tel-00954238, version 1 - 28 Feb 2014

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 5 décembre 2011 devant un jury composé de :

M^{me}	Régine André-Obrecht	Professeur, IRIT, Toulouse	Rapporteur
M.	Guillaume Gravier	Maître de Conférence, IRISA, Rennes	Rapporteur
M.	Yannik Estève	Professeur, LIUM, Le Mans	Examinateur
M.	Benoît Favre	Maître de Conférence, LIF, Marseille	Examinateur
M.	Driss Matrouf	Maître de Conférence, LIA, Avignon	Examinateur
M.	Georges Linarès	Professeur, LIA, Avignon	Directeur de thèse

Laboratoire Informatique d'Avignon

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Table des matières

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Chapitre 1

Introduction

Contents

1.1 Introduction

Ces dernières années, avec l'apparition des 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'intéressante 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 les experts doivent capter en quelques secondes ce que les vidéos valent en termes de buzz, d'infos ou de créativité.

Devant cette quantité d'informations, la gestion de l'information est de plus en plus problématique pour notre société. C'est devenu un enjeu industriel, scientifique et économique. Dans la recherche de vidéos, les utilisateurs gardent un rôle actif, c'est à dire qu'il 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 viendrait à lui. Les résumés sont la solution la plus logique à l'information pléthorique. Il faut arriver à fournir à l'utilisateur un éventail des vidéos disponibles où l'information est condensée et la plus efficace possible.

C'est ce à quoi s'est intéressé le projet 2 [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 s'occupe de 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 automatique multi-documents. Ce résumé devra essayer de sélectionner les vidéos ayant un intérêt et 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. Conccu 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 création d'un Zapping est un véritable challenge puisque c'est une équipe composée de 12 personnes, de tous horizons (art, danse, photographie, etc...) qui regardent la télé toute la journée pour isoler des séquences considérées comme "intéressantes". La sélection des vidéos marquantes n'est qu'une étape, puisqu'il faut 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 cca", 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 sur le modèle du "Zapping" proposé par Canal+ que nous avons essayé de nous rapprocher dans nos travaux. Le zapping est une forme de résumé dans lequel nous voulons sélectionner l'information *importante*. Contrairement aux résumés texte, nous avons 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 modéle du résumé texte, des segments sont extraits des différentes vidéos et aglomérés dans

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

tel-00954238, version 1 - 28 Feb 2014

une vidéo "résumé". La création d'un document sous forme de zapping est un véritable challenge scientifique puisque, afin de réaliser le dit document, plusieurs verrous scientifiques devront être levés. Les verrous débordent du cadre du résumé et posent des problèmes plus généraux, de caractérisation de vidéo 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 vidéo ayant un intérêt notable et évaluation de cet intérêt
3. Agrégation des différentes séquences

Pour collecter et sélectioner des vidéos, nous nous trouvons ici dans un contexte web ce qui pose deux problèmes : la taille de la collection disponible et la structuration de ces données. La collection des vidéos disponible sur le Web est gigantesque. Selon Youtube, 10 milliards de vidéos seraient hébergées par la plateforme. De plus cette collection n'est pas une collection fermée, elle augmente considérablement chaque jour. Ce sont environ, toujours selon Youtube, plus de 65 000 vidéos postés quotidiennement soit environ 20 heures de vidéos par minute ! Une recherche de vidéo la plus efficace doit obligatoirement reposer sur une structuration des collections par le contenu et/ou par les méta-données.

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. La recherche se base sur des méta-données 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 imparfaite ou partiellement remplies. La structuration des bases de données ne doit pas uniquement se faire sur des données remplies 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 contient bien évidemment cet intérêt notable. 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 personnalités car elle vise directement leur compétence qu'elles occupent. Ainsi, lors de l'interview de Luc Chatel (ministre de l'éducation), 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 continue par la solution du journaliste [2]. Nous sommes donc là au cœur de notre problème : comment

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

tel-00954238, version 1 - 28 Feb 2014

réussir à détecter que la séquence ayant un intérêt notable dans cet interview est précisément la question ainsi que la réponse de notre ministre ?

Une fois les sous-séquences obtenues, il faut les agréger afin de constituer notre document. Cette dernière étape devra faire attention à deux points : les documents parlant d'un même sujet devront être agrégés les uns à la suite des autres 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, consacrées à la création d'un document sous forme de zapping. 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 le domaine texte, 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 ASR permettant ainsi d'améliorer, dans des conditions acoustiques bruitées, la transcription automatique. Dans la troisième partie nous parlerons de la structuration et catégorisation de larges bases de données. Le chapitre 5, parlera de la classification de larges bases de données selon le genre vidéo. Dans le chapitre 6, nous parlerons de la structuration de bases de données audio selon le niveau de spontanéité. Dans la dernière partie, le chapitre 7 nous proposerons 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.

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Première partie

Etat de l'art

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Chapitre 2

Etat de l'art du résumé automatique

Contents

2.1 Introduction

Le but d'un système de résumé automatique est de produire une représentation condensée d'une source d'information dans laquelle les informations "importantes" du contenu original sont préservées. Les sources d'information pouvant être résumées sont nombreuses et hétérogènes : documents, vidéos, sonores ou textuels. Un résumé peut être produit à partir d'un seul ou de 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) qu'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 resté une des approches les plus répandues pendant les 50 années qui suivirent. 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).

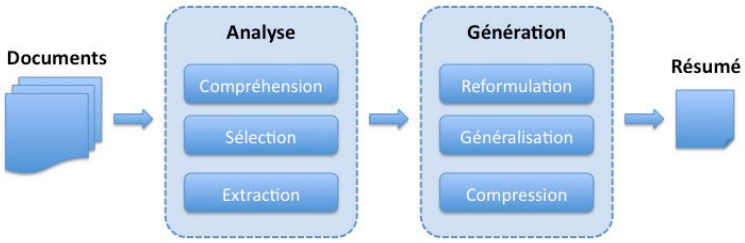

FIGURE 2.1 – *Présentation d'un système de résumé automatique audio.*

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). Il propose d'utiliser la fréquence d'un terme pour mesurer la pertinence des phrases, l'idée étant qu'une personne aura tendance a 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 de stop-liste. Les bases du résumé automatique sont lancées puisque l'auteur utilise les statistiques pour la production des résumés automatiques. Cette faccon de procéder a eu un impact sur la grande majorité des systèmes d'aujourd'hui puisqu'ils sont basés sur le même principe.

tel-00954238, version 1 - 28 Feb 2014

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 certain domaine, partagent des termes communs dans ce domaine mais qu'ils n'apportent pas d'informations saillantes. La pertinence des termes devrait donc être réduite. (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 inverse dans le document, 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 mot titre, où certain 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 compté quand le mot "bicycle" apparait mais elle est aussi comptée pour les mots "vélo", "pédale", "guidon", etc... Les concepts peuvent être déterminé en utilisant les liens sémantiques de la base de donnée de *WordNet*.

2.2.2 Approche basé 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 hors contexte.

Les propriétés de cohésion de texte ont été explorées par différentes approches de résumé. Dans (Barzilay and Elhadad, 1997), l'auteur introduit une méthode appelée "chaîne lexicale" (Lexical chains). Il utilise 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 type de relations

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...

tel-00954238, version 1 - 28 Feb 2014

dans la chaîne. Les phrases ou les chaînes les plus concentrées sont sélectionnées pour produire le résumé.

2.2.3 Approche basé 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, l'analyse du nombre de citations ou l'étude de la structure du Web. Ces algorithmes peuvent être vus comme les éléments clés 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 représenté 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 :

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

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 à des documents bruités ainsi que la redondance due à la parole spontanée. Cette méthode, appelé ClusterRank, propose dans une première étape de regrouper certaines phrases en classe selon leur score de similarité cosine. Le graphe est construit selon ces clusters.

tel-00954238, version 1 - 28 Feb 2014

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 étayant cette affirmation 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 proposent 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 qui sont 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.

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

tel-00954238, version 1 - 28 Feb 2014

Dans la formulation originelle de MMR, $Sim_1()$ et $Sim_2()$ sont la similarité $cosine()$ qui a fait ses 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'appuie 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 avant 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 de nombre entier. En effet la programmation linéaire en nombre entier 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. Il 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 la taille de la contrainte.

Les concepts sont représentés par des éléments d'informations comme par exemple 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 d'être extraite automatiquement, il faut ramener ces concepts à des mots plus simples, les n-grams, qui peuvent être utilisés pour représenter la structure du document. Cependant, les n-grams se recoupent souvent avec des marqueurs de discours ("en fait", "vous savez") lesquels peuvent rajouter du bruit. Un algorithme d'extraction de mot-clef est proposé pour représenter la séquence des mots ainsi que le contenu :

1. Extraction de tous les n-grams pour $n = 1, 2, 3$
2. Suppression du bruit : Suppression des n-grams qui apparaissent seulement une fois
3. Ré-évaluation des poids des Bi-gram et Tri-gram : $w_i = frequence(g_i) \cdot n \cdot idf(g_i)$ ou w_i est le poids final du n-gram, n la taille du n-gram et idf le poids Fréquence Inverse de Document (IDF) du mot.

Formellement, désignons c_i qui dénote la présence d'un concept i dans le résumé et s_j qui dénote la présence de la phrase j dans le résumé. Chaque concept peut apparaître dans des multiples phrases et les phrases peuvent contenir des

tel-00954238, version 1 - 28 Feb 2014

concepts multiples. L'occurrence du concept i dans la phrase j est notée par la variable binaire o_{ij}. Le score du résumé est exprimé comme la somme des poids positifs de w_i du concept présent dans le résumé. La taille du résumé est limitée par la constante L par-dessus la somme des longueurs l_j de ces phrases. Ainsi la recherche d'un résumé automatique peut être exprimée sous forme de problème ILP :

$$
\begin{aligned}
\text{Maximize} \quad & \sum_i w_i c_i \\
\text{Subject To} \quad & \sum_j l_x n_x \\
& n_x Occ_{ix} \leq c_i, \qquad \forall i, x \\
& \sum_x n_x Occ_{ix} \geq c_i, \quad \forall i, x \\
& c_i \in \{0,1\} \qquad \forall i \\
& n_x \in \{0,1\} \qquad \forall x
\end{aligned}
$$

Dans ce cadre, la fonction objectif permet de maximiser la somme pondérée des concepts présents dans le résumé compte tenu de la contrainte de la 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 que si un concept 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 aux résumés de texte automatique. La nécessité de résumer le contenu parlé s'est faite ressentir lorsque les bases de données audio/vidéo ont commencé a 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 débat) ainsi que les erreurs issues des systèmes de transcription automatique de parole.

2.3.1 Approche basé sur la prosodie

Le résumé de texte par extraction sélectionne les segments les plus représentants 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

tel-00954238, version 1 - 28 Feb 2014

verbale, car elle permet d'exprimer une information non-linguistique comme une intention, un changement de sujet, pouvoir mettre l'accent 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 à priori).

L'intégration de jeu de paramètre acoustique/prosodique a été principalement conduit 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 les domaines journalistique où le style de parole du locuteur varie peu, l'intégration des jeux de paramètre acoustique/prosodique n'apporte rien en général et peut même dégrader les résultats.

De manière général les auteurs proposent d'extraire des paramètres prosodiques issus du F0 (maximum, minimum, moyenne, médiane et variance) et l'énergie du signal (max, min, 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. De manière générale les paramètres prosodiques sont utilisés en complément de la transcription donnée par un ASR (Xie et al., 2009; Maskey and Hirschberg, 2005) ; mais 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 ASR. Selon le corpus, le taux d'erreurs-mots peut osciller généralement entre 10% et 50%. Ces taux d'erreurs-mots peuvent être imputables aux langages utilisés dans le document, les conditions acoustiques, etc...

Intuitivement le taux d'erreur-mot a un impact négatif sur les performances du système de résumé. Des précédentes recherches ont évalué le système de résumé utilisant la transcription humaine et la sortie d'un ASR. La plupart des travaux ont montré que les erreurs d'un ASR 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 de l'ASR pour créer le résumé. Les n-meilleures hypothèses, treillis de mots et réseau de confusion ont été largement utilisés comme une interface entre un ASR 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 ASR pour le résumé de parole. Plusieurs études utilisent les scores de confiance acoustique de la meilleure hypothèse de la sortie de l'ASR afin de rescorer 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

tel-00954238, version 1 - 28 Feb 2014

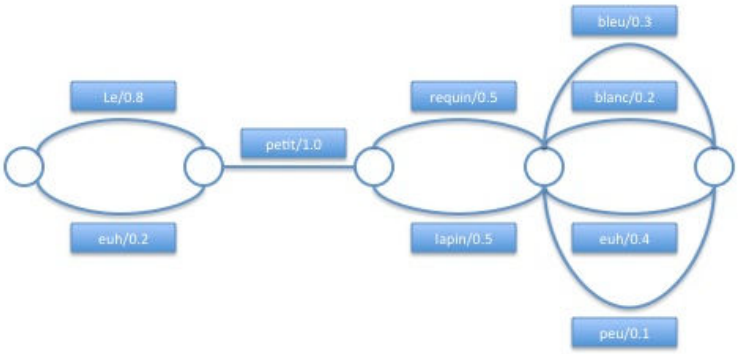

FIGURE 2.2 – *Exemple d'un treillis de mot.*

cadre de génération de résumé automatique pour le broadcast news chinois. 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.

2.4 Résumé vidéo

Le résumé vidéo n'est pas le sujet principal de notre thèse. Dans ce document nous évoquons certains objets, certaines techniques, venant du résumé vidéo. Nous proposons donc un bref aperccu 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 vidéos, sans visualisation et interprétation de l'ensemble de celles-ci. Les méthodes développées consistent à extraireun ensemble d'image fixes, apellées vignettes, qui misent 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.

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

Une variété de métriques a été proposée 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

tel-00954238, version 1 - 28 Feb 2014

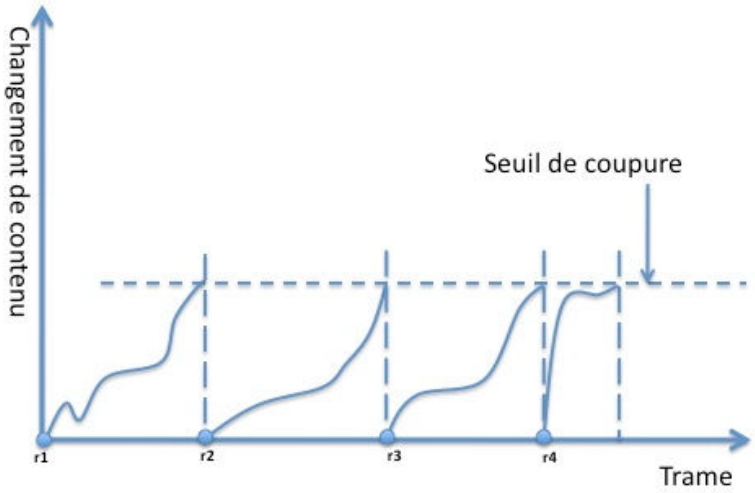

FIGURE 2.3 – *Illustration de la méthode de changement de contenu.*

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...

2.4.2 Approche basée sur le cluster

Cette approche représente les trames vidéo comme étant des points dans l'espace et fait l'hypothèse que le point représentatif d'un cluster formé 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 taille des dimensions et en retenant uniquement les variations significatives. Cette réduction de dimension se fait via une Principal Component Analysis (PCA).

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

Certains clusters peuvent être bruités ou ne contenir aucune information significative. (Zhang, 1997), Zhang considère les clusters qui ont une taille plus grande que la moyenne de la taille des clusters. Dans (Girgensohn and Boreczky, 1999), l'auteur propose qu'un cluster a une trame clef si elle contient quelques trames qui se suivent afin de supprimer les trames contenant des artefacts.

Finalement l'extraction des points représentatifs d'un cluster s'effectue en sé-

tel-00954238, version 1 - 28 Feb 2014

lectionnant le point le plus proche du centroïde d'un cluster. Ainsi, la trame peut être ajoutée à la séquence des trames du résumé (Yu et al., 2004b).

2.4.3 Video-MR

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 :

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

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 :

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

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 mot visuel" (baf of visual word). Un sac de mot visuel 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 clusterisés en n groupes par un algorithme de K-Means, où n représente le nombre de mots visuels dans le document.

2.5 Métrique d'évaluation

Évaluer un résumé est une tâche difficile 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 entre plusieurs personnes selon l'importance que chacune d'entre elles 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

tel-00954238, version 1 - 28 Feb 2014

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. Sur 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 phrase pertinentes dans le résumé de référence. La F-Mesure (F) est une mesure qui combine la précision et le rappel. La meilleure fa**c**con 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 fa**c**con égale ; il s'agit d'un cas particulier de la F-Mesure. L'équation de la F-Mesure s'écrit :

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

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

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 pour lequel la phrase donnée 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 un nombre de juges d'assigner un score d'utilité à

tel-00954238, version 1 - 28 Feb 2014

toutes les phrases dans un document. Nous pouvons définir le système de mesure suivant :

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

où u_{ij} est le score d'utilité de la phrase j de l'annotateur i, $\in_j$ est à 1 pour les meilleures e phrases selon la somme des scores d'utilité pour tous les juges, d'un autre côté sa valeur est 0, et λ_j est égale à 1 pour la meilleure e phrase extraite par le système, d'un autre côté, sa valeur est 0.

2.5.3 Similarité cosine

Les mesures présentées jusqu'à présent comptent combien de phrases 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 manière semi-automatique d'évaluer le 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 cosine :

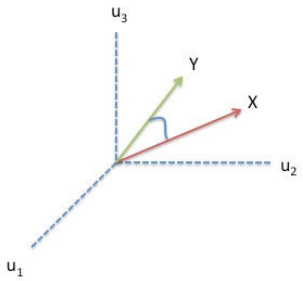

FIGURE 2.4 – .

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

où X représente le système de résumé candidat et Y celui de résumé de référence. Le modèle pour le résumé est le modèle d'espace de vecteur.

