifications portent sur la pondération des concepts. L'algorithme procède en 6 étapes :

1. Extraction de tous les n-grammes pour $n = 1, 2, 3$
2. Suppression du bruit 1 : suppression des n-grammes qui apparaissent seulement une fois
3. Suppression du bruit 2 : suppression des n-grammes si un des mots du n-gramme a un IDF plus bas qu'un seuil donné
4. Suppression du bruit 3 : suppression des n-grammes qui sont contenus dans d'autres n-grammes et qui ont la même fréquence (par exemple supprimer le n-gramme "chat noir" si la fréquence est la même que le n-gramme "petit chat noir").
5. Réévaluation des poids des bi-grammes et tri-grammes : $w_i = n \cdot idf(g_i)$ ou w_i est le poids du n-gramme, n la taille du n-gramme et idf le poids IDF [4] du mot.
6. Réévaluation des poids des n-grammes : $w_i = \frac{\exp(pw_i)}{\sum_i \exp(pw_i)}$, où w_i est le poids final des n-grammes et p une constante fixée à 7 dans nos expériences.

La modification de l'algorithme porte uniquement sur l'étape 6 du processus. Elle est motivée par le fait que les poids attribués aux différents concepts sont très proches les uns des autres dans la version intiale. Il y a une discrimination faible entre les différents concepts. L'étape 6 permet de réévaluer la pondération avec une fonction exponentielle qui accentue la séparation des concepts entre eux par renforcement ou diminution de leur poids.

Par ailleurs, l'intérêt d'un contenu sémantique (et des concepts associés) peut dépendre de l'actualité. Par exemple, lors de l'affaire Nafitassalou Diallou les concepts comme "chambre 2806", "Sofitel", "DSK" revenaient assez souvent dans l'actualité. Ainsi, l'actualité peut nous donner une information sur la pertinence des phrases à inclure dans un résumé. Nous proposons dans notre modèle de pondérer les phrases en fonction de leur proximité à l'actualité telle qu'elle est diffusée sur le Web. Le poids d'une phrase est calculé via la similarité cosinus entre la phrase et la dépêche d'actualité présente sur Internet qui lui est la plus proche. Bien entendu, cette dernière méthode n'est pas universelle ; elle n'est *a priori* pertinente que dans le contexte de résumés d'actualités.

Ainsi, l'algorithme peut s'écrire comme ceci :

$$
\begin{aligned}
\text{Maximize} \quad & F^{obj} = (1-\lambda)(\sum_x w_x c_x) + \lambda(\sum_x \sum_j web_{x,j} n_{x,j}) & \\
\text{Subject To} \quad & n_x Occ_{ix} \leq c_i, & \forall i, x(1) \\
& \sum_x n_x Occ_{ix} \geq c_i, & \forall i, x(2) \\
& c_i \in \{0, 1\} & \forall i
\end{aligned}
$$

4. L'IDF a été calculé sur le corpus Wikipedia

où c_x dénote la présence du concept x dans le résumé, w_x est le poids associé au concept x, $web_{x,j}$ est le poids associé à la séquence x, j. Le paramètre λ est utilisé pour équilibrer les scores attribués aux phrases et ceux des concepts. Nous rajoutons dans notre modèle de détection de sous-séquence d'intérêt deux nouvelles contraintes : si une phrase est sélectionnée, tous les concepts contenus dans cette phrase sont aussi sélectionnés (1) et si un concept est sélectionné, au moins une phrase qui contient ce concept est sélectionnée également (2).

TABLE 7.1 – *Les résultats obtenus en utilisant le critère de significativité.*

	Concept (old)	Concept (new)	Web	Significativité (Concept/Web)
Résultat	0.27	0.42	0.38	0.44

Les résultats obtenus avec ce modèle sont reportés dans le Tableau 7.1. Ils correspondent à l'intersection entre la séquence d'intérêt détectée automatiquement et la séquence d'intérêt de référence (cf équation 7.1). Les champs *Concept (old)* et *Concept (new)* font référence respectivement à l'ancienne façon de calculer les poids sur les concepts et celle que nous proposons. Nous constatons que notre nouvelle méthode permet dans le cadre du zapping d'améliorer nettement les résultats (elle passe de 27% à 42% de détection de séquence d'intérêt). En utilisant seulement les concepts pour détecter les sous-séquences d'intérêt, 42% des sous-séquences d'intérêt de notre corpus ont été correctement détectées. En combinant les concepts et le poids des phrases par rapport au web, notre taux de détection est amélioré puisqu'il passe de 42% à 44% des sous-séquences d'intérêt détectées.

Dans le tableau 7.2, nous proposons de comparer l'algorithme MMR (typiquement utilisé pour le résumé automatique) et l'algorithme de PLNE. Pour cela, nous avons du réaliser une version modifiée de l'algorithme MMR, puisque, dans le cadre du zapping, les sous-segments extraits doivent être contigüs à ceux précédemment sélectionnés (c'est-à-dire à ceux présents dans l'historique).

TABLE 7.2 – *Les résultats obtenus en utilisant les algorithmes mmr et plne.*

	MMR	PLNE
Résultat	0.37	0.42

Les résultats obtenus montrent que l'algorithme PLNE obtient de meilleurs résultats que l'algorithme MMR. Nous pensons qu'étant donné la nature gloutonne de l'algorithme MMR et la contigüité voulue des sous-segments sélectionnés, le choix de la première phrase va en grande partie guider la sélection du reste du résumé. Pour des algorithmes gloutons, tels que le MMR, le choix de la première phrase est primordial ; à l'inverse, l'algorithme PLNE remet en cause tout au long du processus les sous-segments sélectionnés.

7.5.1.3 Fonction d'intérêt : Expressivité

La forme d'un discours peut être une caractéristique d'un segment d'intérêt. Nous proposons de détecter l'expressivité par la charge émotive et le niveau de spontanéité.

Dans (Liscombe et al., 2005), l'auteur explique que les mots que les personnes emploient jouent un rôle important sur l'état émotionnel dans lequel la personne se trouve. L'auteur Baudouin Labrique propose sur son site Internet une liste de mots corrélés avec la charge émotive. Nous proposons de créer un vecteur contenant l'ensemble de ces mots, puis d'utiliser une mesure de similarité (le cosinus) entre une phrase issue de la transcription automatique de la vidéo et le vecteur contenant l'ensemble des mots expressifs :

$$expressivite(D_1, D_2) = \frac{\sum_i t_{1i} \sum_i t_{2i}}{\sqrt{\sum_i t_{1i}^2} \sqrt{\sum_i t_{2i}^2}} \tag{7.2}$$

où t_i est le poids TF-IDF d'un mot. Le vecteur D_1 correspond aux mots présents dans la transcription, le vecteur D_2 correspond à l'ensemble des mots chargés émotionnellement d'après le dictionnaire de référence. Le score obtenu permet de savoir à quel point une phrase est chargée émotionnellement.