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

La mesure cosine 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 cosine que le résumé candidat initial alors que celui-ci sera complétement illisible. Dans (yew Lin, 2004), l'auteur propose la méthode

tel-00954238, version 1 - 28 Feb 2014

Recall-Oriented Undestudy for Gisting Evaluation (ROUGE) : cette mesure permet de connaître la similarité de n-grams 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-grams qui co-occure entre un résumé candidat et le résumé de référence. $Nombre(N - grammes)$ est le nombre de n-grams 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 regroupés et pondérés 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 le 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 est considéré comme important par les annotateurs. Cette méthode intéressante demande toutefois l'intervention d'un être humain pour annoter les corpus.

tel-00954238, version 1 - 28 Feb 2014

TABLE 2.2 – *Les phrases du résumé de référence sont indexées avec une lettre et un numéro : la lettre montre de quel résumé la phrase vient et le nombre indique la position de la phrase dans son résumé respectif. 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.

Ainsi dans les exemples du Tableau 2.2, l'annotation commence en identifiant les phrases similaires, comme les 4 phrases sous-lignées. Les phrases sélectionnées nous permettent d'obtenir deux SCU. Chaque SCU a un poids correspondant au nombre de résumés dans lequel il apparait.

Le premier SCU parle des Libyens qui ont été officiellement accusés de l'attentat de Lockerbie. Ce SCU est parlé dans les résumés A1 (two Lybyans, indicted), B1 (Two Libyans were indicted), C1 (Two Libyans, accused) et D2 (Two Libyan suspects were indicted). Le SCU obtient un poids égale a 4.

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

2.6 Conclusion

Nous avons pu constater, que ce soit dans le domaine de la vidéo, du texte ou de la parole, la création d'un résumé automatique consistait a essayé de sélectionner les informations les plus importantes tout en essayant de minimiser la redondance d'information. La plupart des travaux dans le domaine se sont focalisés sur deux points :

- la maniére d'extraire et de juger de la pertinence d'une information
- proposer un algorithme qui permet dans un même cadre de sélectionner et minimiser la redondance d'information

Le but du zaping est de sélectionner, dans un document, la séquence ayant un intérêt notable tout en essayant que cette sous-séquence soit unique dans le zapping. On peut constater, que le zapping est une forme particulière du résumé

tel-00954238, version 1 - 28 Feb 2014

automatique. Nous retrouvons bien cette question de redondance, mais la fonction d'intérêt est différente, puisque nous voulons sélectionner une séquence ayant un intérêt notable. Nous verrons, dans le Chapitre 6, comment nous allons intégrer toutes ces contraintes et comment nous évaluerons les résultats obtenues.

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Deuxième partie

Extraction du contenu

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Les systèmes de résumés automatiques audio 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 fourni au deuxième niveau qui va lui appliquer une méthode de résumé texte. Malheureusement, la transcription automatique fournie par un système de RAP (Reconnaissance Automatique de la Parole) 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 et à la tâche qu'il décode. Généralement, les systèmes de transcription obtiennent sur des données de radio journalistique un 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 diverses, suivant les types de documents. Les corpus disponibles sur Youtube ou Dailymotion nous observons deux principaux problèmes.

Le premier problème est lié au vocabulaire utilisé dans les vidéos. Le vocabulaire peut appartenir à un domaine scientifique, avoir des termes politiques, etc... et ce vocabulaire n'est pas forcément présent dans le lexique d'un système de RAP à grand vocabulaire. On appelle cela des mots Out Of Vocabulary (OOV), ce sont des mots qui sont absents du vocabulaire, du lexique, d'un système de RAP. Les mots peuvent être OOV soit à cause de la taille limitée du vocabulaire, soit parce que le mot 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, nouveau produit, nouveau film, etc...
- Des termes politique : nom de politicien, nom de législation, etc...
- Des termes étranger : 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 In Vocabulary (IV), perturbant ainsi la transcription. Nous proposons comme solution de récupérer des mots, des expressions revenant souvent dans l'actualité (site Internet, dépêche, blog....) et d'essayer de détecter ces mots, expressions dans le flux audio afin d'améliorer la transcription proposée par le système de RAP.

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 d'un système de RAP.

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. Pour cela, il est nécessaire d'intégrer des méthodes permettant de tolérer ou enlever ces variabilités à différent niveaux :

- 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 disponible 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 terme rapide dans des milieux bruités puis, dans le chapitre 4, un nouveau cadre de normalisation robuste de données acoustiques.

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Chapitre 3

Détection de terme à la volée

Contents

3.1 Introduction

La recherche des mots clefs dans un flux audio (Word Spotting) est l'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 un document audio, le but étant de rechercher des termes dans de vastes documents audio hétérogènes. Dans la détection de termes, on peut citer deux approches différentes : le Spoken Term Detection (STD) ou la détection de mots dans un flux audio continu (Keyword Spotting).

Le STD a été définie par le National Institute of Standards and Technology (NIST). Pour encourager la recherche et le développement de cette technologie, NIST organise une série d'évaluations pour le STD. La première évaluation pilote réalisée en 2006 se fit sur trois conditions : broadcast news, conversation téléphone 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

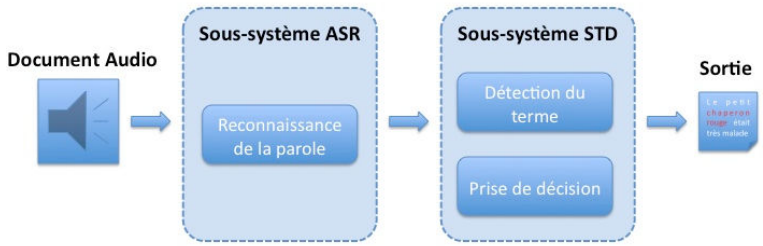

FIGURE 3.1 – *Présentation d'un système de détection de terme dans un document audio.*

La Figure 3.1 illustre le framework standard d'un système STD. Dans ce framework, le signal de la parole est en premier transcrit par une reconnaissance de la parole (mot, phonèmes, réseau de confusion...) et ensuite une recherche du terme [1] est effectuée sur cette transcription.

Une faccon 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 terme en anglais sur de la parole téléphonique.

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

tel-00954238, version 1 - 28 Feb 2014

Les différentes approches de système STD proposaient pour détecter des mots permettent d'obtenir une bonne précision parce que l'information lexicale est utilisés ; cependant, ils souffrent d'une lacune importante la détection des termes OOV. Les OOV n'apparaîtront jamais dans un treillis généré par le système de transcription de mots, et par conséquence ne peuvent pas être détectés. Pour résoudre ce problème la plupart des systèmes font de la reconnaissance sur des sous-unités 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 unité phonétique 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 condition propres, spécialement sur les données de radio qui ont été largement utilisées par les systèmes de RAP (Fiscus, 1997). Mais dans des conditions plus difficiles, comme un enregistrement dans un contexte bruité ou un discours spontané, les performances sont alors 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. Et la détection des termes doivent se faire sur de très grandes bases de données dans un temps raisonnables. Nous proposons un système de détection de terme en temps réel.

Nous proposons un système avec une architecture à deux niveaux dans lequel 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'ils maximisent séquentiellement le rappel et la précision respectivement au premier et second niveau.

Au premier niveau, une recherche rapide est vue comme une tâche de filtrage qui a pour but d'accepter ou rejeter les segments selon la probabilité de cibler le terme. Nous présentons un schéma général dans lequel le terme prononcé est projeté dans un graphe de filtre phonétique. 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 terme basé sur la RAP, ayant pour but de raffiner la détection du terme. Nous proposons d'améliorer le taux de détection en intégrant la requête dans le système, cette intégration est basée sur

tel-00954238, version 1 - 28 Feb 2014

l'algorithme de décodage guidé (DDA) qui a été précédemment proposé (Lecouteux et al., 2006).

Ainsi cette section est organisée comme suit : la section 3.3.1 présente l'architecture globale de notre détection de terme, la section 3.3.2 décrit le premier niveau, qui a pour but d'identifier les segments de parole dans lesquels la requête est probablement présente. Nous présentons un système de filtrage acoustique ou différent 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 terme à la volée est supposé rechercher de manière synchrone dans un flux de parole toutes les occurrences et les notifier une fois la séquence détectée.

L'architecture du système est composée de deux niveaux dans laquelle la précision et le rappel sont séquentiellement optimisés. Le premier niveau, strictement acoustique, est composé d'un outil qui identifie les segments de paroles 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 en premier transcrite phonétiquement en utilisant un lexique de prononciation et un phonétiseur à base de règles qui produit un graphe phonétique de 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 filtre phonétique. Les filtres phonétiques peuvent être basés sur des méthodes statistiques de classification comme le Gaussian Mixture Model (GMM) ou le Multi-Layer Perceptron (MLP). Dans la suite, le filtrage de graphe phonétique est appelé *filtre acoustique*, tandis que les *filtres phonétiques* opèrent au niveau du nœud.

A ce point, 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.

tel-00954238, version 1 - 28 Feb 2014

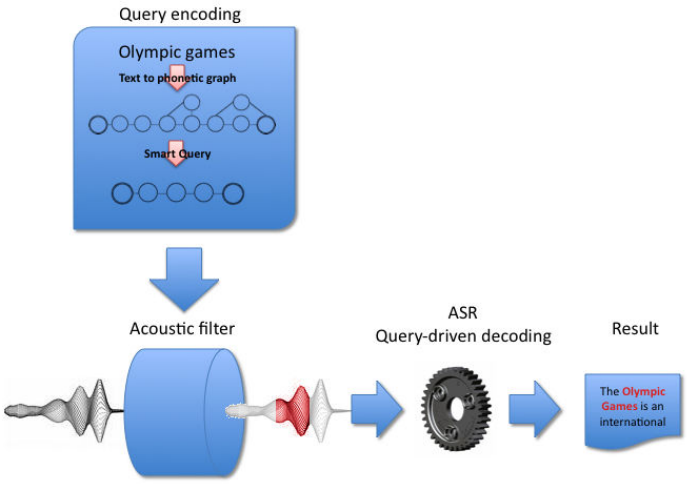

FIGURE 3.2 – *Architecture d'un système de détection de terme à la volée.*

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, ou 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 phonème où chaque chemin représente une variante de prononciation comme illustré dans la Figure 3.3.

L'approche utilisée pour détecter le terme est d'aligner le graphe et le signal dans une fenêtre glissante ainsi la probabilité du chemin total est utilisée pour prendre la décision de détecter ce terme. Cette approche est sous-optimale en terme de consommation et de ressource CPU : 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 afin de 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 plus profondément dans le prochain paragraphe, ou nous présenterons des filtres à base de GMM et de MLP.

Considérant cette stratégie d'élagage, il est clair que la partie la plus discriminante du graphe doit être évaluée en premier dans le but de réduire le temps de calcul tout en préservant le taux de précision. Par conséquent, le graphe peut être réduit en fonction de la complexité et de la capacité discriminative des sous-

tel-00954238, version 1 - 28 Feb 2014

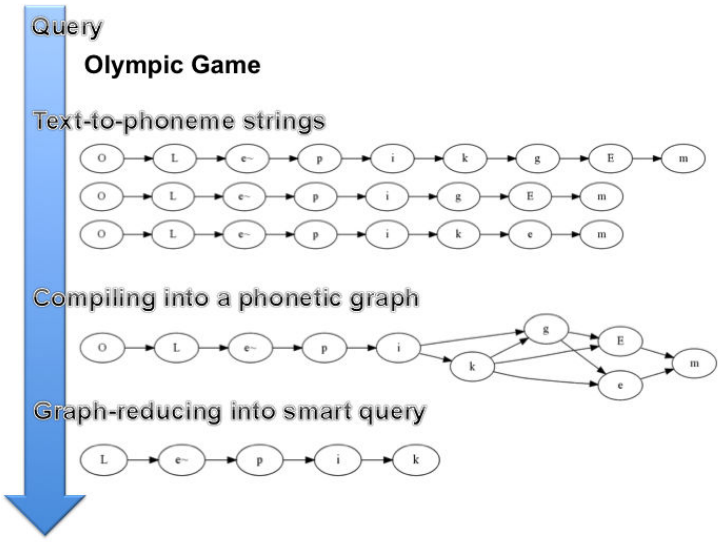

FIGURE 3.3 – *De la requête écrite à la requête compressée : La requête écrite est transcrite en graphe de prononciation. Le meilleur sous-graphe qui maximise la précision tout en minimisant les coûts CPU, est extrait pour construire la requête compressée.*

graphes. Nous proposons un algorithme de réduction de graphes des variantes de prononciation décrit dans le prochain paragraphe.

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 Hidden Markov Model (HMM) du système de RAP. Le graphe phonétique est développé selon la topologie HMM, chaque nœud de phonème dépendant du contexte est ensuite découpé suivant une séquence de n nœud d'état dépendant du contexte.

Les filtres d'état dépendant du contexte doivent être capables d'arrêter l'exploration 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 composée de 12 coefficients Perceptual Linear Preditive (PLP) plus l'énergie et de leurs dérivés première et seconde. L'Universal Background Model (UBM) est un modèle générique qui représente le signal de la parole, indépendamment de l'unité phonétique. Ici, l'UBM est un GMM composé de 64 composantes, estimé en utilisant la procédure d'Expectation-Maximization sur le corpus d'apprentissage.

tel-00954238, version 1 - 28 Feb 2014

Le seuil de coupure du filtre-dépendant 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.

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 en premier, 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 $S = \{S_i\}$ est l'état correspondant à la séquence de la chaîne phonétique, et C est la requête-dépendant du seuil.

3.3.2.3 Filtre phonétique basé sur un MLP

Des méthodes discriminatives pour la détection de mots clés ont été récemment traités par plusieurs auteurs dans (Keshet et al., 2009; Ezzat and Poggio, 2008; Y. et al., 2004) (Keshet et al. 2009, Ezzat et 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 segment. Nous proposons d'utiliser le MLP comme filtre phonétique discriminative.

Le filtrage à base de MLP intègre 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.

Nous utilisons comme classifieur un MLP. Chaque sortie du MLP correspond a un état S_i, un jeu de phonème contexte-indépendant. Le filtre à base de MLP opère aux niveaux des trames. Le vecteur d'entrée est composé de 351 coefficients, résultant de la concaténation de 9 trames de 39 coefficients chacune. La couche cachée est composée de 2 000 neurones et la couche de sortie de 108 neurones. Chaque neurone représente un état du phonème-indépendant. Le MLP est entraîné sur un grand corpus en utilisant l'approche de back-propagation.

Chaque neurone de la couche de sortie du MLP est supposé être une estimation de la probabilité $P(X_t|S_i)$ que la trame X_t soit l'état S_i. Le filtre phonétique à base de MLP est ensuite intégré dans le graphe de filtrage d'une faccon similaire aux filtre GMM : un seuil de coupure c_i est associé à chacune de ces 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

tel-00954238, version 1 - 28 Feb 2014

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 depuis 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 compressée

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", et celui-ci obtiendra un gain significatif en termes de calcul sans avoir un impact sur la précision. Il est important de noter que le taux de précision 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 en terme 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 a une requête W.

Pour simplifier, nous linéarisons le graphe en concaténant les modèles en compétition en un filtre phonétique commun. Les filtres résultants $f = \{f_i\}_{i=0,\dots,n-1}$ sont composés en cascade de n filtres phonétiques f_i, correspondant a une séquence phonétique h et à la séquence d'état associé S_i. La pertinence de f est d'estimer via une fonction objective $F_{ab}(f)$ 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 précédent $f_{k,i>k>0}$ dans le filtre de cascade. 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 a passé le filtre ou pas. Ensuite, $D_i(X_t)$ est mis à 1 quand l'inégalité $ll(X_t|S_i) > c_i$ est vrai, et $D_i(X_t)$ et 0 dans l'autre cas. La probabilité *a priori* de passer f_i est désignée par $P(D_i(X_t) = 1)$. Les probabilités *a priori* sont estimées en comptant le nombre de trames qui passent le filtre dans le corpus d'entraînement par le nombre de trames total.

tel-00954238, version 1 - 28 Feb 2014

La probabilité *a priori* d'atteindre le filtre phonétique i est le produit de la probabilité *a piori* $P(D_i(X_t = 1)), k < i$ 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 ce filtre :

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

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

La précision du filtre $f = \{f_i, f_{i-1}, ..., f_0\}$ 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 trouver une requête incorrecte 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, mais 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 rencontrés. Cette valeur est calculé :

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

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

D'une manière similaire, la précision du filtre phonétique $P(S_i|D_i(X_t = 1)$ 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 est passé par le filtre alors effectivement émise 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 objectif 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.

tel-00954238, version 1 - 28 Feb 2014

Cette fonction est utilisée pour déterminer le rang des sous-requêtes. La meilleure *sous-requête ecompressé* 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 atteinte par une évaluation exhaustive de toutes les parties des filtres en cascade f. Ensuite, la requête est substituée par la sous-requête f^{sq} qui est utilisée dans le premier niveau de notre système.

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.

Finalement, la mise au point d'une requête compressée s'effectue en cinq étapes :

- Transcrire de 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'état
- Estimer les filtres acoustiques qui sont attachés dans un graphe de nœud
- 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.

3.3.3 Décodage guidé de la requête

Le but de cette étape est d'affiner la détection atteinte dans le premier niveau. Les segments de parole qui sont passés par le processus de filtrage sont envoyés au système de RAP pour une passe de décodage. Afin d'être sûr que le segment de parole contient le terme complet de la cible, même si une partie seulement de la chaîne phonétique est recherchée (dû à la requête minimale), nous élargissons le segment avant et après la zone sélectionnée. Dans nos expérimentations, nous utilisons une valeur de 0.5 seconde sur les bords des segments.

Rechercher en utilisant un système de RAP est connu pour avoir un bon rappel 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 est grand, plus le risque de probabilité de rencontrer une erreur de mot est grand. Afin de limiter le risque, la probabilité *a priori* de

tel-00954238, version 1 - 28 Feb 2014

la requête est légèrement boostée par l'algorithme de décodage : Driven Decoding Algorithm (DDA) (Lecouteux et al., 2006).

Cet algorithme a pour but d'aligner une transcription *a priori* en utilisant le moteur de reconnaissance de la parole. L'algorithme procède en deux étapes. Premièrement, la transcription fournit h_p. L'hypothèse courante h_x est synchronisés en utilisant un algorithme d'alignement en minimisant la distance d'édition entre les deux chaînes h_p et h_c.

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 trigram courant, un alignement complet du bi-grammes courant et un alignement du mot seulement. Les valeurs de α sont empiriquement déterminées sur un corpus de développement. Ensuite, les probabilités de tri-grammes sont ré-estimées :

$$\tilde{P}(w_i|w_{i-2}, w_{i-3}) = P^{1-\alpha}(w_i|w_{i-2}, w_{i-3}) \tag{3.8}$$

où $\tilde{P}(w_i|w_{i-2}, w_{i-3})$ est la probabilité de tri-gram ré-estimé d'un mot w_i connaissant l'historique w_{i-1}, w_{i-2} et $P(w_i|w_{i-2}, w_{i-3})$ est la probabilité initiale du tri-gram.

Ici, nous utilisons le DDA comme un post-processus opérant sur un segment précédemment identifié comme un bon candidat par le filtre acoustique. Le terme ciblé utilisé comme une transcription *a priori* tend à doucement booster le score linguistique de l'hypothèse qui trouve la requête.

A cette étape, les probabilités de l'OOV sont interpolées par la probabilité du mot inconnu. La probabilité du mot inconnu est estimée classiquement : nous appelons *inc* tous les mots dans le jeu d'entraînement qui sont en dehors du vocabulaire de la reconnaissance. Ensuite, *inc* est vu comme un mot et sa probabilité linguistique est estimée classiquement.

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} :

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

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}.

tel-00954238, version 1 - 28 Feb 2014

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, (Linarès et al., 2007). Ce système utilise un algorithme A⋆ pour le décodage et des HMM pour la modélisation acoustique. Le lexique contient 65 mille 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 extraits sont composés de 12 coefficients PLP et l'énergie, de leurs dérivées première et seconde, soit 39 dimensions pour la transcription. Deux configurations sont réalisées dans ces expériences, selon leur vitesse de décodage exprimé 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 par été et un schéma d'élagage stricte, 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 franccais. 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 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 de la parole lue et le niveau 10 pour de la parole très spontanée. Ici, nous considérons 2 classes : moyenne, correspond aux niveaux 1 à 4, et élevée qui correspond au niveau 5 et plus.

Dans la suite, le corpus EPAC est utilisé seulement comme corpus de test. Les filtres acoustiques et les requêtes compressées sont calibrés 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, ce dernier 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 : moyennement et hautement spontanée. La sous-partie moyennement spontanée obtient un WER de 33.2% et la sous-partie

tel-00954238, version 1 - 28 Feb 2014

hautement spontanée obtient un WER de 47.2%. Dans la configuration 3xRT, ce taux décroit a 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 de contexte indépendant) entraînés sur le corpus de données ESTER. Nous allons étudier en premier l'impact des techniques de coupure (A-GMM), puis de la requête compressée (A+SR-GMM). Finalement, nous évaluerons le système à base de MLP avec le système de coupure et de requête compressée (A+SR-MLP). Dans le Tableau 3.1 nous montrerons les résultats sur le corpus de parole spontanée EPAC en termes de taux de rappel, facteur de temps et taux de filtrage, ce dernier étant la durée cumulée de segments de parole, normalisée 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 a 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, ce facteur est pratiquement réduit par deux. Le MLP démontre l'efficacité d'utiliser des approches discriminantes dans une tâche de filtre. Comme attendu, les performances MLP ont un filtrage plus sélectif (de 37% a 23%) à un taux de 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.

tel-00954238, version 1 - 28 Feb 2014

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 l'entière durée de la 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 :

$$F = \frac{2 * precision * rappel}{precision + rappel} \tag{3.10}$$

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 celui obtenu avec le système ASR-3xRT, lequel est clairement en dehors des limites de temps réel requis pour un processus de détection de terme à la volée. Les deux systèmes bénéficient de filtrage acoustique et de l'algorithme de DDA. Comparé a celui de l'ASR-1xRT on observe, sur les requêtes IV, un gain de F-Mesure de 20%. Booster, la probabilité linguistique semble être réellement efficace pour rechercher un terme dans un sytème de RAP. En intégrant la requête elle-même dans le processus de reconnaissance, cela permet d'améliorer les informations qui tendent à limiter les erreurs sur les énoncés cibles.

3.3.5.3 Détection des performances selon le niveau de spontanéité

Les expériences suivantes cherchent à voir 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 décodage de requête guidée (DDA-1xRT) sont reportés dans le Tableau 3.3. Comme attendu, les performances sont affectées par les disfluences de la parole, la F-Mesure décroît de 0.76 a 0.63 ; le taux de rappel se stabilise, mais le taux de précision décroît d'environ 0.32 en valeur absolue. Le filtrage acoustique prévoit clairement un gain dans toutes les conditions

tel-00954238, version 1 - 28 Feb 2014

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 composés de 1 a 4 mots (70 requêtes OOV, 70 hybrides and 130 requête 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

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 a 0.83) mais la F-Mesure est affectée par le niveau de 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 terme où le processus est guidé par la requête. Le premier niveau repose sur une optimisation de la représentation de la requête comme une cascade de filtres phonétique. Le second niveau effectue un décodage guidé de la requête sur les segments de la parole qui a passé le premier niveau de filtre. Nous évaluons les performances de ces techniques sur de la parole spontanée. Les résultats démontrent que les coupures sur les philtres phonétiques et la compression de requête améliore significativement l'efficacité de recherche de terme, dans toutes les conditions. Plus encore, le décodage de la requête guidée 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 taggs, 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 hypothése plutôt que d'extraire en "aveugle" le contenu de la vidéo, avec des perspectives de robustesse et de vitesse de décodage. Les expériences que nous avons présenté confirme l'idée de la robustesse. Le système pourrait être utilisé conjointement avec un moteur de recherche pour collecter, filtrer les vidéos issues d'une base ouverte comme le web.

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Chapitre 4

Normalisation des données

Contents

4.1 Introduction

Le but d'un système de RAP est d'extraire le contenu linguistique d'un signal de parole enregistrée. Cependant, le signal de parole n'inclut pas seulement l'information linguistique 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'information 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 ses 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 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 par 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 faire face à 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 des formes audio (Audio Pattern Classificatin, APC) inclut de nombreuses tâches telless 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, l'APC 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 l'APC.

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

tel-00954238, version 1 - 28 Feb 2014

citement modélisée dans le processus d'apprentissage car elle n'est pas implicitement capturée dans le corpus d'apprentissage. Récemment, dans le contexte de la tâche de vérification du locuteur basée sur un GMM-UBM, le paradigme de 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 paramètre estimé sur O (O est un jeu de trame 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, genre, 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 mots, la *variabilité du bruit* est censée se trouver dans un sous-espace de faible dimension.

Le succés de la *variabilité du bruit* dépend principalement de l'hypothèse que la variabilité est localisée dans un espace de faible dimension et que les effets liées à l'information utile et inutile sont additives.

Le super-vecteur issu du GMM m_s (s représentant la classe) est statistiquement indépendant et 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 $N(0, I)$. Actuellement, l'équation 4.2 est équivalente à celle obtenue par le MAP de Reynold.

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

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

tel-00954238, version 1 - 28 Feb 2014

où $x_{(h,s)}$ est un vecteur correspondant à une variable latente et ayant une distribution normale standard $N(0, I)$. 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, D est une matrice diagonale (de dimension $MDxMD$), y_s et le vecteur de la classe s (de dimension MD), U est la matrice de faible dimension représentant l'espace des dimensions de l'information inutile (de dimension $MDxR$) et $x_{(h,s)}$ est un vecteur (de dimension R). Les vecteurs y_s et $x_{(h,s)}$ ont théoriquement une distribution normale standard $N(0, I)$. $DD^t = (\Sigma/\tau)$ 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 où plusieurs sessions différentes sont disponibles.

Soit $N(s)$ et $N(h,s)$ les vecteurs contenant l'ordre zéro de la classe s et de la session h :

$$\begin{aligned} N_g(s) &= \sum_{f \in s} \gamma_g(f) \\ N_g(h,s) &= \sum_{f \in (h,s)} \gamma_g(f) \end{aligned} \tag{4.5}$$

où $\gamma(f)$ est la probabilité *a posteriori* d'une Gaussiene g et d'une observation f. $\sum_{f \in s}$ correspond à la somme de toutes les trames de la même classe s et $\sum_{f\ in(h,s)}$ à la somme de toutes les trames des sessions h et des classes s.

Soit $X(s)$ et $X(h,s)$ les vecteurs qui contiennent l'information du première-ordre de la classe s et de la session h :

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

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

tel-00954238, version 1 - 28 Feb 2014

$$\begin{aligned} \overline{X}_g^s &= X_g^s - \sum_{h \in s} N_g^{(h,s)} \cdot \{m + Ux_{(h,s)}\}_g \\ \overline{X}_g^{(h,s)} &= X_g^{(h,s)} - N_g^{(h,s)} \cdot \{m + Dy_s\}_g \end{aligned} \tag{4.7}$$

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 g^{th} composante 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 matrices U peut être estimée ligne par ligne, avec U_g^i comme étant la i^{th} ligne 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 :

tel-00954238, version 1 - 28 Feb 2014

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 $L^{-1}_{(h,s)}$ and $B_{(h,s)}$;
    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 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. Désignons i l'information utile à être modelée et h l'information session (qui représente la variabilité locuteur ou canal). En utilisant le paradigme FA, le modèle de 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 a un état d'un HMM. Ux est la composante de variabilité session. U est composé par les

tel-00954238, version 1 - 28 Feb 2014

vecteurs propres de la variabilité session d'un sous espace. y_i et x_h sont tous les deux normalement distribués selon $N(0, I)$. D est une matrice diagonale de sorte que DD^t soit la matrice de covariance *a priori* de la composante de la phonème. U est une matrice rectangulaire de sorte que UU^t est 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 additive.

Afin d'avoir un compromis entre la précision de la modélisation et la quantité de données conduisant à estimer les paramètres, nous avons choisi i comme étant un phonème modéliser indépendamment 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 suffisantes pour estimer le facteur de session s_h. Dans cette section nous considérerons la variabilité locuteur et canal comme une session. En prenant i comme un phonème de contexte-indépendant, 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 données de phonème produits par différents locuteurs et une diversité de condition acoustique. 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, mais seulement pour compenser les trames de la parole. Il est utilisé sur le corpus d'entraînement et de test.

4.3.1.1 Discussion sur le modèle acoustique G

L'estimation des paramètres FA (décrite dans (Matrouf et al., 2007)) sont basés sur les statistiques de zéro et premier ordre :

$$N_g^{(h,s)} = \sum_{t \in (h,s)} \gamma_g(t) \quad ; \quad X_g^{(h,s)} = \sum_{t \in (h,s)} \gamma_g(t) \cdot t \tag{4.14}$$

où g est un indice de la Gaussienne G. $\gamma_g(t)$ est la probabilité *a posteriori* de la Gaussienne g donnée dans le vecteur cepstral d'observation t. Nous pouvons voir que le rappel de l'estimation des paramètres FA est principalement basé sur la probabilité *a posteriori* $\gamma_g(t)$. L'estimation de ces probabilités dépend des modèles acoustiques. Peut-être qu'une maniére 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

tel-00954238, version 1 - 28 Feb 2014

de l'Equation 4.13 est la concaténation de toutes les moyennes des Gaussiennes contenues dans le HMM : G est le jeu de toutes les Gaussiennes 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 *a posteriori* les probabilités. Dans ce cas les moyennes du super-vecteur sont la concaténation dans le GMM-UBM (G est composé par les Gaussiennes dans l'UBM). Ce GMM-UBM est entraîné en utilisant les données venant d'un large nombre d'utilisateur" et de sources de canaux différents.

4.3.1.2 Estimer le sous-espace de la variabilité session

La matrice U est un paramètre global. Il est estimé en utilisant un large nombre de données contenant la variabilité session. La matrice est itérativement estimée en utilisant l'algorithme d'Expectation Maximization (EM). Pour chaque étape, $x_{session}$ sont estimées ensuite $y_{phoneme}$ est estimé pour chaque phonème (utilisant le nouveau x) et finalement U est estimé globalement, basé sur ces $s_{session}$ et $y_{phoneme}$. Depuis $x_{session}$ et $y_{phoneme}$ dépend aussi de U le processus est itéré. Les étapes de l'algorithme sont décrites plus en profondeur dans (Matrouf et al., 2007).

4.3.1.3 Modèle acoustique

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

$$\hat{t} = t - \sum_{g=1}^{M} \gamma_g(t) \cdot \{U \cdot x_{utterance}\}_{[g]} \tag{4.15}$$

où M est le nombre de Gaussienne 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 $U \cdot x_{utterance}$ est la composante de la variabilité session estimée sur le segment enregistré. C'est un super-vecteur avec $M \times D$ composantes. $\{U \cdot x_{utterance}\}_{[g]}$ est la g^{th} D composante du bloc de vecteur de $U \cdot x_{utterance}$.

Après avoir normalisé tous les segments en utilisant l'Equation 4.15, les HMM du système de RAP sont entraînés en utilisant les données de parole normalisée. Théoriquement, pour chaque segment nous devons estimer la composante de la variabilité session sur chaque phonèmes 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.

tel-00954238, version 1 - 28 Feb 2014

4.3.2 Modéliser variabilité sessions multiples

Dans les précédentes section (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 FA afin de faire face aux multiples effets 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 FA peut-être formulée ainsi :

$$m_{observed} = m_{ubm} + Dy_{phoneme} + Ux_{speaker} + Vz_{channel} \tag{4.16}$$

où, comme précédemment, m sont les moyennes du super-vecteur, y est la partie spécifique au phonème de context-indépendant, 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 où toutes les sessions modélisent une variabilité spécifique. Ici, une estimation de chaque matrice est obtenue de différents corpus. Chaque corpus modélise une variabilité spécifique. La matrice U est estimée sur le corpus de variabilité locuteur où chaque session représente un couplet phonème-locuteur. La matrice V est estimée sur le corpus de variabilité canal. Chaque session représente un couplet phonème-canal.

En vérification du locuteur, les auteurs ont proposé un framework (Kenny, 2006) similaire à l'Equation 4.16. Le framework appelé Joint Factor Analysis (JFA) modélise sur le même corpus, deux variabilités sessions effet locuteur et effet canaux. Cependant, le framework que nous proposons est une variante du JFA. La variabilité session est estimée itérativement et nous proposons de modéliser la variabilité session sur deux corpus différents ce qui permet d'étendre le framework à d'autres variabilités.

4.3.2.1 Estimer le sous-espace de variabilité locuteur et canal

Les matrices U et V sont communs à tous les phonèmess. Les matrices sont jointement optimisées. La procédure d'estimation est présentée dans l'algorithme (Matrouf et al., 2007). Dans une première étape, la matrice U est optimisée dans les données du corpus locuteur. Les $x_{speaker}$ et z_{cannal} vecteurs 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 $x_{speaker}$ et $z_{channel}$ vecteurs sont estimés, ensuite $y_{phoneme}$ est estimé pour chaque phonème (utilisant le nouveau x et z) et finalement, V est estimée globalement, utilisant ces x, z et y variables.

Les statistiques sont calculées pour prendre en compte les matrices U et V :

tel-00954238, version 1 - 28 Feb 2014

$$\begin{aligned}
\overline{X}_g^s &= X_g^s - \sum_{h \in s} N_g^{(h,s)} \cdot \{m + Ux_{(h,s)} + Vz_{(h,s)}\}_g \\
\overline{X}_g^{(h,s)} &= X_g^{(h,s)} - N_g^{(h,s)} \cdot \{m + Dy_s + Vz_s\}_g \\
\overline{Z}_g^s &= Z_g^s - \sum_{h \in s} M_g^{(h,s)} \cdot \{m + Ux_{(h,s)} + Vz_{(h,s)}\}_g \\
\overline{Z}_g^{(h,s)} &= Z_g^{(h,s)} - M_g^{(h,s)} \cdot \{m + Dy_s + Ux_s\}_g
\end{aligned} \tag{4.17}$$

où N^s, $N^{(h,s)}$, X^s, $X^{(h,s)}$ sont les statistiques de zéro$^t h$ et premier-ordre, calculées sur le corpus de variabilité locuteur et M^s, $M^{(h,s)}$, Z^s, $Z^{(h,s)}$ sont les statistiques de zeroth et premier-ordre, 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 :

$$\begin{aligned}
B_{(h,s)} &= \sum_{g \in UBM} U_g^T \cdot \Sigma_g^{-1} \cdot \overline{X_g^{h,s}} \\
L_{(h,s)} &= I + \sum_{g \in UBM} N_g^{(h,s)} \cdot U_g^T \cdot \Sigma_g^{-1} \cdot U_g \\
Q_{(h,s)} &= \sum_{g \in UBM} V_g^T \cdot \Sigma_g^{-1} \cdot \overline{Z_g^{h,s}} \\
P_{(h,s)} &= I + \sum_{g \in UBM} M_g^{(h,s)} \cdot V_g^T \cdot \Sigma_g^{-1} \cdot V_g
\end{aligned} \tag{4.18}$$

où Σ_g est la matrice de covariance de la g^{th} composante 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 :

$$\begin{aligned}
z_{h,s} &= P_{(h,s)}^{-1} \cdot Q_{(h,s)} \\
x_{h,s} &= L_{(h,s)}^{-1} \cdot B_{(h,s)} \\
y_s &= \frac{\tau}{\tau + N_g} \cdot D_g \Sigma_g^{-1} \cdot \overline{X_g^{h,s}}
\end{aligned} \tag{4.19}$$

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 :

$$\begin{aligned}
U_g^i &= \mathcal{L}(g)^{-1} \cdot \mathcal{R}^i(g) \\
V_g^i &= \mathcal{P}(g)^{-1} \cdot \mathcal{Q}^i(g)
\end{aligned} \tag{4.20}$$

tel-00954238, version 1 - 28 Feb 2014

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

$$\begin{aligned}
\mathcal{L}(g) &= \sum_s \sum_{h \in s} N_g^{(h,s)} \cdot (L_{(h,s)}^{-1} + \mathbf{x}_{(h,s)} \mathbf{x}_{(h,s)}^T) \\
\mathcal{R}^i(g) &= \sum_s \sum_{h \in s} \overline{X}_g^{(h,s)}[i] \cdot \mathbf{x}_{(h,s)} \\
\mathcal{P}(g) &= \sum_s \sum_{h \in s} M_g^{(h,s)} \cdot (P_{(h,s)}^{-1} + \mathbf{x}_{(h,s)} \mathbf{x}_{(h,s)}^T) \\
\mathcal{Q}^i(g) &= \sum_s \sum_{h \in s} \overline{Z}_g^{(h,s)}[i] \cdot \mathbf{x}_{(h,s)}
\end{aligned} \tag{4.21}$$

4.3.2.2 Modèle acoustique normalisé sur variabilités multiples

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

$$\hat{t} = t - \sum_{g=1}^{M} \gamma_g(t) \cdot (\{U \cdot x_{utterance}\}_{[g]} + \{V \cdot z_{utterance}\}_{[g]}) \tag{4.22}$$

où $U \cdot x_{utterance}$ et $V \cdot z_{utterance}$ sont les composantes canaux et locuteurs estimées sur l'enregistrement. Les variables latentes estimées $x_{utterance}$, $z_{utterance}$ sont respectivement calculées depuis U et V.