Le niveau de spontanéité est calculé selon la méthode proposée dans le chapitre 6. Nous proposons d'attribuer à chaque segment un poids selon son niveau de spontanéité détecté.

Ainsi le modèle de détection de moment d'intérêt basé sur l'émotion et la spontanéité va chercher un sous-segment dans lequel les phrases sont chargées émotionnellement et fortement spontanées :

$$\text{Maximize} \quad F^{obj} = (1-\lambda)(\sum_x \sum_j expressivite_{x,j} n_{x,j}) + \lambda(\sum_x \sum_j spont_{x,j} n_{x,j})$$

où $emotion_{x,j}$ correspond au poids lié à l'émotion du segment x, j et $spont_{x,j}$ correspond au poids lié à la spontanéité du segment x, j.

TABLE 7.3 – *Les résultats obtenus en utilisant le critère d'expressivité.*

	Expressivité	Spontanéité	Expressivité/Spontanéité
Résultat	0.22	0.14	0.34

Le Tableau 7.3 présente les résultats obtenus sur la détection des moments d'intérêt en utilisant la charge émotive et la spontanéité. Globalement, les résultats montrent que la fonction objective utilisée obtient de moins bons résultats que les

autres fonctions objectives : nous n'obtenons que 34% de bonne détection de moments d'intérêt. Nous espérons par la suite que cette fonction objective soit complémentaire des autres.

7.5.1.4 Fonction d'intérêt : saillance

Une sous-séquence peut être saillante parce qu'il y a eu dans la vidéo (que ce soit à l'image ou dans le son) quelque chose d'atypique, d'inattendu... Une séquence est atypique car il y a eu l'apparition d'une nouvelle information non prévisible. Cette nouvelle information est en rupture avec le document.

Pour détecter une sous-séquence atypique, la première étape consiste à modéliser les informations présentes dans le signal audio et vidéo. Une fois celles-ci modélisées, la deuxième étape consiste à rechercher la séquence qui est en rupture avec la vidéo.

Dans le domaine de la recherche et de la classification d'images, le contenu d'une trame peut être représenté par un modèle de sac de mots (Sivic and Zisserman, 2003). Le modèle de sac de mots consiste à décrire une image au moyen d'un histogramme des occurrences d'un certain nombre de motifs de référence prédéfinis.

Pour construire le modèle de sac de mots, nous détectons d'abord dans l'image les points d'intérêt LIP. Ceux-ci sont extraits par une différence de gaussiennes et un Laplacien de gausiennes. Ensuite nous calculons les descripteurs SIFT pour chaque région d'intérêt, puis tous les descripteurs SIFT de la vidéo sont alors regroupés en 500 classes à l'aide de l'algorithme des *k-moyennes*. Ainsi, les paramètres du sac de mots d'une trame est l'histogramme du nombre de mots visuels (classe) qui apparait dans la trame.

Notre but est de calculer un score de rupture vidéo. La mesure de similarité permet de savoir à quel point deux vecteurs sont proches, c'est-à-dire, contiennent une information similaire. Ce qui nous intéresse c'est de savoir à quel point une sous-séquence est en rupture par rapport à l'ensemble du document. Nous proposons de calculer le score de rupture ainsi :

$$video(D_1, D_2) = 1 - \frac{\sum_i w_{1i} \sum_i w_{2i}}{\sqrt{\sum_i w_{1i}^2} \sqrt{\sum_i w_{2i}^2}} \tag{7.3}$$

où D_1 correspond à l'histogramme du nombre de mots visuels du sous-segment et D_2 à l'histogramme du nombre de mots visuels de la vidéo.

Nous proposons de modéliser les MFCC d'un segment via un GMM. Ce dernier permet de modéliser l'information présente dans un segment, puis de calculer la vraisemblance de ce GMM sur l'ensemble du signal audio de notre vidéo. Le score de vraisemblance obtenu permet de savoir à quel point un segment est en

rupture avec le reste du document. Chaque segment est pondéré avec ce score de vraisemblance de façon à déterminer lequel est le plus dissemblable du document.

Le but du modèle est de chercher les sous-segments contigüs qui sont le plus en rupture avec la vidéo.

$$\text{Maximize} \quad F^{obj} = (1-\lambda)(\sum_x \sum_j image_{x,j} n_{x,j}) + \lambda(\sum_x \sum_j son_{x,j} n_{x,j})$$

où $image_{x,j}$ et $son_{x,j}$ correspondent respectivement aux scores de rupture vidéo et de rupture audio de chaque segment x, j.

TABLE 7.4 – *Les résultats obtenus en utilisant le critère de saillance.*

	Vidéo	Audio	Saillance (Vidéo/Audio)
Résultat	0.34	0.28	0.38

Le Tableau 7.4 présente les résultats obtenus sur la détection des moments saillant. En utilisant seulement les informations audio et vidéo, les fonctions permettent de détecter respectivement 34% et 28% des sous-séquences d'intérêt. La combinaison de ces deux informations dans la même fonction objective permet d'obtenir 38% de bonne détection de moments d'intérêt. Nous espérons par la suite que cette fonction objective soit complémentaire des autres.

7.5.1.5 Classification

Fort de l'idée de définir différentes fonctions objectives pour détecter des moments d'intérêt dans la vidéo, nous devons au préalable choisir la fonction objective qui sera utilisée pour chaque vidéo. Les paramètres sur la structure du document nous paraissent de bons indicateurs pour faire ce choix. Par exemple, si la vidéo parle d'un fait d'actualité, il y a de fortes chances que le contenu de cette vidéo soit expressive. Nous proposons d'extraire 8 paramètres sur la structure du document :

1. le temps de parole de chaque locuteur sur la durée totale de la vidéo
2. le nombre de locuteurs
3. le nombre de plans de la vidéo
4. le temps de parole des locuteurs ayant un niveau de spontanéité élevé sur la durée totale de la vidéo
5. la similarité entre le document et l'actualité sur le web (nous utilisons comme mesure de similarité : le cosinus)
6. l'énergie audio d'une vidéo (moyenne, maximum, minimum)

Ces paramètres sont utilisés dans un classifieur de type SVM. Nous définissons 3 classes, chacune étant attribuée à une fonction objective.

Pour attribuer une classe à chaque vidéo, nous faisons tourner dans un premier temps les 3 systèmes qui extraient le segment d'intérêt pour chacun des critères. Nous calculons pour chaque vidéo le taux correspondant à l'intersection entre la sous-séquence d'intérêt issue de la référence et celle obtenue par le système (Equation 7.1). Enfin, nous attribuons à chaque vidéo la classe qui appartient au système qui maximise ce taux.