Comme précédemment, après normalisation de tous les segments utilisant l'Equation 4.22, les HMM du système de RAP sont entraînés en utilisant les segments de parole normalisée.

4.3.3 Système description et résultat

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 genre et bande passante détectés par le processus de segmentation et utilisant un modèle de langage trigram.

tel-00954238, version 1 - 28 Feb 2014

– 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 trigram que la *PASS-1*.

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

4.3.3.2 Entraînement modèle acoustique

Le modèle acoustique a été appris sur un corpus d'entraînement où toutes les trames sont normalisées par l'Equation 4.15 or 4.22. La normalisation est aussi appliquée aux trames de test pour être décodé. Pour tous ces résultats le rang des matrices U et V est fixé à 60. Le GMM-UBM dans l'approche du 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 4.15.

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 où 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 4.16. Comparé a notre baseline, *Norm-speaker-channel* le système 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 nuisances variabilité et permet de la supprimer dans l'espace acoustique. Dans ces

tel-00954238, version 1 - 28 Feb 2014

expériences nous limitons la variabilité aux locuteurs et aux canaux mais il est tout à fait possible d'étendre le paradigme FA pour supprimer d'autres variabilités.

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 intervales de WER. Chaque phrase du système *Norm-speaker-channel* est mis dans la même rangée que la phrase de la baseline. Les différences entre les deux systèmes sont principalement basées sur la normalisation des trames de données. Ce tableau permet de comparer les phrases selon leur condition acoustique.

Nous pouvons observer, pour l'interval 0-10, que le WER entre *Norm-speaker-channel* et *Baseline* est augmenté. Nous obtenons sur la rangée 0-5 une augmentation du WER absolu de 2.11%. Sur les rangées 30-100, nous observons quelques gains pour le système *Norm-speaker-channel*. Ce gain est particulièrement important sur les rangées entre 50-100 (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 importantes. Cependant, sur les phrases avec un WER bas la normalisation n'apporte aucune amélioration.

4.4 Conclusion

Dans ces travaux, nous proposons un framework de normalisation de données basé sur le paradigme du FA. Nous avons aussi présenté une extension pour faire face aux multiples et différents types de variabilité. 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. Une robustesse particulièrement intéressante, surtout quand on sait que les documents issus du WEB (principalement les données issues de Youtube et/ou Dailymotion), sont enregistrés dans de mauvaises conditions (enregistrement depuis les téléphones portables, les personnes enregistrés, etc...).

tel-00954238, version 1 - 28 Feb 2014

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

Pour chaque phonèmes 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 $L^{-1}_{(h,s)}$ and $B_{(h,s)}$;
    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

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

tel-00954238, version 1 - 28 Feb 2014

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

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

Interval 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

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Troisième partie

Structuration et catégorisation de collections multimédia

tel-00954238, version 1 - 28 Feb 2014

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 hétérogènes. Les documents peuvent être hétérogènes sur la forme, comme par exemple le genre vidéo ; ou sur le fond, comme par exemple des thèmes. Cette collection hétérogène est un phénomène particulièrement important lorsque les bases sont générées par les utilisateurs. De plus, ces collections sont aussi rarement ou trés mal structurées soit parce qu'une annotation fine serait très coûteuse à cette échelle, soit parce que la production des données elle-même échappe à toute structure claire, soit parce que la compréhension d'une structure par une personne peut être différente d'une autre personne.

Le succès croissant de sites web 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 multimédia 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 structurer de grandes bases de données, nous nous sommes focalisés dans le chapitre 5 à classifier les vidéos selon leur genre vidéo et, dans le chapitre 6 à caractériser, à détecter le niveau de spontanéité d'un locuteur.

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Chapitre 5

Catégorisation selon le genre vidéo

Contents

tel-00954238, version 1 - 28 Feb 2014

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 ou le résumé...).

Il existe potentiellement une multitude d'éléments structurants qui peuvent être appliqués au contenu de la vidéo, à son type, etc... La structuration automatique de telles collections requière une catégorisation de haut niveau par des descripteurs qui ne sont pas reliés non seulement sur le contenu mais aussi sur la forme du document. Le genre est une de ces métadonnées, qui peut aider 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 TrecVid [2].

Nous présenterons dans la section 5.2.1 une taxonomie du genre vidéo puis, dans la section 5.2, nous proposons 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. TrecVid : 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 vidéo : bulletin météo, sport (courses de voiture 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 la détection du genre à 5 classes : sport, actualité, publicité, bande dessiné 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 changés. On a commencé en 1995 avec 3 classes principalement focalisées sur les bulletins météo et la publicité et, en 2011, on est passé 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. Il n'est pas impossible qu'au fil du temps, ce nombre augmente encore avec l'apparition de nouveaux types de programmes télévisés.

5.2.2 Approche basée sur le texte

Les textes extraits à partir d'une vidéo peuvent être classés en 2 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 d'OCR (Optical Character Recognition) (Hauptmann et al., 2002).

La seconde catégorie concerne les informations textuelles de la transcription de la vidéo, lesquelles peuvent être obtenues à partir du sous-titre ou à partir de l'audio. 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... ou, 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 conduites sur la classification de documents de textes (Sebastiani, 2002). De plus, la relation entre les paramètres (les mots) et le genre

tel-00954238, version 1 - 28 Feb 2014

spécifique est facile à comprendre. Par exemple, on ne sera pas surpris de trouver les mots "stade", "ballon" et "arbitre" dans une transcription pour un jeu de sport.

Cependant, utiliser des approches basées sur le texte pose des problèmes difficile à résoudre et comporte quelques inconvénients. Les sous-titres ne sont pas toujours disponibles avec la vidéo, l'obtenir revient souvent à faire de la transcription manuelle qui est très onéreuse. Ces approches sont inutilisables lorsque ces sous-titres ne sont absolument pas disponibles, par exemple sur les plateformes internet d'échanges de contenus vidéo. Une manière peu coûteuse d'obtenir la transcription des vidéos est d'utiliser un système de reconnaissance automatique de la parole. Or, les vidéos issues de la télévision ou de données web ont un taux d'erreurs-mots assez élevé et donc les méthodes utilisées pour faire de la classification de textes seront affectées. C'est pour cela que les approches basées uniquement sur le texte ne sont pas communes dans la littérature et sont souvent des approches combinées avec les autres approches basées sur l'audio ou la vidéo.

L'approche classique pour représenter le texte comme jeu de paramètres est de construire un vecteur utilisant le modèle de "sac-de-mot" (Forman, 2003). Dans lequel chaque terme du vecteur représente le nombre de fois que le mot est apparu 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 requiert un vecteur de paramètres avec une dimension assez élevée si chaque mot unique est inclus. Pour réduire la dimension de l'espace de représentation, une stop-liste et un stemming de mot sont souvent appliqués : ou la stop-liste contient un ensemble de mots discriminant comme "le", "je", etc... et le stemming supprime le suffixe des mots, laissant ainsi la racine du mot. Par exemple le mot : "construction" et "construirons" ont tous les deux, avec le stemming, la même racine "construi". Ces techniques permettent notamment de réduire la dimensionnalité de notre document, et ainsi pouvoir mieux classer les documents.

Une autre approche est de pondérer les mots correspondant à notre document en utilisant la fréquence du terme et la fréquence inverse de document (Fréquence de Terme (TF)-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 Approche basée sur l'audio

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

tel-00954238, version 1 - 28 Feb 2014

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

5.2.4 Approche basée sur la vidéo

Les approches basées sur la vidéo, comme la couleur, le mouvement, l'interprétation du contenus visuels, ont été très largement étudiées.

5.2.4.1 Paramètre basé sur la couleur

Une trame vidéo est composée d'une matrice de point appelés pixel. 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 (Gupta et al., 1997).

La distribution des couleurs dans une trame vidéo est souvent représentée en utilisant un histogramme : cela consiste à représenter le nombre de fois que la couleur est présente dans une trame. Attention toutefois : cette représentation peut poser deux problèmes. D'une part, avec l'histogramme, il est impossible de déterminer la position d'un pixel et, d'autre part, l'information (l'objet) contenue dans une frame peut être rendue sous différentes conditions (lumière, exposition, etc...).

tel-00954238, version 1 - 28 Feb 2014

Cette approche a permis notamment de différencier facilement les cartoons des autres genres (Ianeva et al., 2003).

5.2.4.2 Paramètre basé 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 les exigences 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 peau.

5.2.4.3 Paramètre basé 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 utiliser les vecteurs liés au mouvement du MPEG ou à calculer le flux optique.

Une autre faccon 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 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 le 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 paramètre 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 descripteur audio de haut niveau.

tel-00954238, version 1 - 28 Feb 2014

5.3.1 Tâche et corpus

Les expériences sont conduites sur un corpus vidéo composé de 7 classes communément utilisé pour l'évaluation des méthodes de classification de genre vidéo : *publicité*, *sport*, *actualité*, *cartoon*, *documentaire*, *musique* et *film* (trailer). La base de données contient 1 822 vidéos collectées depuis des plateformes de partage de vidéo. 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 franccais, la classe *musique* contient des chansons en franccais et en anglais.

La collection est découpée en 2 parties : 1 542 vidéos sont utilisées pour l'apprentissage et 280 composent le jeu de test en équilibrant les vidéos sur chaque classe : 220 vidéos pour chaque classe dans le corpus d'apprentissage et 40 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 décomposant la probabilité globale.

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 paroles exclusivement. Quelques séquences de films peuvent être filmées dans un environnement bruité ou dans une pièce silencieuse. La seconde difficulté est que les classes sont potentiellement mal 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...

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

5.3.3 Analyse Factorielle pour l'identification de genre

5.3.3.1 Introduction

L'approche GMM-UBM est un framework standard dans le domaine de la vérification de locuteur (Bimbot et al., 2004). Dans ce travail, nous l'utilisons pour

tel-00954238, version 1 - 28 Feb 2014

la classification de genre vidéo : chaque genre (actualité, film, cartoon, musique, sport, documentaire et publicité) est modélisé par un GMM spécifique.

Un modèle du monde (GMM-UBM) représente l'espace acoustique tandis que le GMM spécifique aux genres est obtenu en adaptant le GMM-UBM. La technique utilisée pour l'adaptation est l'approche MAP (Gauvain and Lee, 1994) de la même faccon que celui de la vérification de locuteur, seulement les vecteurs des moyennes sont adaptés, le poids et les variances ne changent pas.

L'analyse factorielle 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. Le super-vecteur est défini comme la concaténation des moyennes du GMM. Désignons D la dimension de l'esapce acoustique (39 dans notre cas), la dimension des moyennes d'un supervecteur est $M \cdot D$, où M est le nombre de Gaussienne dans l'UBM. Le modèle d'un genre est une session indépendante, il est habituellement estimé pour représenter l'hypothèse inverse : le modèle UBM. Désignons ce modèle pouvant être paramétrisé par $\theta = \{m, \Sigma, \alpha\}$. Dans $(h; GE)$, le genre de l'enregistrement correspond à GE et la session est h. Deux différentes sessions correspondent aux même genres constitués de différentes observations dues à plusieurs raisons : différents locuteurs, différents environnements acoustiques, différentes sortes de musique...

Le modèle de l'analyse factorielle peut être écrit :

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

où $\mathbf{m}_{(h,GE)}$ est un super-vecteur aléatoire de moyenne de session-genre dépendante, $\mathbf{D}$ est la matrice diagonale $MD \times MD$, $\mathbf{y}_{GE}$ un vecteur de genre aléatoire (un MD vecteur), U est la matrice de variabilité session de rang R (une matrice de taille $MD \times R$) et $\mathbf{x}_{(h,GE)}$ une variable aléatoire. $\mathbf{y}_{GE}$ et $\mathbf{x}_{(h,GE)}$ sont normalement distribués autour de $\mathcal{N}(0, I)$. $\mathbf{D}$ satisfait l'équation suivante $\mathbf{I} = \tau \mathbf{D}^t \mathbf{\Sigma}^{-1} \mathbf{D}$ où τ est le facteur de pertinence requis pour l'adaptation MAP, et $\mathbf{D}\mathbf{D}^t$ représente la matrice de covariance *a priori* de $\mathbf{y}_{GE}$.

5.3.3.2 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 de l'analyse factorielle dans

tel-00954238, version 1 - 28 Feb 2014

les données de test, nous pouvons écrire :

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

Le genre 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 ayant pour 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 M est le nombre de Gaussiennes dans l'UBM, $\gamma_g(x)$ 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 $M \times D$ composante. $\{\mathbf{U} \cdot x_{\mathbf{h}_{test}}\}_{[g]}$ est la g^{th}.

5.3.3.3 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 de vraisemblance de toutes les trames. Ici, les GMM partagent leur matrice de covariance ainsi que le poids des mixtures (les deux ont été supprimées de l'équation pour plus de clarté). La soustraction de l'information inutile dans le jeu de test est effectuée au niveau des trames (domaine cepstral).

5.3.3.4 SVM

En utilisant l'équation 5.7, le modèle de l'analyse factorielle estime le super-vecteur contenant seulement l'information du genre, normalisé en respectant la variabilité inutile. Dans (Campbell et al., 2006), les auteurs proposent un noyau qui calcule une distance entre des GMM, adaptée pour les GMM. Désignons $\mathcal{X}_{\mathbf{s}}$ et $\mathcal{X}_{\mathbf{s}'}$ deux séquences de données audio correspondant aux genres GE et GE', l'équation du noyau peut s'écrire ainsi :

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

tel-00954238, version 1 - 28 Feb 2014

Ce noyau est valide seulement si les moyennes du modèle GMM varie (poids et covariance sont pris du modèle du monde). m_{GE} est pris ici du modèle **??**, *i.e.* $m_{GE} = m + \mathbf{D}y_{GE}$.

5.3.3.5 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, extrait utilisant une fenêtre Hamming de 25ms. Chaque trame est composée de 39 coefficients (MFCC 13, δ MFCC 13 et $\delta\delta$ MFCC 13) toutes les 10ms. La prochaine section décrit tous les différents systèmes que nous avons testés dans nos expériences.

5.3.3.6 GMM-UBM-FA

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

D'un UBM et un genre donné, la décomposition de la FA est réalisée avec l'Equation **??**. 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 expliqués dans la section 5.3.3.2.

5.3.3.7 SVM-UBM et FA

Un Support Vector Machine (SVM) est un classifieur à deux classes construit autour d'un noyau. Afin d'utiliser un SVM sur un problème multi-classe, nous proposons d'utiliser le SVM LaRank (Bordes et al., 2007). L'algorithme LaRank est inspiré de l'algorithme de perceptron où l'algorithme va descendre suivant une exploration aléatoire.

5.3.3.8 Résultat

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 d'un rang de 40.

La première ligne montre les résultats obtenus avec une approche GMM-UBM sans compensation de session. La seconde ligne du Tableau 5.1 montre les résultats obtenus avec une approche GMM-UBM-FA. On peut constater que les performances sont grandement améliorées grâce à la FA avec une réduction de l'erreur relative d'environ 66%. Pour le système SVM-UBM-FA, nous observons une réduction de l'erreur relative d'environ 70% comparée aux GMM-UBM système.

tel-00954238, version 1 - 28 Feb 2014

TABLE 5.1 – *Paramètre cepstral pour la classification de genre : taux correct de classification (%) par genre donné par un SVM.*

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	21
SVM-UBM-FA	4	18	14	14	23	9	17	18

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 (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].

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

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

Nos expériences démontrent que la décomposition en FA peut correspondre au problème VGI : le taux de classification est réduit d'environ 70% en erreur relative.

5.3.4 Paramètre acoustique 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. Ces paramètres reposent sur la structure du document ou sur son contenu. Les prochaines sections étudient les paramètres reliés à la morphologie du document, spécialement aux paramètres de l'interactivité du locuteur et

tel-00954238, version 1 - 28 Feb 2014

de la qualité du contenu vidéo. Pour chacun de ces paramètres, nous étudions les performances seules et en combinant à celles précédemment évalués.

5.3.5 Paramètres d'interactivité

Le nombre de personnes et la faccon dont ils communiquent pourraient 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 paroles. Le paramètre d'interactivité a pour but de représenter le profil du locuteur. Le paramètre du locuteur est lui-même composé de 3 paramètres : le nombre de tour de parole, le nombre de locuteur et le temps de parole du principal locuteur.

Ces données sont extraites en utilisant un système de segmentation et regroupement en locuteur. La première étape, effectue une segmentation en Viterbi basée sur les classes suivants : "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 plus l'énergie ainsi que de leur dérivée première et seconde. Ensuite, les deux dernières étapes effectuent une détection du tour de locuteur et un regroupement en locuteur. 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 tour de parole pour chaque vidéo.

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

TABLE 5.3 – *Les paramètres d'interactivité pour la classification de genre : CER par genre avec un classifieur SVM.*

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. Cependant, comme montré dans la matrice de confusion (Tableau 5.4), la distribution des erreurs est tout à fait différente de celle obtenue par la classification cepstrale : la plus fréquente concerne *documentaire* et *musique* tandis que *actualité* est la classe la plus susceptible d'être confondue avec le domaine cepstral. 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.

tel-00954238, version 1 - 28 Feb 2014

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

5.3.6 Paramètre 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, . 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 x_m sont la moyenne sur une fenêtre glissante de taille N et où K représente le modèle phonétique et q_k la séquence phonétique sortant d'un système ASR. Cette mesure est supposée être élevée sur les paroles de basse qualité, et décroissante sur les paroles propres.

TABLE 5.5 – *Qualité de la parole pour la classification de genre : CER par genre en utilisant un classifieur SVM.*

System	Doc.	Actu.	Film	Musique	Cartoon	Pub.	Sport	Total
Q	22	21	52	3	5	71	24	39

Les paramètres de qualité de parole sont proches de ceux obtenus à l'interactivité en terme de taux d'erreurs : nous obtenons environ 39% CER tandis que

tel-00954238, version 1 - 28 Feb 2014

les paramètres d'interactivité ont un taux d'erreur d'environ 38%. La distribution des erreurs est très différente comme montré dans le Tableau 5.6 : les meilleures classes sont actualité et documentaire qui contiennent normalement de la parole correspondant au condition d'entraînement du système ASR.

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

5.3.7 Paramètre linguistique

5.3.7.1 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 ASR 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 nécessaires. Nous proposons donc d'utiliser un modèle de langage standard avec un lexique peu adapté à certains genres ce 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, 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 (**?**). Cette approche sera notre système de base. En effet, nous proposons ici une approche différente et inhabituelle, dans lesquels les fréquences des mots-outils peuvent être utilisées pour identifier le genre écrit.

tel-00954238, version 1 - 28 Feb 2014

5.3.7.2 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.

5.3.7.3 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 aux erreurs lexicales d'un système ASR.

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

5.3.7.4 Evaluation

Nous avons choisi de tester deux types de classifieurs : Boosting et Artificial Neural Network (ANN). Concernant l'extraction des paramètres TF-IDF, nous avons essayé toutes les valeurs n pour la taille de notre vecteur et nous avons trouvé que, dans notre cas, la taille de notre vecteur doit être constituée 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 présent dans

tel-00954238, version 1 - 28 Feb 2014

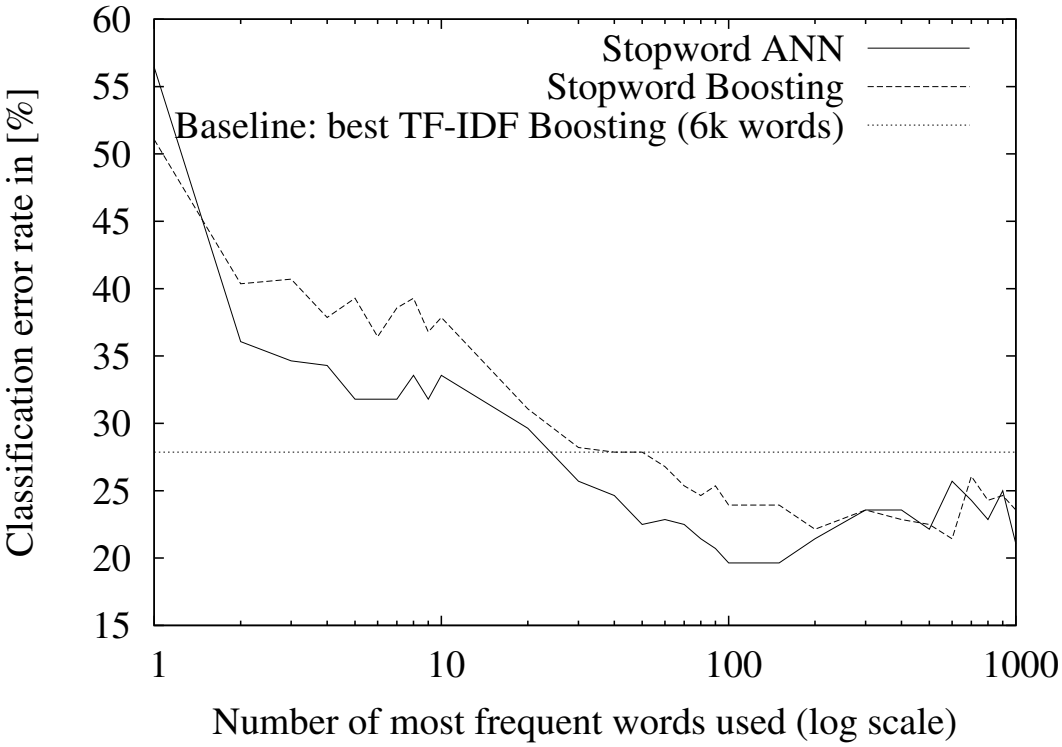

FIGURE 5.1 – *CER (%) du classifieur ANN et Boosting en utilisant les paramètres mots-outils selon le nombre de mots utilisés.*

le vecteur. Nous notons que l'ANN est un MLP doté d'une seule couche cachée ; la taille de la couche caché est optimisée sur le corpus d'entraînement. Dans le vecteur des paramètres, les fréquences sont brutes. Nous avons observé les résultats sont meilleurs lorsque nous normalisons les données.

Les performances obtenues sont comparables avec celles de notre système de base des 6 000 paramètres et elles sont 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, comme entrée, le meilleur classifieur obtient un CER a 51.4% et, en ajoutant le second mot cela donne un CER a 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'informations pour l'identification selon le genre. Le Tableau 5.7 contient les neufs mots les plus fréquents dans le corpus d'apprentissage, associés avec 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 d'environ 8% comparé à notre baseline TF-IDF, tandis que l'espace de représentation est réduit de 98%.

tel-00954238, version 1 - 28 Feb 2014

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

TABLE 5.8 – *Paramètre linguistique pour la classification de genre : CER par genre en utilisant un classifieur ANN.*

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

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

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

5.3.8 Combinaison de paramètres audio

Cette section présente le système qui intègre tous les paramètres audio. Afin de combiner tous les paramètres audio décrits précédemment, nous groupons les paramètres dépendants du score 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 prochains coefficients sont les sorties du classifieur linguistiques (les 7 sorties du réseau de neurone). Les 6 derniers coefficients sont respectivement les 3 paramètres 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

tel-00954238, version 1 - 28 Feb 2014

en 6 parties. 5 parties sont utilisées pour entraîner les différents modèles (modèle cepstral, linguistique) et la dernière partie pour entraîner le méta-modèle.

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

5.3.8.1 Résultats

Les résultats de la combinaison globale sont reportés dans le Tableau 5.10. Nous observons un gain absolu de 3% comparé au meilleur descripteur (descripteur cepstral). Ces résultats montrent que tous les paramètres proposés sont globalement complémentaires pour la classification de genre.

TABLE *5.10* – *CER (%) sur la combinaison des paramètres audios combinés.*

System	Doc.	Actu.	Film	Musique	Cartoon	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	1	07	03	08	19	09
AS+L+Int+Q	04	14	07	02	05	06	18	08

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 5.11 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 voiture, football...).

TABLE *5.11* – *La matrice de confusion en (%) sur la combinaison des paramètres audios.*

System	Doc.	Actu.	Film	Musique	Cartoon	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

tel-00954238, version 1 - 28 Feb 2014

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), ce qui correspondait à environ 350 heures de données. Pour chaque vidéo étaient associées 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'a 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 mélangé la forme (documentaire, film, music...) et le fond (politique, religion, technologie...). Cette catégorisation nous a obligé a proposé un système "allégé". La paramétrisation linguistique était focalisé sur les mots porteurs de sens et il n'y avait aucun paramétre sur l'interactivité du locuteur.

Lors de cette campagne nous avions la possibilité d'utiliser les méta-données associées a chaque vidéo. Nous avons observé que le nom de la personne qui a posté la vidéo (présent dans les méta-données) 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*. En utilisant uniquement notre système "allégé" (sans utilisation de la donnée de la personne qui télécharge) nous nous serions classé 3^{ime}. L'information sur l'utilisateur nous a permis d'améliorer notre système est de remporter cette campagne d'évaluation 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 personnes 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 ceps-

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

tel-00954238, version 1 - 28 Feb 2014

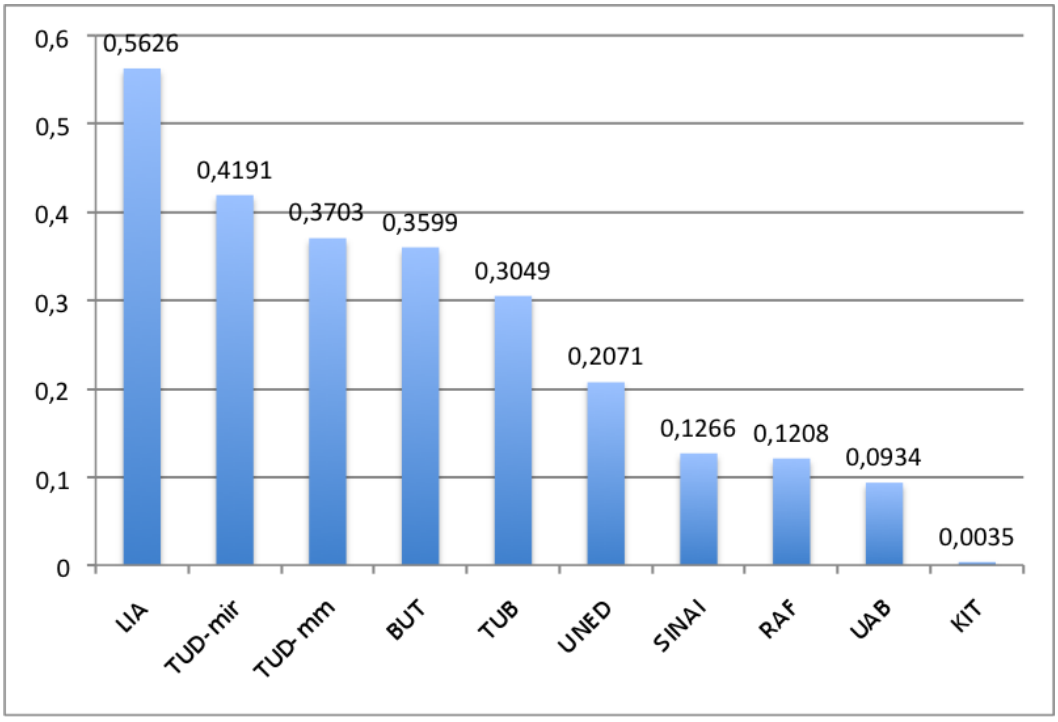

FIGURE 5.2 – *Résultat de la campagne d'évaluation MediaEval 2011 sur la tâche détection du genre (Genre Tagging). Les résultats sont exprimés en MAP (%).*

tral 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 drastiquement la précision du classifieur.

L'extraction automatique de paramètre linguistique est normalement fortement dépendante des performances du système ASR spécialement sur la couverture du lexique qui peut être critique dans de tels 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 plus 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 critères 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 qui alterne les genres (afin de donner une certaine dynamique à notre document).

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Chapitre 6

Structuration de document : détection du niveau de spontanéité

Contents

6.1 Introduction

La détection du niveau de spontanéité a été étudiée ces dernières années par plusieurs auteurs dans le domaine de la reconnaissance du locuteur, mais aussi dans le domaine de la compréhension. 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 où l'auteur entraîne des modèles de langages et acoustiques spécifiques selon le niveau de spontanéité (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... C'est par exemple le cas d'une question embarrassante posée à Nicolas Sarkozy lors d'une conférence au G20 sur un éventuel accueil d'Abdelaziz Bouteflika s'il était chassé du pouvoir [1]. C'est d'abord par un long silence que le chef de l'état a répondu à la question posée avant de commencer sa phrase en bégayant : on peut alors voir que le chef de l'état réfléchit vraiment à sa réponse, ce qui implique un débit de parole faible.

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.

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 ASR. 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 détecter la parole spontanée uniquement sur l'acoustique. Nous proposons de combiner des paramètres acoustiques différent et complémentaire, où chaque paramètre détecte une disfluence caractéristique de la parole spontanée.

6.2 Contribution

6.2.1 Tâche et corpus

Les expériences ont été conduites sur le corpus Franccais EPAC composé de parties spontanées issues de la radio (Estève et al., 2010). Chaque parole de segment est annotée avec un jeu de 10 étiquettes, chacune correspondant à un niveau

1. http ://www.youtube.com/watch ?v=RgMAOwtB$_{ng}$

tel-00954238, version 1 - 28 Feb 2014

de spontanéité : le niveau 1 correspond a de la parole préparée (souvent similaire a de la parole lue) et le niveau 10 correspond à de très grandes disfluences dans la parole (souvent non compréhensible). 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 au niveau 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. Pour ces 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.

6.2.2 Architecture et Principe 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.

Le second niveau consiste à estimer le niveau de spontanéité dans un modèle global, les probabilités estimées localement. Ce modèle 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.

tel-00954238, version 1 - 28 Feb 2014

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 conccu 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 (**?**), les auteurs montrent la difficulté à définir a 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.

L'approche typique pour faire de la classification dans la parole est d'utiliser le couple MFCC/GMM. Le classifieur GMM estime la probabilité de la classe (préparée, faiblement et hautement spontanée) en fonction d'une observation acoustique (une trame). Les scores obtenus pour chaque trame sont ensuite cumulés pour évaluer l'hypothèse de classification sur la séquence en entier. Dans le cadre de la classification du niveau de spontanéité, pour une trame donnée, il faut arriver à différencier par exemple le phonème *e* du mot *euh* et du mot *euler* ; de la même faccon, il faut différencier la pause courte d'une pause longue (l'une est caractéristique de la parole lue et l'autre de la parole spontanée).

Pour arriver à détecter ces pauses (silencieuses ou remplies), nous avons besoin d'avoir une paramétrisation acoustique qui capture pour chaque trame une dynamique du cepstre sur une fenêtre temporelle assez large. Une des faccons d'avoir un apercu 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 delta sont concaténés pour former le paramètre final, et P est le décalage de temps entre blocs consécutifs. En conséquence, kN paramètres sont utilisés pour chaque paramètre SDC, comparés aux $2N$ pour les paramètres conventionnels. Par exemple, le vecteur avec la trame t est donné pour la concaténation de tous les $c(t+iP)$, où :

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

Nous comparons les MFCC et SDC-MFCC sur un classifieur GMM. Puis nous

tel-00954238, version 1 - 28 Feb 2014

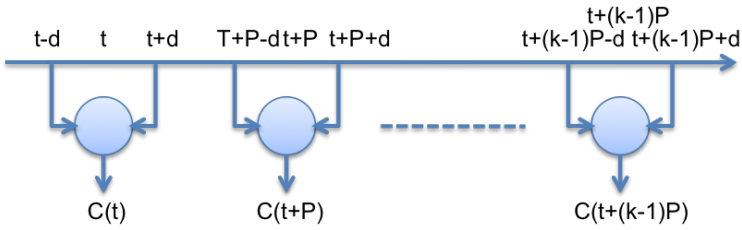

FIGURE 6.1 – *Calcul du Shift Delta Cepstrum.*

utilisons la FA pour essayer de supprimer la variabilité inutile dans l'acoustique. Pour les trois, nous utilisons un modéle de mixture composé de 256 gaussiennes entrainées par un maximum de vraissemblance avec un algorithme d'expectation-maximisation. Le SDC est calculé avec les paramétres 11-3-5 (N-d-P-k). 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.*

	E1	E2	E3	Total
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 de classification augmente 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 que la FA, permet d'améliorer significativement les résultats puisqu'en utilisant la paramétrisation SDC-MFCC, avec et sans FA, nous passons de 41% a 56% de taux de classification.

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 ce qui a pour conséquence d'influer sur la manière dont la phrase va s'articuler. 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 mots associés. Or, dans

tel-00954238, version 1 - 28 Feb 2014

un contexte spontané, cet état émotionnel peut rendre complexe la construction et l'organisation des idées du locuteur ; il peut avoir plus de mal à parler, la construction de ses phrases peut être beaucoup plus difficile et moins compréhensible (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 fa**c**con que la détection d'un état émotionnel. La plupart des approches pour détecter les émotions dans un locuteur sont basées sur les paramètres prosodiques (pitch, énergie et débit d'élocution) et paramètres cepstrals (MFCC et autre). Dans (Neiberg et al., 2006), l'auteur propose d'utiliser comme paramètre cepstraux le MFCC-Low. Les paramètres acoustiques sont calculés de la même fa**c**con 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 paramétres 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.*

	E1	E2	E3	Total
? ?MFCC-Low	0.46 *(0.52/0.42)*	0.28 *(0.25/0.33)*	0.42 *(0.42/0.42)*	**0.39**
? ?SDC-MFCC-Low	0.47 *(0.50/0.44)*	0.28 *(0.24/0.33)*	0.48 *(0.52/0.44)*	**0.41**
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 les meilleurs résultats, avec 54% de taux correct de classification. Le résultat obtenu est certes moins bien que celui obtenu avec 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

tel-00954238, version 1 - 28 Feb 2014

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 la préparation de son contenu, un locuteur varie son débit (ou utilise des pauses) jusqu'à ce que le prochain discours du contenu résultant de la réflexion arrive. 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 *a priori*. Malheureusement, cette méthode n'a pas donné des résultats concluants puisqu'elle a été réalisée sur des transcriptions *a priori*.