Etant donné le manque de données d'apprentissage, les modèles SVM sont entraînés par une stratégie de *leave-one-out*. Le corpus d'apprentissage a été découpé en 8 parties (correspondant aux 8 jours du corpus) : 7 parties sont utilisées pour entraîner les différents modèles et une pour l'évaluation.

TABLE 7.5 – *Les performances en utilisant un classifieur SVM.*

	Significativité	Expressivité	Saillance	Classif	Oracle
Résultat	0.44	0.34	0.38	0.51	0.68

En utilisant les paramètres sur la structure du document, le classifieur arrive à attribuer à 68% des documents la bonne fonction objective, ce qui permet d'obtenir 51% de détection des séquences d'intérêt. Cette méthode est loin d'obtenir les performances de l'Oracle (68% de détection des séquences d'intérêt) : celui-ci n'utilise pas de classifieur, mais pour l'ensemble des vidéos il regarde et compare les résultats de chacun des trois fonctions objectives et sélectionne celle qui obtient le meilleur résultat par rapport à la référence. Les performances obtenues via l'oracle tendent à montrer que ces fonctions objectives sont complémentaires.

7.5.2 Sélection des sous-séquences d'intérêt d'une collection de vidéos

Cette dernière étape d'agrégation des différents contenus vidéos consiste à concaténer les différents segments vidéos, de façon à ce qu'il y ait le moins possible de redondances d'informations. En effet, toujours dans le but de créer un zapping de la journée, si plusieurs vidéos traitent de la même information, il serait souhaitable d'avoir des informations complémentaires et non redondantes. Dans notre étude, la minimisation de la redondance concerne uniquement la redondance linguistique.

Pour modéliser la redondance linguistique, nous proposons d'extraire les concepts et de pénaliser la fonction objective si le concept est partagé par plusieurs phrases. Ainsi, notre modèle peut s'écrire ainsi :

$$
\begin{aligned}
\text{Maximize} \quad & F^{obj} = (1-\lambda)\sum_i f_i^{obj} - \lambda \sum_u w_u c_u & \\
\text{Subject To} \quad & \sum_x \sum_j \sum_i l_{x,i} n_{x,j,i} \leq \delta & \forall x \\
& (\sum_j n_{x,j,i}) - (\sum_j n_{x+1,j,i}) - o_{x,i} <= 0 & \\
& \sum_j n_{x,j,i} = 1 & \forall x \forall i \\
& \sum_x \sum_i o_{x,i} <= 1 & \\
& n_{x,j,i} \in \{0,1\} & \forall x \forall j \forall i \\
& o_{x,i} \in \{0,1\} & \forall x \forall i \\
& c_{x,i} \in \{0,1\} & \forall x \forall i
\end{aligned}
$$

où i correspond à la i^{ime} vidéo, $\sum_i f_i^{obj}$ est la somme de toutes les fonctions objectives propres à nos vidéos.

L'évaluation des résumés issus de cette dernière étape devrait reposer sur la comparaison des vidéos zappings obtenues avec un (ou plusieurs) zapping de référence. La constitution d'une telle base de test serait extrêmement coûteuse : la diversité des zappings produits manuellement obligerait à faire composer un grand nombre de références par divers "compositeurs", ou à demander à un grand nombre d'utilisateurs d'apprécier la qualité du résultat obtenu. Une telle évaluation n'était malheureusement pas à notre portée dans le cadre de ce travail et cette étape ultime n'a pas pu être évaluée.

7.6 Conclusion

Dans ce travail, nous avons présenté un modèle permettant de faire du résumé automatique de vidéo par extraction pour une collection de documents. L'objectif était de développer des méthodes permettant de produire des vues à la fois représentatives et concises d'une collection de documents.

Dans un premier temps, nous avons essayé de définir ce qui est *essentiel* dans un document. Nous avons proposé de modéliser cet "essentiel" selon trois modes : la significativité, la saillance et l'expressivité. Pour chacun d'eux, nous avons proposé des descripteurs et des méthodes qui permettent de les extaire.

Puis dans un deuxième temps, nous avons proposé une alternative à l'algorithme MMR généralement utilisé en résumé de texte ; nous proposons un algorithme de programmation linéaire en nombres entiers dérivé de celui évalué dans (Gillick and Favre, 2009). Cette formalisation du problème nous permet d'obte-

nir des résumés globalement de meilleure qualité en termes de compromis intérêt/concision.

L'application visée était la construction des résumés vidéos par extraction, avec une perspective applicative qui serait la composition d'une vidéo résumant l'actualité d'une journée. Cependant, les questions et les solutions que nous avons apportées tout au long de ce travail sont généralisables et peuvent être appliquées à d'autres types de problèmes.

Chapitre 8

Conclusion et perspectives

Sommaire

8.0.1 Conclusion

Ces dernières années, nous avons pu constater que le nombre de données multimédias a considérablement augmenté. Internet est l'un des principaux acteurs de cette montée en puissance, notamment avec la nouvelle vague du Web 2.0 qui permet aux utilisateurs les plus néophytes de partager des documents hétérogènes. Aujourd'hui, un internaute ne peut pas visionner l'ensemble des documents disponibles sur la toile et il lui est souvent difficile de trouver ceux qui pourraient l'intéresser.

Le résumé automatique est un des outils permettant d'avoir une vue synthétique d'une grande masse de documents. La réalisation de tels résumés ne peut se faire sans une caractérisation riche des contenus des collections multimédias.

La seconde partie de ce document a été consacrée à l'extraction des contenus audios. Nous avons proposé dans le chapitre 3 un système rapide pour la détection de termes dans de grandes bases multimédias. Ce système peut permettre de valider une hypothèse sur le thème de la vidéo plutôt que d'extraire en "aveugle" l'ensemble des contenus parlés. La conclusion principale de cette partie du travail est que cette stratégie de validation orientée par la requête est significativement plus robuste que la stratégie classique qui consiste à faire de la transcription automatique sans *a priori* sur la nature de ce qui est cherché. Dans le contexte du traitement de donnés issues du Web, cet avantage pourrait devenir critique. D'une façon plus générale, l'application de systèmes de reconnaissance de la parole à des vidéos issues du Web souffre du manque de robustesse des systèmes. Nous

avons proposé dans le chapitre 4 une normalisation des données acoustiques issue de l'analyse factorielle. Contrairement aux autres approches qui utilisent le paradigme FA uniquement en modélisant l'information utile, nous avons ici proposé de modéliser l'information inutile pour pouvoir la supprimer des trames acoustiques. Nos résultats montrent que l'analyse factorielle peut être utilisée efficacement comme une méthode supervisée de filtrage dans le domaine cepstral.

Nous avons proposé dans la troisième partie des méthodes pour structurer de grandes collections multimédias. Le chapitre 5 permet de classer les vidéos selon leur genre (actualité, cartoon, publicité....). Nos contributions ont porté sur deux domaines : la catégorisation dans le domaine cepstral utilisé avec une réduction de variabilité par FA et l'extraction de descripteurs audios de haut niveau. Une version modifiée de ce système a été utilisée pour participer à la campagne d'évaluation MediaEval et a obtenu les meilleurs résultats. La conclusion majeure de cette partie est que le canal audio porte une information caractéristique du genre de la vidéo, alors que les précédentes études suggéraient que l'essentiel de cette information était contenu dans l'image, qu'elle soit fixe ou animée. Dans le chapitre 6, nous avons proposé de caractériser le niveau de spontanéité d'un discours. Notre approche a été de considérer que l'information caractéristique de la spontanéité était concentrée dans des zones de l'espace acoustique mal représentées par les paramètres couramment utilisés en reconnaissance de la parole. Nous avons évalué différents paramètres utilisés pour la reconnaissance des émotions, du locuteur, de la langue. Cette aproche a permis d'améliorer très sensiblement les résultats obtenus auparavant pour la détection la parole spontanée par analyse de l'acoustique.

La quatrième partie a proposé de créer un résumé automatique vidéo par extraction. Nos recherches ont consisté à définir ce qui est *essentiel* dans un document. Nous avons proposé dans ce modèle de reproduire le comportement subjectif d'un utilisateur en définissant 3 types de critères (la saillance, l'expressivité et la significativité), puis de modéliser notre problème de résumé vidéo par extraction en programmation linéaire en nombres entiers. Cette étude montre que l'intérêt des utilisateurs peut reposer sur des critères de natures très diverses, qui nécessitent des modélisations spécifiques.

En conclusion, dans l'état actuel de la recherche dans ce domaine, il est important de mentionner que la production automatique de résumés sous forme de zapping est encore loin d'être comparable à ce qui peut être fait par des professionnels. Que l'on aime ou pas, le Zapping tel qu'il est proposé par Canal+ peut être considéré comme un objet lié à la sensibilité, aux intentions de son auteur et qui peut possèder une structure narrative... autant d'éléments qu'il semble très difficile de simuler actuellement.

8.0.2 Perspectives

Cette section propose quelques pistes pour compléter les études que nous avons réalisées ou pour améliorer les méthodes proposées. Ces pistes concernent essen-

tiellement trois aspects complémentaires :

1. l'amélioration de la transcription automatique de la parole sur des données Web.
2. la détection des vidéos ayant un intérêt particulier.
3. l'ordonancement des segments d'intérêt pour le résumé vidéo par extraction.

Le premier point est relatif à la nécessité d'amélioration des performances de la transcription automatique et cela est d'autant plus vrai dans le traitement de données Web qui sont très variables. Un des problèmes majeurs des systèmes de transcription automatique de la parole sur ce type de données est que les modèles de langage sont relativement figés, et que l'évolution permanente des contenus est difficilement gérable par des systèmes statiques dont les connaissances sont acquises définitivement par apprentissage sur des corpus fermés. La question des capacités d'adaptation, voire d'auto-adaptation des systèmes est essentielle dans ce contexte. Dans le cadre d'un résumé vidéo de type zapping, avoir un modèle de langage adapté quotidiennement à l'actualité serait très important.

Dans le processus de création du zapping, nous avons sélectionné les vidéos ayant un intérêt présumé pour l'utilisateur. Nous avons fait un certain nombre de propositions pour l'évaluation de l'intérêt d'une vidéo ou d'un moment particulier, basées sur la popularité des vidéos, sur le fond ou la forme des contenus et sur la notion de saillance. Une des difficultés majeures à laquelle nous nous sommes heurtés est la définition et la caractérisation de cet intérêt. Cette question reste ouverte ; l'aborder plus complètement supposerait probablement de mieux connaître le comportement des utilisateurs, sujet qui relève plutôt des sciences humaines.

La question de l'ordre des segments d'intérêt dans le résumé automatique multi-vidéos se pose. En effet dans un résumé multi-documents, les segments sont extraits de documents épars et une structure discursive doit être recomposée afin de rendre le résumé compréhensible par le lecteur. Des questions se posent alors sur l'ordre d'insertion des segments d'intérêt dans le résumé.

Acronyms

ACP	Analyse en Composantes Principales.
ANN	Artificial Neural Network.
BIC	Bayesian Information Criterion.
CER	Taux d'Erreur de Classification – Correct Error Rate.
CFA	Classification de Forme Audio.
CSM	Canonical State Models.
DDA	Algorithme de Décodage Guidé – Driven Decoding Algorithm.
EM	Expectation Maximization.
EPAC	Exploration de masse de documents audio pour l'extraction et le traitement de la parole conversationnelle.
ESTER	Evaluation des Systèmes de Transcription Enrichie d'Emissions Radiodiffusées.
FA	Analyse Factorielle – Factor Analysis.
GMM	Modèle de Mélanges Gaussiens – Gaussian Mixture Model.
HMM	Modèle de Markov Caché – Hidden Markov Model.
HSV	Teinte Saturation Valeur – Hue Saturation Value.
IDF	Fréquence Inverse de Document.
IGV	Identification du Genre Vidéo.
IV	Dans le Vocabulaire – In Vocabulary.

JFA	Joint Factor Analysis.
LIA	Laboratoire Informatique d'Avignon.
LIP	Points d'Intérêts Locaux.
LIUM	Laboratoire d'Informatique de l'Université du Maine.
MAP	Maximum A Posteriori.
MAP	Mean Average Precision.
MFCC	Mel Frequency Cepstral Coefficients.
MLLR	Maximum Likelihood Linear Regression.
MLP	Multi-Layer Perceptron.
MMR	Maximal Marginal Relevance.
NIST	National Institute of Standards and Technology.
OCR	Reconnaissance Optique de Caractères – Optical Character Recognition.
OOV	Hors Vocabulaires – Out-Of Vocabulary.
PLNE	Programmation Linéaire en Nombres Entiers.
PLP	Perceptual Linear Predictive.
RAP	Reconnaissance Automatique de la Parole.
ROUGE	Recall-Oriented Undestudy for Gisting Evaluation.
RPM2	Résumé Pluri-Média Multidocument.
RST	Théorie de la Structure Rhétorique.
RVB	Rouge Vert Bleu.
SDC	Shifted Delta Cepstra.
SGMM	Subspace Gaussian Mixture Model.
SPEERAL	Speech RAL.
STD	Détection de termes dans un document audio - Spoken Term Detection.
SVM	Support Vector Machine.
TF	Fréquence de Terme.
UBM	Universal Background Model.
WER	Taux d'Erreur de Mots – Word Error Rate.

ZCR Zero Crossing Rate.