Une autre faccon 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 :

$$\phi \bigtriangledown (O;\lambda) = \frac{1}{T}\left[\bigtriangledown_{\lambda} \log p(O;\lambda)\right] \tag{6.2}$$

où λ est un modéle acoustique et O représente les trames d'un segment.

$$\bigtriangledown_{\mu_m} \log p(O;\lambda) = \sum_{t=1}^{T} \gamma_m^{(k)}(t)\Sigma_m^{-1}(o_t - \mu_m^{(k)}) \tag{6.3}$$

où $\gamma_m^{(k)}(t)$ est la probabilité *a posteriori* d'une composante m, μ_m^k est la moyenne et $\gamma_m^{(k)}$ est la matrice de covariance, tous les deux étant associés à la composante m d'un .

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 spontanéité préparé et chaque trame de notre segment. Les vecteurs obtenus avec le noyau de Fisher sont utilisés avec un classifieur SVM. Les résultats sont reportés dans le Tableau 6.3 :

tel-00954238, version 1 - 28 Feb 2014

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

	E1	E2	E3	Total
? ?MFCC-kd	0.46 *(0.52/0.42)*	0.28 *(0.25/0.33)*	0.42 *(0.42/0.42)*	**0.39**
SDC-MFCC-kd	0.57 *(0.60/0.54)*	0.36 *(0.31/0.42)*	0.63 *(0.68/0.59)*	**0.53**

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

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 meilleure 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 gradient descendant.

Dans le Tableau 6.4, nous rappelons la F-Mesure obtenue par les différentes meilleures paramétrisatione acoustiques.

TABLE 6.4 – *Rappel des scores exprimés en F-Mesure sur le niveau de spontanéité selon les paramétres 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, une F-Meure d'environ 55%. Dans

tel-00954238, version 1 - 28 Feb 2014

le Tableau 6.5, nous combinons les différentes paramétrisations acoustiques pour estimer la complémentarité de ces 3 paramètres acoustiques.

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

	E1	E2	E3	Total
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**

En étudiant les résultats obtenus dans le Tableau 6.5 en combinant deux paramètres acoustiques pour n'importe quel ensemble de paramètres (MFCC, MFCC-Low et MFCC-kd), nous observons une amélioration des résultats d'environ 2 points en valeur absolue (55% a 57%) et en combinant les paramètres acoustiques, nous observons une autre amélioration des résultats d'environ 2 points en valeur absolue également (57% a 59%). La combinaison des paramètres acoustiques permet bien d'améliorer le système de classification de 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 parole de segment audio 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 $P(s_i|s_{i-1}, si+1)$ 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 $P(s_i|s_{i-1}, si+1)$ pour chaque parole de segment i détectée sur le fichier audio. La séquence $\hat{S}$ est calculée en utilisant la formule suivante :

tel-00954238, version 1 - 28 Feb 2014

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

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

Le tableau 7.5 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.*

	E1	E2	E3	Total
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'une personne selon 3 classes : 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'étant uniquement focalisée sur l'acoustique. Nous avons proposé 3 différents jeux de paramètres tous centrés sur la détection d'une disfluence particulière.

Structurer un document par niveau de spontanéité nous semble, pour le zapping, l'un des descripteurs les plus importants : en effet, nous espérons pour la création de notre zapping, que les sous-séquences vidéo comportant un fort taux de spontanéité soient d'un intérêt notable pour les utilisateurs.

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Quatrième partie

Résumé automatique sous forme de Zapping

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Chapitre 7

Résumé automatique sous forme de Zapping

Contents

7.1 Introduction

Si les recherches menées dans le résumé automatique s'inscrivent dans une tradition longue de plus de 50 ans (Luhn, 1958), elles ont connu ces dernières années, grâce au Web, un fort renouveau. Le résumé automatique a su évoluer et répondre aux nouvelles problématiques du Web à savoir le traitement de gros corpus et de documents hétérogènes (texte, audio et vidéo).

Avec l'avènement du Web 2.0, nous avons vu l'apparition sur le Web de nouveaux médias audio et vidéo. Dans le domaine de l'audio, les podcasts ont permis aux utilisateurs l'écoute ou le téléchargement automatique d'émissions audio

pour les baladeurs numériques en vue d'une utilisation immédiate ou ultérieure. Le traitement de ces documents a posé quelques problèmes puisqu'il est difficile d'exploiter ces documents audio en raison du temps nécessaire à leur écoute. Benoit Favre () propose comme solution de faire du résumé automatique pour un accès efficace aux bases de données audio.

Avec l'arrivé des services communautaires vidéos (Youtube, Dailymotion, etc...) et les sites dits de "réseaux sociaux" (Facebook, Google+, etc..), la quantité de données vidéo disponibles sur le Web a explosé à tel point qu'il est impossible pour nous de connaître les grands événements quotidiens dans le monde entier. En effet, la quantité d'informations quotidiennes disponibles est tellement grande que nous ne pouvons pas tout visionner. Nous tentons de réduire cet inconvénient en produisant un résumé automatique vidéo qui sera une vision synthétique et subjective d'un événement en se focalisant sur les éléments saillants qui le caractérisent : une sorte de zapping.

Nous proposons donc un modèle de résumé automatique multi-vidéo qui va essayer de rechercher les moments intéressants dans les vidéos tout en minimisant la redondance d'informations entre les vidéos. En donnant quelques notions de moment intéressant et de redondance, le modèle proposé peut être exprimé sous forme de programmation linéaire en nombre entier. Un solveur de programme linéaire en nombre entier peut être utilisé pour maximiser le résultat de la fonction objectif, lequel va chercher efficacement sur l'ensemble des solutions de notre problème.

Dans la section 7.2, nous présenterons l'architecture de notre système de zapping et dans 7.3, le corpus utilisé. Ensuite, dans la section 7.5 nous définirons la saillance ainsi que les méthodes utilisées pour la détecter. Enfin, nous présenterons dans la section 7.6 une solution pour détecter et agréger les moments saillant d'une vidéo.

7.2 Architecture du système

Dans nos travaux, nous avons essayé de nous rapprocher du modèle du "Zapping" proposé par Canal+ : notre document devra contenir l'actualité de la journée et ne devra pas excéder plus de 5 minutes de vidéo. Les différentes étapes de la création du zapping sont :

1. Acquisition de vidéos d'actualité de la veille
2. Sélection des vidéos ayant un intérêt
3. Détection de l'intérêt dans la vidéo
4. Agrégation des différents contenus pour créer notre zapping

L'acquisition des vidéos d'actualité sur des services communautaires est un réel problème. Les vidéos disponibles sur ces plateformes sont pour la plupart très mal

tel-00954238, version 1 - 28 Feb 2014

indexées et très mal structurées rendant la recherche difficile sans des outils automatiques efficaces. 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 et, d'autre part, les informations laissées par un utilisateur peuvent être très mal ou partiellement remplies. Même pour une tâche aussi simple que la recherche de vidéos d'actualité, nous nous apercevons que la plupart d'entre elles sont mal indexées. Nous proposons de récupérer l'ensemble des vidéos disponibles sur des services communautaires et de les filtrer en utilisant notre système de détection du genre vidéo en sélectionnant uniquement les vidéos d'actualité.

La sélection des vidéos ayant un intérêt revient à sélectionner les vidéos qui ont "buzzé" sur Internet. Le terme "buzz" est un terme anglophone signifiant bourdonnement. En marketing, c'est une technique qui consiste à faire beaucoup de bruit autour d'un produit (souvent avant sa sortie) afin d'en assurer une certaine promotion. Lorsqu'il s'agit d'un contenu audio ou vidéo on pourra dire que celui-ci "a fait le buzz" pour exprimer le fait qu'on en a beaucoup parlé, qu'il a été énormément "ébruité" au point d'avoir été entendu ou vu par beaucoup de gens en un temps très court. On peut prendre l'exemple du clip vidéo d'Adele *Someone Like You* qui a fait un buzz, puisqu'en moins de 4 jours le clip aura été visionné plus de 3 millions de fois. Il aura été vu parce que le document contient une information insolite, drôle, émouvante... Actuellement, il est très difficile de détecter si une vidéo est "buzzable[1]". Dans nos travaux nous n'avons pas proposé de méthode de détection de buzz de vidéo, nous sommes juste partis du principe que si une vidéo buzze c'est qu'elle aura été visionnée par un grand nombre de personnes, et que donc, *a contrario*, le nombre de visites d'une vidéo sur la journée nous permet de savoir si la vidéo a buzzé ou pas. Ainsi la sélection des vidéos ayant un intérêt pour l'utilisateur revient à sélectionner les n vidéos d'actualités les plus vues par des utilisateurs de services communautaires.

L'étape de détection d'une sous-séquence vidéo ayant un intérêt pour l'utilisateur revient à rechercher le segment qui maximise une fonction objective. Cette partie consiste à sélectionner l'information dite "importante".

L'agrégation des différents contenus consiste à concaténer les uns à la suite des autres les documents traitant du même sujet et, dans ce cas, à faire attention à ne pas présenter plusieurs fois la même chose. Cette dernière partie consiste à éviter de faire de la redondance d'information.

1. Buzzable : Une vidéo ayant une grande capacité de buzz

tel-00954238, version 1 - 28 Feb 2014

FIGURE 7.1 – *Plateforme permettant de sélectionner les moments saillants des vidéos.*

7.3 Corpus et Evaluation

7.3.1 Corpus

Les vidéos présentes dans notre corpus ont été téléchargées sur le site Dailymotion [2] 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.

Pour évaluer le moment saillant des vidéos, nous avons mis en place un site Internet [3] 7.1, pour demander à des personnes de visionner des vidéos et de sélectionner, selon elles, le meilleur moment saillant.

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 saillant de la vidéo. Une fois le moment saillant sélectionné, l'utilisateur devait cliquer sur "Sauvegarder" afin d'enregistrer la sélection et passer à la vidéo suivante. Si aucun moment saillant n'apparaissait dans la vidéo, l'utilisateur pouvait cliquer sur le bouton : "Aucun moment saillant".

35 utilisateurs ont participé à l'évaluation. Chacun a, en moyenne, évalué un

2. http ://www.dailymotion.com/fr
3. http ://zapping.mickael-rouvier.fr/

tel-00954238, version 1 - 28 Feb 2014

peu moins de 6 vidéos. Il y a eu 40 réponses pour lesquelles les utilisateurs ne savaient pas où se trouvait le moment intéressant dont 22 réponses attribuées par 4 utilisateurs.

Le corpus (les annotations ainsi que les vidéos) est librement téléchargeable depuis cette adresse [4].

7.3.2 Evaluation

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

$$R = \frac{\text{Nombre de trame de la sous-séquence saillante correctement détectées}}{\text{Nombre de trame de la sous-séquence saillante}} \tag{7.1}$$

La sous-séquence saillante correctement détectée correspond à l'intersection entre la sous-séquence saillante issue de la référence et celle obtenue par le système. Cette mesure permet de connaître le ratio entre le nombre de trames correctement trouvé et celui de la référence.

7.4 Segmentation audio et vidéo

La segmentation d'un document est la première des étapes dans le domaine du résumé vidéo. Elle consiste à identifier les limites d'une unité sémantique dans laquelle l'information est pertinente. Cette unité sémantique, suivant la tâche et le document, peut être différente : par exemple, dans un document audio, l'unité sémantique peut être un locuteur, un thème, etc... La segmentation demande une attention particulière puisqu'une mauvaise segmentation produirait une forte réduction de la qualité des résumés automatiques par rapport à une segmentation manuelle ().

Traditionnellement, dans la vidéo, la segmentation se fait par scène (). Les techniques de segmentation par scène sont basées sur une combinaison des segmentations issues de l'audio et de l'image (). Cette combinaison permet, d'une part, de rattraper les erreurs faites par les différentes segmentations automatiques et, 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 de renforcer les limites des segments de scène fournies uniquement par une seule des deux techniques.

Dans le résumé automatique vidéo, dans le cadre de l'extraction d'une sous-séquence saillante, l'unité sémantique peut être différente suivant le type de document. Par exemple, si la sous-séquence qui nous intéresse est l'intervention d'une

4. http ://zapping.mickael-rouvier.fr/corpus.zip

tel-00954238, version 1 - 28 Feb 2014

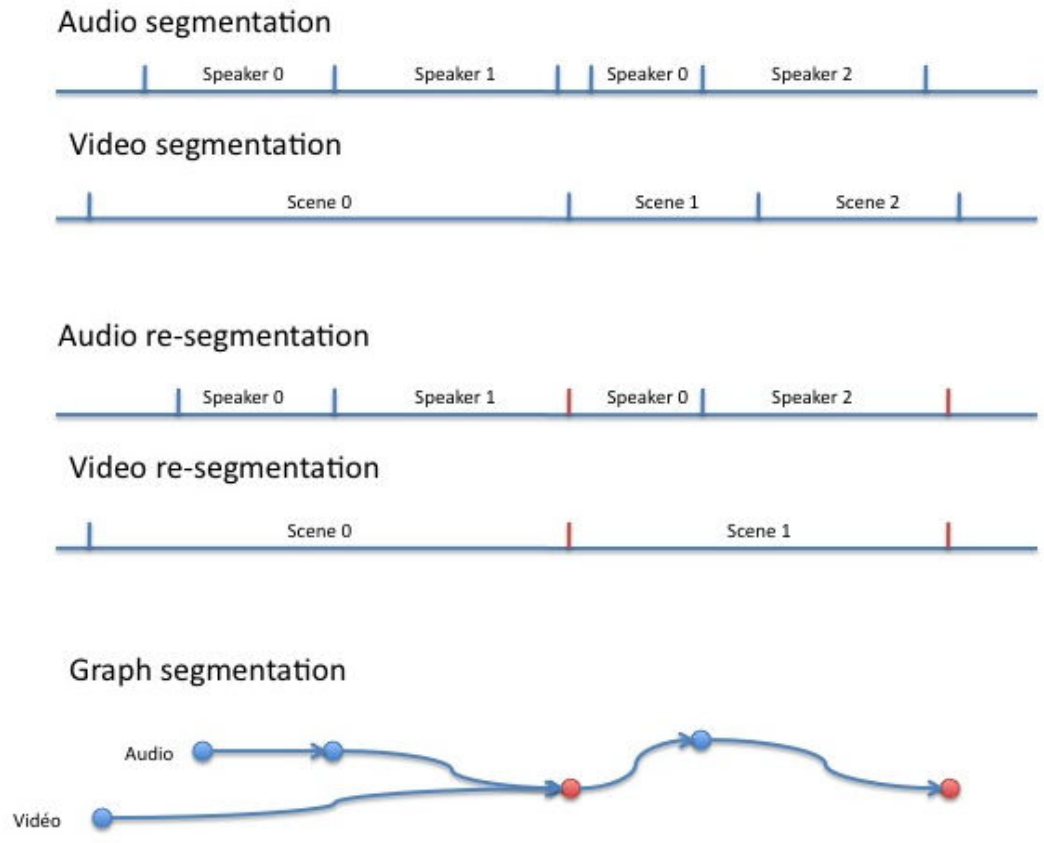

FIGURE 7.2 – *Processus de création du graphe de segmentation.*

personne, alors l'unité sémantique sera le locuteur ; de même pour la vidéo où l'unité sera la scène. Afin de pallier ce problème, nous proposons de créer un graphe issu de la segmentation audio et vidéo. L'algorithme de recherche de la meilleure sous-séquence vidéo sélectionnera dans les différents chemins de notre graphe la meilleure sous-séquence.

Le processus de création de graphe de segmentation, illustré par la Figure 7.2, se déroule en 4 étapes :

1. Segmentation en locuteur et en plan de la vidéo.
2. Détection des points d'accroche : lorsque la fin d'un segment locuteur et la fin d'un segment en plan se terminent plus ou moins en même temps, nous les regroupons. Ainsi dans le graphe, le point qui relie la fin d'un segment locuteur et la fin d'un segment de plan est le même. Nous appelons ce point : un point d'accroche.
3. Nous supprimons deux scènes si elles coupent la parole à un locuteur afin d'éviter que la fin de l'extrait d'une sous-séquence se fasse au milieu d'une phrase d'un locuteur.

tel-00954238, version 1 - 28 Feb 2014

4. Création du graphe.

7.5 Détection d'une sous-séquence vidéo saillante

7.5.1 Algorithme de recherche de sous-séquence saillante

Le but de l'algorithme de recherche de sous-séquence saillante est de détecter dans le graphe de segmentation la sous-séquence vidéo la plus saillante en respectant certaines contraintes : les segments sélectionnés, formant la sous-séquence vidéo, doivent être contigus et, de plus, la sous-séquence vidéo sélectionnée ne doit pas dépasser une certaine durée de temps. Notre problème revient à rechercher une sous-séquence vidéo qui maximise une fonction objectif sujette à une série de contraintes.

Nous proposons de modéliser notre problème sous forme de problème quadratique en variable binaire. Ainsi la résoluton de celui-ci peut se faire de manière globale sur un large éventail de solutions possibles. Tout le contraire des algorithmes gloutons, tels que le MMR, qui va prendre pour chaque itération une solution localement optimale. Dans le cadre d'un résumé automatique vidéo où les segments sélectionnés doivent être contigus, en utilisant un algorithme glouton, le choix du premier segment est primordial.

Pour simplifier l'explication du modèle de recherche de sous-séquence saillante, nous nous placcons dans un cadre ou nous n'avons qu'une segmentation en plan ou en locuteur. Le modèle peut 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 objectif, 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 contigus. Dans l'équation 2 o_x est un critère d'arrêt pour la sélection des segments contigus. Dans le cas où n_x est sélectionné, les deux choix possibles sont la sélection du prochain segment n_{x+1} ou arrêter la sélection du prochain segment o_x. Nous nous assurons

tel-00954238, version 1 - 28 Feb 2014

dans l'équation 3 du nombre de sous-segment à sélectionner dans la vidéo, en faisant la somme des critères d'arrêt.

Dans le cas d'un graphe de segmentation. Le modèle de recherche de sous-séquence saillante s'écrit 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 $n_{x,audio} = n_{x,video}$.

7.5.2 Moment saillant

Le but de la fonction objectif est de rechercher dans le graphe de segment les moments les plus saillants. Un moment saillant dans une vidéo est une sous-séquence qui peut résumer l'actualité, être inhabituelle, inattendue, insolite, drôle, navrante, etc... Définir dans une fonction objectif l'ensemble de ces critères reviendrait à rechercher dans la vidéo une sous-séquence qui représenterait l'ensemble de ces critères : il y a effectivement moins de chance de trouver dans une vidéo un fait combinant l'ensemble de ces critères que de trouver un fait correspondant à l'un de ces critères.