Liste des publications personnelles

Chapitres de livres

– S. Oger, M. Rouvier, N. Camelin, R. Kessler, F. Lefèvre et J-M. Torres-Moreno, Le système du LIA pour la campagne DEFT'10 : datation et localisation d'articles de presse francophones, dans *Campagnes DEFT, Systèmes d'information et organisations documentaires, Hermes, 2012*

Revues Internationales

– D. Matrouf, F. Verdet, M. Rouvier, J-F. Bonastre et G. Linarès, Modeling Nuisance Variabilities with Factor Analysis for GMM-based Audio Pattern Classification, dans *Computer Speech and Language*

– M. Rouvier, G. Linarès et B. Lecouteux, Query driven strategy for onthefly term spotting in spontaneous speech, dans *EURASIP Journal on Audio*

Conférences Internationales

– G. Dupuy, M. Rouvier, S. Meignier et Y. Estève, i-vectors and ILP clustering adapted to cross-show speaker diarization, dans *InterSpeech 2012*

– M. Bouallegue, M. Rouvier, D. Matrouf et G. Linarès, Subspace Gaussian Mixture Models Based on Noise Compensation for Speech Recognition, dans *InterSpeech 2012*

– F. Bougares, M. Rouvier, Y. Estève et G. Linarès, Low latency combination of parallelized single-pass LVCSR systems, dans *InterSpeech 2012*

– M. Rouvier et S. Meignier, A Global Optimization Framework For Speaker Diarization, dans *Speaker Odyssey 2012*

- M. Rouvier, M. Bouallegue, D. Matrouf et G. Linarès, Factor Analysis Based Session Variability Compensation for Automatic Speech Recognition, dans *ASRU 2011*

- M. Bouallegue, D. Matrouf, M. Rouvier et G. Linarès, Subspace Gaussian Mixture Models for Vectorial HMM-states Representation, dans *ASRU 2011*

- Y. Li, B. Merialdo, M. Rouvier et G. Linarès, Static and Dynamic Video Summaries, dans *ACM Multimedia 2011*

- T. Bazillon, B. Maza, M. Rouvier, F. Bechet et A. Nasr, Speaker Role Recognition using question detection and characterization, dans *InterSpeech 2011*

- M. Rouvier, R. Dufour, G. Linarès et Y. Estève, A Language identification inspired method for spontaneous speech detection, dans *InterSpeech 2010*

- M. Rouvier, G. Linarès et D. Matrouf, On-the-fly video genre classification by combination of audio features, dans *ICASSP 2010*

- S. Oger, M. Rouvier et G. Linarès, Transcription-based video genre classification, dans *ICASSP 2010*

- M. Rouvier, G. Linarès et D. Matrouf, Robust Audio-based Classification of Video Genre, dans *InterSpeech 2009*

- M. Rouvier, D. Matrouf et G. Linarès, Factor Analysis for Audio-based Video Genre Classification, dans *InterSpeech 2009*

- M. Rouvier, G. Linarès et B. Lecouteux, On-the-fly term spotting by phonetic filtering and request-driven decoding, dans *Spoken Language Technology 2008*

Conférences Nationales

- G. Dupuy, M. Rouvier, S. Meignier et Y. Estève, Segmentation et Regroupement en Locuteurs d'une collection de documents audio, dans *JEP 2012*

- M. Rouvier et S. Meignier, Nouvelle approche pour le regroupement des locuteurs dans des émissions radiophoniques et télévisuelles, dans *JEP 2012*

- F. Bougares, Y. Estève, P. Deléglise, M. Rouvier et G. Linarès, Avancés dans le domaine de la transcription automatique par décodage guidé, dans *JEP 2012*

- T. Bazillon, B. Maza, M. Rouvier, F. Bechet, A. Nasr, Qui êtes vous ? Catégoriser les questions pour déterminer le rôle des locuteurs dans des conversa-

tions orales, dans *TALN 2011*

– S. Oger, M. Rouvier, N. Camelin, R. Kessler, F. Lefèvre et J-M. Torres-Moreno, Système du LIA pour la campagne DEFT'10 : datation et localisation d'articles de presse francophones, dans *DEFT 2010*

– S. Oger, M. Rouvier et G. Linarès, Classification du genre vidéo reposant sur des transcriptions automatiques, dans *TALN 2010*

– M. Rouvier, G. Linarès et D. Matrouf, Identification du genre vidéo à la volée par combinaison de paramètres acoustiques, dans *JEP 2010*

– M. Rouvier, G. Linarès et D. Matrouf, Identification robuste du genre vidéo par l'audio, dans *MajecSTIC 2009*

Autres

– M. Rouvier et G. Linarès, LIA @ MediaEval 2011 : Compact Representation of Heterogeneous Descriptors for Video Genre Classification, dans *MediaEval 2011*

Bibliographie

(Adda-Decker et al., 2004) M. Adda-Decker, B. Habert, C. Barras, G. Adda, P. B. de Mareuil, & P. Paroubek, 2004. Une étude des disfluences pour la transcription automatique de la parole spontanée et l'amélioration des modèles de langage. In Proc. of *Journées d'Étude sur le Parole (JEP)*.

(Barzilay and Elhadad, 1997) R. Barzilay & M. Elhadad, 1997. Using lexical chains for text summarization. *Workshop on Intelligent Scalable Text Summarization (ISTS)*.

(Baxendale, 1958) P. B. Baxendale, 1958. Machine-made index for technical literature : An experiment. *IBM Journal of Research and Development* *2*(4), 354–361.

(Bazillon et al., 2008) T. Bazillon, V. Jousse, F. Béchet, Y. Estève, G. Linarès, & D. Luzzati, 2008. La parole spontanée : transcription et traitement. In Proc. of *Revue Traitement Automatique des Langues (TAL)*.

(Benayed et al., 2004) Y. Benayed, D. Fohr, J.-P. Haton, & G. Chollet, 2004. Confidence measure for keyword spotting using support vector machines. In Proc. of *International Conference on Acoustics Speech and Signal Processing (ICASSP)*.

(Benzeghiba et al., 2007) M. Benzeghiba, R. D. Mori, O. Deroo, S. Dupont, T. Erbes, D. Jouvet, L. Fissore, P. Laface, A. Mertins, C. Ris, R. Rose, V. Tyagi, & C. Wellekens, 2007. Automatic speech recognition and speech variability : A review. *Speech Communication* *49*(10-11), 763–786.

(Bonastre et al., 2005) J.-F. Bonastre, F. Wils, & S. Meignier, 2005. Alize, a free toolkit for speaker recognition. In Proc. of *International Conference on Acoustics Speech and Signal Processing (ICASSP)*.

(Bordes et al., 2007) A. Bordes, L. Bottou, P. Gallinari, & J. Weston, 2007. Solving multiclass support vector machines with larank. In Proc. of *International Conference on Machine Learning (ICML)*, 89–96.

(Bouallegue et al., 2011) M. Bouallegue, D. Matrouf, & G. Linares, 2011. A simplified subspace gaussian mixture to compact acoustic models for speech recognition. In Proc. of *International Conference on Acoustics Speech and Signal Processing (ICASSP)*.

(Brin and Page, 1998) S. Brin & L. Page, 1998. The anatomy of a large-scale hypertextual web search engine. *Computer Networks and ISDN Systems 30*(1-7), 107–117.

(Caelen-Haumont, 2002) G. Caelen-Haumont, 2002. Perlocutory values and functions of melisms in spontaneous dialogue. In Proc. of *International Conference on Speech Prosody (ICSP)*, 195–198.

(Campbell et al., 2006) W. Campbell, J. Campbell, D. Reynolds, E. Singer, & P. Torres-Carrasquillo, 2006. Support vector machines for speaker and language recognition. *Computer Speech and Language 20*(2-3), 210–229.

(Campione and Véronis, 2004) E. Campione & J. Véronis, 2004. Pauses et hésitations en français spontané. In Proc. of *Journées d'Étude sur le Parole (JEP)*.

(Carbonell and Goldstein, 1998) J. Carbonell & J. Goldstein, 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In Proc. of *ACM SIGIR Conference on Research and development in information retrieval*, 335–336.

(Charton et al., 2008) E. Charton, T. Merlin, C. Lévy, A. Larcher, S. Meignier, J.-F. Bonastre, L. Besacier, J. Farinas, & B. Ravera, 2008. Mistral : Plate-forme open source d'authentification biométrique. In Proc. of *Journées d'Étude sur le Parole (JEP)*.

(Chen et al., 2002) S.-C. Chen, M.-L. Shyu, W. Liao, & C. Zhang, 2002. Scene change detection by audio and video clues. In Proc. of *IEEE International Conference on Multimedia and Expo (ICME)*, 365–368.

(Cohen, 1960) J. Cohen, 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement 20*, 37–46.

(Dan Istrate, 2005) N. S. C. F. J.-F. B. Dan Istrate, 2005. Conference of the international speech communication association (interspeech). In Proc. of *Systems, Man, and Cybernetics*.

(Di Eugenio and Glass, 2004) B. Di Eugenio & M. Glass, 2004. The kappa statistic : a second look. *Computationel Linguistic 30*, 95–101.

(Dimitrova et al., 2000) N. Dimitrova, L. Agnihotri, & G. Wei, 2000. Video classification based on hmm using text and faces. In Proc. of *European Signal Processing Conference (EUSIPCO)*.

(Dufour, 2010) R. Dufour, 2010. *Transcription Automatique de la Parole Spontanée*. Ph. D. thesis, LIUM.

(Dufour et al., 2010a) R. Dufour, F. Bougares, Y. Estève, & P. Deléglise, 2010a. Unsupervised model adaptation on targeted speech segments for LVCSR system combination. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*, 885–888.

(Dufour et al., 2010b) R. Dufour, F. Bougares, Y. Estève, & P. Deléglise, 2010b. Unsupervised model adaptation on targeted speech segments for lvcsr system combination. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Dufour et al., 2009) R. Dufour, V. Jousse, Y. Estève, F. Béchet, & G. Linarès, 2009. Spontaneous speech characterization and detection in large audio database. In Proc. of *International Conference on Speech and Computer (SPECOM)*.

(Edmundson, 1969) H. P. Edmundson, 1969. New methods in automatic extracting. *Journal of the ACM (JACM) 16*, 264–285.

(Eide and Gish, 1996) E. Eide & H. Gish, 1996. A parametric approach to vocal tract length normalization. *International Conference on Acoustics Speech and Signal Processing (ICASSP) 1*, 346–348.

(Estève et al., 2010) Y. Estève, T. Bazillon, J.-Y. Antoine, F. Béchet, & J. Farinas, 2010. The epac corpus : manual and automatic annotations of conversational speech in french broadcast news. In Proc. of *Language Resources and Evaluation (LREC)*.

(Ezzat and Poggio, 2008) T. Ezzat & T. Poggio, 2008. Discriminative word-spotting using ordered spectro-temporal patch features. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Favre, 2007) B. Favre, 2007. *Résumé automatique de parole pour un accès efficace aux bases de données audio*. Ph. D. thesis, LIA.

(Fischer et al., 1995) S. Fischer, R. Lienhart, & W. Effelsberg, 1995. Automatic recognition of film genres. In Proc. of *ACM Multimedia*.

(Fiscus et al., 1998) J. G. Fiscus, J. Ajot, & J. S. Garofolo, 1998. The rich transcription 2007 meeting recognition evaluation.

(Fiscus et al., 2007) J. G. Fiscus, J. Ajot, & J. S. Garofolo, 2007. The rich transcription 2007 meeting recognition evaluation. In Proc. of *CLEAR : Classification of Events, Activities and Relationships*.

(Forman, 2003) G. Forman, 2003. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research (JMLR) 3*, 1289–1305.

(Gales and Yu, 2010) M. Gales & K. Yu, 2010. Canonical state models for automatic speech recognition. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Garg et al., 2009) N. Garg, B. Favre, K. Reidhammer, & D. Hakkani-Tür, 2009. ClusterRank : A Graph Based Method for Meeting Summarization. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Gibson et al., 2002) D. P. Gibson, N. W. Campbell, & B. T. Thomas, 2002. Visual abstraction of wildlife footage using gaussian mixture models and the minimum description length criterion. In Proc. of *International Conference on Pattern Recognition (ICPR)*, 814–817.

(Gillick and Favre, 2009) D. Gillick & B. Favre, 2009. A Scalable Global Model for Summarization. In Proc. of *Human Language Technology conference (HLT-NAACL)*.

(Gupta et al., 1997) A. Gupta, R. A. Gupta, & R. Jain, 1997. Visual information retrieval.

(Hauptmann et al., 2002) A. G. Hauptmann, R. Yan, Y. Qi, R. Jin, M. G. Christel, M. Derthick, M. yu Chen, R. V. Baron, W.-H. Lin, & T. D. Ng, 2002. Video classification and retrieval with the informedia digital video library system. In Proc. of *Text REtrieval Conference (TREC)*.

(Hori and Furui, 2003) C. Hori & S. Furui, 2003. A new approach to automatic speech summarization. *IEEE Transactions on Multimedia*, 368–378.

(Hovy and Lin, 1998) E. Hovy & C.-Y. Lin, 1998. Automated text summarization and the summarist system. In Proc. of *TIPSTER Text Summarization Evaluation Conference*, 197–214.