Pour pallier ce problème nous proposons de définir trois fonctions objectifs recherchant chacunes un fait saillant spécifique : un fait saillant peut être une sous-séquence qui résume la vidéo ou en rapport avec l'actualité, il peut être un moment chargé d'émotion ou encore un moment atypique.

7.5.2.1 Résumé

Dans un zapping, un moment saillant peut être une sous-séquence vidéo qui résumé une vidéo et/ou un fait d'actualité. Pour extraire une sous-séquence qui résume une vidéo, nous proposons d'utiliser le résumé automatique texte, tel que proposé dans (). Celui-ci propose de construire le résumé automatique en estimant

tel-00954238, version 1 - 28 Feb 2014

globalement la pertinence et la redondance dans un cadre basé sur la programmation linéaire de nombre entier. Pour cela, l'auteur explique que chaque phrase est constituée de concepts ; il faut donc de sélectionner les concepts les plus pertinents tout en minimisant leur redondance dans le résumé automatique.

Les concepts sont représentés par des éléments d'information comme, par exemple pour un meeting, une décision prise à une réunion, ou l'opinion d'un participant sur un sujet. Mais l'abstraction de tels concepts rend difficile une extraction automatique, il faut ramener ces concepts à des mots plus simples, les n-grams, qui peuvent être utilisés pour représenter la structure du document. Cependant, les n-grams se recoupent souvent avec des marqueurs de discours ("en fait", "vous savez") lesquels peuvent rajouter du bruit.

Pour représenter la séquences des mots ainsi que le contenu, nous proposons une version modifiée de l'algorithme d'extraction de mot-clef proposé initialement dans (). En effet, nous nous sommes aper**c**cus que les poids donnés aux différents concepts étaient très proche les uns des autres. Il n'y avait aucune discrimination forte entre les différents concepts. Nous proposons donc de réévaluer chaque poids des concepts avec une fonction exponentielle (exp) qui permettra de mieux séparer les concepts entre eux tout en renfor**c**cant ou diminuant leur poids. L'algorithme d'extraction de mot-clef se déroule ainsi :

1. Extraction de tous les n-grams pour $n = 1, 2, 3$
2. Suppression du bruit : Suppression des n-grammes qui apparaissent seulement une fois
3. Suppression du bruit : Suppression des n-grammes si un des mots du n-gramme a un IDF plus bas qu'un seuil
4. Suppression du bruit : 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-gram et Tri-gram : $w_i = n \cdot idf(g_i)$ ou w_i est le poids du n-gramme, n la taille du n-gramme et idf le poids IDF [5] 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.

Mais selon les jours, certains concepts, peuvent être importants à inclure dans le résumé. Par exemple lors de l'affaire Nafitassalou Diallou, les concepts comme "chambre 2806", "Sofitel", "DSK" revenaient assez souvent dans l'actualité. Les phrases contenant ces concepts avaient de fortes chances d'être incluses dans le résumé. Ainsi donc, l'actualité peut nous donner une information sur la pertinence des phrases à inclure dans notre résumé. Au plus une phrase, présente dans la vidéo, est similaire à un fait d'actualité, au plus elle a de chances d'être incluse dans ce

5. L'IDF a été calculé sur Wikipedia

tel-00954238, version 1 - 28 Feb 2014

résumé. Nous proposons, dans notre modèle, de donner un poids à chaque phrase de notre vidéo. Ce poids est calculé via une mesure de similarité entre la phrase et la meilleure dépêche d'actualité présent sur Internet. La mesure de similarité utilisée est le cosine.

Ainsi l'algorithme de recherche de sous-séquence vidéo en essayant de résumer la vidéo peut s'écrire comme ceci :

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

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, y. Le paramètre λ est utilisé pour équilibrer les scores attribués aux phrases, avec ceux des concepts. Nous rajoutons, dans notre modèle de détection de sous-séquence saillante, 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 – .

	Concept	Web	Concept/Web
Résultat	0.42	0.38	0.44

Les résultats obtenus avec ce modèle sont reportés dans le Tableau 7.1. En utilisant seulement les concepts pour détecter les sous-séquences saillantes, 42% des sous-séquences saillantes de notre corpus ont été correctement détectées. En couplant les concepts, ainsi que le poids des phrases par rapport au web, notre taux de détection est amélioré puisqu'il passe de 42% a 44% des sous-séquences saillantes détectées.

7.5.2.2 Chargé en émotion, en spontanéité

Un segment peut être saillant, par exemple, parce qu'une personne est interviewée à chaud sur un sujet et parle avec une certaine émotion et qu'il y a donc une certaine spontanéité dans sa réponse. Nous proposons d'utiliser la charge émotive et la spontanéité pour détecter les faits saillants.

Dans (), l'auteur explique que les mots que les personnes emploient jouent un rôle important sur l'état émotionnel dans laquelle la personne se trouve : les mots

tel-00954238, version 1 - 28 Feb 2014

sont corrélés avec l'émotion. 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 cosine) entre une phrase issue de la transcription et le vecteur contenant l'ensemble des mots d'émotion :

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

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 d'émotion. 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 pour chaque segment un poids selon le niveau de spontanéité détecté.

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

$$\text{Maximize} \quad (1-\lambda)(\sum_x \sum_j emotion_{x,j} n_{x,j}) + \lambda(\sum_x \sum_j spont_{x,j} n_{x,j})$$

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 7.2 – .

	Emotion	Spontanéité	Emotion/Spontanéité
Résultat	0.22	0.14	0.34

Le Tableau 7.2 présente les résultats obtenus sur la détection des fait saillants en utilisant la charge émotive et la spontanéité. Globalement, les résultats montrent que la fonction objectif utilisée obtient de moins bons résultats que les autres fonctions objectifs : nous n'obtenons que 34% de bonne détection de fait saillant. Nous espérons, par la suite, que cette fonction objectif soit complémentaire des autres.

7.5.2.3 Atypique

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, quelque chose d'inattendu. Une séquence est atypique car il y eu l'apparition d'une nouvelle information non

tel-00954238, version 1 - 28 Feb 2014

prévisible. Cette nouvelle information est en rupture avec le document. On peut prendre l'exemple de la vidéo : "JK Wedding Entrance Dance" [6]. La scène se passe dans une église du Minnesota lors de la célébration d'un mariage. Tout se passe normalement, les portes de l'église se ferment pour faire entrer les mariés. Et à la surprise général, les garccons et demoiselles d'honneurs du mariage ont organisé une chorégraphie au beau milieu de la nef sur la musique "Forever" de Chris Brown. Notre but est donc d'extraire la chorégraphie qui est un moment atypique de cette vidéo.

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. L'histogramme est ensuite utilisé comme vecteur de forme.

Pour construire le modèle de sac de mots, nous détectons, en premier, dans l'image, les points d'intérêt (Local Interest Point, LIP). Les points d'intérêt sont extraits par une différence de gaussienne et un Laplacien de Gausienne. Ensuite nous calculons les descripteurs SIFT pour chaque région d'intérêt. Tous les descripteurs SIFT de la vidéo sont alors regroupés en 500 classes à l'aide de l'algorithme de k-moyenne. 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 partagent la même information. Ce qui nous intéresse c'est de savoir à quel point une sous-séquence ne partage pas la même information que notre document vidéo. Nous proposons de calculer le score de rupture ainsi :

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

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

Pour détecter, dans le signal audio, si un segment est en rupture avec la vidéo, nous proposons de modéliser les MFCC d'un segment via un GMM. Le GMM permet de caractériser l'information présente dans un segment. Ce qui nous intéresse, c'est de voir sur l'ensemble de la vidéo à quel point l'information modélisée dans le GMM se répète ou pas. Ainsi chaque segment sera pondéré par un score de rupture audio :

6. http ://www.youtube.com/watch ?v=4-94JhLEiN0

tel-00954238, version 1 - 28 Feb 2014

$$son = 1 - \frac{1}{N}\sum_{t} P(x|M) \tag{7.4}$$

où $P(x|M)$ est la probabilité des trames acoustiques de la trame x sachant le modèle M, et N est le nombre de trames de la vidéo.

Nous proposons de pondérer chaque segment par les scores de rupture audio et vidéo. Le but du modèle est de chercher les sous-segments contigus qui sont le plus en rupture avec la vidéo.

$$\text{Maximize} \quad (1-\lambda)(\sum_{x}\sum jimage_{x,j}n_{x,j}) + \lambda(\sum_{x}\sum json_{x,j}n_{x,j})$$

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

TABLE 7.3 – .

	Image	Son	Image/Son
Résultat	0.34	0.28	0.38

7.5.3 Classification

Fort de l'idée de définir différentes fonctions objectifs pour détecter des moments saillants spécifiques dans la vidéo, nous devons au préalable choisir quelle fonction objectif va être utilisée pour chaque vidéo. Les paramètres sur la structure du document nous paraissent de bons indicateurs pour choisir la fonction objectif. Par exemple, il y a de fortes chances que le fait saillant dans une vidéo soit un résumé si celui-ci contient une information dont l'actualité parle en ce moment. Nous proposons donc d'extraire 8 paramètres sur la structure du document qui sont :

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

tel-00954238, version 1 - 28 Feb 2014

Ces paramètres sont utilisés dans un classifieur de type SVM. Nous définissons 3 classes, chacune étant attribué à une fonction objectif. 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 la dernière partie pour entraîner le modèle.

TABLE 7.4 – *Les performances en utilisant un classifieur SVM.*

	Résumé	Emotion/Spontanéité	Atypique	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 objectif à utiliser ce qui permet de détecter 51% des sous-séquences saillantes. Le système est loin d'obtenir les performances de l'oracle qui est à 68% de détection de bonne sous-séquence saillante mais les performances obtenues via l'oracle tendent à prouver que les différentes fonctions objectifs proposées sont complémentaires.

7.6 Agrégation des segments

Cette dernière étape d'agrégation des différents contenus vidéo consiste à concaténer les différents segments vidéos, issus de la même journée, de faccon à ce qu'il n'y ait aucune redondance de l'information. 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.

Actuellement le modèle proposé recherche la meilleure sous-séquence saillante d'une vidéo. Nous proposons de modifier notre modèle pour que l'extraction de la sous-séquence saillante et la redondance de l'information soit faite au même niveau c'est à dire que pour un ensemble de vidéos, le modèle devra rechercher les sous-séquences saillantes tout en minimisant la redondance inter-vidéo. Nous devons donc modifier le modèle pour qu'il traite non pas une vidéo, mais une collection de vidéos et nous devons également définir dans le modèle la notion de redondance. Il est a noter que laà redondance d'informations a été uniquement appliquée aux textes.

Nous proposons d'utiliser la même notion de redondance que celle utilisée dans la fonction objectif résumé (section 7.5.2.1) : chaque phrase contient des concepts et si un concept est sélectionné seulement une et une seule de ces phrases est sélectionnée dans n'importe quelle vidéo.

Ainsi, pour éviter la redondance d'informations inter-vidéos, notre modèle peut s'écrire ainsi :

tel-00954238, version 1 - 28 Feb 2014

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

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

TABLE 7.5 – .

	Local	Global
Résultat	0.51	0.52

Le Tableau 7.5 montre les résultats obtenus avec le nouveau modèle (Global). Nous obtenons 52% de détection correcte de faits saillants, soit une augmentation de 1 point ce qui est un gain faible. Malheureusement, cette faible augmentation est liée à notre corpus. En effet lors de la création de notre corpus, nous avions demandé aux personnes volontaires d'annoter individuellement chaque vidéo, sans prendre en compte les informations liées aux vidéos de la même journée.

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Chapitre 8

Conclusion et perspectives

Contents

8.0.1 Conclusion

Ces dernières années, on a pu constater que le nombre de données à notre disposition, a considérablement augmenté. Internet est l'un des principaux acteurs de cette montée en puissance de documents disponibles, notamment la nouvelle vague du Web 2.0 qui permet aux utilisateurs les plus néophytes de faciliter le partage de documents hétérogènes. Ainsi, le web s'est petit à petit transformé en une sorte de web communautaire. Le problème inhérent à ces énormes masses de données disponibles est la restitution de la connaissance. C'est d'ailleurs devenu un problème dans notre société. Il devient, de plus en plus, aussi vital de savoir comment accéder à une information que d'en détenir la connaissance. La gestion de l'information est devenue un enjeu industriel, scientifique et économique.

On a pu voir que les problématiques liées à la gestion de masse de l'information sont souvent résolues avec des approches liées au résumé automatique. On s'aperccoit que le résumé automatique évolue au fil du temps. Dans les années 50, le résumé automatique s'est attaqué aux documents de type texte puis ces dernières années à d'autres types de document tel que l'audio et la vidéo. Mais on s'aperccoit aussi que le but d'un résumé automatique, n'est plus seulement de produire uniquement une synthèse de l'information : le système doit aussi dégager des tendances, identifier des opinions, des moments intéressants, etc...

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 L'exploitation de ces collections multimé-

dia ne peut se faire que par une caractérisation riche des contenus. Nous avons proposé dans cette étude différentes méthodes pour structurer de grandes bases de donnée audio-visuels afin d'améliorer le résumé automatique vidéo.

La seconde partie de ce document est consacrée à l'extraction du contenu audio d'un document. Dans le chapitre 3 permet de détecter des termes rapide. Cette détection de termes, dans des milieux bruités, nous permettra de valider une hypothése plutôt que d'extraire en "aveugle" le contenu de la vidéo, avec des perspectives de robustesse et de vitesse de décodage. Parce que la qualité de la transcription obtenue avec un système de RAP est très importante dans un système de résumé automatique. Nous avons proposé dans le chapitre 4 une nouvelle normalisation des données acoustiques issue de la FA. Contrairement aux autres approches qui essayé d'utiliser le paradigme FA en modélisant l'information utile, nous avons ici proposé de modéliser l'information inutile pour la supprimer des trames acoustiques.

Nous proposons, dans la troisième partie, des méthodes pour structurer de grandes collections multimédias. Le chapitre 5 permet de classifier 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 descripteur audio de haut niveau. Une version "allégée" de ce système a été utilisée pour participer à la campagne d'évaluation MediaEval et a obtenu les meilleurs résultats. Dans le chapitre 6 nous proposons de caractériser le niveau de spontanéité d'une personne. Nous avons proposé une série de paramètre acoustique essayant de modéliser pour chacun une disfluence caractéristique de la parole spontanée. Contrairement aux autres approches, cette caractérisaton est focalisé uniquement sur les paramètres acoustique.

La quatrième partie propose un modèle de résumé vidéo basé sur celui du résumé automatique texte. Le but de ce résumé vidéo était de fournir un document proche du Zapping proposé par Canal+.

Pour l'instant, 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 que proposé par Canal+ peut être considéré comme une véritable œuvre d'art et, pour l'instant, dans l'état actuel de la recherche dans ce domaine, un ordinateur peut difficilement parvenir à de meilleurs résultats.

8.0.2 Perspectives

Nous allons explorer les perspectives de ce travail dans le but d'améliorer le résumé automatique de vidéo :

1. Améliorer la transcription automatique du contenu parlé

tel-00954238, version 1 - 28 Feb 2014

2. Prédire de manière automatique des vidéos qui vont faire le buzz
3. Essayer de comprendre et de contextualiser les vidéos

C'est un peu un leitmotiv dans ce domaine mais le résumé automatique nécessite une amélioration des performances de la transcription automatique et cela est d'autant plus vrai dans le résumé vidéo puisque l'étude proposée dans ce document n'est pas uniquement restreinte à un type de contenu radiophonique. De plus, nous nous sommes aperccus que malgré les nombreux problèmes d'un système de RAP (liés à la robustesse, vocabulaire, etc...), ceux-ci restaient figés. Les systèmes de RAP souffrent d'une certaine plasticité. 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, voir 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 est très important et permettrait par exemple de sortir plus facilement des noms propres, lieux, citations, etc...

Dans un zapping, l'extraction des moments saillant se fait sur des vidéos qui ont buzzé. La détection du buzz est actuellement basée sur le nombre de personne qui ont visionnée la vidéo. Dans nos recherches, nous nous sommes rendus compte de deux phénomènes : certaines vidéos avaient tendance à buzzer durant quelques jours (voire quelques semaines) après avoir été postées et certaines vidéos peuvent buzzer plusieurs fois. Le fait de pouvoir prédire si un document va buzzer permettrait de ne pas "rater" la vidéo concernée et de l'inclure dans le zapping. La prédiction de buzz devient, ces dernières années, l'une des grandes problématiques des entreprises d'agrégation de flux Internet : ces entreprises pourraient proposer à l'avance ces documents audio-visuels à leurs utilisateurs de manière automatique et sans être obligé de suivre l'actualité.

Dans nos travaux les moments saillant sont détectés : soit parce qu'une sous-séquence résume la vidéo, soit parce qu'elle est atypique, soit parce que la sous-séquence est chargée émotivement. Nous pensons que l'amélioration des moments saillants ne peut se faire sans une compréhension et une contextualisation des vidéos. La compréhension pourrait se faire par l'extraction des concepts a un haut niveau (comme par exemple : football, G20, guerre en irak, etc....). Le but serait de comprendre si des concepts ne sont pas en contradition dans la vidéo (par exemple une actrice de X parlant de littérature) ou dans l'actualité (par exemple Madame Joly proposant la suppression du défilé le 14 juillet).

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Acronyms

ANN	Artificial Neural Network.
BIC	Bayesian Information Criterion.
CER	taux d'erreur de classification.
CSM	Canonical State Models.
DDA	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	Factor Analysis.
GMM	Gaussian Mixture Model.
HMM	Hidden Markov Model.
IDF	Fréquence Inverse de Document.
IV	In Vocabulary.
JFA	Joint Factor Analysis.
LIA	Laboratoire Informatique d'Avignon.
LIUM	Laboratoire d'Informatique de l'Université du Maine.
MAP	Maximum A Posteriori.
MAP	Mean Average Precision.
MFCC	Mel Frequency Cepstral Coefcients.