(Ianeva et al., 2003) T. Ianeva, A. de Vries, & H. Rohrig, 2003. Detecting cartoons : a case study in automatic video-genre classification. *IEEE International Conference on Multimedia and Expo (ICME) 1*, 449–452.

(Jitendra Ajmera and Bourlard, 2002) I. A. M. Jitendra Ajmera & H. Bourlard, 2002. Robust hmm-based speech/music segmentation. In Proc. of *International Conference on Acoustics Speech and Signal Processing (ICASSP)*.

(Jones, 1972) K. S. Jones, 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation 28*, 11–21.

(Jousse et al., 2008) V. Jousse, Y. Estève, F. Béchet, T. Bazillon, & G. Linarès, 2008. Caractérisation et détection de parole spontanée dans de larges collections de documents audio. In Proc. of *Journées d'Étude sur le Parole (JEP)*.

(Kazemian et al., 2008) S. Kazemian, F. Rudzicz, G. Penn, & C. Munteanu, 2008. A critical assessment of spoken utterance retrieval through approximate lattice representations. In Proc. of *International Conference on Multimedia Information Retrieval (MIR)*, 83–88.

(Kenny, 2006) P. Kenny, 2006. Joint factor analysis of speaker and session variability : Theory and algorithms. Technical report, CRIM.

(Kenny et al., 2007) P. Kenny, G. Boulianne, P. Ouellet, & P. Dumouchel, 2007. Speaker and session variability in gmm-based speaker verification. *IEEE Transactions on Audio, Speech, and Language Processing 15*(4), 1448–1460.

(Keshet et al., 2009) J. Keshet, D. Grangier, & S. Bengio, 2009. Discriminative keyword spotting. *Speech Communication 51*(4), 317 – 329.

(Knight and Marcu, 2000) K. Knight & D. Marcu, 2000. Statistics-based summarization - step one : Sentence compression. In Proc. of *Conference on Artificial Intelligence and Conference on Innovative Applications of Artificial Intelligence*, 703–710.

(Kobla et al., 2000) V. Kobla, D. DeMenthon, & D. S. Doermann, 2000. Identifying sports videos using replay, text, and camera motion features. In Proc. of *Storage and Retrieval for Media Databases*, Volume 3972, 332–343.

(Kohler and Kennedy, 2002) M. Kohler & M. Kennedy, 2002. Language identification using shifted delta cepstra. In Proc. of *International Midwest Symposium on Circuits and Systems (MWSCAS)*.

(Le Nguyen et al., 2004) M. Le Nguyen, A. Shimazu, S. Horiguchi, B. T. Ho, & M. Fukushi, 2004. Probabilistic sentence reduction using support vector machines. In Proc. of *Conference on Computational Linguistics (COLING)*.

(Lecouteux et al., 2006) B. Lecouteux, G. Linarès, P. Nocera, & J.-F. Bonastre, 2006. Imperfect transcript driven speech recognition. In Proc. of *International Conference on Spoken Language Processing (ICSLP)*.

(Lie and Merialdo, 2010) Y. Lie & B. Merialdo, 2010. Multi-video summarization based on video-mmr. In Proc. of *International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*.

(Lin, 2003) C.-Y. Lin, 2003. Improving summarization performance by sentence compression : a pilot study. In Proc. of *International workshop on Information retrieval with Asian languages*, 1–8.

(Lin, 2004) C.-Y. Lin, 2004. Rouge : a package for automatic evaluation of summaries. 25–26.

(Lin and Hovy, 2003) C.-Y. Lin & E. Hovy, 2003. The potential and limitations of automatic sentence extraction for summarization. In Proc. of *Human Language Technology conference (HLT-NAACL)*, 73–80.

(Lin and Chen, 2009) S.-H. Lin & B. Chen, 2009. Improved speech summarization with multiple-hypothesis representations and kullback-leibler divergence measures. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*, 1847–1850.

(Linarès et al., 2007) G. Linarès, P. Nocera, D. Massonié, & D. Matrouf, 2007. The lia speech recognition system : from 10xrt to 1xrt. In Proc. of *International conference on Text, Speech and Dialogue*, 302–308.

(Liscombe et al., 2005) J. Liscombe, G. Riccardi, & D. Z. Hakkani-Tür, 2005. Using context to improve emotion detection in spoken dialog systems. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*, 1845–1848.

(Liu et al., 2010) Y. Liu, S. Xie, & F. Liu, 2010. Using n-best recognition output for extractive summarization and keyword extraction in meeting speech. In Proc. of *International Conference on Acoustics Speech and Signal Processing (ICASSP)*.

(Liu et al., 1998) Z. Liu, Y. Wang, & T. Chen, 1998. Audio feature extraction and analysis for scene segmentation and classification. In Proc. of *Journal of VLSI Signal Processing System*, 61–79.

(Longworth and Gales, 2008) C. Longworth & M. Gales, 2008. A generalised derivative kernel for speaker verification. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*, 1381–1384.

(Luhn, 1958) H. P. Luhn, 1958. The automatic creation of literature abstracts. *IBM Journal of Research and Development 2*, 159–165.

(Marcu, 1997) D. Marcu, 1997. From discourse structures to text summaries. In Proc. of *Workshop on Intelligent Scalable Text Summarization (ISTS)*, 82–88.

(Maskey and Hirschberg, 2005) S. Maskey & J. Hirschberg, 2005. Comparing lexical, acoustic/prosodic, structural and discourse features for speech summarization. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Maskey and Hirschberg, 2006) S. Maskey & J. Hirschberg, 2006. Summarizing speech without text using hidden markov models. In Proc. of *Human Language Technology conference (HLT-NAACL)*, 89–92.

(Matrouf et al., 2007) D. Matrouf, N. Scheffer, B. Fauve, & J.-F. Bonastre, 2007. A straightforward and efficient implementation of the factor analysis model for speaker verification. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Meignier and Merlin, 2010) S. Meignier & T. Merlin, 2010. Lium spkdiarization : An open source toolkit for diarization. In Proc. of *CMU SPUD Workshop*.

(Mihalcea and Tarau, 2004) R. Mihalcea & P. Tarau, 2004. Textrank : Bringing order into texts. In Proc. of *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

(Miller et al., 2007) D. R. H. Miller, M. Kleber, C. lin Kao, O. Kimball, T. Colthurst, S. A. Lowe, R. M. Schwartz, & H. Gish, 2007. Rapid and accurate spoken term detection. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Murray et al., 2005) G. Murray, S. Renals, & J. Carletta, 2005. Extractive summarization of meeting recordings. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Neiberg et al., 2006) D. Neiberg, K. Elenius, & K. Laskowski, 2006. Emotion recognition in spontaneous speech using gmms. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Nenkova et al., 2007) A. Nenkova, R. Passonneau, & K. McKeown, 2007. The pyramid method : Incorporating human content selection variation in summarization evaluation. *ACM Transaction Speech Langage Processing 4*.