MLLR	Maximum Likelihood Linear Regression.
MLP	Multi-Layer Perceptron.
MMR	Maximal Marginal Relevance.
NIST	National Institute of Standards and Technology.
OOV	Out Of Vocabulary.
PCA	Principal Component Analysis.
PLP	Perceptual Linear Preditive.
ROUGE	Recall-Oriented Undestudy for Gisting Evaluation.
RST	Théorie de la Structure Rhétorique.
RVB	Rouge Vert Bleu.
SDC	Shifted Delta Cepstra.
SGMM	Subspace Gaussian Mixture Model.
STD	Spoken Term Detection.
SVM	Support Vector Machine.
TF	Fréquence de Terme.
UBM	Universal Background Model.
WER	Word Error Rate.
ZCR	Zero Crossing Rate.

tel-00954238, version 1 - 28 Feb 2014

tel-00954238, version 1 - 28 Feb 2014

Bibliographie

(Barzilay and Elhadad, 1997) R. Barzilay & M. Elhadad, 1997. Using lexical chains for text summarization. *ISTS 1997*. 2.2.2

(Baxendale, 1958) P. B. Baxendale, 1958. Machine-made index for technical literature : An experiment. *IBM Journal of Research and Development* *2*(4), 354 –361. 2.2.1

(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)*, Volume 49, 47–67. 6.2.3.1

(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. 4.1

(Bimbot et al., 2004) F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, D. Petrovska, & D. A. Reynolds, 2004. A tutorial on text-independent speaker verification. *EURASIP Journal on Applied Signal Processing, Special issue on biometric signal processing*. 5.3.3.1

(Bonastre et al., 2005) J.-F. Bonastre, F. Wils, & S. Meignier, 2005. Alize, a free toolkit for speaker recognition. In Proc. of *ICASSP'05, IEEE*, Philadelphia, PA (USA). 5.3.3.5

(Bordes et al., 2007) A. Bordes, L. Bottou, P. Gallinari, & J. Weston, 2007. Solving multiclass support vector machines with larank. In Z. Ghahramani (Ed.), *Proceedings of the 24th International Machine Learning Conference*, Corvallis, Oregon, 89–96. OmniPress. 5.3.3.5, 5.3.3.7

(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)*. 4.1

(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. 2.2.3

(Caelen-Haumont, 2002) G. Caelen-Haumont, 2002. Perlocutory values and functions of melisms in spontaneous dialogue. In Proc. of *First International Conference on Speech Prosody*, Aix-En-Provence, France, 195–198. 6.2.3.2

(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. 5.3.3.4

(Campione and Véronis, 2004) E. Campione & J. Véronis, 2004. Pauses et hésitations en frans spontané. In Proc. of *Journée d'étude sur le Parole (JEP)*, Fès, Maroc. 6.2.3.1

(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 *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, New York, NY, USA, 335–336. ACM. 2.2.5

(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 *XXVIIe Journées d'étude sur la parole (JEP 2008)*, Avignon (France). 5.3.3.5

(Cohen, 1960) J. Cohen, 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement 20*, 37–46. 6.2.1

(Dan Istrate, 2005) N. S. C. F. J.-F. B. Dan Istrate, 2005. Interspeech 2005. In Proc. of *Systems, Man, and Cybernetics*. 5.3.5

(Di Eugenio and Glass, 2004) B. Di Eugenio & M. Glass, 2004. The kappa statistic : a second look. *Comput. Linguist. 30*, 95–101. 6.2.1

(Dimitrova et al., 2000) N. Dimitrova, L. Agnihotri, & G. Wei, 2000. Video classification based on hmm using text and faces. In Proc. of *In European Signal Processing Conference*. 5.2.1, 5.2.4.2

(Dufour, 2010) R. Dufour, 2010. *Transcription Automatique de la Parole Spontanée*. Ph. D. thesis, LIUM. 6.2.5

(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 T. Kobayashi, K. Hirose, & S. Nakamura (Eds.), *INTERSPEECH*, 885–888. ISCA. 6.1

(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)*, Makuhari (Japan). 4.1

tel-00954238, version 1 - 28 Feb 2014

(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)*, Saint-Pétersbourg, Russie. 6.1

(Edmundson, 1969) H. P. Edmundson, 1969. New methods in automatic extracting. *J. ACM 16*, 264–285. 2.2.1

(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. 4.1

(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)*, Malta. 3.3.4.2, 6.2.1

(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)*, Brisbane, Australia, 35–40. 3.3.2.3

(Fischer et al., 1995) S. Fischer, R. Lienhart, & W. Effelsberg, 1995. Automatic recognition of film genres. In Proc. of *ACM Multimedia*. 5.2.1

(Fiscus, 1997) J. Fiscus, 1997. A post processing system to yield reduced word error rates : Recognizer output voting error reduction (rover). In Proc. of *Automatic Speech Recognition and Understanding (ASRU)*, 347–352. 3.3

(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*, xx–yy. II

(Forman, 2003) G. Forman, 2003. An extensive empirical study of feature selection metrics for text classification. *Journal of Machine Learning Research 3*, 1289–1305. 5.2.2

(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)*, Makuhari (Japan). 4.1

(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 *Interspeech, Brighton (UK)*. 2.2.3

(Gauvain and Lee, 1994) J.-L. Gauvain & C.-H. Lee, 1994. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *2*(2), 291–298. 5.3.3.1

tel-00954238, version 1 - 28 Feb 2014

(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 *ICPR*, 814–817. 2.4.2

(Gillick and Favre, 2009) D. Gillick & B. Favre, 2009. A Scalable Global Model for Summarization. In Proc. of *NAACL/HLT 2009 Workshop on Integer Linear Programming for Natural Language Processing*. 2.2.6

(Girgensohn and Boreczky, 1999) A. Girgensohn & J. Boreczky, 1999. Time-constrained keyframe selection technique. In Proc. of *Multimedia Computing and Systems, 1999. IEEE International Conference on*, Volume 1, 756 –761 vol.1. 2.4.2

(Gupta et al., 1997) A. Gupta, R. A. Gupta, & R. Jain, 1997. Visual information retrieval. 5.2.4.1

(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 *TREC*. 5.2.2

(Hori and Furui, 2003) C. Hori & S. Furui, 2003. A new approach to automatic speech summarization. *IEEE Transactions on Multimedia*, 368–378. 2.3.2

(Hovy and Lin, 1998) E. Hovy & C.-Y. Lin, 1998. Automated text summarization and the summarist system. In Proc. of *Proceedings of a workshop on held at Baltimore, Maryland : October 13-15, 1998*, TIPSTER '98, Stroudsburg, PA, USA, 197–214. Association for Computational Linguistics. 2.2.1

(Ianeva et al., 2003) T. Ianeva, A. de Vries, & H. Rohrig, 2003. Detecting cartoons : a case study in automatic video-genre classification. *Multimedia and Expo, IEEE International Conference on 1*, 449–452. 5.2.4.1

(Jitendra Ajmera and Bourlard, 2002) I. A. M. Jitendra Ajmera & H. Bourlard, 2002. Robust hmm-based speech/music segmentation. In Proc. of *ICASSP 2002*. 5.3.6

(Jones, 1972) K. S. Jones, 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation 28*, 11–21. 2.2.1

(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)*, Avignon, France. 6.1, 6.2.3.3

(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 *Proceeding of the 1st ACM international conference on Multimedia information retrieval*, MIR '08, New York, NY, USA, 83–88. ACM. 2.3.1

tel-00954238, version 1 - 28 Feb 2014

(Kenny, 2006) P. Kenny, 2006. Joint factor analysis of speaker and session variability : Theory and algorithms. Technical report, CRIM. 4.3.2

(Kenny et al., 2007) P. Kenny, G. Boulianne, P. Ouellet, & P. Dumouchel, 2007. Speaker and session variability in gmm-based speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on* *15*(4), 1448–1460. 4.1

(Keshet et al., 2009) J. Keshet, D. Grangier, & S. Bengio, 2009. Discriminative keyword spotting. *Speech Communication* *51*(4), 317 – 329. 3.3.2.3

(Knight and Marcu, 2000) K. Knight & D. Marcu, 2000. Statistics-based summarization - step one : Sentence compression. In Proc. of *Proceedings of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence*, 703–710. AAAI Press. 2.1

(Kobla et al., 2000) V. Kobla, D. DeMenthon, & D. S. Doermann, 2000. Identifying sports videos using replay, text, and camera motion features. In M. M. Yeung, B.-L. Yeo, & C. A. Bouman (Eds.), *Storage and Retrieval for Media Databases*, Volume 3972 of *SPIE Proceedings*, 332–343. SPIE. 5.2.2

(Kohler and Kennedy, 2002) M. Kohler & M. Kennedy, 2002. Language identification using shifted delta cepstra. In Proc. of *Circuits and Systems, 2002. MWSCAS-2002. The 2002 45th Midwest Symposium on*, Volume 3, III – 69–72 vol.3. 6.2.3.1

(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 *Proceedings of the 20th international conference on Computational Linguistics*, COLING '04, Stroudsburg, PA, USA. Association for Computational Linguistics. 2.1

(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)*. 3.3, 3.3.3

(Lie and Merialdo, 2010) Y. Lie & B. Merialdo, 2010. Multi-video summarization based on video-mmr. In Proc. of *WIAMIS 2010, 11th International Workshop on Image Analysis for Multimedia Interactive Services, April 12-14, 2010, Desenzano del Garda, Italy*. 2.4.3

(Lin, 2003) C.-Y. Lin, 2003. Improving summarization performance by sentence compression : a pilot study. In Proc. of *Proceedings of the sixth international workshop on Information retrieval with Asian languages - Volume 11*, AsianIR '03, Stroudsburg, PA, USA, 1–8. Association for Computational Linguistics. 2.1

(Lin and Hovy, 2003) C.-Y. Lin & E. Hovy, 2003. The potential and limitations of automatic sentence extraction for summarization. In Proc. of *Proceedings of the HLT-NAACL 03 on Text summarization workshop - Volume 5*, HLT-NAACL-DUC '03, Stroudsburg, PA, USA, 73–80. Association for Computational Linguistics. 2.1

tel-00954238, version 1 - 28 Feb 2014

(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 *INTERSPEECH*, 1847–1850. 2.3.2

(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*, Berlin, Heidelberg, 302–308. Springer-Verlag. 3.3.4.1

(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. 5.2.3.1

(Longworth and Gales, 2008) C. Longworth & M. J. F. Gales, 2008. A generalised derivative kernel for speaker verification. In Proc. of *INTERSPEECH*, 1381–1384. ISCA. 6.2.3.3

(Luhn, 1958) H. P. Luhn, 1958. The automatic creation of literature abstracts. *IBM J. Res. Dev. 2*, 159–165. 2.2.1, 7.1

(Marcu, 1997) D. Marcu, 1997. From discourse structures to text summaries. In Proc. of *In Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, 82–88. 2.2.4

(Maskey and Hirschberg, 2005) S. Maskey & J. Hirschberg, 2005. Comparing lexical, acoustic/prosodic, structural and discourse features for speech summarization. In Proc. of *Interspeech, Portugal (Lisboa)*. 2.3.1

(Maskey and Hirschberg, 2006) S. Maskey & J. Hirschberg, 2006. Summarizing speech without text using hidden markov models. In Proc. of *Proceedings of the Human Language Technology Conference of the NAACL, Companion Volume : Short Papers*, NAACL-Short '06, Stroudsburg, PA, USA, 89–92. Association for Computational Linguistics. 2.3.1

(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)*. 4.3.1.1, 4.3.1.2, 4.3.2.1

(Mihalcea and Tarau, 2004) R. Mihalcea & P. Tarau, 2004. TextRank : Bringing order into texts. In Proc. of *Proceedings of EMNLP-04and the 2004 Conference on Empirical Methods in Natural Language Processing*. 2.2.3

(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)*, 314–317. 3.2

tel-00954238, version 1 - 28 Feb 2014

(Neiberg et al., 2006) D. Neiberg, K. Elenius, & K. Laskowski, 2006. Emotion recognition in spontaneous speech using GMMs. In Proc. of *INTERSPEECH*. ISCA. 6.2.3.2

(Nenkova et al., 2007) A. Nenkova, R. Passonneau, & K. McKeown, 2007. The pyramid method : Incorporating human content selection variation in summarization evaluation. *ACM Trans. Speech Lang. Process. 4*. 2.5.5

(Ono et al., 1994) K. Ono, K. Sumita, & S. Miike, 1994. Abstract generation based on rhetorical structure extraction. In Proc. of *Proceedings of the 15th conference on Computational linguistics - Volume 1*, COLING '94, Stroudsburg, PA, USA, 344–348. Association for Computational Linguistics. 2.2.4

(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*. IDIAP-RR 08-45. 3.3

(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)*, 4330–4333. 4.1

(Radev and Tam, 2003) D. R. Radev & D. Tam, 2003. Summarization evaluation using relative utility. In Proc. of *Proceedings of the twelfth international conference on Information and knowledge management*, CIKM '03, New York, NY, USA, 508–511. ACM. 2.5.2

(Roach and Mason, 2001) M. Roach & J. Mason, 2001. Classification of video genre using audio. In Proc. of *In Proc. Eurospeech*, 2693–2696. 5.2.3.2

(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)*, Boston, MA, USA, 129–136. 3.3

(Sebastiani, 2002) F. Sebastiani, 2002. Machine learning in automated text categorization. *ACM Computing Surveys 34*(1), 1–47. 5.2.2

(Sivic and Zisserman, 2003) J. Sivic & A. Zisserman, 2003. Video google : A text retrieval approach to object matching in videos. In Proc. of *ICCV*, 1470–1477. 7.5.2.3

(Snoek and Worring, 2005) C. G. M. Snoek & M. Worring, 2005. Multimodal video indexing : A review of the state-of-the-art. *Multimedia Tools Appl. 25*(1), 5–35. 5.2.1

(Sroka and Braida, 2005) J. J. Sroka & L. D. Braida, 2005. Human and machine consonant recognition. *Speech Communication 45*(4), 401 – 423. 4.1

tel-00954238, version 1 - 28 Feb 2014

(Truong and Dorai, 2000) B. T. Truong & C. Dorai, 2000. Automatic genre identification for content-based video categorization. In Proc. of *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, Volume 4, 230 –233 vol.4. 5.2.1

(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*. 2.3.2

(Wang et al., 2003) P. Wang, R. Cai, & S.-Q. Yang, 2003. A hybrid approach to news video classification multimodal features. In Proc. of *Information, Communications and Signal Processing, 2003 and the Fourth Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint Conference of the Fourth International Conference on*, Volume 2, 787 – 791 vol.2. 5.2.2, 5.2.4.2

(Watson, 2003) D. Watson, 2003. *Death sentence : the decay of public language*, Volume 9. Random House Australia, Milsons Point, N.S.W. :. II

(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*, SIGIR '98, New York, NY, USA, 20–27. ACM. 3.2

(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 Exposition*, III : 1345–1348. 5.2.4.2, 5.2.4.3

(Wei and Sethi, 1999) G. Wei & I. K. Sethi, 1999. Face detection for image annotation. *Pattern Recognition Letters 20*(11-13), 1313–1321. 5.2.4.2

(Wold et al., 1996) E. Wold, T. Blum, D. Keislar, & J. Wheaten, 1996. Content-based classification, search, and retrieval of audio. *Multimedia, IEEE 3*(3), 27 –36. 5.2.3.1

(Xie et al., 2009) S. Xie, D. Hakkani-Tür, B. Favre, & Y. Liu, 2009. Integrating Prosodic Features in Extractive Meeting Summarization. In Proc. of *ASRU, Merano (Italy)*. 2.3.1

(Xie and Liu, 2010) S. Xie & Y. Liu, 2010. Using confusion networks for speech summarization. In Proc. of *Human Language Technologies : The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, Stroudsburg, PA, USA, 46–54. Association for Computational Linguistics. 2.3.2

(Y. et al., 2004) B. Y., F. D., H. J.-P., & G. Chollet, 2004. Confidence measure for keyword spotting using support vector machines. In Proc. of *International Conference on Acoustics Speech and Signal Processing (ICASSP)*, Montréal, Canada, 588–591. 3.3.2.3

tel-00954238, version 1 - 28 Feb 2014

(Yeung and Liu, 1995) M. M. Yeung & B. Liu, 1995. Efficient matching and clustering of video shots. In Proc. of *Proceedings of the 1995 International Conference on Image Processing (Vol. 1)-Volume 1 - Volume 1*, ICIP '95, Washington, DC, USA, 338–. IEEE Computer Society. 2.4.1

(yew Lin, 2004) C. yew Lin, 2004. Rouge : a package for automatic evaluation of summaries. 25–26. 2.5.4

(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)*. 3.3

(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 *Proceedings of the 10th International Multimedia Modelling Conference*, MMM '04, Washington, DC, USA, 117–. IEEE Computer Society. 2.4.2

(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 *ICIP*, 2905–2908. 5.2.4.2

(Zechner and Waibel, 2000) K. Zechner & A. Waibel, 2000. Minimizing word error rate in textual summaries of spoken language. In Proc. of *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, San Francisco, CA, USA, 186–193. Morgan Kaufmann Publishers Inc. 2.3.2

(Zhang, 1997) H. Zhang, 1997. An integrated system for content-based video retrieval and browsing. *Pattern Recognition 30*(4), 643–658. 2.4.1, 2.4.2

(Zhang and Fung, 2007) J. Zhang & P. Fung, 2007. Speech summarization without lexical features for mandarin broadcast news. In Proc. of *Human Language Technologies 2007 : The Conference of the North American Chapter of the Association for Computational Linguistics ; Companion Volume, Short Papers*, NAACL-Short '07, Stroudsburg, PA, USA, 213–216. Association for Computational Linguistics. 2.3.1

(Zhu and Penn, 2006) X. Zhu & G. Penn, 2006. Summarization of spontaneous conversations. In Proc. of *INTERSPEECH*. ISCA. 6.1

tel-00954238, version 1 - 28 Feb 2014