(Ono et al., 1994) K. Ono, K. Sumita, & S. Miike, 1994. Abstract generation based on rhetorical structure extraction. In Proc. of *Conference on Computational Linguistics (COLING)*, 344–348.

(Pinto et al., 2008) J. Pinto, I. Szoke, S. Prasanna, & H. Hermansky, 2008. Fast Approximate Spoken Term Detection from Sequence of Phonemes. In Proc. of *Workshop on Searching Spontaneous Conversational Speech at SIGIR*.

(Povey et al., 2010) D. Povey, L. Burget, M. Agarwal, P. Akyazi, K. Feng, A. Ghoshal, O. Glembek, N. Goel, M. Karafiá andt, A. Rastrow, R. Rose, P. Schwarz, & S. Thomas, 2010. Subspace gaussian mixture models for speech recognition. In Proc. of *International Conference on Acoustics Speech and Signal Processing (ICASSP)*.

(Radev and Tam, 2003) D. R. Radev & D. Tam, 2003. Summarization evaluation using relative utility. In Proc. of *International Conference on Information and Knowledge Management (CIKM)*, 508–511.

(Roach and Mason, 2001) M. Roach & J. Mason, 2001. Classification of video genre using audio. In Proc. of *European Conference on Speech Communication and Technology (EUROSPEECH)*, 2693–2696.

(Saraclar and Sproat, 2004) M. Saraclar & R. Sproat, 2004. Lattice-based search for spoken utterance retrieval. In Proc. of *Human Language Technology conference (HLT-NAACL)*, 129–136.

(Sebastiani, 2002) F. Sebastiani, 2002. Machine learning in automated text categorization. *ACM Computing Surveys (CSUR) 34*(1), 1–47.

(Sivic and Zisserman, 2003) J. Sivic & A. Zisserman, 2003. Video google : A text retrieval approach to object matching in videos. 1470–1477.

(Snoek and Worring, 2005) C. G. M. Snoek & M. Worring, 2005. Multimodal video indexing : A review of the state-of-the-art. *Multimedia Tools Application 25*(1), 5–35.

(Sroka and Braida, 2005) J. J. Sroka & L. D. Braida, 2005. Human and machine consonant recognition. *Speech Communication 45*(4), 401–423.

(Sundaram and Chang, 2000) H. Sundaram & S. F. Chang, 2000. Video scene segmentation using video and audio features. In Proc. of *IEEE International Conference on Multimedia and Expo (ICME)*.

(Takenobu and Makoto, 1994) T. Takenobu & I. Makoto, 1994. Text categorization based on weighted inverse document frequency. Technical report.

(Truong and Dorai, 2000) B. T. Truong & C. Dorai, 2000. Automatic genre identification for content-based video categorization. In Proc. of *International Conference on Pattern Recognition (ICPR)*, 230–233.

(Valenza et al., 1999) R. Valenza, T. Robinson, M. Hickey, R. Tucker, F. Rd, & S. Gifford, 1999. Summarisation of spoken audio through information extraction. In Proc. of *ESCA Workshop on Accessing Information in Spoken Audio*.

(Wang et al., 2003) P. Wang, R. Cai, & S.-Q. Yang, 2003. A hybrid approach to news video classification multimodal features. In Proc. of *International Conference on Information, Communication and Signal Processing (ICICS)*, 787–791.

(Watson, 2003) D. Watson, 2003. *Death sentence : the decay of public language*, Volume 9.

(Wechsler et al., 1998) M. Wechsler, E. Munteanu, & P. Schäuble, 1998. New techniques for open-vocabulary spoken document retrieval. In Proc. of *ACM SIGIR Conference on Research and development in information retrieval*, 20–27.

(Wei et al., 2000) G. Wei, L. Agnihotri, & N. Dimitrova, 2000. Tv program classification based on face and text processing. In Proc. of *IEEE International Conference on Multimedia and Expo (ICME)*, 1345–1348.

(Wei and Sethi, 1999) G. Wei & I. K. Sethi, 1999. Face detection for image annotation. *Pattern Recognition Letters 20*(11-13), 1313–1321.

(Wold et al., 1996) E. Wold, T. Blum, D. Keislar, & J. Wheaten, 1996. Content-based classification, search, and retrieval of audio. *IEEE Mutlimedia 3*(3), 27–36.

(Xie et al., 2009a) S. Xie, B. Favre, D. Hakkani-Tür, & Y. Liu, 2009a. Leveraging Sentence Weights in Concept-based Optimization Framework for Extractive Meeting Summarization. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.

(Xie et al., 2009b) S. Xie, D. Hakkani-Tür, B. Favre, & Y. Liu, 2009b. Integrating prosodic features in extractive meeting summarization. In Proc. of *Automatic Speech Recognition and Understanding (ASRU)*.

(Xie and Liu, 2010) S. Xie & Y. Liu, 2010. Using confusion networks for speech summarization. 46–54.

(Yeung and Liu, 1995) M. M. Yeung & B. Liu, 1995. Efficient matching and clustering of video shots. In Proc. of *International Conference on Image Processing (ICIP)*, 338–.

(Yu et al., 2004a) P. Yu, K. Chen, C. Ma, & F. Seide, 2004a. Vocabulary-independent indexing of spontaneous speech. *International Conference on Acoustics Speech and Signal Processing (ICASSP)*.

(Yu et al., 2004b) X.-D. Yu, L. Wang, Q. Tian, & P. Xue, 2004b. Multi-level video representation with application to keyframe extraction. In Proc. of *International Multimedia Modelling Conference (MMM)*, 117–.

(Yuan et al., 2006) X. Yuan, W. Lai, T. Mei, X. S. Hua, X. Q. Wu, & S. P. Li, 2006. Automatic video genre categorization using hierarchical svm. In Proc. of *International Conference on Image Processing (ICIP)*, 2905–2908.

(Zechner and Waibel, 2000) K. Zechner & A. Waibel, 2000. Minimizing word error rate in textual summaries of spoken language. In Proc. of *Human Language Technology conference (HLT-NAACL)*, 186–193.

(Zhang, 1997) H. Zhang, 1997. An integrated system for content-based video retrieval and browsing. *International Conference on Pattern Recognition (ICPR) 30*(4), 643–658.

(Zhang and Fung, 2007) J. Zhang & P. Fung, 2007. Speech summarization without lexical features for mandarin broadcast news. In Proc. of *Human Language Technology conference (HLT-NAACL)*, 213–216.

(Zhu and Penn, 2006) X. Zhu & G. Penn, 2006. Summarization of spontaneous conversations. In Proc. of *Conference of the International Speech Communication Association (INTERSPEECH)*.